
TODO zz a
Release TODO zz a

unknown

31.05.2021

Inhaltsverzeichnis

1	Lizenz	3
2	Up-to-date?	5
3	Abhängigkeiten	7
4	Benutzerhandbuch	9
5	Kommandozeilenschnittstellen-Referenz	11
5.1	Positional Arguments	11
5.2	Named Arguments	11
5.3	Sub-commands:	11
6	API-Referenz	13
6.1	annize namespace	13
	Python-Modulindex	99
	Stichwortverzeichnis	101

TODO jooh

KAPITEL 1

Lizenz

Annize wird unter den Bedingungen der AGPL 3 Lizenz verteilt. Das betrifft ebenso alle Dateien ohne Lizenzkopf (Nichtquelldateien, wie Grafiken), außer sie sind ausdrücklich als Drittinhalt gekennzeichnet. Lesen Sie die Sektion ‚Abhängigkeiten‘ bezüglich enthaltener Drittinhalte.

KAPITEL 2

Up-to-date?

Lesen Sie diesen Text von einer anderen Quelle als der Homepage? Falls Sie unsicher sind ob Ihr Paket aktuell ist, sollten Sie die Homepage besuchen und das prüfen. Sie lesen derzeit die Dokumentation für Version 5.0.6379.

Abhängigkeiten

Annize nutzt einige Teile von Drittanbietern.



Benötigt: **Python 3.7**

Benötigt: **GTK 3.24**



Empfohlen: **GNU/Linux**



Enthalten: **banner image** (License: <http://creativecommons.org/licenses/by-sa/3.0>; from here)



Enthalten: **font ,Inconsolata‘** (for websites; by Raph Levien; License: OFL; [from here](#))



Enthalten: **font ,Khula‘** (for websites; by Erin McLaughlin; License: OFL; [from here](#))



Enthalten: **font ,Symbola‘** (for logo symbol; License: free for use; [from here](#))



Enthalten: **icon set ,Oxygen‘** (some icons; [from here](#))



Enthalten: **third-party project logos** ([from here](#))

KAPITEL 4

Benutzerhandbuch

TODO zz

Kommandozeilenschnittstellen-Referenz

```
usage: annize [-h] [--project PROJECT] {do} ...
```

5.1 Positional Arguments

command	Possible choices: do Das auszuführende Kommando.
----------------	---

5.2 Named Arguments

--project	Projektverzeichnis oder -datei (sonst: das aktuelle Arbeitsverzeichnis). Annize wird die übergeordneten Verzeichnisse automatisch mit absuchen.
------------------	---

5.3 Sub-commands:

5.3.1 do

Führe einen Task aus.

```
annize do [-h] [--with-answers-from-json-file WITH_ANSWERS_FROM_JSON_FILE]
          [--with-answers-from-json-string WITH_ANSWERS_FROM_JSON_STRING]
          [--with-answer WITH_ANSWER WITH_ANSWER]
          [taskname]
```

Positional Arguments

taskname	Taskname. Default: „taskchooser“
-----------------	-------------------------------------

Named Arguments

--with-answers-from-json-file	an_Cli_withAnswersFromJsonFile Default: []
--with-answers-from-json-string	an_Cli_withAnswersFromJsonString Default: []
--with-answer	an_Cli_withAnswer Default: []

6.1 annize namespace

6.1.1 Subpackages

annize.aux package

Submodules

annize.aux._project_infos module

annize.callconv package

exception `annize.callconv.MultipleValuesForSingleArgumentError` (*argname*)
Bases: `TypeError`

Parameter `argname` (*str*) –

class `annize.callconv.ParametersFromChildNodesInfos` (*parameter_name: str, multiple: Union[bool, NoneType], if_passing: Union[Callable[[Any], bool], NoneType], if_typehint_matches: bool*)
Bases: `object`

Parameter

- **parameter_name** (*str*) –
- **multiple** (*Optional[bool]*) –
- **if_passing** (*Optional[Callable[[Any], bool]]*) –
- **if_typehint_matches** (*bool*) –

```

    Rückgabetyp None
    if_passing: Optional[Callable[[Any], bool]]
    if_typehint_matches: bool
    multiple: Optional[bool]
    parameter_name: str
class annize.callconv._CallHelper
    Bases: object
    static _CallHelper__convert_kwargs_from_string (argumentinfos, args, kwargs)
    static _CallHelper__shift_args_to_kwargs (calltype, argumentinfos, args, kwargs)
    classmethod call (calltype, args, kwargs)

    Parameter
        • args (Iterable) –
        • kwargs (Dict) –
annize.callconv.always_true (–)
    Parameter _ (Any) –
    Rückgabetyp bool
annize.callconv.call (calltype, args, kwargs)
    Parameter
        • args (Iterable) –
        • kwargs (Dict) –
annize.callconv.get_parameters_from_childnodes_infos (calltype)
    Parameter calltype (Type) –
    Rückgabetyp List[annize.callconv.ParametersFromChildNodesInfos]
annize.callconv.parameter_from_childnodes (paramname, multiple=None, if_passing=None,
                                          if_typehint_matches=True)
    Parameter
        • paramname (str) –
        • multiple (Optional[bool]) –
        • if_passing (Optional[Callable[[Any], bool]]) –
        • if_typehint_matches (bool) –

```

Submodules

annize.callconv.typehints module

```

class annize.callconv.typehints.ListTypeInfo (name, innertypeinfo)
    Bases: annize.callconv.typehints.TypeInfo
        Parameter name (str) –
        property inner
class annize.callconv.typehints.TypeInfo (name, resolvedtype, constructfromstringfct)
    Bases: object
        Parameter
            • name (str) –
            • resolvedtype (Type) –
            • constructfromstringfct (Optional[Callable[[str], Any]]) –
        construct_from_string (strvalue)
            Parameter strvalue (str) –
            Rückgabetyt Any
        property is_constructable_from_string
        property name
        property resolved_type
annize.callconv.typehints.get_arguments_infos (*, for_callable)
    Parameter for_callable (Callable) –
    Rückgabetyt Dict[str, annize.callconv.typehints.TypeInfo]
annize.callconv.typehints.get_typeinfo (for_type)
    Rückgabetyt annize.callconv.typehints.TypeInfo

```

annize.context package

```

class annize.context.Context
    Bases: object
        _METADATA_IS_TOPLEVEL_OBJECT = 'annize..is_toplevel_object'
        __object_metadata_dict (obj)
            Parameter obj (Any) –
            Rückgabetyt Dict[str, Optional[Any]]
        __object_raw_name (obj)
            Parameter obj (Any) –
            Rückgabetyt str
        __put_object (name, obj)
            Parameter

```

- **name** (*str*) –
- **obj** (*Any*) –

Rückgabotyp `None`

add_object (*obj*)

Parameter **obj** (*Any*) –

Rückgabotyp `str`

get_object_metadata (*obj*, *key*, *defaultvalue=None*)

Parameter

- **obj** (*Any*) –
- **key** (*str*) –
- **defaultvalue** (*Optional[Any]*) –

Rückgabotyp `Optional[Any]`

is_friendly_name (*name*)

Parameter **name** (*str*) –

Rückgabotyp `bool`

is_toplevel_object (*obj*)

Parameter **obj** (*Any*) –

Rückgabotyp `bool`

mark_object_as_toplevel (*obj*)

Parameter **obj** (*Any*) –

Rückgabotyp `None`

object_by_name (*name*, *defaultvalue=None*, *, *create_nonexistent=False*)

Parameter

- **name** (*str*) –
- **defaultvalue** (*Optional[Any]*) –
- **create_nonexistent** (*bool*) –

Rückgabotyp `Optional[Any]`

object_name (*obj*)

Parameter **obj** (*Any*) –

Rückgabotyp `str`

object_names (*obj*)

Parameter **obj** (*Any*) –

Rückgabotyp `List[str]`

objects_by_type (*objtype*, *toplevel_only=True*)

Parameter

- **objtype** (*Type[_T]*) –

- **toplevel_only** (*bool*) –

Rückgabetyp List[_T]

set_object_metadata (*obj*, *key*, *value=None*)

Parameter

- **obj** (*Any*) –
- **key** (*str*) –
- **value** (*Optional[Any]*) –

Rückgabetyp None

set_object_name (*obj*, *name*)

Parameter

- **obj** (*Any*) –
- **name** (*str*) –

Rückgabetyp None

exception `annize.context.OutOfContextError`
 Bases: `TypeError`

`annize.context.add_object` (*obj*)

Parameter **obj** (*Any*) –

Rückgabetyp str

`annize.context.current` ()

Rückgabetyp *annize.context.Context*

`annize.context.get_object_metadata` (*obj*, *key*, *defaultvalue=None*)

Parameter

- **obj** (*Any*) –
- **key** (*str*) –
- **defaultvalue** (*Optional[Any]*) –

Rückgabetyp *Optional[Any]*

`annize.context.is_friendly_name` (*name*)

Parameter **name** (*str*) –

Rückgabetyp bool

`annize.context.is_toplevel_object` (*obj*)

Parameter **obj** (*Any*) –

Rückgabetyp bool

`annize.context.object_by_name` (*name*, *defaultvalue=None*, *, *create_nonexistent=False*)

Parameter

- **name** (*str*) –
- **defaultvalue** (*Optional[Any]*) –

- **create_nonexistent** (*bool*) –

Rückgabotyp Optional[Any]

`annize.context.object_name` (*obj*)

Parameter *obj* (*Any*) –

Rückgabotyp str

`annize.context.object_names` (*obj*)

Parameter *obj* (*Any*) –

Rückgabotyp List[str]

`annize.context.objects_by_type` (*objtype*, *toplevel_only=True*)

Parameter

- **objtype** (*Type[_T]*) –
- **toplevel_only** (*bool*) –

Rückgabotyp List[_T]

`annize.context.set_object_metadata` (*obj*, *key*, *value=None*)

Parameter

- **obj** (*Any*) –
- **key** (*str*) –
- **value** (*Optional[Any]*) –

Rückgabotyp None

`annize.context.set_object_name` (*obj*, *name*)

Parameter

- **obj** (*Any*) –
- **name** (*str*) –

Rückgabotyp None

annize.data package

Submodules

annize.data.color module

class `annize.data.color.Color` (*, *red*, *green*, *blue*)

Bases: `object`

Parameter

- **red** (*float*) –
- **green** (*float*) –
- **blue** (*float*) –

property `blue`

```

property green
property html_color_spec
property red
scalehue (*, brightness, saturation=None)
    Parameter
        • brightness (float) –
        • saturation (Optional[float]) –
    Rückgabotyp annize.data.color.Color
transformed (*, brightness=None, saturation=None)
    Parameter
        • brightness (Optional[float]) –
        • saturation (Optional[float]) –
    Rückgabotyp annize.data.color.Color

```

annize.data.container module

```

class annize.data.container.Basket (*a, **b)
    Bases: list

```

annize.data.unique module

```

class annize.data.unique.UniqueId (localid=None)
    Bases: object
        Parameter localid (Optional[str]) –
        __long_str (txt)
            Parameter txt (str) –
            Rückgabotyp str
        __uniqueid_counter = 0
        __uniqueid_lock = <unlocked _thread.lock object>
        property long_str
        property processonly_long_str
        property processonly_short_str
        property short_str

```

annize.data.version module

```

class annize.data.version.AbstractVersionPatternSegment
    Bases: object
    _name_anon (namep)
        Parameter namep (str) –
        Rückgabetypp None
    _str_to_val (txt)
        Parameter txt (str) –
        Rückgabetypp object
    regexp_string ()
        Rückgabetypp str
    property segment_names
    segments_tuples_to_text (segments_tuples)
        Parameter segments_tuples (TODO) –
        Rückgabetypp str

class annize.data.version.ConcatenatedVersionPatternSegment (*, segments)
    Bases: annize.data.version.AbstractVersionPatternSegment
        Parameter segments (Iterable[annize.data.version.
            AbstractVersionPatternSegment]) –
    _name_anon (namep)
    regexp_string ()
        Rückgabetypp str
    property segment_names
    segments_tuples_to_text (segments_tuples)
        Parameter segments_tuples (TODO) –
        Rückgabetypp str

class annize.data.version.NumericVersionPatternSegment (*, partname=None)
    Bases: annize.data.version.AbstractVersionPatternSegment
        Parameter partname (Optional[str]) –
    _name_anon (namep)
    _str_to_val (txt)
        Parameter txt (str) –
        Rückgabetypp object
    property partname
    regexp_string ()
        Rückgabetypp str
    property segment_names

```



```

    segments_tuples_to_text (segments_tuples)
        Parameter segments_tuples (TODO) –
        Rückgabotyp str

class annize.data.version.OptionalVersionPatternSegment (*, segments)
    Bases: annize.data.version.AbstractVersionPatternSegment
        Parameter segments (Iterable[annize.data.version.
            AbstractVersionPatternSegment]) –
        _name_anon (namep)
        regexp_string()
            Rückgabotyp str
        property segment_names
        segments_tuples_to_text (segments_tuples)
            Parameter segments_tuples (TODO) –
            Rückgabotyp str

class annize.data.version.SeparatorVersionPatternSegment (*, text)
    Bases: annize.data.version.AbstractVersionPatternSegment
        Parameter text (str) –
        regexp_string()
            Rückgabotyp str
        segments_tuples_to_text (segments_tuples)
            Parameter segments_tuples (TODO) –
            Rückgabotyp str

class annize.data.version.Version (*, text=None, pattern=None, **segment_values)
    Bases: object
        Parameter
            • text (str) –
            • pattern (annize.data.version.VersionPattern) –
        property pattern
        property segments_tuples
        property segments_values
        property text

class annize.data.version.VersionPattern (*, segments)
    Bases: object
        Parameter segments (Iterable[annize.data.version.
            AbstractVersionPatternSegment]) –
        property segment_names
        property segments
        segments_tuples_to_text (segments_tuples)

```

```

    Parameter segments_tuples (TODO) –
    Rückgabetyp str
text_to_segments_tuples (text)
    Parameter text (str) –
    Rückgabetyp TODO

```

annize.features namespace

Subpackages

annize.features.changelog namespace

Submodules

annize.features.changelog.common module

```

class annize.features.changelog.common.ByVersionControlSystemCommitMessagesChangelog
    Bases: annize.features.changelog.common.Changelog
    _S_CHANGE = '##CHANGE:'
    _S_LABEL = '##LABEL:'
    property entries
class annize.features.changelog.common.Changelog (*, entries=())
    Bases: object
    Parameter entries (Iterable[annize.features.changelog.common.Entry]) –
    property entries
class annize.features.changelog.common.Entry (*, version=None, time=None, items)
    Bases: object
    Parameter
        • version (annize.data.version.Version) –
        • time (datetime.datetime) –
        • items (Iterable[annize.features.changelog.common.Item]) –
    property items
    property time
    property version
class annize.features.changelog.common.Item (*, text)
    Bases: object
    Parameter text (annize.i18n.TrStr) –
    property text
annize.features.changelog.common._items_matcher (o)
annize.features.changelog.common.default_changelog ()

```

Rückgabety *annize.features.changelog.common.Changelog*

annize.features.datainjections namespace

Submodules

annize.features.datainjections.common module

```
class annize.features.datainjections.common.Inject (*, injection, destination)
    Bases: object
        Parameter
            • injection (annize.features.datainjections.common.Injection) –
            • destination (annize.fs.FsEntry) –
class annize.features.datainjections.common.Injection
    Bases: annize.fs.AdHocFsEntry
    _abc_impl = <_abc_data object>
    _initialize()
    inject (destination)
        Parameter destination (annize.fs.FsEntry) –
```

annize.features.datainjections.python module

```
class annize.features.datainjections.python.Injection (*, filename=
                                                         '_project_infos.py',
                                                         version=None)
    Bases: annize.features.datainjections.common.Injection
        Parameter
            • filename (str) –
            • version (annize.data.version.Version) –
    _abc_impl = <_abc_data object>
    inject (destination)
        Parameter destination (annize.fs.FsEntry) –
```

annize.features.dependencies namespace

Submodules

annize.features.dependencies.common module

```
class annize.features.dependencies.common.Artwork (*, kind=None, label, origin=None,
                                                         comment=None, license=None)
    Bases: annize.features.dependencies.common.Dependency
        Parameter
```

- **origin**(*str*) –
- **comment**(*str*) –
- **license**(*str*) –

```
class annize.features.dependencies.common.Dependency(*, kind=None, label=None,
                                                    comment=None, icon=None,
                                                    importance=0)
```

Bases: object

Parameter

- **kind**(*annize.features.dependencies.common.Kind*) –
- **label**(*str*) –
- **comment**(*str*) –
- **icon**(*str*) –
- **importance**(*int*) –

property comment

property icon

property importance

property kind

property label

```
class annize.features.dependencies.common.GnuLinux(*, kind=None, comment="")
Bases: annize.features.dependencies.common.Dependency
```

```
class annize.features.dependencies.common.Included(**b)
Bases: annize.features.dependencies.common.Kind
```

```
class annize.features.dependencies.common.Kind(*, label, importance, flags=None)
Bases: object
```

Parameter

- **label**(*str*) –
- **importance**(*int*) –
- **flags**(*List[object]*) –

property flags

property importance

property label

```
class annize.features.dependencies.common.Recommended(**b)
Bases: annize.features.dependencies.common.Kind
```

```
class annize.features.dependencies.common.Required(**b)
Bases: annize.features.dependencies.common.Kind
```

```
annize.features.dependencies.common.dependencies_to_rst_text(dependencies)
```

Parameter **dependencies** (*List[annize.features.dependencies.common.Dependency]*) –

Rückgabety *str*

annize.features.dependencies.python module

class annize.features.dependencies.python.**Python**(* , version, kind=None)
 Bases: *annize.features.dependencies.common.Dependency*

Parameter

- **version**(*str*) –
- **kind**(annize.features.dependencies.common.Kind) –

annize.features.distributables namespace

Subpackages

annize.features.distributables.store namespace

Submodules

annize.features.distributables.store.pypi module

class annize.features.distributables.store.pypi.**Connection**(* , token)
 Bases: object

Parameter **token**(*str*) –

property **token**

class annize.features.distributables.store.pypi.**Upload**(* , source, connection)
 Bases: object

Parameter

- **source**(annize.fs.FsEntry) –
- **connection** (annize.features.distributables.store.pypi.
Connection) –

Submodules

annize.features.distributables.common module

class annize.features.distributables.common.**File**
 Bases: *annize.fs.FsEntry*

_abc_impl = <_abc_data object>

class annize.features.distributables.common.**Group**(* , title, files, description="", packa-
 gestore=None)

Bases: object

Parameter

- **title**(*str*) –
- **files**(*List*[annize.fs.FsEntry]) –
- **description**(*str*) –

```

    • packagestore (annize.features.distributables.common.
      PackageStore) –
__get_files_from_packagestore()
    Rückgabotyp Iterable[annize.fs.FsEntry]
__getname()
    Rückgabotyp str
__store_files_to_packagestore()
    Rückgabotyp None
property description
get_files()
    Rückgabotyp List[annize.fs.FsEntry]
property packagestore
property title
class annize.features.distributables.common.PackageStore
    Bases: object
get_package(*, name)
    Parameter name (str) –
    Rückgabotyp Optional[Iterable[annize.fs.FsEntry]]
get_package_history(*, name, limit=3)
    Parameter
    • name (str) –
    • limit (int) –
    Rückgabotyp Iterable[Iterable[annize.fs.FsEntry]]
store_package(fsentries, *, name)
    Parameter
    • fsentries (Iterable[annize.fs.FsEntry]) –
    • name (str) –
    Rückgabotyp None

```

annize.features.distributables.debian module

```

annize.features.distributables.debian.AdministrationUtilitiesSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection
annize.features.distributables.debian.ApplicationsAccessibilityCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

```

```

annize.features.distributables.debian.ApplicationsAmateurradioCategory
  Alias von annize.features.distributables.debian._debian_category.<locals>.
  ACategory

annize.features.distributables.debian.ApplicationsDatamanagementCategory
  Alias von annize.features.distributables.debian._debian_category.<locals>.
  ACategory

annize.features.distributables.debian.ApplicationsEditorsCategory
  Alias von annize.features.distributables.debian._debian_category.<locals>.
  ACategory

annize.features.distributables.debian.ApplicationsEducationCategory
  Alias von annize.features.distributables.debian._debian_category.<locals>.
  ACategory

annize.features.distributables.debian.ApplicationsEmulatorsCategory
  Alias von annize.features.distributables.debian._debian_category.<locals>.
  ACategory

annize.features.distributables.debian.ApplicationsFilemanagementCategory
  Alias von annize.features.distributables.debian._debian_category.<locals>.
  ACategory

annize.features.distributables.debian.ApplicationsGraphicsCategory
  Alias von annize.features.distributables.debian._debian_category.<locals>.
  ACategory

annize.features.distributables.debian.ApplicationsMobiledevicesCategory
  Alias von annize.features.distributables.debian._debian_category.<locals>.
  ACategory

annize.features.distributables.debian.ApplicationsNetworkCategory
  Alias von annize.features.distributables.debian._debian_category.<locals>.
  ACategory

annize.features.distributables.debian.ApplicationsNetworkCommunicationCategory
  Alias von annize.features.distributables.debian._debian_category.<locals>.
  ACategory

annize.features.distributables.debian.ApplicationsNetworkFiletransferCategory
  Alias von annize.features.distributables.debian._debian_category.<locals>.
  ACategory

annize.features.distributables.debian.ApplicationsNetworkMonitoringCategory
  Alias von annize.features.distributables.debian._debian_category.<locals>.
  ACategory

annize.features.distributables.debian.ApplicationsNetworkWebbrowsingCategory
  Alias von annize.features.distributables.debian._debian_category.<locals>.
  ACategory

annize.features.distributables.debian.ApplicationsNetworkWebnewsCategory
  Alias von annize.features.distributables.debian._debian_category.<locals>.
  ACategory

annize.features.distributables.debian.ApplicationsOfficeCategory
  Alias von annize.features.distributables.debian._debian_category.<locals>.
  ACategory

```

```

annize.features.distributables.debian.ApplicationsProgrammingCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.ApplicationsProjectmanagementCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.ApplicationsScienceAstronomyCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.ApplicationsScienceBiologyCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.ApplicationsScienceCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.ApplicationsScienceChemistryCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.ApplicationsScienceDataanalysisCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.ApplicationsScienceElectronicsCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.ApplicationsScienceEngineeringCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.ApplicationsScienceGeoscienceCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.ApplicationsScienceMathematicsCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.ApplicationsScienceMedicineCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.ApplicationsSciencePhysicsCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.ApplicationsScienceSocialCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.ApplicationsShellsCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

```



```

annize.features.distributables.debian.ApplicationsSoundsCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.ApplicationsSystemAdministrationCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.ApplicationsSystemCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.ApplicationsSystemHardwareCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.ApplicationsSystemLanguageenvironmentCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.ApplicationsSystemMonitoringCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.ApplicationsSystemPackagemanagementCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.ApplicationsSystemSecurityCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.ApplicationsTerminalemulatorsCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.ApplicationsTextCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.ApplicationsTvandradiocategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.ApplicationsVideoCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.ApplicationsViewersCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.ApplicationsWebdevelopmentCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

class annize.features.distributables.debian.Category(*,    debian_name,    free-
                                                    desktop_name)

    Bases: object

    Parameter

```

```

        • debian_name (str) –
        • freedesktop_name (str) –

property debian_name
property freedesktop_name

annize.features.distributables.debian.CommunicationProgramsSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

annize.features.distributables.debian.DatabasesSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

annize.features.distributables.debian.DebianInstallerUdebPackagesSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

annize.features.distributables.debian.DebugPackagesSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

annize.features.distributables.debian.DevelopmentSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

annize.features.distributables.debian.DocumentationSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

annize.features.distributables.debian.EditorsSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

annize.features.distributables.debian.EducationSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

annize.features.distributables.debian.ElectronicsSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

annize.features.distributables.debian.EmbeddedSoftwareSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

class annize.features.distributables.debian.ExecutableLink (*, path, name=None)
    Bases: object

        Parameter

        • path (str) –
        • name (str) –

        property name
        property path

annize.features.distributables.debian.FontsSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

```

```

annize.features.distributables.debian.GamesActionCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.GamesAdventureCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.GamesBlocksCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.GamesBoardCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.GamesCardCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.GamesPuzzlesCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.GamesSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

annize.features.distributables.debian.GamesSimulationCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.GamesStrategyCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.GamesToolsCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.GamesToysCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.GnomeSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

annize.features.distributables.debian.GnuRSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

annize.features.distributables.debian.GnustepSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

annize.features.distributables.debian.GraphicsSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

```

```
annize.features.distributables.debian.HamRadioSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

annize.features.distributables.debian.HaskellSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

annize.features.distributables.debian.HelpCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.InterpretersSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

annize.features.distributables.debian.IntrospectionSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

annize.features.distributables.debian.JavaSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

annize.features.distributables.debian.JavascriptSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

annize.features.distributables.debian.KdeSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

annize.features.distributables.debian.KernelsSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

annize.features.distributables.debian.LanguagePacksSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

annize.features.distributables.debian.LibrariesSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

annize.features.distributables.debian.LibraryDevelopmentSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

annize.features.distributables.debian.LispSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

annize.features.distributables.debian.MailSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

annize.features.distributables.debian.MathematicsSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection
```

```
class annize.features.distributables.debian.MenuEntry (*, name, title, category, command, is_gui, icon=None)
```

Bases: object

Parameter

- **name** (*str*) –
- **title** (*str*) –
- **category** (annize.features.distributables.debian.Category) –
- **command** (*str*) –
- **is_gui** (*bool*) –
- **icon** (annize.fs.FsEntry) –

property category

property command

property icon

property is_gui

property name

property title

```
annize.features.distributables.debian.MetaPackagesSection
```

```
Alias von annize.features.distributables.debian._debian_section.<locals>.
ASection
```

```
annize.features.distributables.debian.MiscellaneousSection
```

```
Alias von annize.features.distributables.debian._debian_section.<locals>.
ASection
```

```
annize.features.distributables.debian.MonoCliSection
```

```
Alias von annize.features.distributables.debian._debian_section.<locals>.
ASection
```

```
annize.features.distributables.debian.NetworkSection
```

```
Alias von annize.features.distributables.debian._debian_section.<locals>.
ASection
```

```
annize.features.distributables.debian.NewsgroupsSection
```

```
Alias von annize.features.distributables.debian._debian_section.<locals>.
ASection
```

```
annize.features.distributables.debian.OcamlSection
```

```
Alias von annize.features.distributables.debian._debian_section.<locals>.
ASection
```

```
annize.features.distributables.debian.OldLibrariesSection
```

```
Alias von annize.features.distributables.debian._debian_section.<locals>.
ASection
```

```
annize.features.distributables.debian.OtherOSsAndFSsSection
```

```
Alias von annize.features.distributables.debian._debian_section.<locals>.
ASection
```

```
class annize.features.distributables.debian.Package(*, source, menuentries, executable_links, packagename=None, description=None, summary=None, section=None, homepage_url=None, version=None, documentation=None, author=None, prerm="", postinst="", architecture='all')
```

Bases: `annize.fs.AdHocFsEntry`

Parameter

- **source** (`annize.fs.FsEntry`) –
- **menuentries** (`List[annize.features.distributables.debian.MenuEntry]`) –
- **executable_links** (`List[annize.features.distributables.debian.ExecutableLink]`) –
- **packagename** (`Optional[str]`) –
- **description** (`Optional[str]`) –
- **summary** (`Optional[str]`) –
- **section** (`annize.features.distributables.debian.Section`) –
- **homepage_url** (`Optional[str]`) –
- **version** (`annize.data.version.Version`) –
- **documentation** (`annize.fs.FsEntry`) –
- **author** (`annize.features.authors.Author`) –
- **prerm** (`str`) –
- **postinst** (`str`) –
- **architecture** (`str`) –

```
class _BuildInfo(source: annize.fs.FsEntry, executable_links: List[annize.features.distributables.debian.ExecutableLink], menuentries: List[annize.features.distributables.debian.MenuEntry], services: List[annize.features.distributables.debian.ServiceDescription], description: str, name: str, version: annize.data.version.Version, homepage: str, author: annize.features.authors.Author, licensename: str, section: Union[annize.features.distributables.debian.Section, NoneType], summary: str, documentation_source: Union[annize.fs.FsEntry, NoneType], prerm: str, postinst: str, architecture: str, authorstring: str = None, pkgrootpath: str = None, pkgpath_debian: str = None, pkgpath_documentation: str = None, pkgpath_pixmap: str = None, pkgpath_usrbin: str = None, config_files: List[str] = None, pkgsize: int = None, result: annize.fs.FsEntry = None)
```

Bases: `object`

```
_BuildInfo(source: annize.fs.FsEntry, executable_links: List[annize.features.distributables.debian.ExecutableLink], menuentries: List[annize.features.distributables.debian.MenuEntry], services: List[annize.features.distributables.debian.ServiceDescription], description: str, name: str, version: annize.data.version.Version, homepage: str, author: annize.features.authors.Author, licensename: str, section: Union[annize.features.distributables.debian.Section, NoneType], summary: str, documentation_source: Union[annize.fs.FsEntry, NoneType], prerm: str, postinst: str, architecture: str, authorstring: str = None, pkgrootpath: str = None, pkgpath_debian: str = None, pkgpath_documentation: str = None,
```

pkgpath_pixmap: str = None, pkgpath_usrbin: str = None, config_files: List[str] = None, pkgsize: int = None, result: annize.fs.FsEntry = None)

Parameter

- **source** (annize.fs.FsEntry) –
- **executable_links** (List[annize.features.distributables.debian.ExecutableLink]) –
- **menuentries** (List[annize.features.distributables.debian.MenuEntry]) –
- **services** (List[annize.features.distributables.debian.ServiceDescription]) –
- **description** (str) –
- **name** (str) –
- **version** (annize.data.version.Version) –
- **homepage** (str) –
- **author** (annize.features.authors.Author) –
- **licensename** (str) –
- **section** (Optional[annize.features.distributables.debian.Section]) –
- **summary** (str) –
- **documentation_source** (Optional[annize.fs.FsEntry]) –
- **preinst** (str) –
- **postinst** (str) –
- **architecture** (str) –
- **authorstring** (str) –
- **pkgrootpath** (str) –
- **pkgpath_debian** (str) –
- **pkgpath_documentation** (str) –
- **pkgpath_pixmap** (str) –
- **pkgpath_usrbin** (str) –
- **config_files** (List[str]) –
- **pkgsize** (int) –
- **result** (annize.fs.FsEntry) –

Rückgabetyyp None

architecture: str

author: [annize.features.authors.Author](#)

authorstring: str = None

config_files: List[str] = None

description: str

```

documentation_source: Optional[annize.fs.FsEntry]
executable_links: List[annize.features.distributables.debian.ExecutableLink]
homepage: str
licensename: str
menuentries: List[annize.features.distributables.debian.MenuEntry]
name: str
pkgpath_debian: str = None
pkgpath_documentation: str = None
pkgpath_pixmaps: str = None
pkgpath_usrbin: str = None
pkgrootpath: str = None
pkgsize: int = None
postinst: str
prerm: str
result: annize.fs.FsEntry = None
section: Optional[annize.features.distributables.debian.Section]
services: List[annize.features.distributables.debian.ServiceDescription]
source: annize.fs.FsEntry
summary: str
version: annize.data.version.Version

_abc_impl = <_abc_data object>

_initialize()

classmethod _mkpackage (build)
    Parameter build      (annize.features.distributables.debian.Package.
                          _BuildInfo) –
    Rückgabety annize.fs.FsEntry

classmethod _mkpackage_correctbuildsourcepermissions (build)
    Parameter build      (annize.features.distributables.debian.Package.
                          _BuildInfo) –
    Rückgabety None

classmethod _mkpackage_determinesize (build)
    Parameter build      (annize.features.distributables.debian.Package.
                          _BuildInfo) –
    Rückgabety None

classmethod _mkpackage_dpkg (build)
    Parameter build      (annize.features.distributables.debian.Package.
                          _BuildInfo) –

```



```

    Rückgabetyyp None
classmethod _mkpackage_mkchangelog (build)
    Parameter build      (annize.features.distributables.debian.Package.
                          _BuildInfo)-
    Rückgabetyyp None
classmethod _mkpackage_mkcopyright (build)
    Parameter build      (annize.features.distributables.debian.Package.
                          _BuildInfo)-
    Rückgabetyyp None
classmethod _mkpackage_mkdebianconffilesfile (build)
    Parameter build      (annize.features.distributables.debian.Package.
                          _BuildInfo)-
    Rückgabetyyp None
classmethod _mkpackage_mkdebiancontrolfile (build)
    Parameter build      (annize.features.distributables.debian.Package.
                          _BuildInfo)-
    Rückgabetyyp None
classmethod _mkpackage_mkexeclinks (build)
    Parameter build      (annize.features.distributables.debian.Package.
                          _BuildInfo)-
    Rückgabetyyp None
classmethod _mkpackage_mkmenuentries (build)
    Parameter build      (annize.features.distributables.debian.Package.
                          _BuildInfo)-
    Rückgabetyyp None
classmethod _mkpackage_mkprepostcmds (build)
    Parameter build      (annize.features.distributables.debian.Package.
                          _BuildInfo)-
    Rückgabetyyp None
classmethod _mkpackage_mkservices (build)
    Parameter build      (annize.features.distributables.debian.Package.
                          _BuildInfo)-
    Rückgabetyyp None
classmethod _mkpackage_prepareinfos (build, tmpdir)
    Parameter
        • build          (annize.features.distributables.debian.Package.
                          _BuildInfo)-
        • tmpdir (str)-
    Rückgabetyyp None

```

```

annize.features.distributables.debian.PerlSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

annize.features.distributables.debian.PhpSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

annize.features.distributables.debian.PythonSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

annize.features.distributables.debian.RubySection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

annize.features.distributables.debian.RustSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

annize.features.distributables.debian.ScienceSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

annize.features.distributables.debian.ScreenLockingCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

annize.features.distributables.debian.ScreenSavingCategory
    Alias von annize.features.distributables.debian._debian_category.<locals>.
    ACategory

class annize.features.distributables.debian.Section(*name)
    Bases: object

    property name

class annize.features.distributables.debian.ServiceDescription(name,      com-
                                                                mand)
    Bases: object

    Description for Debian services to be included in a package.

    Parameter

    • name (str) – The display name.

    • command (str) – The command to be executed.

annize.features.distributables.debian.ShellsSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

annize.features.distributables.debian.SoundSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

annize.features.distributables.debian.TasksSection
    Alias von annize.features.distributables.debian._debian_section.<locals>.
    ASection

```

```

annize.features.distributables.debian.TexSection
    Alias von annize.features.distributables.debian.__debian_section.<locals>.
    ASection

annize.features.distributables.debian.TextProcessingSection
    Alias von annize.features.distributables.debian.__debian_section.<locals>.
    ASection

annize.features.distributables.debian.UtilitiesSection
    Alias von annize.features.distributables.debian.__debian_section.<locals>.
    ASection

annize.features.distributables.debian.VersionControlSystemsSection
    Alias von annize.features.distributables.debian.__debian_section.<locals>.
    ASection

annize.features.distributables.debian.VideoSection
    Alias von annize.features.distributables.debian.__debian_section.<locals>.
    ASection

annize.features.distributables.debian.VirtualPackagesSection
    Alias von annize.features.distributables.debian.__debian_section.<locals>.
    ASection

annize.features.distributables.debian.WebServersSection
    Alias von annize.features.distributables.debian.__debian_section.<locals>.
    ASection

annize.features.distributables.debian.WebSoftwareSection
    Alias von annize.features.distributables.debian.__debian_section.<locals>.
    ASection

annize.features.distributables.debian.XWindowSystemSoftwareSection
    Alias von annize.features.distributables.debian.__debian_section.<locals>.
    ASection

annize.features.distributables.debian.XfceSection
    Alias von annize.features.distributables.debian.__debian_section.<locals>.
    ASection

annize.features.distributables.debian.ZopePloneFrameworkSection
    Alias von annize.features.distributables.debian.__debian_section.<locals>.
    ASection

annize.features.distributables.debian.__debian_category (debian_name,      free-
                                                         desktop_name)

```

Parameter

- **debian_name** (*str*) –
- **freedesktop_name** (*str*) –

Rückgabetype `Type[annize.features.distributables.debian.Category]`

```
annize.features.distributables.debian.__debian_section (name)
```

Parameter **name** (*str*) –

Rückgabetype `Type[annize.features.distributables.debian.Section]`

annize.features.distributables.flatpak module

```
class annize.features.distributables.flatpak.EnvironmentVariable(**b)
    Bases: object

class annize.features.distributables.flatpak.Filesystem(**b)
    Bases: object

class annize.features.distributables.flatpak.FlatpakImage(*,
                                                         source,
                                                         package_name,
                                                         sockets='x11',          fi-
                                                         lesystems='home',
                                                         shares='network')
```

Bases: *annize.fs.AdHocFsEntry*

Parameter

- **source** (*annize.fs.FsEntry*) –
- **package_name** (*str*) –
- **sockets** (*Iterable[str]*) –
- **filesystems** (*Iterable[str]*) –
- **shares** (*Iterable[str]*) –

```
class _BuildInfo(source: annize.fs.FsEntry, name: str, sdk: str, platform: str, kitversion:
                 Union[str, NoneType], command: Union[str, NoneType], sockets: Iter-
                 able[str], filesystems: Iterable[str], shares: Iterable[str], menu_entries:
                 Iterable[annize.features.distributables.flatpak.MenuEntry], environment:
                 Dict[str, str], pkgrootpath: str = None, pkgpath_share: str = None, pkg-
                 path_share_applications: str = None, pkgpath_share_icons: str = None, result:
                 annize.fs.FsEntry = None)
```

Bases: object

_BuildInfo(source: *annize.fs.FsEntry*, name: *str*, sdk: *str*, platform: *str*, kitversion: *Union[str, NoneType]*, command: *Union[str, NoneType]*, sockets: *Iterable[str]*, filesystems: *Iterable[str]*, shares: *Iterable[str]*, menu_entries: *Iterable[annize.features.distributables.flatpak.MenuEntry]*, environment: *Dict[str, str]*, pkgrootpath: *str* = *None*, pkgpath_share: *str* = *None*, pkgpath_share_applications: *str* = *None*, pkgpath_share_icons: *str* = *None*, result: *annize.fs.FsEntry* = *None*)

Parameter

- **source** (*annize.fs.FsEntry*) –
- **name** (*str*) –
- **sdk** (*str*) –
- **platform** (*str*) –
- **kitversion** (*Optional[str]*) –
- **command** (*Optional[str]*) –
- **sockets** (*Iterable[str]*) –
- **filesystems** (*Iterable[str]*) –
- **shares** (*Iterable[str]*) –
- **menu_entries** (*Iterable[annize.features.distributables.flatpak.MenuEntry]*) –

```

    • environment (Dict[str, str]) –
    • pkgrootpath (str) –
    • pkgpath_share (str) –
    • pkgpath_share_applications (str) –
    • pkgpath_share_icons (str) –
    • result (annize.fs.FsEntry) –

    Rückgabetyp None
    command: Optional[str]
    environment: Dict[str, str]
    filesystems: Iterable[str]
    kitversion: Optional[str]
    menu_entries: Iterable[annize.features.distributables.flatpak.MenuEntry]
    name: str
    pkgpath_share: str = None
    pkgpath_share_applications: str = None
    pkgpath_share_icons: str = None
    pkgrootpath: str = None
    platform: str
    result: annize.fs.FsEntry = None
    sdk: str
    shares: Iterable[str]
    sockets: Iterable[str]
    source: annize.fs.FsEntry

    _abc_impl = <_abc_data object>
    _initialize()

    classmethod _mkpackage (build)
        Parameter build (annize.features.distributables.flatpak.
            FlatpakImage._BuildInfo) –
        Rückgabetyp annize.fs.FsEntry

    classmethod _mkpackage_applysource (build)
        Parameter build (annize.features.distributables.flatpak.
            FlatpakImage._BuildInfo) –
        Rückgabetyp None

    classmethod _mkpackage_flatpak_build_export (build)
        Parameter build (annize.features.distributables.flatpak.
            FlatpakImage._BuildInfo) –
        Rückgabetyp None

```

```

classmethod _mkpackage_flatpak_build_finish (build)

    Parameter build (annize.features.distributables.flatpak.
        FlatpakImage._BuildInfo) –

    Rückgabetyp None

classmethod _mkpackage_flatpak_build_init (build)

    Parameter build (annize.features.distributables.flatpak.
        FlatpakImage._BuildInfo) –

    Rückgabetyp None

classmethod _mkpackage_prepareinfos (build, tmpdir)

    Parameter

        • build (annize.features.distributables.flatpak.FlatpakImage.
            _BuildInfo) –

        • tmpdir (str) –

    Rückgabetyp None

classmethod _mkpackage_share (build)

    Parameter build (annize.features.distributables.flatpak.
        FlatpakImage._BuildInfo) –

    Rückgabetyp None

class annize.features.distributables.flatpak.FlatpakrefFile (*, refname=None,
                                                                    package_name,
                                                                    title=None,
                                                                    branch='master',
                                                                    runti-
                                                                    me_repository_url=None,
                                                                    gpgkey=None, repo-
                                                                    sitory)

Bases: annize.fs.AdHocFsEntry

    Parameter

        • refname (str) –

        • package_name (str) –

        • title (str) –

        • branch (str) –

        • runtime_repository_url (str) –

        • gpgkey (bytes) –

        • repository (annize.features.distributables.flatpak.Repository)
            –

    _abc_impl = <_abc_data object>

    _initialize()

class annize.features.distributables.flatpak.GpgFile (*, refname=None,    gpg-
                                                                    key=None)

Bases: annize.fs.AdHocFsEntry

```

Parameter

- **refname** (*str*) –
- **gpgkey** (*bytes*) –

_abc_impl = <**_abc_data** object>

_initialize()

```
class annize.features.distributables.flatpak.Group(*, source, title, repository,
                                                    package_name, project_short_hint_name=None,
                                                    package_short_name=None)
```

Bases: *annize.features.distributables.common.Group*

Parameter

- **source** (*annize.fs.FsEntry*) –
- **title** (*str*) –
- **repository** (*annize.features.distributables.flatpak.Repository*) –
- **package_name** (*str*) –
- **project_short_hint_name** (*str*) –
- **package_short_name** (*str*) –

property description

get_files()

```
class annize.features.distributables.flatpak.LocalRepository(*, public_url,
                                                                friendly_name_suggestion=None,
                                                                upload_path=None,
                                                                upload_fsentry=None)
```

Bases: *annize.features.distributables.flatpak.Repository*

Parameter

- **public_url** (*str*) –
- **friendly_name_suggestion** (*str*) –
- **upload_path** (*str*) –
- **upload_fsentry** (*annize.fs.FsEntry*) –

upload (*source*)

Parameter **source** (*annize.fs.FsEntry*) –

```
class annize.features.distributables.flatpak.MenuEntry(*, name, title, category, command, gui, icon)
```

Bases: *object*

Parameter

- **name** (*str*) –
- **title** (*str*) –
- **category** (*Tuple*) –

- **command**(*str*) –
- **gui**(*bool*) –
- **icon**(*annize.fs.FsEntry*) –

class *annize.features.distributables.flatpak.Repository*(*, *public_url*, *friendly_name_suggestion=None*)

Bases: *object*

Parameter

- **public_url**(*str*) –
- **friendly_name_suggestion**(*str*) –

property *friendly_name_suggestion*

property *public_url*

upload(*source*)

Parameter *source* (*annize.fs.FsEntry*) –

class *annize.features.distributables.flatpak.Share*(***b*)

Bases: *object*

annize.features.distributables.pythonwheel module

class *annize.features.distributables.pythonwheel.ExecutableLink*(*, *linkname*, *modulename*, *methodname*, *is_gui*)

Bases: *object*

Parameter

- **linkname**(*str*) –
- **modulename**(*str*) –
- **methodname**(*str*) –
- **is_gui**(*bool*) –

property *is_gui*

property *linkname*

property *methodname*

property *modulename*

class *annize.features.distributables.pythonwheel.Package*(*, *source*, *executable_links*, *packagename=None*, *description=None*, *homepage_url=None*, *long_description=None*, *version=None*, *keywords=None*, *license=None*, *author=None*)

Bases: *annize.fs.AdHocFsEntry*

Parameter

- **source** (`annize.fs.FsEntry`) –
- **executable_links** (`List[annize.features.distributables.pythonwheel.ExecutableLink]`) –
- **packagename** (`Optional[str]`) –
- **description** (`Optional[str]`) –
- **homepage_url** (`Optional[str]`) –
- **long_description** (`Optional[str]`) –
- **version** (`annize.data.version.Version`) –
- **keywords** (`annize.features.base.Keywords`) –
- **license** (`annize.features.licensing.License`) –
- **author** (`annize.features.authors.Author`) –

```
class _BuildInfo (source: annize.fs.FsEntry, description: str, long_description:
                    str, keywords: annize.features.base.Keywords, name: str, versi-
                    on: annize.data.version.Version, homepage: str, author: anni-
                    ze.features.authors.Author, license: annize.features.licensing.License, exe-
                    cutable_links: List[annize.features.distributables.pythonwheel.ExecutableLink],
                    pkgrootpath: str = None, pkgpath_setuppy: str = None, setuppy_conf: Dict[str,
                    Union[Any, NoneType]] = None, result: annize.fs.FsEntry = None)
```

Bases: `object`

`_BuildInfo(source: annize.fs.FsEntry, description: str, long_description: str, keywords: annize.features.base.Keywords, name: str, version: annize.data.version.Version, homepage: str, author: annize.features.authors.Author, license: annize.features.licensing.License, executable_links: List[annize.features.distributables.pythonwheel.ExecutableLink], pkgrootpath: str = None, pkgpath_setuppy: str = None, setuppy_conf: Dict[str, Union[Any, NoneType]] = None, result: annize.fs.FsEntry = None)`

Parameter

- **source** (`annize.fs.FsEntry`) –
- **description** (`str`) –
- **long_description** (`str`) –
- **keywords** (`annize.features.base.Keywords`) –
- **name** (`str`) –
- **version** (`annize.data.version.Version`) –
- **homepage** (`str`) –
- **author** (`annize.features.authors.Author`) –
- **license** (`annize.features.licensing.License`) –
- **executable_links** (`List[annize.features.distributables.pythonwheel.ExecutableLink]`) –
- **pkgrootpath** (`str`) –
- **pkgpath_setuppy** (`str`) –
- **setuppy_conf** (`Dict[str, Optional[Any]]`) –
- **result** (`annize.fs.FsEntry`) –

```

    Rückgabetyp None
author: annize.features.authors.Author
description: str
executable_links: List[annize.features.distributables.pythonwheel.ExecutableLink]
homepage: str
keywords: annize.features.base.Keywords
license: annize.features.licensing.License
long_description: str
name: str
pkgpath_setuppy: str = None
pkgrootpath: str = None
result: annize.fs.FsEntry = None
setuppy_conf: Dict[str, Optional[Any]] = None
source: annize.fs.FsEntry
version: annize.data.version.Version

_abc_impl = <_abc_data object>
_initialize()

classmethod _mkpackage (build)
    Parameter build      (annize.features.distributables.pythonwheel.
                          Package._BuildInfo) –
    Rückgabetyp annize.fs.FsEntry

classmethod _mkpackage_bdist_wheel (build)
    Parameter build      (annize.features.distributables.pythonwheel.
                          Package._BuildInfo) –
    Rückgabetyp None

classmethod _mkpackage_mkexeclinks (build)
    Parameter build      (annize.features.distributables.pythonwheel.
                          Package._BuildInfo) –
    Rückgabetyp None

classmethod _mkpackage_mkmanifestin (build)
    Parameter build      (annize.features.distributables.pythonwheel.
                          Package._BuildInfo) –
    Rückgabetyp None

classmethod _mkpackage_mksetuppyconf (build)
    Parameter build      (annize.features.distributables.pythonwheel.
                          Package._BuildInfo) –
    Rückgabetyp None

classmethod _mkpackage_prepareinfos (build, tmpdir)

```

Parameter

- **build** (`annize.features.distributables.pythonwheel.Package._BuildInfo`) –
- **tmpdir** (`str`) –

Rückgabetyyp `None`

classmethod `_mkpackage_setuppyconf_classifiers` (*build*)

Parameter **build** (`annize.features.distributables.pythonwheel.Package._BuildInfo`) –

Rückgabetyyp `None`

classmethod `_mkpackage_setuppyconf_install_requires` (*build*)

Parameter **build** (`annize.features.distributables.pythonwheel.Package._BuildInfo`) –

Rückgabetyyp `None`

classmethod `_mkpackage_setuppyconf_prepare` (*build*)

Parameter **build** (`annize.features.distributables.pythonwheel.Package._BuildInfo`) –

Rückgabetyyp `None`

`annize.features.distributables.tar` module

```
class annize.features.distributables.tar.Package(*, packagename=None, packa-
                                         genamepostfix=None,      source,
                                         version=None,      license=None,
                                         documentation=None)
```

Bases: `annize.fs.AdHocFsEntry`

Parameter

- **packagename** (`Optional[str]`) –
- **packagenamepostfix** (`Optional[str]`) –
- **source** (`annize.fs.FsEntry`) –
- **version** (`annize.data.version.Version`) –
- **license** (`annize.features.licensing.License`) –
- **documentation** (`annize.fs.FsEntry`) –

`_abc_impl` = `<_abc_data object>`

`_initialize` ()

annize.features.documentation namespace

Subpackages

annize.features.documentation.sphinx namespace

Subpackages

annize.features.documentation.sphinx.output namespace

Submodules

annize.features.documentation.sphinx.output.common module

class annize.features.documentation.sphinx.output.common.**OutputGenerator** (*outputspec*)
Bases: abc.ABC

Base class for documentation output specifications. See `render()`.

_abc_impl = <_abc_data object>

abstract formatname()
Returns the Sphinx format name.

Rückgabetyyp str

abstract classmethod is_compatible_for (*outputspec*)

Parameter **outputspec** (annize.features.documentation.common.
OutputSpec) –

Rückgabetyyp bool

multilanguage_frame (*document*)

Parameter **document** (*documentationsphinx.Document*) –

Rückgabetyyp *annize.features.documentation.common._DocumentGenerateAllLanguagesResult*

property outputspec

postproc (*preresult*)

Parameter **preresult** (*annize.fs.FsEntry*) –

Rückgabetyyp *annize.fs.FsEntry*

prepare_generate (*geninfo*)

Parameter **geninfo** (*documentationsphinx.Document.GenerateInfo*) –

Rückgabetyyp None

annize.features.documentation.sphinx.output.common.**find_output_generator_for_outputspec** (*outputspec*)

annize.features.documentation.sphinx.output.common.**register_output_generator** (*outputgenerator*)

annize.features.documentation.sphinx.output.html module

```

class annize.features.documentation.sphinx.output.html.HtmlOutputGenerator(outputspec)
    Bases: annize.features.documentation.sphinx.output.common.OutputGenerator
    HTML documentation output.
    _abc_impl = <_abc_data object>
    formatname()
        Returns the Sphinx format name.
    classmethod is_compatible_for(outputspec)
    multilanguage_frame(document)
    prepare_generate(geninfo)
class annize.features.documentation.sphinx.output.html.HtmlOutputSpec(*,
    is_homepage=False,
    title=None,
    short_title=None,
    short_desc=None,
    masterlink=None,
    logo_image=None,
    background_image=None)
    Bases: annize.features.documentation.common.HtmlOutputSpec
    Parameter
    • path – The output path.
    • theme (Optional[str]) – The sphinx html theme name.
    • homepage_mode – If to render output for a homepage with slight different stylings and behaviors.
    • title (Optional[str]) – The html title.
    • short_title (Optional[str]) – The short html title.
    • short_desc (Optional[str]) – The short description. Ignored by most themes.
    • masterlink (Optional[str]) – Url that overrides the target of the main heading (which is also a link).
    • is_homepage (bool) –
    • logo_image (Optional[annize.fs.FsEntry]) –
    • background_image (Optional[annize.fs.FsEntry]) –
    _abc_impl = <_abc_data object>
    property background_image
    property logo_image

```

```

property masterlink
property short_desc
property short_title
property theme

```

annize.features.documentation.sphinx.output.pdf module

```

class annize.features.documentation.sphinx.output.pdf.PdfOutputGenerator (outputspec)
    Bases: annize.features.documentation.sphinx.output.common.OutputGenerator
    PDF documentation output.
    _abc_impl = <_abc_data object>
    formatname ()
        Returns the Sphinx format name.
    classmethod is_compatible_for (outputspec)
    postproc (preresult)
    resultpath (projectname, buildpath)

```

annize.features.documentation.sphinx.output.plaintext module

```

class annize.features.documentation.sphinx.output.plaintext.PlaintextOutputGenerator (outputspec)
    Bases: annize.features.documentation.sphinx.output.common.OutputGenerator
    Plaintext documentation output.
    _abc_impl = <_abc_data object>
    formatname ()
        Returns the Sphinx format name.
    classmethod is_compatible_for (outputspec)

```

Submodules

annize.features.documentation.sphinx._utils module

```

annize.features.documentation.sphinx._utils.serialize_for_confpy (obj)
    Parameter obj (Optional [Any]) –
    Rückgabotyp str

```

annize.features.documentation.sphinx.common module

```
class annize.features.documentation.sphinx.common.AboutProjectDocument (*,
                                                                    pro-
                                                                    ject-
                                                                    na-
                                                                    me=None,
                                                                    aut-
                                                                    hors=None,
                                                                    ver-
                                                                    si-
                                                                    on=None,
                                                                    re-
                                                                    lea-
                                                                    se=None,
                                                                    de-
                                                                    pen-
                                                                    den-
                                                                    cies,
                                                                    pro-
                                                                    ject_version=None)
```

Bases: *annize.features.documentation.sphinx.common.Document*

Parameter

- **projectname** (*Optional[str]*) –
- **authors** (*Optional[str]*) –
- **version** (*Optional[str]*) –
- **release** (*Optional[str]*) –
- **dependencies** (*List[annize.features.dependencies.common.Dependency]*) –
- **project_version** (*annize.data.version.Version*) –

_abc_impl = *<_abc_data object>*

_generate_master_doc (*geninfo*)

available_languages ()

Rückgabety *List[str]*

```
class annize.features.documentation.sphinx.common.ApiReferenceDocument (*,
                                                                    lan-
                                                                    gua-
                                                                    ge,
                                                                    hea-
                                                                    ding=None,
                                                                    sour-
                                                                    ce=None)
```

Bases: *annize.features.documentation.sphinx.common.Document*

An api reference.

Parameter

- **language** (*annize.features.documentation.sphinx.common.ApiReferenceLanguage*) –

```

    • heading (Optional[str]) –
    • source (annize.fs.FsEntry) –
__get_refgeninfo (geninfo)
    Parameter geninfo      (annize.features.documentation.sphinx.common.
                             Document.GenerateInfo) –
__abc_impl = <_abc_data object>
__generate_master_doc (geninfo)
__prepare_generate (geninfo)
available_languages ()
    Rückgabetyt List[str]
class annize.features.documentation.sphinx.common.ApiReferenceLanguage
    Bases: abc.ABC
    Base class for a programming language in api references. See ApiReferencePiece.
class ApiReferenceGenerateInfo (geninfo, source, heading)
    Bases: annize.features.documentation.sphinx.common.Document.GenerateInfo
    Parameter
        • geninfo      (annize.features.documentation.sphinx.common.
                         Document.GenerateInfo) –
        • source (annize.fs.FsEntry) –
        • heading (str) –
    property heading
    property source
__abc_impl = <_abc_data object>
abstract generate_master_doc (rgeninfo)
    Parameter rgeninfo      (annize.features.documentation.sphinx.common.
                             ApiReferenceLanguage.ApiReferenceGenerateInfo) –
    Rückgabetyt str
abstract prepare_generate (rgeninfo)
    Parameter rgeninfo      (annize.features.documentation.sphinx.common.
                             ApiReferenceLanguage.ApiReferenceGenerateInfo) –
    Rückgabetyt None

```



```
class annize.features.documentation.sphinx.common.ArgparseCommandLineInterfaceDocument (*,
```

par-
ser-
fac-
to-
ry,
pro-
gram
me,
hea-
ding-
sour-
ce=N

Bases: *annize.features.documentation.sphinx.common.Document*

Parameter

- **parserfactory** (*str*) –
- **programname** (*str*) –
- **heading** (*Optional[str]*) –
- **source** (*annize.fs.FsEntry*) –

_abc_impl = *<_abc_data object>*

_generate_master_doc (*geninfo*)

_prepare_generate (*geninfo*)

available_languages ()

Rückgabety *List[str]*

```
class annize.features.documentation.sphinx.common.CompositeDocument (*, projectna-  
me=None,  
aut-  
hors=None,  
versi-  
on=None,  
relea-  
se=None,  
docu-  
ments)
```

Bases: *annize.features.documentation.sphinx.common.Document*

Parameter

- **projectname** (*Optional[str]*) –
- **authors** (*Optional[str]*) –
- **version** (*Optional[str]*) –
- **release** (*Optional[str]*) –
- **documents** (*Iterable[annize.features.documentation.sphinx.common.Document]*) –

__get_inner_generateinfo (*geninfo, innerdoc*)

Parameter

```

    • geninfo (annize.features.documentation.sphinx.common.
      Document.GenerateInfo)–
    • innerdoc (annize.features.documentation.sphinx.common.
      Document)–
  _abc_impl = <_abc_data object>
  _generate_master_doc (geninfo)
  _prepare_generate (geninfo)
class annize.features.documentation.sphinx.common.Document (*, projectname=None,
      authors=None,
      version=None, relea-
      se=None)

Bases: annize.features.documentation.common.Document

  Parameter

    • projectname (Optional[str])–
    • authors (Optional[str])–
    • version (Optional[str])–
    • release (Optional[str])–

class GenerateInfo (intstruct, outdir, confdir, language, *, configvalues=None, configlines=None,
      entry_path="")
  Bases: object

  Parameter

    • intstruct (annize.fs.FsEntry)–
    • outdir (annize.fs.FsEntry)–
    • confdir (annize.fs.FsEntry)–
    • language (str)–
    • configvalues (Optional[Dict])–
    • configlines (Optional[List])–
    • entry_path (str)–

  _to_dict ()
  property confdir
  property configlines
  property configvalues
  property entry_path
  property intstruct
  property language
  property outdir
  _abc_impl = <_abc_data object>
  property _authors
  abstract _generate_master_doc (geninfo)

```

```

    Parameter geninfo      (annize.features.documentation.sphinx.common.
                             Document.GenerateInfo)–
    Rückgabotyp str
    _prepare_generate (geninfo)
    Parameter geninfo      (annize.features.documentation.sphinx.common.
                             Document.GenerateInfo)–
    Rückgabotyp None
    property _projectname
    property _release
    property _version
    generate (outputspec, *, language='?')
    generate_all_languages (outputspec)
class annize.features.documentation.sphinx.common.ReadmeDocument (*, projectna-
                                                                    me=None,
                                                                    aut-
                                                                    hors=None,
                                                                    versi-
                                                                    on=None, re-
                                                                    lease=None,
                                                                    documents,
                                                                    title=None,
                                                                    depen-
                                                                    dencies, pro-
                                                                    ject_version=None)
Bases: annize.features.documentation.sphinx.common.CompositeDocument
A reStructuredText formatted file or a directory of such files.

```

Parameter

- **projectname** (Optional[str])–
- **authors** (Optional[str])–
- **version** (Optional[str])–
- **release** (Optional[str])–
- **documents** (Iterable[annize.features.documentation.sphinx.common.Document])–
- **title** (Optional[str])–
- **dependencies** (List[annize.features.dependencies.common.Dependency])–
- **project_version** (annize.data.version.Version)–

```

_abc_impl = <_abc_data object>
available_languages ()
    Rückgabotyp List[str]
    property title

```

```
class annize.features.documentation.sphinx.common.RstDocument (*, variants,
                                                                **kwargs)
```

Bases: *annize.features.documentation.sphinx.common.Document*

A reStructuredText formatted file or a directory of such files.

Parameter variants (*List [annize.features.documentation.sphinx.common.RstDocumentVariant]*) –

__get_variant (*language*)

Parameter language (*str*) –

Rückgabetyp Optional[*annize.features.documentation.sphinx.common.RstDocumentVariant*]

_abc_impl = *<_abc_data object>*

_generate_master_doc (*geninfo*)

available_languages ()

```
class annize.features.documentation.sphinx.common.RstDocumentVariant (*, lan-
                                                                           gua-
                                                                           ge='?',
                                                                           sour-
                                                                           ce)
```

Bases: object

Parameter

- **language** (*str*) –
- **source** (*annize.fs.FsEntry*) –

property language

property source

annize.features.documentation.sphinx.cpp module

```
class annize.features.documentation.sphinx.cpp.CppApiReferenceLanguage (**kwargs)
Bases: annize.features.documentation.sphinx.doxygencompat.DxygenSupportedApiReferenceLanguage
```

C++ language support for api references.

_abc_impl = *<_abc_data object>*

annize.features.documentation.sphinx.doxygencompat module

class annize.features.documentation.sphinx.doxygencompat.DoxygenSupportedApiReferenceLanguage

Bases: *annize.features.documentation.sphinx.common.ApiReferenceLanguage*

Language support for api references powered by Doxygen.

Parameter

- **_doxygenopts** (*Optional[Dict[str, str]]*) –
- **extract_all** (*bool*) –
- **extract_private** (*bool*) –
- **extract_package** (*bool*) –
- **extract_static** (*bool*) –
- **extract_local_classes** (*bool*) –

- **extract_local_methods** (*bool*) –
- **extract_anon_nspaces** (*bool*) –
- **extract_priv_virtual** (*bool*) –
- **file_patterns** (*str*) –
- **inline_inherited_memb** (*bool*) –
- **inherit_docs** (*bool*) –
- **hide_undoc_members** (*bool*) –
- **hide_undoc_classes** (*bool*) –
- **exclude_patterns** (*str*) –
- **predefined** (*Optional[List[str]]*) –

__info (*outpath*)

_abc_impl = <_abc_data object>

_generate_master_doc (*srcpath, intstructpath, confdirpath, heading*)

prepare_generate (*rootpath, outpath*)

annize.features.documentation.sphinx.javascript module

class annize.features.documentation.sphinx.javascript.**JavaScriptApiReferenceLanguage**

Bases: *annize.features.documentation.sphinx.common.ApiReferenceLanguage*

JavaScript language support for api references.

__jsfiles (*dirf*)

Parameter **dirf** (*str*) –

Rückgabetyyp *List[str]*

__scanjsfile (*f*)

Parameter **f** (*str*) –

Rückgabetyyp *str*

_abc_impl = <_abc_data object>

generate_master_doc (*rgeninfo*)

prepare_generate (*rgeninfo*)

annize.features.documentation.sphinx.python module

class annize.features.documentation.sphinx.python.**Python3ApiReferenceLanguage** (*,

show_undoc_mem
show_protected_m
ex-
clu-
de=None)

Bases: *annize.features.documentation.sphinx.common.ApiReferenceLanguage*

Python 3 language support for api references.

Parameter

- **show_undoc_members** (*bool*) –
- **show_protected_members** (*bool*) –
- **exclude** (*Optional[List[str]]*) –

__patch_property_types_in_docstrings (*pdir*)

Parameter **pdir** (*str*) –

Rückgabotyp *None*

_abc_impl = *<_abc_data object>*

generate_master_doc (*rgeninfo*)

prepare_generate (*rgeninfo*)

annize.features.documentation.sphinx.rst module

class annize.features.documentation.sphinx.rst.**RstGenerator**

Bases: *object*

Generator for some documentation source parts.

static heading (*text*, *, *variant=''*, *sub=False*, *anchor=None*)

Generates documentation source for a section heading.

Parameter

- **text** (*str*) –
- **variant** (*str*) –
- **sub** (*bool*) –
- **anchor** (*Optional[str]*) –

Rückgabotyp *str*

Submodules

annize.features.documentation.common module

class annize.features.documentation.common.**Document**

Bases: *abc.ABC*

_abc_impl = *<_abc_data object>*

abstract available_languages ()

Rückgabotyp *List[str]*

abstract generate (*outputspec*, *, *language='?'*)

Parameter **language** (*str*) –

Rückgabotyp *annize.features.documentation.common._DocumentGenerateResult*

abstract generate_all_languages (*outputspec*)

Rückgabotyp *annize.features.documentation.common._DocumentGenerateAllLanguagesResult*

```
class annize.features.documentation.common.GeneratedDocument (*,      document,
                                                                outputspec,  lan-
                                                                guage=None,
                                                                filename=None)
```

Bases: *annize.fs.AdHocFsEntry*

Parameter

- **document** (*annize.features.documentation.common.Document*) –
- **outputspec** (*annize.features.documentation.common.OutputSpec*) –
- **language** (*str*) –
- **filename** (*str*) –

```
_abc_impl = <_abc_data object>
```

```
_initialize()
```

```
class annize.features.documentation.common.HtmlOutputSpec (*, is_homepage=False,
                                                                title=None)
```

Bases: *annize.features.documentation.common.OutputSpec*

HTML documentation output.

Parameter

- **is_homepage** (*bool*) – If to render output for a homepage with slight different stylings and behavior.
- **title** (*Optional[str]*) – The html title. TODO higher level thing here?!

```
_abc_impl = <_abc_data object>
```

```
property is_homepage
```

```
property title
```

```
class annize.features.documentation.common.OutputSpec
```

Bases: *abc.ABC*

Base class for documentation output specifications. See `render()`.

```
_abc_impl = <_abc_data object>
```

```
class annize.features.documentation.common.PdfOutputSpec
```

Bases: *annize.features.documentation.common.OutputSpec*

PDF documentation output.

```
_abc_impl = <_abc_data object>
```

```
class annize.features.documentation.common.PlaintextOutputSpec
```

Bases: *annize.features.documentation.common.OutputSpec*

Plaintext documentation output.

```
_abc_impl = <_abc_data object>
```

```
class annize.features.documentation.common._DocumentGenerateAllLanguagesResult (file,
                                                                                      ent-
                                                                                      ry_path,
                                                                                      ent-
                                                                                      ry_paths_for_lan
```

Bases: *annize.features.documentation.common._DocumentGenerateResult*

Parameter


```

    • file(annize.fs.FsEntry) –
    • entry_path(str) –
    • entry_paths_for_languages(Dict[str, str]) –
entry_path_for_language(language)
    Parameter language(str) –
    Rückgabotyp str
property languages
class annize.features.documentation.common._DocumentGenerateResult(file, entry_path)
    Bases: object
    Parameter
        • file(annize.fs.FsEntry) –
        • entry_path(str) –
    _set_entry_path(entry_path)
    Parameter entry_path(str) –
    Rückgabotyp None
property entry_path
property file

```

annize.features.files namespace

Subpackages

annize.features.files.transfer namespace

Submodules

annize.features.files.transfer.common module

```

class annize.features.files.transfer.common.Endpoint
    Bases: object
    access_filesystem(rootpath)
    Parameter rootpath(str) –
    Rückgabotyp AbstractContextManager[annize.fs.FsEntry]
class annize.features.files.transfer.common.FsEndpoint(*, fsentry=None, path=None)
    Bases: annize.features.files.transfer.common.Endpoint
    Parameter
        • fsentry(annize.fs.FsEntry) –
        • path(str) –

```

access_filesystem (*rootpath*)

Parameter **rootpath** (*str*) –

Rückgabetyyp AbstractContextManager[*annize.fs.FsEntry*]

class *annize.features.files.transfer.common.Upload* (*, *source*, *destination_endpoint*,
destination_path)

Bases: object

Parameter

- **source** (*annize.fs.FsEntry*) –
- **destination_endpoint** (*annize.features.files.transfer.common.Endpoint*) –
- **destination_path** (*str*) –

annize.features.files.transfer.ssh module

class *annize.features.files.transfer.ssh.Endpoint* (*, *host*, *port*=22, *user-*
name, *identity_file*=None,
has_shell_access=False)

Bases: *annize.features.files.transfer.common.Endpoint*

Parameter

- **host** (*str*) –
- **port** (*int*) –
- **username** (*str*) –
- **identity_file** (*Optional[str]*) –
- **has_shell_access** (*bool*) –

access_filesystem (*rootpath*)

exec (*cmdline*)

Parameter **cmdline** (*str*) –

Rückgabetyyp *TODO*

property **has_shell_access**

property **host**

property **identity_file**

locationstring (*path*=None)

Parameter **path** (*Optional[str]*) –

Rückgabetyyp *str*

property **port**

property **username**

Submodules

annize.features.files.common module

```
class annize.features.files.common.Dir(*, excludes, parts, root=None, path=None, name=None)
    Bases: annize.features.files.common.FsEntry
        Parameter
            • excludes (Iterable[annize.features.files.common.Exclude]) –
            • parts (Iterable[annize.features.files.common.DirPart]) –
            • root (annize.fs.FsEntry) –
            • path (str) –
            • name (str) –

        classmethod __transfer_predicate_and(*predicates)
            Parameter predicates (Callable[[str, FsEntry, FsEntry], bool]) –
            Rückgabotyp Callable[[str, annize.features.files.common.FsEntry, annize.features.files.common.FsEntry], bool]

        classmethod __transfer_predicate_for_exclude(exclude)
            Parameter exclude (annize.features.files.common.Exclude) –
            Rückgabotyp Callable[[str, annize.features.files.common.FsEntry, annize.features.files.common.FsEntry], bool]

        classmethod __transfer_predicate_for_excludes(excludes)
            Parameter excludes (Iterable[annize.features.files.common.Exclude]) –
            Rückgabotyp Callable[[str, annize.features.files.common.FsEntry, annize.features.files.common.FsEntry], bool]

        _abc_impl = <_abc_data object>
        _initialize()

class annize.features.files.common.DirPart(*, excludes=None, root=None, source_path=None, destination_path=None, path=None, destination_is_parent=False)
    Bases: object
        Parameter
            • excludes (Iterable[annize.features.files.common.Exclude]) –
            • root (annize.fs.FsEntry) –
            • source_path (str) –
            • destination_path (str) –
            • path (str) –
            • destination_is_parent (bool) –
        property destination_is_parent
```

```

property destination_path
property excludes
property path
property root
class annize.features.files.common.Exclude(*, by_path_pattern=None, by_path=None,
                                             by_name_pattern=None, by_name=None)
    Bases: object
        Parameter
            • by_path_pattern(str) –
            • by_path(str) –
            • by_name_pattern(str) –
            • by_name(str) –
        __does_exclude(by_text, by_pattern)
            Parameter
                • text(str) –
                • by_text(str) –
                • by_pattern(re.Pattern) –
            Rückgabetyyp bool
        does_exclude(relative_path, source, destination)
            Parameter
                • relative_path(str) –
                • source(annize.fs.FsEntry) –
                • destination(annize.fs.FsEntry) –
            Rückgabetyyp bool
class annize.features.files.common.File(*, root=None, path=None)
    Bases: annize.features.files.common.FsEntry
        Parameter
            • root(annize.fs.FsEntry) –
            • path(str) –
        _abc_impl = <_abc_data object>
class annize.features.files.common.FsEntry(*, root=None, path=None)
    Bases: annize.fs.AdHocFsEntry
        Parameter
            • root(annize.fs.FsEntry) –
            • path(str) –
        _abc_impl = <_abc_data object>
        _initialize()
        property relative_path

```

property root

```
class annize.features.files.common.MachineRootDirectory
    Bases: annize.fs.AdHocFsEntry
    _abc_impl = <_abc_data object>
    _initialize()

class annize.features.files.common.ProjectDir
    Bases: annize.fs.AdHocFsEntry
    _abc_impl = <_abc_data object>
    _initialize()
```

annize.features.homepage namespace

Subpackages

annize.features.homepage.sections namespace

Submodules

annize.features.homepage.sections.about module

```
class annize.features.homepage.sections.about.Section(*,
    head=<annize.i18n.ProvidedTrStr
    object>, sortindex=10000)
    Bases: annize.features.homepage.common.HomepageSection
    _generate(info)
```

annize.features.homepage.sections.changelog module

```
class annize.features.homepage.sections.changelog.Section(*, changelog=None,
    head=<annize.i18n.ProvidedTrStr
    object>, sortindex=60000)
    Bases: annize.features.homepage.common.HomepageSection
    Parameter changelog (annize.features.changelog.common.Changelog) –
    _generate(info)
```

annize.features.homepage.sections.documentation module

```
class annize.features.homepage.sections.documentation.Section(*, documentation,
    head=<annize.i18n.ProvidedTrStr
    object>, sortindex=30000)
    Bases: annize.features.homepage.common.HomepageSection
    Parameter documentation (List[annize.features.documentation.common.
    Document]) –
```

```
_generate (info)
preproc_generate (info)
```

annize.features.homepage.sections.download module

```
class annize.features.homepage.sections.download.Section (*,      distributables,
                                                             dependencies,
                                                             head=<annize.i18n.ProvidedTrStr
                                                             object>,      sortin-
                                                             dex=40000)
```

Bases: *annize.features.homepage.common.HomepageSection*

Parameter

- **distributables** (*List [annize.features.distributables.common.Group]*) –
- **dependencies** (*List [annize.features.dependencies.common.Dependency]*) –

```
__generate_packagelist (info)
```

```
    Parameter info      (annize.features.homepage.common.HomepageSection.
                          _GenerateInfo) –
```

```
_generate (info)
```

```
postproc_generate (info)
```

```
preproc_generate (info)
```

```
annize.features.homepage.sections.download.filehash (filepath)
```

```
    Parameter filepath (str) –
```

```
    Rückgabotyp str
```

```
annize.features.homepage.sections.download.friendly_filesize (isize)
```

```
    Parameter isize (int) –
```

```
    Rückgabotyp str
```

annize.features.homepage.sections.gallery module

```
class annize.features.homepage.sections.gallery.Section (*,
                                                             head=<annize.i18n.ProvidedTrStr
                                                             object>, sortindex=50000,
                                                             mediagalleries)
```

Bases: *annize.features.homepage.common.HomepageSection*

```
    Parameter mediagalleries      (List [annize.features.mediagalleries.
                                   Gallery]) –
```

```
_generate (info)
```

```
preproc_generate (info)
```

annize.features.homepage.sections.imprint module

```
class annize.features.homepage.sections.imprint.Section(*, imprint,
                                                         head=<annize.i18n.ProvidedTrStr
                                                         object>,      sortin-
                                                         dex=70000)

Bases: annize.features.homepage.common.HomepageSection

    Parameter imprint (str) –
    _generate (info)
```

annize.features.homepage.sections.license module

```
class annize.features.homepage.sections.license.Section(*,
                                                         head=<annize.i18n.ProvidedTrStr
                                                         object>,      sortin-
                                                         dex=20000)

Bases: annize.features.homepage.common.HomepageSection

    _generate (info)
```

Submodules

annize.features.homepage.common module

```
class annize.features.homepage.common.GeneratedHomepage(*, homepage)
    Bases: annize.fs.AdHocFsEntry
        Parameter homepage (annize.features.homepage.common.Homepage) –
        _abc_impl = <_abc_data object>
        _initialize()

class annize.features.homepage.common.Homepage(*, title=None, short_desc=None, secti-
                                                         ons, languages='')
    Bases: object
        Parameter
            • title (Optional[str]) –
            • short_desc (Optional[str]) –
            • sections (List[annize.features.homepage.common.
                          HomepageSection]) –
            • languages (str) –
        __generate_postpreproc(*, is_pre, document_root_directory, document_root_url)
        Parameter
            • customargs (Dict) –
            • is_pre (bool) –
            • document_root_directory (annize.fs.FsEntry) –
            • document_root_url (str) –
```

```
__generate_section (*, customargs, culture, document_root_directory, document_root_url, document_variant_directory)
```

Parameter

- **customargs** (*Dict*) –
- **culture** (*annize.i18n.Culture*) –
- **document_root_directory** (*annize.fs.FsEntry*) –
- **document_root_url** (*str*) –
- **document_variant_directory** (*annize.fs.FsEntry*) –

```
_append_section (section)
```

Parameter **section** (*annize.features.homepage.common.HomepageSection*) –

Rückgabety *None*

```
generate ()
```

```
property languages
```

```
property sections
```

```
property short_desc
```

```
property title
```

```
class annize.features.homepage.common.HomepageSection (*, head, sortindex=0)
```

Bases: *object*

Parameter

- **head** (*str*) –
- **sortindex** (*int*) –

```
class Product (*, rst_text="", media_files=())
```

Bases: *object*

Parameter

- **rst_text** (*str*) –
- **media_files** (*List [annize.fs.FsEntry]*) –

```
append_rst (rst_text)
```

Rückgabety *None*

```
attach_media_file (file)
```

Parameter **file** (*annize.fs.FsEntry*) –

Rückgabety *None*

```
property media_files
```

```
property rst_text
```

```
class _GenerateInfo (culture: annize.i18n.Culture, customarg: Union [object, NoneType], document_root_url: str, document_variant_directory: annize.fs.FsEntry, document_variant_url: str)
```

Bases: *object*

```
_GenerateInfo (culture: annize.i18n.Culture, customarg: Union [object, NoneType], document_root_url: str, document_variant_directory: annize.fs.FsEntry, document_variant_url: str)
```


Parameter

- **culture** (*annize.i18n.Culture*) –
- **customarg** (*Optional[object]*) –
- **document_root_url** (*str*) –
- **document_variant_directory** (*annize.fs.FsEntry*) –
- **document_variant_url** (*str*) –

Rückgabetyyp *None*

culture: *annize.i18n.Culture*

customarg: *Optional[object]*

document_root_url: *str*

document_variant_directory: *annize.fs.FsEntry*

document_variant_url: *str*

```
class _PrePostprocGenerateInfo (document_root_directory: annize.fs.FsEntry, document_root_url: str, customarg: Union[object, NoneType])
```

Bases: *object*

```
_PrePostprocGenerateInfo(document_root_directory: annize.fs.FsEntry, document_root_url: str, customarg: Union[object, NoneType])
```

Parameter

- **document_root_directory** (*annize.fs.FsEntry*) –
- **document_root_url** (*str*) –
- **customarg** (*Optional[object]*) –

Rückgabetyyp *None*

customarg: *Optional[object]*

document_root_directory: *annize.fs.FsEntry*

document_root_url: *str*

_generate (*info*)

Parameter info (*annize.features.homepage.common.HomepageSection._GenerateInfo*) –

Rückgabetyyp *Union[annize.features.homepage.common.HomepageSection.Product, str]*

generate (*info*)

Parameter info (*annize.features.homepage.common.HomepageSection._GenerateInfo*) –

Rückgabetyyp *Optional[annize.features.homepage.common.HomepageSection.Product]*

property head

postproc_generate (*info*)

Parameter info (*annize.features.homepage.common.HomepageSection._PrePostprocGenerateInfo*) –

Rückgabetyyp *None*

```

preproc_generate (info)
    Parameter info      (annize.features.homepage.common.HomepageSection.
                          _PrePostprocGenerateInfo) –
    Rückgabetyp None

property sortindex

class annize.features.homepage.common.PinoHomepageAuxInfo
    Bases: annize.features.homepage.common.HomepageSection
    _generate (info)
    preproc_generate (info)

class annize.features.homepage.common.SimpleProjectHomepage (*, changelog=None,
                                                                dependencies,
                                                                distributables,
                                                                documentation,
                                                                imprint=None,
                                                                mediagalleries,
                                                                **kwargs)

    Bases: annize.features.homepage.common.Homepage
    Parameter
        • changelog (annize.features.changelog.common.Changelog) –
        • dependencies      (List [annize.features.dependencies.common.
                              Dependency]) –
        • distributables    (List [annize.features.distributables.common.
                              Group]) –
        • documentation    (List [annize.features.documentation.common.
                              Document]) –
        • imprint (str) –
        • mediagalleries (List [annize.features.mediagalleries.Gallery]) –

```

annize.features.i18n namespace

Submodules

annize.features.i18n.common module

```

class annize.features.i18n.common.ProjectDefinedTranslationProvider
    Bases: annize.i18n.TranslationProvider
    __translations_for_stringname (stringname)
        Parameter stringname (str) –
        Rückgabetyp Dict[str, str]
    add_translations (stringname, variants)
        Parameter
            • stringname (str) –

```

- **variants** (*Dict[str, str]*) –

Rückgabetyyp None

translate (*stringname*, *, *culture*)

class `annize.features.i18n.common.String` (*, *stringname*=None, *stringtr*=None, ***variants*)

Bases: `annize.i18n.ProvidedTrStr`

Parameter

- **stringname** (*Optional[str]*) –
- **stringtr** (*Optional[str]*) –
- **variants** (*str*) –

annize.features.i18n.gettext module

class `annize.features.i18n.gettext.GenerateMOs` (*, *po_directory*, *mo_directory*)

Bases: `object`

Parameter

- **po_directory** (`annize.fs.FsEntry`) –
- **mo_directory** (`annize.fs.FsEntry`) –

class `annize.features.i18n.gettext.UpdatePOs` (*, *po_directory*)

Bases: `object`

Parameter **po_directory** (`annize.fs.FsEntry`) –

annize.features.testing namespace

Submodules

annize.features.testing.common module

class `annize.features.testing.common.RunTests` (***b*)

Bases: `object`

class `annize.features.testing.common.Test`

Bases: `object`

run ()

class `annize.features.testing.common.TestGroup` (*, *tests*)

Bases: `annize.features.testing.common.Test`

Parameter **tests** (*List[annize.features.testing.common.Test]*) –

run ()

annize.features.testing.pylint module

```
class annize.features.testing.pylint.Test (*, src)
    Bases: annize.features.testing.common.Test

    Parameter src (annize.fs.FsEntry) –

    run ()
```

annize.features.testing.pytest module

```
class annize.features.testing.pytest.Test (*, testdir, src)
    Bases: annize.features.testing.common.Test

    Parameter

        • testdir (str) –

        • src (str) –

    run ()
```

annize.features.testing.pyunit module

```
class annize.features.testing.pyunit.Test (*, src)
    Bases: annize.features.testing.common.Test

    Parameter src (str) –

    run ()
```

annize.features.versioncontrol namespace

Submodules

annize.features.versioncontrol.common module

```
class annize.features.versioncontrol.common.BuildVersion (*, base_version, vcs)
    Bases: annize.data.version.Version

    Parameter

        • base_version (annize.data.version.Version) –

        • vcs (annize.features.versioncontrol.common.
            VersionControlSystem) –

    __effversion ()

    property segments_tuples

    property text

class annize.features.versioncontrol.common.VersionControlSystem
    Bases: object

    get_commit_message (revision)
```

```

        Parameter revision (str) –
        Rückgabetyyp str
    get_commit_time (revision)
        Parameter revision (str) –
        Rückgabetyyp datetime.datetime
    get_current_revision ()
        Rückgabetyyp str
    get_revision_list ()
        Rückgabetyyp List[str]
    get_revision_number (revision)
        Parameter revision (str) –
        Rückgabetyyp int
annize.features.versioncontrol.common.default_version_control_system ()
        Rückgabetyyp Optional[annize.features.versioncontrol.common.VersionControlSystem]

```

annize.features.versioncontrol.git module

```

class annize.features.versioncontrol.git.ExcludeByGitIgnores
    Bases: annize.features.files.common.Exclude
    does_exclude (relative_path, source, destination)
class annize.features.versioncontrol.git.VersionControlSystem (*, path="")
    Bases: annize.features.versioncontrol.common.VersionControlSystem
        Parameter path (str) –
    __call_git (cmdline)
        Parameter cmdline (List[str]) –
        Rückgabetyyp str
    get_commit_message (revision)
    get_commit_time (revision)
    get_current_revision ()
    get_revision_list ()
    get_revision_number (revision)
    property path

```

Submodules

annize.features.authors module

class annize.features.authors.**Author** (*, *fullname*, *email=None*)

Bases: object

Parameter

- **fullname** (*str*) –
- **email** (*str*) –

property email

property fullname

annize.features.authors.**default_project_authors** ()

Rückgabetyyp List[annize.features.authors.Author]

annize.features.base module

class annize.features.base.**Basket** (*, *objects*)

Bases: annize.data.container.Basket

Parameter *objects* (List[object]) –

class annize.features.base.**BrandColor** (*, *red*, *green*, *blue*)

Bases: annize.data.color.Color

Parameter

- **red** (*float*) –
- **green** (*float*) –
- **blue** (*float*) –

class annize.features.base.**Data** (*, *project_name=None*, *pretty_project_name=None*, *summary=None*, *long_description=None*, *homepage_url=None*, *imprint=None*, *project_directory=None*)

Bases: object

Parameter

- **project_name** (Union[*str*, annize.il8n.TrStr]) –
- **pretty_project_name** (Union[*str*, annize.il8n.TrStr]) –
- **summary** (Union[*str*, annize.il8n.TrStr]) –
- **long_description** (Union[*str*, annize.il8n.TrStr]) –
- **homepage_url** (Union[*str*, annize.il8n.TrStr]) –
- **imprint** (Union[*str*, annize.il8n.TrStr]) –
- **project_directory** (*str*) –

property homepage_url

property imprint

property long_description

```

    property pretty_project_name
    property project_directory
    property project_name
    property summary
class annize.features.base.DateTime(*, iso=None)
    Bases: datetime.datetime
        Parameter iso(str) –
class annize.features.base.FirstOf(*, objects)
    Bases: annize.data.container.Basket
        Parameter objects(List[object]) –
class annize.features.base.Keyword(text)
    Bases: annize.features.base.Keywords
        Parameter text(str) –
class annize.features.base.Keywords(*, from_string="", split_by=',', keywords=())
    Bases: object
        Parameter
            • from_string(str) –
            • split_by(str) –
            • keywords(List[str]) –
    property keywords
annize.features.base._get_data(key, defaultvalue)
    Parameter
        • key(str) –
        • defaultvalue(Optional[Any]) –
    Rückgabetyp Optional[Any]
annize.features.base.brand_color(*, none_on_undefined=False)
    Parameter none_on_undefined(bool) –
    Rückgabetyp annize.data.color.Color
annize.features.base.homepage_url()
    Rückgabetyp str
annize.features.base.imprint()
    Rückgabetyp str
annize.features.base.long_description()
    Rückgabetyp str
annize.features.base.pretty_project_name()
    Rückgabetyp str
annize.features.base.project_directory()

```

Rückgabotyp str

`annize.features.base.project_keywords()`

Rückgabotyp *annize.features.base.Keywords*

`annize.features.base.project_name()`

Rückgabotyp str

`annize.features.base.summary()`

Rückgabotyp str

annize.features.licensing module

`annize.features.licensing.AFLv3`

Alias von `annize.features.licensing._license.<locals>.ALicense`

`annize.features.licensing.AGPLv3`

Alias von `annize.features.licensing._license.<locals>.ALicense`

`annize.features.licensing.Apache2`

Alias von `annize.features.licensing._license.<locals>.ALicense`

`annize.features.licensing.Artistic1`

Alias von `annize.features.licensing._license.<locals>.ALicense`

`annize.features.licensing.BSD2clause`

Alias von `annize.features.licensing._license.<locals>.ALicense`

`annize.features.licensing.BSD3clause`

Alias von `annize.features.licensing._license.<locals>.ALicense`

`annize.features.licensing.Cc0v1`

Alias von `annize.features.licensing._license.<locals>.ALicense`

`annize.features.licensing.CcBy3`

Alias von `annize.features.licensing._license.<locals>.ALicense`

`annize.features.licensing.CcByNc3`

Alias von `annize.features.licensing._license.<locals>.ALicense`

`annize.features.licensing.CcByNcNd3`

Alias von `annize.features.licensing._license.<locals>.ALicense`

`annize.features.licensing.CcByNcSa3`

Alias von `annize.features.licensing._license.<locals>.ALicense`

`annize.features.licensing.CcByNd3`

Alias von `annize.features.licensing._license.<locals>.ALicense`

`annize.features.licensing.CcBySa3`

Alias von `annize.features.licensing._license.<locals>.ALicense`

`annize.features.licensing.GPLv3`

Alias von `annize.features.licensing._license.<locals>.ALicense`

`annize.features.licensing.LGPLv3`

Alias von `annize.features.licensing._license.<locals>.ALicense`

class `annize.features.licensing.License(*, name, text=None, **additional_info)`

Bases: object

Parameter

- **name** (*str*) –
- **text** (*str*) –

additional_info (*key*, *, *default=None*)

Parameter

- **key** (*str*) –
- **default** (*Optional[Any]*) –

Rückgabotyp *Any*

property name

property text

`annize.features.licensing.MIT`

Alias von `annize.features.licensing._license.<locals>.ALicense`

`annize.features.licensing.MPLv11`

Alias von `annize.features.licensing._license.<locals>.ALicense`

`annize.features.licensing.MPLv2`

Alias von `annize.features.licensing._license.<locals>.ALicense`

`annize.features.licensing.PublicDomain`

Alias von `annize.features.licensing._license.<locals>.ALicense`

`annize.features.licensing._license` (*name*)

Parameter **name** (*str*) –

Rückgabotyp *Type[annize.features.licensing.License]*

`annize.features.licensing.project_licenses` ()

Rückgabotyp *List[annize.features.licensing.License]*

annize.features.medialogalleries module

class `annize.features.medialogalleries.Gallery` (*, *source*, *title=""*)

Bases: *object*

Parameter

- **source** (`annize.fs.FsEntry`) –
- **title** (*str*) –

class `Item` (*file*, *description*)

Bases: *object*

property description

property file

_description_for_mediafile (*itemfile*)

Parameter **itemfile** (`annize.fs.FsEntry`) –

Rückgabotyp *str*

property items

property title

annize.features.task module

class annize.features.task.Task(*, innertasks)

Bases: object

Parameter innertasks (List[object]) –

annize.features.version module

class annize.features.version.CommonVersionPattern

Bases: *annize.data.version.VersionPattern*

class annize.features.version.Line(*, version)

Bases: object

Parameter version (*annize.data.version.Version*) –

property version

class annize.features.version.Version(*, text=None, pattern=None, **segment_values)

Bases: *annize.data.version.Version*

Parameter

- **text** (str) –
- **pattern** (*annize.data.version.VersionPattern*) –

annize.features.version.default_version_pattern()

Rückgabotyp *annize.data.version.VersionPattern*

annize.features.version.project_versions()

Rückgabotyp List[*annize.data.version.Version*]

annize.fs package

class annize.fs.AdHocFsEntry

Bases: *annize.fs.FsEntry*

__getinner()

_abc_impl = <_abc_data object>

_initialize()

Rückgabotyp *annize.fs.FsEntry*

readable_path()

exception annize.fs.BadDisowningError(path)

Bases: RuntimeError

Parameter path (str) –

class annize.fs.ChDir(path)

Bases: object

Temporarily changes the current working directory in a scope of the with keyword.

Parameter `path (str)` – The path to the new working directory.

```
class annize.fs.FsEntry (*, is_ownable=False, is_owned=False)
    Bases: abc.ABC

    Parameter
        • is_ownable (bool) –
        • is_owned (bool) –

    _abc_impl = <_abc_data object>
    basename ()
        Rückgabotyp str
    child_by_relative_path (name)
        Rückgabotyp annize.fs.FsEntry
    children ()
        Rückgabotyp List[annize.fs.FsEntry]
    copy_to (path, *, path_is_parent=False, merge=False, overwrite=False, transfer_predicate=None)
        Parameter
            • path (str) –
            • path_is_parent (bool) –
            • merge (bool) –
            • overwrite (bool) –
            • transfer_predicate (Optional[Callable[[str, FsEntry,  

               FsEntry], bool]) –
        Rückgabotyp annize.fs.FsEntry
    created_time ()
        Rückgabotyp datetime.datetime
    disowned ()
        Rückgabotyp annize.fs.FsEntry
    exists ()
        Rückgabotyp bool
    filesize ()
        Rückgabotyp int
    is_dir (follow_symlinks=True)
        Parameter follow_symlinks (bool) –
        Rückgabotyp bool
    is_file (follow_symlinks=True)
        Parameter follow_symlinks (bool) –
        Rückgabotyp bool
    is_link ()
```

```

    Rückgabetyb bool
property is_ownable
property is_owned
mkdir (*, exist_ok=True)
    Parameter exist_ok (bool) –
    Rückgabetyb None
modified_time ()
    Rückgabetyb datetime.datetime
move_to (path, *, path_is_parent=False, merge=False, overwrite=False, transfer_predicate=None)
    Parameter
        • path (str) –
        • path_is_parent (bool) –
        • merge (bool) –
        • overwrite (bool) –
        • transfer_predicate (Optional[Callable[[str, FsEntry,
            FsEntry], bool]]) –
    Rückgabetyb annize.fs.FsEntry
owned (allow_clone=True)
    Parameter allow_clone (bool) –
    Rückgabetyb annize.fs.FsEntry
parent ()
    Rückgabetyb Optional[annize.fs.FsEntry]
path ()
    Rückgabetyb str
read_file ()
    Rückgabetyb bytes
readable_path ()
    Rückgabetyb str
remove ()
    Rückgabetyb None
temp_clone (*, temprootpath=None, basename=None)
    Parameter
        • temprootpath (Optional[str]) –
        • basename (Optional[str]) –
    Rückgabetyb annize.fs.FsEntry
write_file (content)

```

```

    Parameter content (AnyStr) -
    Rückgabetyt None
exception annize.fs.MissingOwnershipError (path)
    Bases: RuntimeError
    Parameter path (str) -
class annize.fs.Mount (src, dst, *, options=(), mounter=None, unmounter=None)
    Bases: object
    Parameter
        • src (str) -
        • dst (str) -
        • options (Iterable[str]) -
        • mounter (Optional[Iterable[str]]) -
        • unmounter (Optional[Iterable[str]]) -
    property destination
exception annize.fs.OriginalOwnershipImpossibleError (path)
    Bases: RuntimeError
    Parameter path (str) -
class annize.fs._SimpleFsEntry (path, *, is_ownable=False, is_owned=False)
    Bases: annize.fs.FsEntry
    Parameter
        • path (str) -
        • is_ownable (bool) -
        • is_owned (bool) -
    _abc_impl = <_abc_data object>
    readable_path()
class annize.fs._TransferHelper
    Bases: object
    _TransferHelper__transfer_dirmerge (action, transfer_predicate)
    Parameter
        • action (Callable) -
        • transfer_predicate (Optional[Callable[[str, FsEntry,
            FsEntry], bool]]) -
    Rückgabetyt Callable
    _TransferHelper__transfer_prepare (srcf, dstf, overwrite)
    Parameter
        • srcf (annize.fs.FsEntry) -
        • dstf (annize.fs.FsEntry) -
        • overwrite (bool) -

```

Rückgabotyp `None`

transfer_action_copy (*srcf, dstf, overwrite*)

Parameter

- **srcf** (*annize.fs.FsEntry*) –
- **dstf** (*annize.fs.FsEntry*) –
- **overwrite** (*bool*) –

Rückgabotyp `None`

transfer_action_move (*srcf, dstf, overwrite*)

Parameter

- **srcf** (*annize.fs.FsEntry*) –
- **dstf** (*annize.fs.FsEntry*) –
- **overwrite** (*bool*) –

Rückgabotyp `None`

transfer_to (*srcf, path, *, merge, overwrite, action, transfer_predicate=None*)

Parameter

- **srcf** (*annize.fs.FsEntry*) –
- **path** (*str*) –
- **merge** (*bool*) –
- **overwrite** (*bool*) –
- **action** (*Callable*) –
- **transfer_predicate** (*Optional[Callable[[str, FsEntry, FsEntry], bool]]*) –

Rückgabotyp *annize.fs.FsEntry*

annize.fs.fsentry_by_path (*path, *, is_ownable=False, is_owned=False*)

Parameter

- **path** (*str*) –
- **is_ownable** (*bool*) –
- **is_owned** (*bool*) –

Rückgabotyp *annize.fs.FsEntry*

Submodules

annize.fs.temp module

class annize.fs.temp.**TempFileByContent** (*, content, filename='afile')

Bases: *annize.fs.AdHocFsEntry*

Parameter

- **content** (*AnyStr*) –
- **filename** (*Optional[str]*) –

_abc_impl = <_abc_data object>

_initialize()

Rückgabetyyp *annize.fs.FsEntry*

class annize.fs.temp.**_TempDir** (dirname=None, temprootpath=None)

Bases: *annize.fs.FsEntry*

Parameter

- **dirname** (*Optional[str]*) –
- **temprootpath** (*Optional[str]*) –

_TempDir__cleanup()

_abc_impl = <_abc_data object>

readable_path()

restrict_path(path)

annize.fs.temp.**fresh_temp_dir** (dirname=None, *, temprootpath=None)

Parameter

- **dirname** (*Optional[str]*) –
- **temprootpath** (*Optional[str]*) –

Rückgabetyyp *annize.fs.temp._TempDir*

annize.i18n package

class annize.i18n.**Culture** (english_lang_name, iso_639_1_lang_code)

Bases: object

Parameter

- **english_lang_name** (*str*) –
- **iso_639_1_lang_code** (*str*) –

property english_lang_name

static get_from_iso_639_1_lang_code (iso_639_1_lang_code)

property iso_639_1_lang_code

class annize.i18n.**GettextTranslationProvider** (mopath)

Bases: *annize.i18n.TranslationProvider*

```

translate (stringname, *, culture)

    Parameter
        • stringname (str) –
        • culture (annize.i18n.Culture) –

class annize.i18n.IndependentTrStr (**variants)
    Bases: annize.i18n.TrStr

        Parameter variants (str) –

    translate (culture=None)

        Rückgabotyp str

exception annize.i18n.NoCurrentCultureError
    Bases: TypeError

class annize.i18n.ProvidedTrStr (stringname)
    Bases: annize.i18n.TrStr

        Parameter stringname (str) –

    translate (culture=None)

        Rückgabotyp str

class annize.i18n.TODO (txt)
    Bases: annize.i18n.TrStr

    translate (culture=None)

        Parameter culture (Optional[Union[Culture, str]]) –

        Rückgabotyp str

class annize.i18n.TrStr
    Bases: object

    format (*args, **kwargs)

        Rückgabotyp annize.i18n.TrStr

    static tr (stringname)

        Parameter stringname (str) –

        Rückgabotyp annize.i18n.TrStr

    translate (culture=None)

        Parameter culture (Optional[Union[Culture, str]]) –

        Rückgabotyp str

annize.i18n.TrStrOrStr
    Type annotation for something that can be either a str or a TrStr.
    alias of Union[str, annize.i18n.TrStr]

class annize.i18n.TranslationProvider
    Bases: object

    translate (stringname, *, culture)

        Parameter

```


- **stringname** (*str*) –
- **culture** (*annize.i18n.Culture*) –

Rückgabety *str*

exception *annize.i18n.TranslationUnavailableError* (*stringname, language*)

Bases: *TypeError*

Parameter

- **stringname** (*str*) –
- **language** (*str*) –

annize.i18n.add_translation_provider (*provider, *, priority=0*)

Parameter

- **provider** (*annize.i18n.TranslationProvider*) –
- **priority** (*int*) –

Rückgabety *None*

annize.i18n.annize_user_interaction_culture ()

Rückgabety *annize.i18n.Culture*

annize.i18n.current_culture ()

Rückgabety *annize.i18n.Culture*

annize.i18n.get_culture (*culture*)

Parameter **culture** (*Optional[Union[Culture, str]]*) –

Rückgabety *Optional[annize.i18n.Culture]*

annize.i18n.tr (*stringname, *, culture=None*)

Parameter

- **stringname** (*str*) –
- **culture** (*Optional[Union[Culture, str]]*) –

Rückgabety *str*

annize.i18n.tr_if_trstr (*txt, culture=None*)

Parameter

- **txt** (*Union[str, annize.i18n.TrStr]*) –
- **culture** (*Optional[Union[Culture, str]]*) –

Rückgabety *str*

annize.i18n.translation_providers ()

Rückgabety *List[annize.i18n.TranslationProvider]*

annize.project package

Definitions of Annize projects.

```

class annize.project.ArgumentNode
    Bases: annize.project.Node

    _str_helper()

    property append_to

    property arg_name

    property name

exception annize.project.BadStructureError(message)
    Bases: ValueError

    Parameter message (str) –

class annize.project.BlockNode
    Bases: annize.project.Node

    classmethod _allowed_child_types()

class annize.project.CallableNode
    Bases: annize.project.ArgumentNode

    classmethod _allowed_child_types()

    _str_helper()

    property callee

    property feature

exception annize.project.FeatureUnavailableError(featurename)
    Bases: ModuleNotFoundError

    Parameter featurename (str) –

class annize.project.FileNode(path)
    Bases: annize.project.Node

    Parameter path (str) –

    classmethod _allowed_child_types()

    _str_helper()

    property path

exception annize.project.MaterializerError(message)
    Bases: TypeError

    Parameter message (str) –

class annize.project.Node
    Bases: object

    __get_description(indent, with_children, multiline)

    Parameter

        • indent (int) –

        • with_children (bool) –
    
```

```

        • multiline (bool) –
        Rückgabotyp str
    classmethod _allowed_child_types ()
        Rückgabotyp Iterable[Type[annize.project.Node]]
    _str_helper ()
    append_child (node)
        Parameter node (annize.project.Node) –
    property children
    clone (with_children=True)
        Parameter with_children (bool) –
        Rückgabotyp annize.project.Node
    get_description (*, with_children=True, multiline=True)
        Parameter
            • with_children (bool) –
            • multiline (bool) –
        Rückgabotyp str
    insert_child (i, node)
        Parameter
            • i (int) –
            • node (annize.project.Node) –
    property parent
    remove_child (node)
        Parameter node (annize.project.Node) –
class annize.project.OnFeatureUnavailableNode
    Bases: annize.project.ScopableBlockNode
    class Action (value)
        Bases: enum.Enum
        An enumeration.
        FAIL = 'fail'
        SKIP_BLOCK = 'skip_block'
        SKIP_NODE = 'skip_node'
    _str_helper ()
    property do
    property feature
exception annize.project.ParserError (message)
    Bases: ValueError
    Parsing error like bad input xml.

```

Parameter message (*str*) –

```
class annize.project.ProjectDefinition(*nodes)
    Bases: object
```

A data structure that holds an in-memory representation of the project description. It is not the in-memory representation of the live universe object state. It is like the file structure of the project description file, but not exactly the same. There is a parser and a generator for translating between a project description file and an ProjectDefinition.

Parameter nodes (*FileNode*) –

```
static from_files(path)

    Parameter path (str) –
```

Rückgabotyp *annize.project.ProjectDefinition*

property nodes

```
class annize.project.ReferenceNode
    Bases: annize.project.ArgumentNode

    class OnUnresolvableAction(value)
        Bases: enum.Enum

        An enumeration.

        FAIL = 'fail'
        SKIP = 'skip'

    classmethod _allowed_child_types()
    _str_helper()

    property on_unresolvable
    property reference_key

class annize.project.ScalarValueNode
    Bases: annize.project.ArgumentNode

    __shorten(maxlen=100)

    Parameter
        • obj (Any) –
        • maxlen (int) –

    Rückgabotyp str

    _str_helper()

    property value

class annize.project.ScopableBlockNode
    Bases: annize.project.BlockNode

    class Scope(value)
        Bases: enum.Enum

        An enumeration.

        BLOCK = 'block'
        FILE = 'file'
```

```

    PROJECT = 'project'
    _str_helper()
    property scope
exception annize.project.UnresolvableReferenceError(refname)
    Bases: annize.project.MaterializerError
    Parameter refname(str) -

```

Subpackages

annize.project.materializer package

Materializing of Annize projects into a working structure.

```
annize.project.materializer.materialize(nodes, *, featureloader=None)
```

Parameter

- **nodes** (*Iterable*[*annize.project.FileNode*]) -
- **featureloader** (*Optional*[*annize.project.featureloader.FeatureLoader*]) -

Rückgabetyyp *List*[*Optional*[*Any*]]

Subpackages

annize.project.materializer.behaviors package

```

class annize.project.materializer.behaviors.Behavior
    Bases: abc.ABC
    _abc_impl = <_abc_data object>
    abstract node_context(nodemat)
        Parameter nodemat (annize.project.materializer.core.
            NodeMaterialization) -
        Rückgabetyyp AbstractContextManager
    stop_materialization_here_and_try_again()
        Rückgabetyyp None

```

Submodules

annize.project.materializer.behaviors.argument module

```

class annize.project.materializer.behaviors.argument.ArgumentBehavior(callfct,
                                                                    *,
                                                                    featu-
                                                                    reloa-
                                                                    der)
    Bases: annize.project.materializer.behaviors.Behavior

```

```

    Parameter featureloader (annize.project.featureloader.FeatureLoader) –
    _abc_impl = <_abc_data object>
    node_context (nodemat)

```

annize.project.materializer.behaviors.basket module

```

class annize.project.materializer.behaviors.basket.BasketBehavior
    Bases: annize.project.materializer.behaviors.Behavior
    _abc_impl = <_abc_data object>
    node_context (nodemat)

```

annize.project.materializer.behaviors.block module

```

class annize.project.materializer.behaviors.block.BlockBehavior
    Bases: annize.project.materializer.behaviors.Behavior
    _abc_impl = <_abc_data object>
    node_context (nodemat)

```

annize.project.materializer.behaviors.featureunavailable module

```

class annize.project.materializer.behaviors.featureunavailable.FeatureUnavailableBehavior
    Bases: annize.project.materializer.behaviors.Behavior
    __context_catchexceptions (nodemat, featureignorelist)
    __context_skipnode_featureignorelist (node)
    _abc_impl = <_abc_data object>
    node_context (nodemat)

```

annize.project.materializer.behaviors.reference module

```

class annize.project.materializer.behaviors.reference.ReferenceBehavior (rootblock)
    Bases: annize.project.materializer.behaviors.Behavior
    _abc_impl = <_abc_data object>
    detect_retry_useful_node_context (nodemat)
    node_context (nodemat)
    resolver_node_context (nodemat)
    retry_node_context (nodemat)

class annize.project.materializer.behaviors.reference.SkipUnresolvableReferenceBehavior (rootblock)
    Bases: annize.project.materializer.behaviors.Behavior
    classmethod __node_is_hopeless (node, rootnodemat)
        Parameter node (annize.project.Node) –

```

```

    Rückgabetyb bool
__nodes_by_name (*, rootnode)
    Parameter
        • name (str) –
        • rootnode (annize.project.Node) –
    Rückgabetyb List[annize.project.Node]
classmethod __reference_is_hopeless (refnode, *, rootnodemat)
    Parameter refnode (annize.project.ReferenceNode) –
    Rückgabetyb bool
__unresolvable_references ()
    Rückgabetyb List[annize.project.ReferenceNode]
__abc_impl = <_abc_data object>
node_context (nodemat)

```

Submodules

annize.project.materializer.core module

```

class annize.project.materializer.core.NodeMaterialization (materializer, node,
                                                            store)
    Bases: object
    Parameter
        • materializer (ProjectMaterializer) –
        • node (annize.project.Node) –
        • store (Dict) –
    get_materialized_children ()
        Rückgabetyb Iterable[Optional[Any]]
    get_materialized_children_tuples ()
    property has_result
    property node
    property result
    set_materialized_result (resultlist)
    try_get_materialization_for_node (node)
        Parameter node (annize.project.Node) –
class annize.project.materializer.core.ProjectMaterializer (node, *, behaviors)
    Bases: object
    Parameter
        • node (annize.project.Node) –

```

- **behaviors** (*Iterable[annize.project.materializer.behaviors.Behavior]*) –

__materialization_for_node (*node, store*)

Parameter

- **node** (*annize.project.Node*) –
- **store** (*Dict*) –

Rückgabotyp *annize.project.materializer.core.NodeMaterialization*

__materialize (*node, store*)

Parameter

- **node** (*annize.project.Node*) –
- **store** (*Dict*) –

Rückgabotyp *List[Optional[Any]]*

__materialize_hlp_childobjs (*node, store*)

Parameter

- **node** (*annize.project.Node*) –
- **store** (*Dict*) –

Rückgabotyp *List[Tuple[annize.project.Node, List[Optional[Any]]]]*

get_materialized ()

Rückgabotyp *List[Optional[Any]]*

exception *annize.project.materializer.core.TryAgain*
 Bases: *BaseException*

annize.project.materializer.preprocessors module

annize.project.materializer.preprocessors.normalize_blockscopes (*nodes*)

Parameter **nodes** (*Iterable[annize.project.Node]*) –

Rückgabotyp *Iterable[annize.project.Node]*

annize.project.materializer.preprocessors.resolve_appendtonodes (*nodes*)

Parameter **nodes** (*Iterable[annize.project.Node]*) –

Rückgabotyp *Iterable[annize.project.Node]*

Submodules

annize.project.featureloader module

Feature module loader.

```
class annize.project.featureloader.DefaultFeatureLoader
    Bases: annize.project.featureloader.FeatureLoader
        _FEATURES_NAMESPACE = 'annize.features'
        _abc_impl = <_abc_data object>
        load_feature(name)

class annize.project.featureloader.FeatureLoader
    Bases: abc.ABC
        _abc_impl = <_abc_data object>
        abstract load_feature(name)
            Parameter name (str) –
                Rückgabetyp Optional[Any]
```

annize.project.parser module

Reading, processing and writing of Annize projects.

```
class annize.project.parser._XmlParser
    Bases: object
        ATTRIBUTE_COMMAND_END = '~'
        ATTRIBUTE_COMMAND_START = '~'

    class _Context
        Bases: object
            static _Context__nodeshort(xnode, maxlen=100)
                Parameter xnode (xml.etree.ElementTree.Element) –
            in_file(fpath)
            in_node(xnode)

    class _TagParts(name, namespace="")
        Bases: object
            Parameter namespace (str) –

    classmethod _XmlParser__interpret_attribute_string(txt)
        Parameter txt (str) –
        Rückgabetyp Tuple[bool, str]

    classmethod _XmlParser__parse_attrib(key, value)
        Parameter
            • key (str) –
            • value (str) –
```

Rückgabotyp *annize.project.Node*

_XmlParser__parse_child (*node*, *xnode*)

Parameter

- **node** (*annize.project.Node*) –
- **xnode** (*xml.etree.ElementTree.Element*) –

Rückgabotyp *Tuple[annize.project.Node, xml.etree.ElementTree.Element]*

_XmlParser__parse_children (*node*, *xparent*)

Parameter

- **node** (*annize.project.Node*) –
- **xparent** (*xml.etree.ElementTree.Element*) –

_XmlParser__parse_tag (*node*, *argname*, *callname*, *feature*, *xnode*)

Parameter **node** (*annize.project.Node*) –

classmethod escape_attribute_string (*txt*)

Parameter **txt** (*str*) –

Rückgabotyp *str*

parse_file (*fpath*)

Parameter **fpath** (*str*) –

Rückgabotyp *annize.project.FileNode*

annize.project.parser.parse (*path*)

Parameter **path** (*str*) –

Rückgabotyp *List[annize.project.FileNode]*

annize.userfeedback package

class *annize.userfeedback.NullUserFeedbackController*

Bases: *object*

choice_dialog (*_)

input_dialog (*_)

message_dialog (*_)

exception *annize.userfeedback.UnsatisfiableUserFeedbackAttemptError*

Bases: *RuntimeError*

class *annize.userfeedback.UserFeedbackController*

Bases: *object*

choice_dialog (*question*, *choices*, *config_key*)

Parameter

- **question** (*str*) –
- **choices** (*List[str]*) –
- **config_key** (*Optional[str]*) –

Rückgabotyp Optional[int]

input_dialog (*question, suggested_answer, config_key*)

Parameter

- **question** (*str*) –
- **suggested_answer** (*str*) –
- **config_key** (*Optional[str]*) –

Rückgabotyp Optional[str]

message_dialog (*message, answers, config_key*)

Parameter

- **message** (*str*) –
- **answers** (*List[str]*) –
- **config_key** (*Optional[str]*) –

Rückgabotyp int

annize.userfeedback._add_controller_to_context (*, *controller, context, priority_index=0*)

Parameter

- **controller** (*annize.userfeedback.UserFeedbackController*) –
- **context** (*annize.context.Context*) –
- **priority_index** (*int*) –

Rückgabotyp None

annize.userfeedback._controllers_for_context (*context*)

Parameter **context** (*annize.context.Context*) –

Rückgabotyp List[*annize.userfeedback.UserFeedbackController*]

annize.userfeedback._controllers_tuples_for_context (*context*)

Parameter **context** (*annize.context.Context*) –

Rückgabotyp List[Tuple[int, *annize.userfeedback.UserFeedbackController*]]

annize.userfeedback.choice_dialog (*message, choices, *, config_key=None*)

Parameter

- **message** (*Union[str, annize.i18n.TrStr]*) –
- **choices** (*Iterable[Union[str, annize.i18n.TrStr]]*) –
- **config_key** (*Optional[str]*) –

Rückgabotyp Optional[int]

annize.userfeedback.input_dialog (*message, *, suggested_answer, config_key=None*)

Parameter

- **message** (*Union[str, annize.i18n.TrStr]*) –
- **suggested_answer** (*Union[str, annize.i18n.TrStr]*) –
- **config_key** (*Optional[str]*) –

Rückgabotyp Optional[str]

`annize.userfeedback.message_dialog(message, answers, *, config_key=None)`

Parameter

- **message** (*Union[str, annize.i18n.TrStr]*) –
- **answers** (*Iterable[Union[str, annize.i18n.TrStr]]*) –
- **config_key** (*Optional[str]*) –

Rückgabotyp int

Submodules

annize.userfeedback.static module

class `annize.userfeedback.static.StaticUserFeedbackController` (*answers*)

Bases: `annize.userfeedback.UserFeedbackController`

Parameter **answers** (*Dict[str, Any]*) –

`__get_answer` (*config_key*)

Parameter **config_key** (*str*) –

Rückgabotyp Any

`add_answer` (*config_key, value*)

Parameter

- **config_key** (*str*) –
- **value** (*Any*) –

Rückgabotyp None

`choice_dialog` (*question, choices, config_key*)

`input_dialog` (*question, suggested_answer, config_key*)

`message_dialog` (*message, answers, config_key*)

annize.userfeedback.tty module

class `annize.userfeedback.tty.TtyUserFeedbackController`

Bases: `annize.userfeedback.UserFeedbackController`

`__action_line` (*num, text*)

Parameter

- **num** (*Any*) –
- **text** (*str*) –

Rückgabotyp str

`__dialog` (*message, config_key, actiontext*)

Parameter

- **message** (*str*) –

- **config_key** (*str*) –
- **actionstext** (*str*) –

Rückgabety *str*

__dialog_frame_actions (*text*)

Parameter **text** (*str*) –

Rückgabety *str*

__dialog_frame_configkey (*config_key*)

Parameter **config_key** (*str*) –

Rückgabety *str*

__dialog_frame_message (*message*)

Parameter **message** (*str*) –

Rückgabety *str*

choice_dialog (*question, choices, config_key*)

input_dialog (*question, suggested_answer, config_key*)

message_dialog (*message, answers, config_key*)

6.1.2 Submodules

6.1.3 annize.annize_cli module

```
class annize.annize_cli.Cli
    Bases: object

    classmethod __answers_from_json_files (destination, with_answers_from_json_files)
        Parameter
        • destination (Dict) –
        • with_answers_from_json_files (Iterable[str]) –

    classmethod __answers_from_json_strings (destination, with_answers_from_json_strings)
        Parameter
        • destination (Dict) –
        • with_answers_from_json_strings (Iterable[str]) –

    classmethod __answers_from_single_answers (destination, with_answers)
        Parameter
        • destination (Dict) –
        • with_answers (Iterable[Tuple[str, str]]) –

    classmethod do (project, taskname, with_answers_from_json_file, with_answers_from_json_string,
                    with_answer, **_)
        Parameter
        • project (str) –
```

```

        • taskname (str) –
        • with_answers_from_json_file (Iterable[str]) –
        • with_answers_from_json_string (Iterable[str]) –
        • with_answer (Iterable[Tuple[str, str]]) –

    project_default = '/home/pino/projects/annize/src/annize'
annize.annize_cli.get_parser()

    Rückgabetyp argparse.ArgumentParser
annize.annize_cli.main()

```

6.1.4 annize.engine module

```

class annize.engine.LoadedProject (context, null_culture)
    Bases: object

    Parameter

        • context (annize.context.Context) –
        • null_culture (annize.i18n.Culture) –

    property context

annize.engine._prepare_context (context, *, annize_config_rootpath)

    Parameter

        • context (annize.context.Context) –
        • annize_config_rootpath (str) –

    Rückgabetyp None

annize.engine.find_project_annize_config_root_file (project_path)

    Parameter project_path (str) –

    Rückgabetyp Optional[str]

annize.engine.guess_project_root_directory (annize_config_rootpath)

    Parameter annize_config_rootpath (str) –

    Rückgabetyp str

annize.engine.load_project (project_path)

    Parameter project_path (str) –

    Rückgabetyp annize.engine.LoadedProject

annize.engine.load_project_definition (projectdefinition, *, annize_config_rootpath)

    Parameter

        • projectdefinition (annize.project.ProjectDefinition) –
        • annize_config_rootpath (str) –

    Rückgabetyp annize.engine.LoadedProject

```

a

`annize.annize_cli`, 97
`annize.aux`, 13
`annize.aux._project_infos`, 13
`annize.callconv`, 13
`annize.callconv.typehints`, 15
`annize.context`, 15
`annize.data`, 18
`annize.data.color`, 18
`annize.data.container`, 19
`annize.data.unique`, 19
`annize.data.version`, 20
`annize.engine`, 98
`annize.features.authors`, 74
`annize.features.base`, 74
`annize.features.changelog.common`, 22
`annize.features.datainjections.common`, 23
`annize.features.datainjections.python`, 23
`annize.features.dependencies.common`, 23
`annize.features.dependencies.python`, 25
`annize.features.distributables.common`, 25
`annize.features.distributables.debian`, 26
`annize.features.distributables.flatpak`, 40
`annize.features.distributables.pythonwheel`, 44
`annize.features.distributables.store.pypi`, 25
`annize.features.distributables.tar`, 47
`annize.features.documentation.common`, 59
`annize.features.documentation.sphinx._utils`, 50
`annize.features.documentation.sphinx.common`, 51
`annize.features.documentation.sphinx.cpp`, 56
`annize.features.documentation.sphinx.doxygencompat`, 57
`annize.features.documentation.sphinx.javascript`, 58
`annize.features.documentation.sphinx.output.common`, 48
`annize.features.documentation.sphinx.output.html`, 49
`annize.features.documentation.sphinx.output.pdf`, 50
`annize.features.documentation.sphinx.output.plaintext`, 50
`annize.features.documentation.sphinx.python`, 58
`annize.features.documentation.sphinx.rst`, 59
`annize.features.files.common`, 63
`annize.features.files.transfer.common`, 61
`annize.features.files.transfer.ssh`, 62
`annize.features.homepage.common`, 67
`annize.features.homepage.sections.about`, 65
`annize.features.homepage.sections.changelog`, 65
`annize.features.homepage.sections.documentation`, 65
`annize.features.homepage.sections.download`, 66
`annize.features.homepage.sections.gallery`, 66
`annize.features.homepage.sections.imprint`, 67
`annize.features.homepage.sections.license`, 67
`annize.features.i18n.common`, 70
`annize.features.i18n.gettext`, 71
`annize.features.licensing`, 76

annize.features.mediagalleries, 77
 annize.features.task, 78
 annize.features.testing.common, 71
 annize.features.testing.pylint, 72
 annize.features.testing.pytest, 72
 annize.features.testing.pyunit, 72
 annize.features.version, 78
 annize.features.versioncontrol.common,
 72
 annize.features.versioncontrol.git, 73
 annize.fs, 78
 annize.fs.temp, 83
 annize.il8n, 83
 annize.project, 86
 annize.project.featureloader, 93
 annize.project.materializer, 89
 annize.project.materializer.behaviors,
 89
 annize.project.materializer.behaviors.argument,
 89
 annize.project.materializer.behaviors.basket,
 90
 annize.project.materializer.behaviors.block,
 90
 annize.project.materializer.behaviors.featureunavailable,
 90
 annize.project.materializer.behaviors.reference,
 90
 annize.project.materializer.core, 91
 annize.project.materializer.preprocessors,
 92
 annize.project.parser, 93
 annize.userfeedback, 94
 annize.userfeedback.static, 96
 annize.userfeedback.tty, 96

Sonderzeichen

_CallHelper (Klasse in *annize.callconv*), 14
 _CallHelper__convert_kwargs_from_string() (statische Methode von *annize.callconv._CallHelper*), 14
 _CallHelper__shift_args_to_kwargs() (statische Methode von *annize.callconv._CallHelper*), 14
 _Context__nodeshort() (statische Methode von *annize.project.parser._XmlParser._Context*), 93
 _DocumentGenerateAllLanguagesResult (Klasse in *annize.features.documentation.common*), 60
 _DocumentGenerateResult (Klasse in *annize.features.documentation.common*), 61
 _FEATURES_NAMESPACE (Attribut von *annize.project.featureloader.DefaultFeatureLoader*), 93
 _METADATA_IS_TOPLEVEL_OBJECT (Attribut von *annize.context.Context*), 15
 _S_CHANGE (Attribut von *annize.features.changelog.common.ByVersionControlSystemCommitMessagesChangelog*), 22
 _S_LABEL (Attribut von *annize.features.changelog.common.ByVersionControlSystemCommitMessagesChangelog*), 22
 _SimpleFsEntry (Klasse in *annize.fs*), 81
 _TempDir (Klasse in *annize.fs.temp*), 83
 _TempDir__cleanup() (Methode von *annize.fs.temp._TempDir*), 83
 _TransferHelper (Klasse in *annize.fs*), 81
 _TransferHelper__transfer_dirmerge() (Methode von *annize.fs._TransferHelper*), 81
 _TransferHelper__transfer_prepare() (Methode von *annize.fs._TransferHelper*), 81
 _XmlParser (Klasse in *annize.project.parser*), 93
 _XmlParser._Context (Klasse in *annize.project.parser*), 93
 _XmlParser._TagParts (Klasse in *annize.project.parser*), 93
 _XmlParser__interpret_attribute_string() (Klassenmethode von *annize.project.parser._XmlParser*), 93
 _XmlParser__parse_attrib() (Klassenmethode von *annize.project.parser._XmlParser*), 93
 _XmlParser__parse_child() (Methode von *annize.project.parser._XmlParser*), 94
 _XmlParser__parse_children() (Methode von *annize.project.parser._XmlParser*), 94
 _XmlParser__parse_tag() (Methode von *annize.project.parser._XmlParser*), 94
 __action_line() (Methode von *annize.userfeedback.tty.TtyUserFeedbackController*), 96
 __answers_from_json_files() (Klassenmethode von *annize.annize_cli.Cli*), 97
 __answers_from_json_strings() (Klassenmethode von *annize.annize_cli.Cli*), 97
 __answers_from_single_answers() (Klassenmethode von *annize.annize_cli.Cli*), 97
 __call_git() (Methode von *annize.features.versioncontrol.git.VersionControlSystem*), 73
 __context_match_exceptions() (Methode von *annize.project.materializer.behaviors.featureunavailable.FeatureUnavailable*), 90
 __context_skipnode_featureignorelist() (Methode von *annize.project.materializer.behaviors.featureunavailable.FeatureUnavailable*), 90
 __dialog() (Methode von *annize.userfeedback.tty.TtyUserFeedbackController*), 96
 __dialog_frame_actions() (Methode von *annize.userfeedback.tty.TtyUserFeedbackController*), 97
 __dialog_frame_configkey() (Methode von *annize*), 97

ze.userfeedback.tty.TtyUserFeedbackController), __materialize() (Methode von anni-
 97 ze.project.materializer.core.ProjectMaterializer),
 __dialog_frame_message() (Methode von anni- 92
 ze.userfeedback.tty.TtyUserFeedbackController), __node_is_hopeless() (Klas-
 97 senmethode von anni-
 __does_exclude() (Methode von anni- ze.project.materializer.behaviors.reference.SkipUnresolvableReferen-
 ze.features.files.common.Exclude), 64 90
 __effversion() (Methode von anni- __nodes_by_name() (Methode von anni-
 ze.features.versioncontrol.common.BuildVersion), ze.project.materializer.behaviors.reference.SkipUnresolvableReferen-
 72 91
 __generate_packagelist() (Methode von anni- __object_metadata_dict() (Methode von anni-
 ze.features.homepage.sections.download.Section), ze.context.Context), 15
 66 __object_raw_name() (Methode von anni-
 __generate_postpreproc() (Methode von anni- ze.context.Context), 15
 ze.features.homepage.common.Homepage), 67
 __generate_section() (Methode von anni- __patch_property_types_in_docstrings() (Methode von anni-
 ze.features.homepage.common.Homepage), ze.features.documentation.sphinx.python.Python3ApiReferenceLa-
 67 59
 __get_answer() (Methode von anni- __put_object() (Methode von anni-
 ze.userfeedback.static.StaticUserFeedbackController), ze.context.Context), 15
 96 __reference_is_hopeless() (Klassenmethode von anni-
 __get_description() (Methode von anni- ze.project.materializer.behaviors.reference.SkipUnresolvableReferen-
 ze.project.Node), 86 91
 __get_files_from_packagestore() (Methode von anni- __scanjsfile() (Methode von anni-
 ze.features.distributables.common.Group), ze.features.documentation.sphinx.javascript.JavaScriptApiReferen-
 26 58
 __get_inner_generateinfo() __shorten() (Methode von anni-
 (Methode von anni- ze.project.ScalarValueNode), 88
 ze.features.documentation.sphinx.common.CompositeDocument), 53
 __get_refgeninfo() (Methode von anni- __site_files_to_packagestore() (Methode von anni-
 ze.features.documentation.sphinx.common.ApiReferenceDocument), ze.features.distributables.common.Group), 26
 52
 __get_variant() (Methode von anni- __transfer_predicate_and() (Klassenmethode von anni-
 ze.features.documentation.sphinx.common.RstDocument), ze.features.files.common.Dir), 63
 56 __transfer_predicate_for_exclude() (Klassenmethode von anni-
 __getinner() (Methode von anni- ze.features.files.common.Dir), 63
 ze.fs.AdHocFsEntry), 78
 __getname() (Methode von anni- __transfer_predicate_for_excludes() (Klassenmethode von anni-
 ze.features.distributables.common.Group), ze.features.files.common.Dir), 63
 26
 __info() (Methode von anni- __translations_for_stringname() (Methode von anni-
 ze.features.documentation.sphinx.doxygencompat.DoxygenSupportedApiReferenceProvider), ze.features.documentation.sphinx.javascript.JavaScriptApiReferenceProvider), 70
 58
 __jsfiles() (Methode von anni- __uniqueid_counter (Attribut von anni-
 ze.features.documentation.sphinx.javascript.JavaScriptApiReferenceProvider), 19
 58
 __long_str() (Methode von anni- __uniqueid_lock (Attribut von anni-
 ze.data.unique.UniqueId), 19 ze.data.unique.UniqueId), 19
 __materialization_for_node() (Methode von anni-
 (Methode von anni- ze.project.materializer.behaviors.reference.SkipUnresolvableReferen-
 ze.project.materializer.core.ProjectMaterializer), 91
 92
 __abc_impl (Attribut von anni-

`ze.features.datainjections.common.Injection),` 53
`_abc_impl (Attribut von anni-`
`ze.features.datainjections.python.Injection),`
`_abc_impl (Attribut von anni-`
`ze.features.documentation.sphinx.common.CompositeDocument),` 54
`_abc_impl (Attribut von anni-`
`ze.features.documentation.sphinx.common.Document),` 54
`_abc_impl (Attribut von anni-`
`ze.features.documentation.sphinx.common.ReadmeDocument),` 55
`_abc_impl (Attribut von anni-`
`ze.features.documentation.sphinx.common.RstDocument),` 56
`_abc_impl (Attribut von anni-`
`ze.features.documentation.sphinx.cpp.CppApiReferenceLanguage),` 56
`_abc_impl (Attribut von anni-`
`ze.features.documentation.sphinx.doxygencompat.DoxygenSupport),` 58
`_abc_impl (Attribut von anni-`
`ze.features.documentation.sphinx.javascript.JavaScriptApiReference),` 58
`_abc_impl (Attribut von anni-`
`ze.features.documentation.sphinx.output.common.OutputGenerator),` 48
`_abc_impl (Attribut von anni-`
`ze.features.documentation.sphinx.output.html.HtmlOutputGenerator),` 49
`_abc_impl (Attribut von anni-`
`ze.features.documentation.sphinx.output.html.HtmlOutputSpec),` 49
`_abc_impl (Attribut von anni-`
`ze.features.documentation.sphinx.output.pdf.PdfOutputGenerator),` 50
`_abc_impl (Attribut von anni-`
`ze.features.documentation.sphinx.output.plaintext.PlaintextOutputGenerator),` 50
`_abc_impl (Attribut von anni-`
`ze.features.documentation.sphinx.python.Python3ApiReferenceLanguage),` 59
`_abc_impl (Attribut von anni-`
`ze.features.documentation.sphinx.common.PlaintextOutputSpec),` 60
`_abc_impl (Attribut von anni-`
`ze.features.documentation.sphinx.common.AboutProjectDocument),` 51
`_abc_impl (Attribut von anni-`
`ze.features.documentation.sphinx.common.ApiReferenceDocument),` 52
`_abc_impl (Attribut von anni-`
`ze.features.documentation.sphinx.common.ApiReferenceLanguage),` 52
`_abc_impl (Attribut von anni-`
`ze.features.homepage.common.GeneratedHomepage),` 67
`ze.features.documentation.sphinx.common.ArgparseCommandLineInterfaceDocument),`

_abc_impl (Attribut von annize.fs.AdHocFsEntry), 78
 _abc_impl (Attribut von annize.fs.FsEntry), 79
 _abc_impl (Attribut von annize.fs.SimpleFsEntry), 81
 _abc_impl (Attribut von annize.fs.temp.TempFileByContent), 83
 _abc_impl (Attribut von annize.fs.temp.TempDir), 83
 _abc_impl (Attribut von annize.project.featureloader.DefaultFeatureLoader), 93
 _abc_impl (Attribut von annize.project.featureloader.FeatureLoader), 93
 _abc_impl (Attribut von annize.project.materializer.behaviors.Behavior), 89
 _abc_impl (Attribut von annize.project.materializer.behaviors.argument.ArgumentBehavior), 90
 _abc_impl (Attribut von annize.project.materializer.behaviors.basket.BasketBehavior), 90
 _abc_impl (Attribut von annize.project.materializer.behaviors.block.BlockBehavior), 90
 _abc_impl (Attribut von annize.project.materializer.behaviors.featureunavailable.FeatureUnavailableBehavior), 90
 _abc_impl (Attribut von annize.project.materializer.behaviors.reference.ReferenceBehavior), 90
 _abc_impl (Attribut von annize.project.materializer.behaviors.reference.SkipUnresolvableReferencesBehavior), 91
 _add_controller_to_context () (im Modul annize.userfeedback), 95
 _allowed_child_types () (Klassenmethode von annize.project.BlockNode), 86
 _allowed_child_types () (Klassenmethode von annize.project.CallableNode), 86
 _allowed_child_types () (Klassenmethode von annize.project.FileNode), 86
 _allowed_child_types () (Klassenmethode von annize.project.Node), 87
 _allowed_child_types () (Klassenmethode von annize.project.ReferenceNode), 88
 _append_section () (Methode von annize.features.homepage.common.Homepage), 68
 _authors () (annize.features.documentation.sphinx.common.Document property), 54
 _controllers_for_context () (im Modul annize.userfeedback), 95
 _controllers_tuples_for_context () (im Modul annize.userfeedback), 95
 _debian_category () (im Modul annize.features.distributables.debian), 39
 _debian_section () (im Modul annize.features.distributables.debian), 39
 _description_for_mediafile () (Methode von annize.features.media galleries.Gallery), 77
 _generate () (Methode von annize.features.homepage.common.HomepageSection), 69
 _generate () (Methode von annize.features.homepage.common.PinoHomepageAuxInfo), 70
 _generate () (Methode von annize.features.homepage.sections.about.Section), 65
 _generate () (Methode von annize.features.homepage.sections.changelog.Section), 65
 _generate () (Methode von annize.features.homepage.sections.documentation.Section), 65
 _generate () (Methode von annize.features.homepage.sections.download.Section), 66
 _generate () (Methode von annize.features.homepage.sections.gallery.Section), 66
 _generate () (Methode von annize.features.homepage.sections.imprint.Section), 67
 _generate () (Methode von annize.features.homepage.sections.license.Section), 67
 _generate_master_doc () (Methode von annize.features.documentation.sphinx.common.AboutProjectDocument), 51
 _generate_master_doc () (Methode von annize.features.documentation.sphinx.common.ApiReferenceDocument), 52
 _generate_master_doc () (Methode von annize.features.documentation.sphinx.common.ArgparseCommandLineDocument), 53
 _generate_master_doc () (Methode von annize.features.documentation.sphinx.common.CompositeDocument), 54
 _generate_master_doc () (Methode von annize.features.documentation.sphinx.common.Document), 54
 _generate_master_doc () (Methode von annize.features.documentation.sphinx.common.RstDocument), 56
 _generate_master_doc () (Methode von annize.features.documentation.sphinx.doxygencompat.DoxygenSupportDocument), 58

```

_get_data() (im Modul annize.features.base), 75
_initialize() (Methode von annize.features.datainjections.common.Injection), 23
_initialize() (Methode von annize.features.distributables.debian.Package), 36
_initialize() (Methode von annize.features.distributables.flatpak.FlatpakImage), 41
_initialize() (Methode von annize.features.distributables.flatpak.FlatpakrefFile), 42
_initialize() (Methode von annize.features.distributables.flatpak.GpgFile), 43
_initialize() (Methode von annize.features.distributables.pythonwheel.Package), 46
_initialize() (Methode von annize.features.distributables.tar.Package), 47
_initialize() (Methode von annize.features.documentation.common.GeneratedDocument), 60
_initialize() (Methode von annize.features.files.common.Dir), 63
_initialize() (Methode von annize.features.files.common.FsEntry), 64
_initialize() (Methode von annize.features.files.common.MachineRootDirectory), 65
_initialize() (Methode von annize.features.files.common.ProjectDir), 65
_initialize() (Methode von annize.features.homepage.common.GeneratedHomepage), 67
_initialize() (Methode von annize.fs.AdHocFsEntry), 78
_initialize() (Methode von annize.fs.temp.TempFileByContent), 83
_items_matcher() (im Modul annize.features.changelog.common), 22
_license() (im Modul annize.features.licensing), 77
_materialize_hlp_childobjs() (Methode von annize.project.materializer.core.ProjectMaterializer), 92
_mkpakage() (Klassenmethode von annize.features.distributables.debian.Package), 36
_mkpakage() (Klassenmethode von annize.features.distributables.flatpak.FlatpakImage), 41
_mkpakage() (Klassenmethode von annize.features.distributables.flatpak.FlatpakImage), 41
_mkpakage_applysource() (Klassenmethode von annize.features.distributables.flatpak.FlatpakImage), 41
_mkpakage_bdist_wheel() (Klassenmethode von annize.features.distributables.pythonwheel.Package), 46
_mkpakage_correctbuildsourcepermissions() (Klassenmethode von annize.features.distributables.debian.Package), 36
_mkpakage_determinesize() (Klassenmethode von annize.features.distributables.debian.Package), 36
_mkpakage_dpkg() (Klassenmethode von annize.features.distributables.debian.Package), 36
_mkpakage_flatpak_build_export() (Klassenmethode von annize.features.distributables.flatpak.FlatpakImage), 41
_mkpakage_flatpak_build_finish() (Klassenmethode von annize.features.distributables.flatpak.FlatpakImage), 41
_mkpakage_flatpak_build_init() (Klassenmethode von annize.features.distributables.flatpak.FlatpakImage), 42
_mkpakage_mkchangelog() (Klassenmethode von annize.features.distributables.debian.Package), 37
_mkpakage_mkcopyright() (Klassenmethode von annize.features.distributables.debian.Package), 37
_mkpakage_mkdebianconffilesfile() (Klassenmethode von annize.features.distributables.debian.Package), 37
_mkpakage_mkdebiancontrolfile() (Klassenmethode von annize.features.distributables.debian.Package), 37
_mkpakage_mkexeclinks() (Klassenmethode von annize.features.distributables.debian.Package), 37
_mkpakage_mkexeclinks() (Klassenmethode von annize.features.distributables.debian.Package), 37
_mkpakage_mkexeclinks() (Klassenmethode von annize.features.distributables.debian.Package), 37

```


[ze.features.distributables.pythonwheel.Package\)](#), [_name_anon\(\)](#) (Methode von [anni-ze.data.version.NumericVersionPatternSegment](#)),
[46](#) [20](#)
[_mkpackage_mkmanifestin\(\)](#) (Klassenmethode von [anni-ze.features.distributables.pythonwheel.Package](#)),
[46](#) [21](#)
[_mkpackage_mkmenuentries\(\)](#) (Klassenmethode von [anni-ze.features.distributables.debian.Package](#)),
[37](#) [52](#)
[_mkpackage_mkprepostcmds\(\)](#) (Klassenmethode von [anni-ze.features.distributables.debian.Package](#)),
[37](#) [53](#)
[_mkpackage_mkservices\(\)](#) (Klassenmethode von [annize.features.distributables.debian.Package](#)),
[37](#) [54](#)
[_mkpackage_mksetuppyconf\(\)](#) (Klassenmethode von [anni-ze.features.distributables.pythonwheel.Package](#)),
[46](#) [55](#)
[_mkpackage_prepareinfos\(\)](#) (Klassenmethode von [anni-ze.features.distributables.debian.Package](#)),
[37](#) [_projectname\(\)](#) ([anni-ze.features.documentation.sphinx.common.Document](#) property), [55](#)
[_mkpackage_prepareinfos\(\)](#) (Klassenmethode von [anni-ze.features.distributables.flatpak.FlatpakImage](#)),
[42](#) [_release\(\)](#) ([annize.features.documentation.sphinx.common.Document](#) property), [55](#)
[_mkpackage_prepareinfos\(\)](#) (Klassenmethode von [anni-ze.features.distributables.pythonwheel.Package](#)),
[46](#) [_set_entry_path\(\)](#) (Methode von [anni-ze.features.documentation.common._DocumentGenerateResult](#)),
[42](#) [61](#)
[_mkpackage_setuppyconf_classifiers\(\)](#) (Klassenmethode von [anni-ze.features.distributables.pythonwheel.Package](#)),
[47](#) [_str_helper\(\)](#) (Methode von [anni-ze.project.ArgumentNode](#)), [86](#)
[_mkpackage_setuppyconf_install_requires\(\)](#) (Klassenmethode von [anni-ze.features.distributables.pythonwheel.Package](#)),
[47](#) [_str_helper\(\)](#) (Methode von [anni-ze.project.CallableNode](#)), [86](#)
[_mkpackage_setuppyconf_prepare\(\)](#) (Klassenmethode von [anni-ze.features.distributables.pythonwheel.Package](#)),
[47](#) [_str_helper\(\)](#) (Methode von [anni-ze.project.FileNode](#)), [86](#)
[_mkpackage_share\(\)](#) (Klassenmethode von [anni-ze.features.distributables.flatpak.FlatpakImage](#)),
[42](#) [_str_helper\(\)](#) (Methode von [annize.project.Node](#)),
[_name_anon\(\)](#) (Methode von [anni-ze.data.version.AbstractVersionPatternSegment](#)),
[20](#) [87](#)
[_name_anon\(\)](#) (Methode von [anni-ze.data.version.ConcatenatedVersionPatternSegment](#)),
[20](#) [87](#)
[_name_anon\(\)](#) (Methode von [anni-ze.data.version.NumericVersionPatternSegment](#)),
[20](#) [88](#)
[_to_dict\(\)](#) (Methode von [anni-ze.features.documentation.sphinx.common.Document.GenerateIn](#)),
[54](#)
[_version\(\)](#) ([annize.features.documentation.sphinx.common.Document](#)

property), 55

A

AboutProjectDocument (Klasse in *annize.features.documentation.sphinx.common*), 51

AbstractVersionPatternSegment (Klasse in *annize.data.version*), 20

access_filesystem() (Methode von *annize.features.files.transfer.common.Endpoint*), 61

access_filesystem() (Methode von *annize.features.files.transfer.common.FsEndpoint*), 61

access_filesystem() (Methode von *annize.features.files.transfer.ssh.Endpoint*), 62

add_answer() (Methode von *annize.userfeedback.static.StaticUserFeedbackController*), 96

add_object() (im Modul *annize.context*), 17

add_object() (Methode von *annize.context.Context*), 16

add_translation_provider() (im Modul *annize.i18n*), 85

add_translations() (Methode von *annize.features.i18n.common.ProjectDefinedTranslationProvider*), 70

additional_info() (Methode von *annize.features.licensing.License*), 77

AdHocFsEntry (Klasse in *annize.fs*), 78

AdministrationUtilitiesSection (in Modul *annize.features.distributables.debian*), 26

AFLv3 (in Modul *annize.features.licensing*), 76

AGPLv3 (in Modul *annize.features.licensing*), 76

always_true() (im Modul *annize.callconv*), 14

annize.annize_cli
Modul, 97

annize.aux
Modul, 13

annize.aux._project_infos
Modul, 13

annize.callconv
Modul, 13

annize.callconv.typehints
Modul, 15

annize.context
Modul, 15

annize.data
Modul, 18

annize.data.color
Modul, 18

annize.data.container
Modul, 19

annize.data.unique

Modul, 19

annize.data.version
Modul, 20

annize.engine
Modul, 98

annize.features.authors
Modul, 74

annize.features.base
Modul, 74

annize.features.changelog.common
Modul, 22

annize.features.datainjections.common
Modul, 23

annize.features.datainjections.python
Modul, 23

annize.features.dependencies.common
Modul, 23

annize.features.dependencies.python
Modul, 25

annize.features.distributables.common
Modul, 25

annize.features.distributables.debian
Modul, 26

annize.features.distributables.flatpak
Modul, 40

annize.features.distributables.pythonwheel
Modul, 44

annize.features.distributables.store.pypi
Modul, 25

annize.features.distributables.tar
Modul, 47

annize.features.documentation.common
Modul, 59

annize.features.documentation.sphinx._utils
Modul, 50

annize.features.documentation.sphinx.common
Modul, 51

annize.features.documentation.sphinx.cpp
Modul, 56

annize.features.documentation.sphinx.doxygencompat
Modul, 57

annize.features.documentation.sphinx.javascript
Modul, 58

annize.features.documentation.sphinx.output.common
Modul, 48

annize.features.documentation.sphinx.output.html
Modul, 49

annize.features.documentation.sphinx.output.pdf
Modul, 50

annize.features.documentation.sphinx.output.plaintext
Modul, 50

annize.features.documentation.sphinx.python
Modul, 58

annize.features.documentation.sphinx.rst

Modul, 59
 annize.features.files.common
 Modul, 63
 annize.features.files.transfer.common
 Modul, 61
 annize.features.files.transfer.ssh
 Modul, 62
 annize.features.homepage.common
 Modul, 67
 annize.features.homepage.sections.about
 Modul, 65
 annize.features.homepage.sections.change_log
 Modul, 65
 annize.features.homepage.sections.documentation
 Modul, 65
 annize.features.homepage.sections.download
 Modul, 66
 annize.features.homepage.sections.gallery
 Modul, 66
 annize.features.homepage.sections.imprint
 Modul, 67
 annize.features.homepage.sections.license
 Modul, 67
 annize.features.il8n.common
 Modul, 70
 annize.features.il8n.gettext
 Modul, 71
 annize.features.licensing
 Modul, 76
 annize.features.mediagalleries
 Modul, 77
 annize.features.task
 Modul, 78
 annize.features.testing.common
 Modul, 71
 annize.features.testing.pylint
 Modul, 72
 annize.features.testing.pytest
 Modul, 72
 annize.features.testing.pyunit
 Modul, 72
 annize.features.version
 Modul, 78
 annize.features.versioncontrol.common
 Modul, 72
 annize.features.versioncontrol.git
 Modul, 73
 annize.fs
 Modul, 78
 annize.fs.temp
 Modul, 83
 annize.il8n
 Modul, 83
 annize.project
 Modul, 86
 annize.project.featureloader
 Modul, 93
 annize.project.materializer
 Modul, 89
 annize.project.materializer.behaviors
 Modul, 89
 annize.project.materializer.behaviors.argument
 Modul, 89
 annize.project.materializer.behaviors.basket
 Modul, 90
 annize.project.materializer.behaviors.block
 Modul, 90
 annize.project.materializer.behaviors.featureunava
 Modul, 90
 annize.project.materializer.behaviors.reference
 Modul, 90
 annize.project.materializer.core
 Modul, 91
 annize.project.materializer.preprocessors
 Modul, 92
 annize.project.parser
 Modul, 93
 annize.userfeedback
 Modul, 94
 annize.userfeedback.static
 Modul, 96
 annize.userfeedback.tty
 Modul, 96
 annize_user_interaction_culture() (in
 Modul annize.il8n), 85
 Apache2 (in Modul annize.features.licensing), 76
 ApiReferenceDocument (Klasse in anni-
 ze.features.documentation.sphinx.common),
 51
 ApiReferenceLanguage (Klasse in anni-
 ze.features.documentation.sphinx.common),
 52
 ApiReferenceLanguage.ApiReferenceGenerateInfo
 (Klasse in anni-
 ze.features.documentation.sphinx.common),
 52
 append_child() (Methode von annize.project.Node),
 87
 append_rst() (Methode von anni-
 ze.features.homepage.common.HomepageSection.Product),
 68
 append_to() (annize.project.ArgumentNode proper-
 ty), 86
 ApplicationsAccessibilityCategory (in Mo-
 dul annize.features.distributables.debian), 26
 ApplicationsAmateurradioCategory (in Mo-
 dul annize.features.distributables.debian), 26
 ApplicationsDatamanagementCategory (in

<i>Modul annize.features.distributables.debian),</i> 27	<i>(in Modul annize.features.distributables.debian),</i> 28
ApplicationsEditorsCategory (<i>in Modul annize.features.distributables.debian),</i> 27	ApplicationsScienceEngineeringCategory (<i>in Modul annize.features.distributables.debian),</i> 28
ApplicationsEducationCategory (<i>in Modul annize.features.distributables.debian),</i> 27	ApplicationsScienceGeoscienceCategory (<i>in Modul annize.features.distributables.debian),</i> 28
ApplicationsEmulatorsCategory (<i>in Modul annize.features.distributables.debian),</i> 27	ApplicationsScienceMathematicsCategory (<i>in Modul annize.features.distributables.debian),</i> 28
ApplicationsFilemanagementCategory (<i>in Modul annize.features.distributables.debian),</i> 27	ApplicationsScienceMedicineCategory (<i>in Modul annize.features.distributables.debian),</i> 28
ApplicationsGraphicsCategory (<i>in Modul annize.features.distributables.debian),</i> 27	ApplicationsSciencePhysicsCategory (<i>in Modul annize.features.distributables.debian),</i> 28
ApplicationsMobiledevicesCategory (<i>in Modul annize.features.distributables.debian),</i> 27	ApplicationsScienceSocialCategory (<i>in Modul annize.features.distributables.debian),</i> 28
ApplicationsNetworkCategory (<i>in Modul annize.features.distributables.debian),</i> 27	ApplicationsShellsCategory (<i>in Modul annize.features.distributables.debian),</i> 28
ApplicationsNetworkCommunicationCategory (<i>in Modul annize.features.distributables.debian),</i> 27	ApplicationsSoundsCategory (<i>in Modul annize.features.distributables.debian),</i> 28
ApplicationsNetworkFiletransferCategory (<i>in Modul annize.features.distributables.debian),</i> 27	ApplicationsSystemAdministrationCategory (<i>in Modul annize.features.distributables.debian),</i> 29
ApplicationsNetworkMonitoringCategory (<i>in Modul annize.features.distributables.debian),</i> 27	ApplicationsSystemCategory (<i>in Modul annize.features.distributables.debian),</i> 29
ApplicationsNetworkWebbrowsingCategory (<i>in Modul annize.features.distributables.debian),</i> 27	ApplicationsSystemHardwareCategory (<i>in Modul annize.features.distributables.debian),</i> 29
ApplicationsNetworkWebnewsCategory (<i>in Modul annize.features.distributables.debian),</i> 27	ApplicationsSystemLanguageenvironmentCategory (<i>in Modul annize.features.distributables.debian),</i> 29
ApplicationsOfficeCategory (<i>in Modul annize.features.distributables.debian),</i> 27	ApplicationsSystemMonitoringCategory (<i>in Modul annize.features.distributables.debian),</i> 29
ApplicationsProgrammingCategory (<i>in Modul annize.features.distributables.debian),</i> 27	ApplicationsSystemPackagemanagementCategory (<i>in Modul annize.features.distributables.debian),</i> 29
ApplicationsProjectmanagementCategory (<i>in Modul annize.features.distributables.debian),</i> 28	ApplicationsSystemSecurityCategory (<i>in Modul annize.features.distributables.debian),</i> 29
ApplicationsScienceAstronomyCategory (<i>in Modul annize.features.distributables.debian),</i> 28	ApplicationsTerminalemulatorsCategory (<i>in Modul annize.features.distributables.debian),</i> 29
ApplicationsScienceBiologyCategory (<i>in Modul annize.features.distributables.debian),</i> 28	ApplicationsTextCategory (<i>in Modul annize.features.distributables.debian),</i> 29
ApplicationsScienceCategory (<i>in Modul annize.features.distributables.debian),</i> 28	ApplicationsTvandradiocategory (<i>in Modul annize.features.distributables.debian),</i> 29
ApplicationsScienceChemistryCategory (<i>in Modul annize.features.distributables.debian),</i> 28	ApplicationsVideoCategory (<i>in Modul annize.features.distributables.debian),</i> 29
ApplicationsScienceDataanalysisCategory (<i>in Modul annize.features.distributables.debian),</i> 28	ApplicationsViewersCategory (<i>in Modul annize.features.distributables.debian),</i> 29
ApplicationsScienceElectronicsCategory	

- ApplicationsWebdevelopmentCategory (in Modul *annize.features.distributables.debian*), 29
- architecture (Attribut von *annize.features.distributables.debian.Package._BuildInfo*), 35
- arg_name() (*annize.project.ArgumentNode* property), 86
- ArgparseCommandLineInterfaceDocument (Klasse in *annize.features.documentation.sphinx.common*), 52
- ArgumentBehavior (Klasse in *annize.project.materializer.behaviors.argument*), 89
- ArgumentNode (Klasse in *annize.project*), 86
- Artistic1 (in Modul *annize.features.licensing*), 76
- Artwork (Klasse in *annize.features.dependencies.common*), 23
- attach_media_file() (Methode von *annize.features.homepage.common.HomepageSection.Product*), 68
- ATTRIBUTE_COMMAND_END (Attribut von *annize.project.parser.XmlParser*), 93
- ATTRIBUTE_COMMAND_START (Attribut von *annize.project.parser.XmlParser*), 93
- author (Attribut von *annize.features.distributables.debian.Package._BuildInfo*), 35
- author (Attribut von *annize.features.distributables.pythonwheel.Package._BuildInfo*), 46
- Author (Klasse in *annize.features.authors*), 74
- authorstring (Attribut von *annize.features.distributables.debian.Package._BuildInfo*), 35
- available_languages() (Methode von *annize.features.documentation.common.Document*), 59
- available_languages() (Methode von *annize.features.documentation.sphinx.common.AboutProjectDocument*), 51
- available_languages() (Methode von *annize.features.documentation.sphinx.common.ApiReferenceDocument*), 52
- available_languages() (Methode von *annize.features.documentation.sphinx.common.ArgparseCommandLineInterfaceDocument*), 53
- available_languages() (Methode von *annize.features.documentation.sphinx.common.ReadmeDocument*), 55
- available_languages() (Methode von *annize.features.documentation.sphinx.common.RstDocument*), 56
- B**
- background_image() (*annize.features.documentation.sphinx.output.html.HtmlOutputSpec* property), 49
- BadDisowningError, 78
- BadStructureError, 86
- basename() (Methode von *annize.fs.FsEntry*), 79
- Basket (Klasse in *annize.data.container*), 19
- Basket (Klasse in *annize.features.base*), 74
- BasketBehavior (Klasse in *annize.project.materializer.behaviors.basket*), 90
- Behavior (Klasse in *annize.project.materializer.behaviors*), 89
- BLOCK (Attribut von *annize.project.ScopableBlockNode.Scope*), 88
- BlockBehavior (Klasse in *annize.project.materializer.behaviors.block*), 90
- BlockNode (Klasse in *annize.project*), 86
- blue() (*annize.data.color.Color* property), 18
- brand_color() (im Modul *annize.features.base*), 75
- BrandColor (Klasse in *annize.features.base*), 74
- BSD2clause (in Modul *annize.features.licensing*), 76
- BSD3clause (in Modul *annize.features.licensing*), 76
- BuildVersion (Klasse in *annize.features.versioncontrol.common*), 72
- ByVersionControlSystemCommitMessagesChangelog (Klasse in *annize.features.changelog.common*), 22
- C**
- call() (im Modul *annize.callconv*), 14
- call() (Klassenmethode von *annize.callconv._CallHelper*), 14
- CallableNode (Klasse in *annize.project*), 86
- callee() (*annize.project.CallableNode* property), 86
- Category (Klasse in *annize.features.distributables.debian*), 29
- category() (*annize.features.distributables.debian.MenuEntry* property), 33
- Cc0v1 (in Modul *annize.features.licensing*), 76
- CcBy3 (in Modul *annize.features.licensing*), 76
- CcByNc3 (in Modul *annize.features.licensing*), 76
- CcByNcNd3 (in Modul *annize.features.licensing*), 76
- CcByNcSa3 (in Modul *annize.features.licensing*), 76
- CcByNd3 (in Modul *annize.features.licensing*), 76
- CcBySa3 (in Modul *annize.features.licensing*), 76
- Changelog (Klasse in *annize.features.changelog.common*), 22
- ChDir (Klasse in *annize.fs*), 78
- child_by_relative_path() (Methode von *annize.fs.FsEntry*), 79
- children() (*annize.project.Node* property), 87

children() (Methode von `annize.fs.FsEntry`), 79
choice_dialog() (im Modul `annize.userfeedback`), 95
choice_dialog() (Methode von `annize.userfeedback.NullUserFeedbackController`), 94
choice_dialog() (Methode von `annize.userfeedback.static.StaticUserFeedbackController`), 96
choice_dialog() (Methode von `annize.userfeedback.tty.TtyUserFeedbackController`), 97
choice_dialog() (Methode von `annize.userfeedback.UserFeedbackController`), 94
Cli (Klasse in `annize.annize_cli`), 97
clone() (Methode von `annize.project.Node`), 87
Color (Klasse in `annize.data.color`), 18
command (Attribut von `annize.features.distributables.flatpak.FlatpakImage._BuildInfo`), 41
command() (`annize.features.distributables.debian.MenuEntry` property), 33
comment() (`annize.features.dependencies.common.Dependency` property), 24
CommonVersionPattern (Klasse in `annize.features.version`), 78
CommunicationProgramsSection (in Modul `annize.features.distributables.debian`), 30
CompositeDocument (Klasse in `annize.features.documentation.sphinx.common`), 53
ConcatenatedVersionPatternSegment (Klasse in `annize.data.version`), 20
confdir() (`annize.features.documentation.sphinx.common.Document._GenerateInfo` property), 54
config_files (Attribut von `annize.features.distributables.debian.Package._BuildInfo`), 35
configlines() (`annize.features.documentation.sphinx.common.Document._GenerateInfo` property), 54
configvalues() (`annize.features.documentation.sphinx.common.Document._GenerateInfo` property), 54
Connection (Klasse in `annize.features.distributables.store.pypi`), 25
construct_from_string() (Methode von `annize.callconv.typehints.TypeInfo`), 15
Context (Klasse in `annize.context`), 15
context() (`annize.engine.LoadedProject` property), 98
copy_to() (Methode von `annize.fs.FsEntry`), 79
CppApiReferenceLanguage (Klasse in `annize.features.documentation.sphinx.cpp`), 56
created_time() (Methode von `annize.fs.FsEntry`), 79
culture (Attribut von `annize.features.homepage.common.HomepageSection._GenerateInfo`), 69
Culture (Klasse in `annize.i18n`), 83
current() (im Modul `annize.context`), 17
current_culture() (im Modul `annize.i18n`), 85
customarg (Attribut von `annize.features.homepage.common.HomepageSection._GenerateInfo`), 69
customarg (Attribut von `annize.features.homepage.common.HomepageSection._PrePostprocG`), 69
D
Data (Klasse in `annize.features.base`), 74
DatabasesSection (in Modul `annize.features.distributables.debian`), 30
DefaultImage (Klasse in `annize.features.base`), 75
debian_name() (`annize.features.distributables.debian.Category` property), 30
DefaultUdebPackagesSection (in Modul `annize.features.distributables.debian`), 30
DebugPackagesSection (in Modul `annize.features.distributables.debian`), 30
default_changelog() (im Modul `annize.features.changelog.common`), 22
default_project_authors() (im Modul `annize.features.authors`), 74
default_version_control_system() (im Modul `annize.features.versioncontrol.common`), 73
DefaultVersionPattern() (im Modul `annize.features.version`), 78
DefaultFeatureLoader (Klasse in `annize.project.featureloader`), 93
dependencies_to_rst_text() (im Modul `annize.features.dependencies.common`), 24
Dependency (Klasse in `annize.features.dependencies.common`), 24
description (Attribut von `annize.features.distributables.debian.Package._BuildInfo`), 35
description (Attribut von `annize.features.distributables.pythonwheel.Package._BuildInfo`), 46
description() (`annize.features.distributables.common.Group` property), 26
description() (`annize.features.distributables.flatpak.Group` property), 43

description() (anni-
 ze.features.mediagalleries.Gallery.Item proper-
 ty), 77
 destination() (annize.fs.Mount property), 81
 destination_is_parent() (anni-
 ze.features.files.common.DirPart property),
 63
 destination_path() (anni-
 ze.features.files.common.DirPart property),
 63
 detect_retry_useful_node_context() (Methode von anni-
 ze.project.materializer.behaviors.reference.ReferenceBehavior),
 90
 DevelopmentSection (in Modul anni-
 ze.features.distributables.debian), 30
 Dir (Klasse in annize.features.files.common), 63
 DirPart (Klasse in annize.features.files.common), 63
 disowned() (Methode von annize.fs.FsEntry), 79
 do() (annize.project.OnFeatureUnavailableNode pro-
 perty), 87
 do() (Klassenmethode von annize.annize_cli.Cli), 97
 Document (Klasse in anni-
 ze.features.documentation.common), 59
 Document (Klasse in anni-
 ze.features.documentation.sphinx.common),
 54
 Document.GenerateInfo (Klasse in anni-
 ze.features.documentation.sphinx.common),
 54
 document_root_directory (Attribut von anni-
 ze.features.homepage.common.HomepageSection._PrePostProcGenerateInfo),
 69
 document_root_url (Attribut von anni-
 ze.features.homepage.common.HomepageSection._GenerateInfo), 54
 69
 document_root_url (Attribut von anni-
 ze.features.homepage.common.HomepageSection._PrePostProcGenerateInfo),
 69
 document_variant_directory (Attribut von anni-
 ze.features.homepage.common.HomepageSection._GenerateInfo),
 69
 document_variant_url (Attribut von anni-
 ze.features.homepage.common.HomepageSection._GenerateInfo),
 69
 documentation_source (Attribut von anni-
 ze.features.distributables.debian.Package._BuildInfo),
 35
 DocumentationSection (in Modul anni-
 ze.features.distributables.debian), 30
 does_exclude() (Methode von anni-
 ze.features.files.common.Exclude), 64
 does_exclude() (Methode von anni-
 ze.features.versioncontrol.git.ExcludeByGitIgnores),
 73
 DoxygenSupportedApiReferenceLanguage (Klasse in anni-
 ze.features.documentation.sphinx.doxygencompat),
 57
E
 EditorsSection (in Modul anni-
 ze.features.distributables.debian), 30
 EducationSection (in Modul anni-
 ze.features.distributables.debian), 30
 EmacsSection (in Modul anni-
 ze.features.distributables.debian), 30
 email() (annize.features.authors.Author property), 74
 EmbeddedSoftwareSection (in Modul anni-
 ze.features.distributables.debian), 30
 Endpoint (Klasse in anni-
 ze.features.files.transfer.common), 61
 Endpoint (Klasse in annize.features.files.transfer.ssh),
 62
 english_lang_name() (annize.i18n.Culture pro-
 perty), 83
 entries() (annize.features.changelog.common.ByVersionControlSystem
 property), 22
 entries() (annize.features.changelog.common.Changelog
 property), 22
 Entry (Klasse in annize.features.changelog.common),
 22
 entry_path() (anni-
 ze.features.documentation.common._DocumentGenerateResult
 property), 64
 entry_path() (anni-
 ze.features.documentation.sphinx.common.Document.GenerateIn-
 fo), 54
 entry_path_for_language() (Methode von anni-
 ze.features.documentation.common._DocumentGenerateAllLang-
 uage), 64
 environment (Attribut von anni-
 ze.features.distributables.flatpak.FlatpakImage._BuildInfo),
 41
 EnvironmentVariable (Klasse in anni-
 ze.features.distributables.flatpak), 40
 escape_attribute_string() (Klassenmethode
 von annize.project.parser.XmlParser), 94
 Exclude (Klasse in annize.features.files.common), 64
 ExcludeByGitIgnores (Klasse in anni-
 ze.features.versioncontrol.git), 73
 excludes() (annize.features.files.common.DirPart
 property), 64
 exec() (Methode von anni-
 ze.features.files.transfer.ssh.Endpoint), 62
 executable_links (Attribut von anni-
 ze.features.distributables.debian.Package._BuildInfo),

36
executable_links (Attribut von annize.features.distributables.pythonwheel.Package._BuildInfo), 40
46
ExecutableLink (Klasse in annize.features.distributables.debian), 30
ExecutableLink (Klasse in annize.features.distributables.pythonwheel), 44
exists() (Methode von annize.fs.FsEntry), 79

F

FAIL (Attribut von annize.project.OnFeatureUnavailableNode.Action), 87
FAIL (Attribut von annize.project.ReferenceNode.OnUnresolvableAction), 88
feature() (annize.project.CallableNode property), 86
feature() (annize.project.OnFeatureUnavailableNode property), 87
FeatureLoader (Klasse in annize.project.featureloader), 93
FeatureUnavailableBehavior (Klasse in annize.project.materializer.behaviors.featureunavailable), 90
FeatureUnavailableError, 86
FILE (Attribut von annize.project.ScopableBlockNode.Scope), 88
File (Klasse in annize.features.distributables.common), 25
File (Klasse in annize.features.files.common), 64
file() (annize.features.documentation.common._DocumentGeneratorResult property), 61
file() (annize.features.mediagalleries.Gallery.Item property), 77
filehash() (im Modul annize.features.homepage.sections.download), 66
FileNode (Klasse in annize.project), 86
filesize() (Methode von annize.fs.FsEntry), 79
Filesystem (Klasse in annize.features.distributables.flatpak), 40
filesystems (Attribut von annize.features.distributables.flatpak.FlatpakImage._BuildInfo), 41
find_output_generator_for_outputspec() (im Modul annize.features.documentation.sphinx.output.common), 48
find_project_annize_config_root_file() (im Modul annize.engine), 98
FirstOf (Klasse in annize.features.base), 75
flags() (annize.features.dependencies.common.Kind property), 24

FlatpakImage (Klasse in annize.features.distributables.flatpak), 40
FlatpakImage._BuildInfo (Klasse in annize.features.distributables.flatpak), 40
FlatpakrefFile (Klasse in annize.features.distributables.flatpak), 42
FontsSection (in Modul annize.features.distributables.debian), 30
format() (Methode von annize.i18n.TrStr), 84
formatname() (Methode von annize.features.documentation.sphinx.output.common.OutputGenerator), 48
formatname() (Methode von annize.features.documentation.sphinx.output.html.HtmlOutputGenerator), 49
formatname() (Methode von annize.features.documentation.sphinx.output.pdf.PdfOutputGenerator), 50
formatname() (Methode von annize.features.documentation.sphinx.output.plaintext.PlaintextOutputGenerator), 50
freedesktop_name() (annize.features.distributables.debian.Category property), 30
fresh_temp_dir() (im Modul annize.fs.temp), 83
friendly_filesize() (im Modul annize.features.homepage.sections.download), 66
friendly_name_suggestion() (annize.features.distributables.flatpak.Repository property), 44
from_files() (statische Methode von annize.project.ProjectDefinition), 88
FsEndpoint (Klasse in annize.features.files.transfer.common), 61
FsEntry (Klasse in annize.features.files.common), 64
FsEntry (Klasse in annize.fs), 79
fsentry_by_path() (im Modul annize.fs), 82
fullname() (annize.features.authors.Author property), 74

G

Gallery (Klasse in annize.features.mediagalleries), 77
Gallery.Item (Klasse in annize.features.mediagalleries), 77
GamesActionCategory (in Modul annize.features.distributables.debian), 30
GamesAdventureCategory (in Modul annize.features.distributables.debian), 31
GamesBlocksCategory (in Modul annize.features.distributables.debian), 31
GamesBoardCategory (in Modul annize.features.distributables.debian), 31

GamesCardCategory (in Modul *annize.features.distributables.debian*), 31
 GamesPuzzlesCategory (in Modul *annize.features.distributables.debian*), 31
 GamesSection (in Modul *annize.features.distributables.debian*), 31
 GamesSimulationCategory (in Modul *annize.features.distributables.debian*), 31
 GamesStrategyCategory (in Modul *annize.features.distributables.debian*), 31
 GamesToolsCategory (in Modul *annize.features.distributables.debian*), 31
 GamesToysCategory (in Modul *annize.features.distributables.debian*), 31
 generate() (Methode von *annize.features.documentation.common.Document*), 59
 generate() (Methode von *annize.features.documentation.sphinx.common.Document*), 55
 generate() (Methode von *annize.features.homepage.common.Homepage*), 68
 generate() (Methode von *annize.features.homepage.common.HomepageSection*), 69
 generate_all_languages() (Methode von *annize.features.documentation.common.Document*), 59
 generate_all_languages() (Methode von *annize.features.documentation.sphinx.common.Document*), 55
 generate_master_doc() (Methode von *annize.features.documentation.sphinx.common.ApiReferenceLanguage*), 52
 generate_master_doc() (Methode von *annize.features.documentation.sphinx.common.ApiReferenceLanguage*), 58
 generate_master_doc() (Methode von *annize.features.documentation.sphinx.common.ApiReferenceLanguage*), 59
 GeneratedDocument (Klasse in *annize.features.documentation.common*), 59
 GeneratedHomepage (Klasse in *annize.features.homepage.common*), 67
 GenerateMOs (Klasse in *annize.features.i18ngettext*), 71
 get_arguments_infos() (im Modul *annize.callconv.typehints*), 15
 get_commit_message() (Methode von *annize.features.versioncontrol.common.VersionControlSystem*), 72
 get_commit_message() (Methode von *annize.features.versioncontrol.git.VersionControlSystem*), 73
 annize 73
 get_commit_time() (Methode von *annize.features.versioncontrol.common.VersionControlSystem*), 73
 get_commit_time() (Methode von *annize.features.versioncontrol.git.VersionControlSystem*), 73
 get_culture() (im Modul *annize.i18n*), 85
 get_current_revision() (Methode von *annize.features.versioncontrol.common.VersionControlSystem*), 73
 get_current_revision() (Methode von *annize.features.versioncontrol.git.VersionControlSystem*), 73
 get_description() (Methode von *annize.project.Node*), 87
 get_files() (Methode von *annize.features.distributables.common.Group*), 26
 get_files() (Methode von *annize.features.distributables.flatpak.Group*), 43
 get_from_iso_639_1_lang_code() (statische Methode von *annize.i18n.Culture*), 83
 get_materialized() (Methode von *annize.project.materializer.core.ProjectMaterializer*), 92
 get_materialized_children() (Methode von *annize.project.materializer.core.NodeMaterialization*), 91
 get_materialized_children_tuples() (Methode von *annize.project.materializer.core.NodeMaterialization*), 91
 get_object_metadata() (im Modul *annize.features.distributables.common.PackageStore*), 26
 get_object_metadata() (Methode von *annize.context.Context*), 16
 get_package_history() (Methode von *annize.features.distributables.common.PackageStore*), 26
 get_parameters_from_childnodes_infos() (im Modul *annize.callconv*), 14
 get_parser() (im Modul *annize.annize_cli*), 98
 get_revision_list() (Methode von *annize.features.versioncontrol.common.VersionControlSystem*), 73
 get_revision_list() (Methode von *annize.features.versioncontrol.git.VersionControlSystem*), 73

get_revision_number() (Methode von *annize.features.versioncontrol.common.VersionControlSystem*), *ze.features.distributables.pythonwheel.Package._BuildInfo*),
 73 46
 get_revision_number() (Methode von *annize.features.versioncontrol.git.VersionControlSystem*), *ze.features.homepage.common*), 67
 73
 get_typeinfo() (im Modul *annize.callconv.typehints*), 15
 GettextTranslationProvider (Klasse in *annize.i18n*), 83
 GnomeSection (in Modul *annize.features.distributables.debian*), 31
 GnuLinux (Klasse in *annize.features.dependencies.common*), 24
 GnuRSection (in Modul *annize.features.distributables.debian*), 31
 GnustepSection (in Modul *annize.features.distributables.debian*), 31
 GpgFile (Klasse in *annize.features.distributables.flatpak*), 42
 GPLv3 (in Modul *annize.features.licensing*), 76
 GraphicsSection (in Modul *annize.features.distributables.debian*), 31
 green() (*annize.data.color.Color* property), 18
 Group (Klasse in *annize.features.distributables.common*), 25
 Group (Klasse in *annize.features.distributables.flatpak*), 43
 guess_project_root_directory() (im Modul *annize.engine*), 98

H
 HamRadioSection (in Modul *annize.features.distributables.debian*), 31
 has_result() (*annize.project.materializer.core.NodeMaterialization* property), 91
 has_shell_access() (*annize.features.files.transfer.ssh.Endpoint* property), 62
 HaskellSection (in Modul *annize.features.distributables.debian*), 32
 head() (*annize.features.homepage.common.HomepageSection* property), 69
 heading() (*annize.features.documentation.sphinx.common.ApiReferenceLanguage.ApiReferenceGenerateInfo* property), 52
 heading() (statische Methode von *annize.features.documentation.sphinx.rst.RstGenerator*), 59
 HelpCategory (in Modul *annize.features.distributables.debian*), 32
 homepage (Attribut von *annize.features.distributables.debian.Package._BuildInfo*), 36
 homepage (Attribut von *annize.features.homepage.common.Homepage* (Klasse in *annize.features.homepage.common*), 67
 homepage_url() (*annize.features.base.Data* property), 74
 homepage_url() (im Modul *annize.features.base*), 75
 HomepageSection (Klasse in *annize.features.homepage.common*), 68
 HomepageSection._GenerateInfo (Klasse in *annize.features.homepage.common*), 68
 HomepageSection._PrePostprocGenerateInfo (Klasse in *annize.features.homepage.common*), 69
 HomepageSection.Product (Klasse in *annize.features.homepage.common*), 68
 host() (*annize.features.files.transfer.ssh.Endpoint* property), 62
 html_color_spec() (*annize.data.color.Color* property), 19
 HtmlOutputGenerator (Klasse in *annize.features.documentation.sphinx.output.html*), 49
 HtmlOutputSpec (Klasse in *annize.features.documentation.common*), 60
 HtmlOutputSpec (Klasse in *annize.features.documentation.sphinx.output.html*), 49

I
 icon() (*annize.features.dependencies.common.Dependency* property), 24
 icon() (*annize.features.distributables.debian.MenuEntry* property), 33
 identity_file() (*annize.features.files.transfer.ssh.Endpoint* property), 62
 if_passing (Attribut von *annize.callconv.ParametersFromChildNodesInfos*), 14
 if_typehint_matches (Attribut von *annize.callconv.ParametersFromChildNodesInfos*), 14
 import_reference() (*annize.features.dependencies.common.Dependency* property), 24
 importance() (*annize.features.dependencies.common.Kind* property), 24
 imprint() (*annize.features.base.Data* property), 74
 imprint() (im Modul *annize.features.base*), 75
 info_file() (Methode von *annize.project.parser._XmlParser._Context*), 93

[in_node\(\)](#) (Methode von [annize.project.parser.XmlParser.Context](#)), 93
[Included](#) (Klasse in [annize.features.dependencies.common](#)), 24
[IndependentTrStr](#) (Klasse in [annize.i18n](#)), 84
[Inject](#) (Klasse in [annize.features.datainjections.common](#)), 23
[inject\(\)](#) (Methode von [annize.features.datainjections.common.Injection](#)), 23
[inject\(\)](#) (Methode von [annize.features.datainjections.python.Injection](#)), 23
[Injection](#) (Klasse in [annize.features.datainjections.common](#)), 23
[Injection](#) (Klasse in [annize.features.datainjections.python](#)), 23
[inner\(\)](#) ([annize.callconv.typehints.ListTypeInfo](#) property), 15
[input_dialog\(\)](#) (im Modul [annize.userfeedback](#)), 95
[input_dialog\(\)](#) (Methode von [annize.userfeedback.NullUserFeedbackController](#)), 94
[input_dialog\(\)](#) (Methode von [annize.userfeedback.static.StaticUserFeedbackController](#)), 96
[input_dialog\(\)](#) (Methode von [annize.userfeedback.tty.TtyUserFeedbackController](#)), 97
[input_dialog\(\)](#) (Methode von [annize.userfeedback.UserFeedbackController](#)), 95
[insert_child\(\)](#) (Methode von [annize.project.Node](#)), 87
[InterpretersSection](#) (in Modul [annize.features.distributables.debian](#)), 32
[IntrospectionSection](#) (in Modul [annize.features.distributables.debian](#)), 32
[intstruct\(\)](#) ([annize.features.documentation.sphinx.common.Document.GenerateInfo](#) property), 54
[is_compatible_for\(\)](#) (Klassenmethode von [annize.features.documentation.sphinx.output.common.OutputGenerator](#)), 48
[is_compatible_for\(\)](#) (Klassenmethode von [annize.features.documentation.sphinx.output.html.HtmlOutputGenerator](#)), 49
[is_compatible_for\(\)](#) (Klassenmethode von [annize.features.documentation.sphinx.output.pdf.PdfOutputGenerator](#)), 50
[is_compatible_for\(\)](#) (Klassenmethode von [annize.features.documentation.sphinx.output.plaintext.PlaintextOutputGenerator](#)), 50
[is_constructable_from_string\(\)](#) ([annize.callconv.typehints.TypeInfo](#) property), 15
[is_dir\(\)](#) (Methode von [annize.fs.FsEntry](#)), 79
[is_file\(\)](#) (Methode von [annize.fs.FsEntry](#)), 79
[is_friendly_name\(\)](#) (im Modul [annize.context](#)), 17
[is_friendly_name\(\)](#) (Methode von [annize.context.Context](#)), 16
[is_gui\(\)](#) ([annize.features.distributables.debian.MenuEntry](#) property), 33
[is_gui\(\)](#) ([annize.features.distributables.pythonwheel.ExecutableLink](#) property), 44
[is_homepage\(\)](#) ([annize.features.documentation.common.HtmlOutputSpec](#) property), 60
[is_link\(\)](#) (Methode von [annize.fs.FsEntry](#)), 79
[is_ownable\(\)](#) ([annize.fs.FsEntry](#) property), 80
[is_owned\(\)](#) ([annize.fs.FsEntry](#) property), 80
[is_toplevel_object\(\)](#) (im Modul [annize.context](#)), 17
[is_toplevel_object\(\)](#) (Methode von [annize.context.Context](#)), 16
[iso_639_1_lang_code\(\)](#) ([annize.i18n.Culture](#) property), 83
[Item](#) (Klasse in [annize.features.changelog.common](#)), 22
[items\(\)](#) ([annize.features.changelog.common.Entry](#) property), 22
[items\(\)](#) ([annize.features.mediagalleries.Gallery](#) property), 77
J
[JavaScriptApiReferenceLanguage](#) (Klasse in [annize.features.documentation.sphinx.javascript](#)), 58
[JavascriptSection](#) (in Modul [annize.features.distributables.debian](#)), 32
[JavaSection](#) (in Modul [annize.features.distributables.debian](#)), 32
K
[KdeSection](#) (im Modul [annize.features.distributables.debian](#)), 32
[KernelsSection](#) (in Modul [annize.features.distributables.debian](#)), 32
[Keyword](#) (Klasse in [annize.features.base](#)), 75
[keywords](#) (Attribut von [annize.features.distributables.pythonwheel.Package._BuildInfo](#)), 46
[Keywords](#) (Klasse in [annize.features.base](#)), 75
[keywords\(\)](#) ([annize.features.base.Keywords](#) property), 75
[Kind](#) (Klasse in [annize.features.dependencies.common](#)), 24
[kind\(\)](#) ([annize.features.dependencies.common.Dependency](#) property), 24

kitversion (Attribut von annize.features.distributables.flatpak.FlatpakImage._BuildInfo), 49
 ze.features.distributables.flatpak.FlatpakImage._BuildInfo (Attribut von annize.features.distributables.flatpak.FlatpakImage), 41
 ze.features.distributables.pythonwheel.Package._BuildInfo (Attribut von annize.features.distributables.pythonwheel.Package), 46
L
 long_description() (annize.features.base.Data property), 74
 label() (annize.features.dependencies.common.Dependency property), 24
 long_description() (im Modul annize.features.base), 75
 label() (annize.features.dependencies.common.Kind property), 24
 long_str() (annize.data.unique.UniqueId property), 19
 language() (annize.features.documentation.sphinx.common.Document.GenerateInfo property), 54
 language() (annize.features.documentation.sphinx.common.RstDocumentVariant property), 56
M
 MachineRootDirectory (Klasse in annize.features.files.common), 65
 LanguagePacksSection (in Modul annize.features.distributables.debian), 32
 MailSection (in Modul annize.features.distributables.debian), 32
 languages() (annize.features.documentation.common.Document.GenerateInfo property), 61
 main() (im Modul annize.annize_cli), 98
 languages() (annize.features.homepage.common.HomepagePageObjectAsToplevel() (Methode von annize.context.Context), 16
 masterlink() (annize.features.documentation.sphinx.output.html.HtmlOutputSpec property), 49
 LGPLv3 (in Modul annize.features.licensing), 76
 LibrariesSection (in Modul annize.features.distributables.debian), 32
 materialize() (im Modul annize.project.materializer), 89
 LibraryDevelopmentSection (in Modul annize.features.distributables.debian), 32
 MaterializerError, 86
 license (Attribut von annize.features.distributables.pythonwheel.Package._BuildInfo), 46
 License (Klasse in annize.features.licensing), 76
 LicenseSection (in Modul annize.features.distributables.debian), 32
 license_name (Attribut von annize.features.distributables.debian.Package._BuildInfo), 36
 media_files() (annize.features.homepage.common.HomepageSection.Product property), 68
 menu_entries (Attribut von annize.features.distributables.flatpak.FlatpakImage._BuildInfo), 44
 Line (Klasse in annize.features.version), 78
 linkname() (annize.features.distributables.pythonwheel.ExecutableLink property), 44
 menuentries (Attribut von annize.features.distributables.debian.Package._BuildInfo), 36
 LispSection (in Modul annize.features.distributables.debian), 32
 MenuEntry (Klasse in annize.features.distributables.debian), 32
 ListTypeInfo (Klasse in annize.callconv.typehints), 15
 MenuEntry (Klasse in annize.features.distributables.flatpak), 43
 load_feature() (Methode von annize.project.featureloader.DefaultFeatureLoader), 93
 message_dialog() (im Modul annize.userfeedback), 96
 load_feature() (Methode von annize.project.featureloader.FeatureLoader), 93
 message_dialog() (Methode von annize.userfeedback.NullUserFeedbackController), 94
 load_project() (im Modul annize.engine), 98
 message_dialog() (Methode von annize.userfeedback.static.StaticUserFeedbackController), 96
 load_project_definition() (im Modul annize.engine), 98
 message_dialog() (Methode von annize.userfeedback.tty.TtyUserFeedbackController), 97
 LoadedProject (Klasse in annize.engine), 98
 message_dialog() (Methode von annize.userfeedback.UserFeedbackController), 95
 LocalRepository (Klasse in annize.features.distributables.flatpak), 43
 locationstring() (Methode von annize.features.files.transfer.ssh.Endpoint), 62
 logo_image() (annize.features.documentation.sphinx.output.html.HtmlOutputSpec property), 95

MetaPackagesSection (in Modul *annize.features.distributables.debian*), 33
 methodname () (in Modul *annize.features.distributables.pythonwheel.ExecutableLink* property), 44
 MiscellaneousSection (in Modul *annize.features.distributables.debian*), 33
 MissingOwnershipError, 81
 MIT (in Modul *annize.features.licensing*), 77
 mkdir () (Methode von *annize.fs.FsEntry*), 80
 modified_time () (Methode von *annize.fs.FsEntry*), 80
 Modul
 annize.annize_cli, 97
 annize.aux, 13
 annize.aux._project_infos, 13
 annize.callconv, 13
 annize.callconv.typehints, 15
 annize.context, 15
 annize.data, 18
 annize.data.color, 18
 annize.data.container, 19
 annize.data.unique, 19
 annize.data.version, 20
 annize.engine, 98
 annize.features.authors, 74
 annize.features.base, 74
 annize.features.changelog.common, 22
 annize.features.datainjections.common, 23
 annize.features.datainjections.python, 23
 annize.features.dependencies.common, 23
 annize.features.dependencies.python, 25
 annize.features.distributables.common, 25
 annize.features.distributables.debian, 26
 annize.features.distributables.flatpak, 40
 annize.features.distributables.pythonwheel, 44
 annize.features.distributables.store.pyproject.toml, 25
 annize.features.distributables.tar, 47
 annize.features.documentation.common, 59
 annize.features.documentation.sphinx._utils, 50
 annize.features.documentation.sphinx.common, 51
 annize.features.documentation.sphinx.cpp, 56
 annize.features.documentation.sphinx.doxygencomment, 57
 annize.features.documentation.sphinx.javascript, 58
 annize.features.documentation.sphinx.output.common, 48
 annize.features.documentation.sphinx.output.html, 49
 annize.features.documentation.sphinx.output.pdf, 50
 annize.features.documentation.sphinx.output.plain, 50
 annize.features.documentation.sphinx.python, 58
 annize.features.documentation.sphinx.rst, 59
 annize.features.files.common, 63
 annize.features.files.transfer.common, 61
 annize.features.files.transfer.ssh, 62
 annize.features.homepage.common, 67
 annize.features.homepage.sections.about, 65
 annize.features.homepage.sections.changelog, 65
 annize.features.homepage.sections.documentation, 65
 annize.features.homepage.sections.download, 66
 annize.features.homepage.sections.gallery, 66
 annize.features.homepage.sections.imprint, 67
 annize.features.homepage.sections.license, 67
 annize.features.il8n.common, 70
 annize.features.il8n.gettext, 71
 annize.features.licensing, 76
 annize.features.mediagalleries, 77
 annize.features.task, 78
 annize.features.testing.common, 71
 annize.features.testing.pylint, 72
 annize.features.testing.pytest, 72
 annize.features.testing.pyunit, 72
 annize.features.version, 78
 annize.features.versioncontrol.common, 72
 annize.features.versioncontrol.git, 73
 annize.fs, 78
 annize.fs.temp, 83

annize.i18n, 13
 annize.project, 86
 annize.project.featureloader, 93
 annize.project.materializer, 89
 annize.project.materializer.behaviors, 89
 annize.project.materializer.behaviors.argument, 89
 annize.project.materializer.behaviors.basket, 90
 annize.project.materializer.behaviors.block, 90
 annize.project.materializer.behaviors.featureunavailable, 90
 annize.project.materializer.behaviors.reference, 90
 annize.project.materializer.core, 91
 annize.project.materializer.preprocessors, 92
 annize.project.parser, 93
 annize.userfeedback, 94
 annize.userfeedback.static, 96
 annize.userfeedback.tty, 96
 modulename() (annize.features.distributables.pythonwheel.ExecutableLink property), 44
 MonoCliSection (in Modul annize.features.distributables.debian), 33
 Mount (Klasse in annize.fs), 81
 move_to() (Methode von annize.fs.FsEntry), 80
 MPLv11 (in Modul annize.features.licensing), 77
 MPLv2 (in Modul annize.features.licensing), 77
 multilanguage_frame() (Methode von annize.features.documentation.sphinx.output.common.OutputGenerator), 48
 multilanguage_frame() (Methode von annize.features.documentation.sphinx.output.html.HtmlOutputGenerator), 49
 multiple (Attribut von annize.callconv.ParametersFromChildNodesInfos), 14
 MultipleValuesForSingleArgumentError, 13
N
 name (Attribut von annize.features.distributables.debian.Package_BuildInfo), 36
 name (Attribut von annize.features.distributables.flatpak.FlatpakImage_BuildInfo), 41
 name (Attribut von annize.features.distributables.pythonwheel.Package_BuildInfo), 46
 name() (annize.callconv.typehints.TypeInfo property), 15
 name() (annize.features.distributables.debian.ExecutableLink property), 30
 name() (annize.features.distributables.debian.MenuEntry property), 33
 name() (annize.features.distributables.debian.Section property), 38
 name() (annize.features.licensing.License property), 77
 name() (annize.project.ArgumentNode property), 86
 NetworkSection (in Modul annize.features.distributables.debian), 33
 NestedImportUnavailable (in Modul annize.features.distributables.debian), 33
 NoReferentCultureError, 84
 Node (Klasse in annize.project), 86
 node() (annize.project.materializer.core.NodeMaterialization property), 91
 node_context() (Methode von annize.project.materializer.behaviors.argument.ArgumentBehavior), 90
 node_context() (Methode von annize.project.materializer.behaviors.basket.BasketBehavior), 90
 node_context() (Methode von annize.project.materializer.behaviors.Behavior), 89
 node_context() (Methode von annize.project.materializer.behaviors.block.BlockBehavior), 90
 node_context() (Methode von annize.project.materializer.behaviors.featureunavailable.FeatureUnavailableBehavior), 90
 node_context() (Methode von annize.project.materializer.behaviors.reference.ReferenceBehavior), 90
 node_context() (Methode von annize.project.materializer.behaviors.reference.SkipUnresolvableReferenceBehavior), 91
 NodeMaterialization (Klasse in annize.project.materializer.core), 91
 nodes() (annize.project.ProjectDefinition property), 88
 normalize_blockscopes() (im Modul annize.project.materializer.preprocessors), 92
 NullUserFeedbackController (Klasse in annize.userfeedback), 94
 NumericVersionPatternSegment (Klasse in annize.data.version), 20
O
 object_by_name() (im Modul annize.context), 17
 object_by_name() (Methode von annize.context.Context), 16
 object_name() (im Modul annize.context), 18

object_name() (Methode von *annize.context.Context*), 16
 object_names() (im Modul *annize.context*), 18
 object_names() (Methode von *annize.context.Context*), 16
 objects_by_type() (im Modul *annize.context*), 18
 objects_by_type() (Methode von *annize.context.Context*), 16
 OcamlSection (in Modul *annize.features.distributables.debian*), 33
 OldLibrariesSection (in Modul *annize.features.distributables.debian*), 33
 on_unresolvable() (*annize.project.ReferenceNode* property), 88
 OnFeatureUnavailableNode (Klasse in *annize.project*), 87
 OnFeatureUnavailableNode.Action (Klasse in *annize.project*), 87
 OptionalVersionPatternSegment (Klasse in *annize.data.version*), 21
 OriginalOwnershipImpossibleError, 81
 OtherOSsAndFSsSection (in Modul *annize.features.distributables.debian*), 33
 outdir() (*annize.features.documentation.sphinx.common.Documentation.sphinx.output.pdf* property), 54
 OutOfContextError, 17
 OutputGenerator (Klasse in *annize.features.documentation.sphinx.output.common*), 48
 OutputSpec (Klasse in *annize.features.documentation.common*), 60
 outputspec() (*annize.features.documentation.sphinx.output.common* property), 48
 owned() (Methode von *annize.fs.FsEntry*), 80

P

Package (Klasse in *annize.features.distributables.debian*), 33
 Package (Klasse in *annize.features.distributables.pythonwheel*), 44
 Package (Klasse in *annize.features.distributables.tar*), 47
 Package._BuildInfo (Klasse in *annize.features.distributables.debian*), 34
 Package._BuildInfo (Klasse in *annize.features.distributables.pythonwheel*), 45
 PackageStore (Klasse in *annize.features.distributables.common*), 26
 packagestore() (*annize.features.distributables.common.Group* property), 26
 parameter_from_childnodes() (im Modul *annize.callconv*), 14
 parameter_name (Attribut von *annize.callconv.ParametersFromChildNodesInfos*), 14
 ParametersFromChildNodesInfos (Klasse in *annize.callconv*), 13
 parent() (*annize.project.Node* property), 87
 parent() (Methode von *annize.fs.FsEntry*), 80
 parse() (im Modul *annize.project.parser*), 94
 parse_file() (Methode von *annize.project.parser.XmlParser*), 94
 ParserError, 87
 partname() (*annize.data.version.NumericVersionPatternSegment* property), 20
 path() (*annize.features.distributables.debian.ExecutableLink* property), 30
 path() (*annize.features.files.common.DirPart* property), 64
 path() (*annize.features.versioncontrol.git.VersionControlSystem* property), 73
 path() (*annize.project.FileNode* property), 86
 path() (Methode von *annize.fs.FsEntry*), 80
 pattern() (*annize.data.version.Version* property), 21
 PdfOutputGenerator (Klasse in *annize.features.documentation.sphinx.output.pdf*), 50
 PdfOutputSpec (Klasse in *annize.features.documentation.common*), 60
 PerlSection (in Modul *annize.features.distributables.debian*), 37
 PhpSection (in Modul *annize.features.distributables.debian*), 38
 PinoHomepageAuxInfo (Klasse in *annize.features.homepage.common*), 70
 pkgpath_debian (Attribut von *annize.features.distributables.debian.Package._BuildInfo*), 36
 pkgpath_documentation (Attribut von *annize.features.distributables.debian.Package._BuildInfo*), 36
 pkgpath_pixmap (Attribut von *annize.features.distributables.debian.Package._BuildInfo*), 36
 pkgpath_setuppy (Attribut von *annize.features.distributables.pythonwheel.Package._BuildInfo*), 46
 pkgpath_share (Attribut von *annize.features.distributables.flatpak.FlatpakImage._BuildInfo*), 41
 pkgpath_share_applications (Attribut von *annize.features.distributables.flatpak.FlatpakImage._BuildInfo*), 41
 pkgpath_share_icons (Attribut von *annize.features.distributables.flatpak.FlatpakImage._BuildInfo*), 41

41
 pkgpath_usrbin (Attribut von anni- 49
 ze.features.distributables.debian.Package._BuildInfo),
 36
 pkgrootpath (Attribut von anni- 59
 ze.features.distributables.debian.Package._BuildInfo),
 36
 pkgrootpath (Attribut von anni- 69
 ze.features.distributables.flatpak.FlatpakImage._BuildInfo),
 41
 pkgrootpath (Attribut von anni- 70
 ze.features.distributables.pythonwheel.Package._BuildInfo),
 46
 pkgsize (Attribut von anni- 66
 ze.features.distributables.debian.Package._BuildInfo),
 36
 PlaintextOutputGenerator (Klasse in anni- 66
 ze.features.documentation.sphinx.output.plaintext),
 50
 PlaintextOutputSpec (Klasse in anni- 66
 ze.features.documentation.common), 60
 platform (Attribut von anni- ze.features.distributables.debian.Package._BuildInfo),
 ze.features.distributables.flatpak.FlatpakImage._BuildInfo), 36
 41
 port() (annize.features.files.transfer.ssh.Endpoint pro- ze.features.base.Data property), 74
 perty), 62
 pretty_project_name() (im Modul anni-
 postinst (Attribut von anni- ze.features.base), 75
 ze.features.distributables.debian.Package._BuildInfo),
 36
 processonly_long_str() (anni-
 ze.data.unique.UniqueId property), 19
 postproc() (Methode von anni- processonly_short_str() (anni-
 ze.features.documentation.sphinx.output.common.OutputGenerator), 19
 48
 PROJECT (Attribut von anni-
 postproc() (Methode von anni- ze.project.ScopableBlockNode.Scope), 88
 ze.features.documentation.sphinx.output.pdf.PdfOutputGenerator),
 50
 ze.annize_cli.Cli), 98
 postproc_generate() (Methode von anni- project_directory() (annize.features.base.Data
 ze.features.homepage.common.HomepageSection), property), 75
 69
 project_directory() (im Modul anni-
 postproc_generate() (Methode von anni- ze.features.base), 75
 ze.features.homepage.sections.download.Section),
 66
 project_keywords() (im Modul anni-
 ze.features.base), 76
 prepare_generate() (Methode von anni- project_licenses() (im Modul anni-
 ze.features.documentation.sphinx.common.ApiReferenceLanguage), 77
 52
 project_name() (annize.features.base.Data proper-
 prepare_generate() (Methode von anni- ty), 75
 ze.features.documentation.sphinx.doxygencompat.DoxygenSupport(ApiReferenceLanguage), 76
 58
 project_versions() (im Modul anni-
 prepare_generate() (Methode von anni- ze.features.version), 78
 ze.features.documentation.sphinx.javascript.JavaScriptApiReferenceLanguage),
 58
 ScriptApiReferenceLanguage),
 prepare_generate() (Methode von anni- I18nLocalizationProvider (Klas-
 ze.features.documentation.sphinx.output.common.OutputGenerator), 65
 48
 ProjectDefinition (Klasse in annize.project), 88
 ProjectMaterializer (Klasse in anni-

`ze.project.materializer.core`), 91
ProvidedTrStr (Klasse in *annize.i18n*), 84
`public_url()` (*annize.features.distributables.flatpak.Repository* property), 44
PublicDomain (in Modul *annize.features.licensing*), 77
Python (Klasse in *annize.features.dependencies.python*), 25
Python3ApiReferenceLanguage (Klasse in *annize.features.documentation.sphinx.python*), 58
PythonSection (in Modul *annize.features.distributables.debian*), 38

R

`read_file()` (Methode von *annize.fs.FsEntry*), 80
`readable_path()` (Methode von *annize.fs._SimpleFsEntry*), 81
`readable_path()` (Methode von *annize.fs.AdHocFsEntry*), 78
`readable_path()` (Methode von *annize.fs.FsEntry*), 80
`readable_path()` (Methode von *annize.fs.temp._TempDir*), 83
ReadmeDocument (Klasse in *annize.features.documentation.sphinx.common*), 55
Recommended (Klasse in *annize.features.dependencies.common*), 24
`red()` (*annize.data.color.Color* property), 19
`reference_key()` (*annize.project.ReferenceNode* property), 88
ReferenceBehavior (Klasse in *annize.project.materializer.behaviors.reference*), 90
ReferenceNode (Klasse in *annize.project*), 88
ReferenceNode.OnUnresolvableAction (Klasse in *annize.project*), 88
`regex_string()` (Methode von *annize.data.version.AbstractVersionPatternSegment*), 20
`regex_string()` (Methode von *annize.data.version.ConcatenatedVersionPatternSegment*), 20
`regex_string()` (Methode von *annize.data.version.NumericVersionPatternSegment*), 20
`regex_string()` (Methode von *annize.data.version.OptionalVersionPatternSegment*), 21
`regex_string()` (Methode von *annize.data.version.SeparatorVersionPatternSegment*), 21

`register_output_generator()` (in Modul *annize.features.documentation.sphinx.output.common*), 48
`relative_path()` (*annize.features.files.common.FsEntry* property), 64
`remove()` (Methode von *annize.fs.FsEntry*), 80
`remove_child()` (Methode von *annize.project.Node*), 87
Repository (Klasse in *annize.features.distributables.flatpak*), 44
Required (Klasse in *annize.features.dependencies.common*), 24
`resolve_appendtonodes()` (in Modul *annize.project.materializer.preprocessors*), 92
`resolved_type()` (*annize.callconv.typehints.TypeInfo* property), 15
`resolver_node_context()` (Methode von *annize.project.materializer.behaviors.reference.ReferenceBehavior*), 90
`restrict_path()` (Methode von *annize.fs.temp._TempDir*), 83
`result` (Attribut von *annize.features.distributables.debian.Package._BuildInfo*), 36
`result` (Attribut von *annize.features.distributables.flatpak.FlatpakImage._BuildInfo*), 41
`result` (Attribut von *annize.features.distributables.pythonwheel.Package._BuildInfo*), 46
`result()` (*annize.project.materializer.core.NodeMaterialization* property), 91
`resultpath()` (Methode von *annize.features.documentation.sphinx.output.pdf.PdfOutputGenerator*), 50
`retry_node_context()` (Methode von *annize.project.materializer.behaviors.reference.ReferenceBehavior*), 90
`root()` (*annize.features.files.common.DirPart* property), 64
`root()` (*annize.features.files.common.FsEntry* property), 65
`rst_text()` (*annize.features.homepage.common.HomepageSection.Prod* property), 68
RstDocument (Klasse in *annize.features.documentation.sphinx.common*), 55
RstDocumentVariant (Klasse in *annize.features.documentation.sphinx.common*), 56
RstGenerator (Klasse in *annize.features.documentation.sphinx.rst*), 59

RubySection (in Modul *annize.features.distributables.debian*), 38

run() (Methode von *annize.features.testing.common.Test*), 71

run() (Methode von *annize.features.testing.common.TestGroup*), 71

run() (Methode von *annize.features.testing.pylint.Test*), 72

run() (Methode von *annize.features.testing.pytest.Test*), 72

run() (Methode von *annize.features.testing.pyunit.Test*), 72

RunTests (Klasse in *annize.features.testing.common*), 71

RustSection (in Modul *annize.features.distributables.debian*), 38

S

ScalarValueNode (Klasse in *annize.project*), 88

scalehue() (Methode von *annize.data.color.Color*), 19

ScienceSection (in Modul *annize.features.distributables.debian*), 38

ScopableBlockNode (Klasse in *annize.project*), 88

ScopableBlockNode.Scope (Klasse in *annize.project*), 88

scope() (*annize.project.ScopableBlockNode* property), 89

ScreenLockingCategory (in Modul *annize.features.distributables.debian*), 38

ScreenSavingCategory (in Modul *annize.features.distributables.debian*), 38

sdk (Attribut von *annize.features.distributables.flatpak.FlatpakImage_BuildInfo*), 41

section (Attribut von *annize.features.distributables.debian.Package_BuildInfo*), 36

Section (Klasse in *annize.features.distributables.debian*), 38

Section (Klasse in *annize.features.homepage.sections.about*), 65

Section (Klasse in *annize.features.homepage.sections.changelog*), 65

Section (Klasse in *annize.features.homepage.sections.documentation*), 65

Section (Klasse in *annize.features.homepage.sections.download*), 66

Section (Klasse in *annize.features.homepage.sections.gallery*), 66

Section (Klasse in *annize.features.homepage.sections.imprint*), 67

Section (Klasse in *annize.features.homepage.sections.license*), 67

sections() (*annize.features.homepage.common.Homepage* property), 68

segment_names() (*annize.data.version.AbstractVersionPatternSegment* property), 20

segment_names() (*annize.data.version.ConcatenatedVersionPatternSegment* property), 20

segment_names() (*annize.data.version.NumericVersionPatternSegment* property), 20

segment_names() (*annize.data.version.OptionalVersionPatternSegment* property), 21

segment_names() (*annize.data.version.VersionPattern* property), 21

segments() (*annize.data.version.VersionPattern* property), 21

segments_tuples() (*annize.data.version.Version* property), 21

segments_tuples() (*annize.features.versioncontrol.common.BuildVersion* property), 72

segments_tuples_to_text() (Methode von *annize.data.version.AbstractVersionPatternSegment*), 20

segments_tuples_to_text() (Methode von *annize.data.version.ConcatenatedVersionPatternSegment*), 20

segments_tuples_to_text() (Methode von *annize.data.version.NumericVersionPatternSegment*), 20

segments_tuples_to_text() (Methode von *annize.data.version.OptionalVersionPatternSegment*), 21

segments_tuples_to_text() (Methode von *annize.data.version.SeparatorVersionPatternSegment*), 21

segments_tuples_to_text() (Methode von *annize.data.version.VersionPattern*), 21

segments_values() (*annize.data.version.Version* property), 21

SeparatorVersionPatternSegment (Klasse in *annize.data.version*), 21

serialize_for_confpy() (in Modul *annize.features.documentation.sphinx._utils*), 50

ServiceDescription (Klasse in *annize.features.distributables.debian*), 38

services (Attribut von *annize*)

`ze.features.distributables.debian.Package._BuildInfo`),
 36
`set_materialized_result()` (Methode von `anni-ze.project.materializer.core.NodeMaterialization`),
 91
`set_object_metadata()` (in Modul `anni-ze.context`), 18
`set_object_metadata()` (Methode von `anni-ze.context.Context`), 17
`set_object_name()` (in Modul `anni-ze.context`), 18
`set_object_name()` (Methode von `anni-ze.context.Context`), 17
`setuppy_conf` (Attribut von `anni-ze.features.distributables.pythonwheel.Package._BuildInfo`), 46
`Share` (Klasse in `anni-ze.features.distributables.flatpak`), 44
`shares` (Attribut von `anni-ze.features.distributables.flatpak.FlatpakImage._BuildInfo`), 41
`ShellsSection` (in Modul `anni-ze.features.distributables.debian`), 38
`short_desc()` (in `anni-ze.features.documentation.sphinx.output.html.HtmlOutputSpec`), 50
`short_desc()` (in `anni-ze.features.homepage.common.Homepage`), 68
`short_str()` (`anni-ze.data.unique.UniqueId` property), 19
`short_title()` (in `anni-ze.features.documentation.sphinx.output.html.HtmlOutputSpec`), 50
`SimpleProjectHomepage` (Klasse in `anni-ze.features.homepage.common`), 70
`SKIP` (Attribut von `anni-ze.project.ReferenceNode.OnUnresolvableAction`), 88
`SKIP_BLOCK` (Attribut von `anni-ze.project.OnFeatureUnavailableNode.Action`), 87
`SKIP_NODE` (Attribut von `anni-ze.project.OnFeatureUnavailableNode.Action`), 87
`SkipUnresolvableReferenceBehavior` (Klasse in `anni-ze.project.materializer.behaviors.reference`), 90
`sockets` (Attribut von `anni-ze.features.distributables.flatpak.FlatpakImage._BuildInfo`), 41
`sortindex()` (`anni-ze.features.homepage.common.HomepageSection` property), 70
`SoundSection` (in Modul `anni-ze.features.distributables.debian`), 38
`source` (Attribut von `anni-ze.features.distributables.debian.Package._BuildInfo`), 36
`source` (Attribut von `anni-ze.features.distributables.flatpak.FlatpakImage._BuildInfo`), 41
`source` (Attribut von `anni-ze.features.distributables.pythonwheel.Package._BuildInfo`), 46
`source()` (`anni-ze.features.documentation.sphinx.common.ApiReference` property), 52
`source()` (`anni-ze.features.documentation.sphinx.common.RstDocumentV`), 56
`StaticUserFeedbackController` (Klasse in `anni-ze.userfeedback.static`), 96
`stop_materialization_here_and_try_again()` (Methode von `anni-ze.project.materializer.behaviors.Behavior`), 89
`store_package()` (Methode von `anni-ze.features.distributables.common.PackageStore`), 26
`SummarySpec` (Klasse in `anni-ze.features.il8n.common`), 71
`summary` (Attribut von `anni-ze.features.distributables.debian.Package._BuildInfo`), 36
`summary()` (`anni-ze.features.base.Data` property), 75
`summary()` (in Modul `anni-ze.features.base`), 76

T

`Task` (Klasse in `anni-ze.features.task`), 78
`TasksSection` (in Modul `anni-ze.features.distributables.debian`), 38
`temp_clone()` (Methode von `anni-ze.fs.FsEntry`), 80
`TempFileByContent` (Klasse in `anni-ze.fs.temp`), 83
`Test` (Klasse in `anni-ze.features.testing.common`), 71
`Test` (Klasse in `anni-ze.features.testing.pylint`), 72
`Test` (Klasse in `anni-ze.features.testing.pytest`), 72
`Test` (Klasse in `anni-ze.features.testing.pyunit`), 72
`TestGroup` (Klasse in `anni-ze.features.testing.common`), 71
`TexSection` (in Modul `anni-ze.features.distributables.debian`), 38
`text()` (`anni-ze.data.version.Version` property), 21
`text()` (`anni-ze.features.changelog.common.Item` property), 22
`text()` (`anni-ze.features.licensing.License` property), 77
`text()` (`anni-ze.features.versioncontrol.common.BuildVersion` property), 72
`text_to_segments_tuples()` (Methode von `anni-ze.data.version.VersionPattern`), 22
`TextProcessingSection` (in Modul `anni-ze.features.distributables.debian`), 39

theme() (*annize.features.documentation.sphinx.output.html.HtmlOutputSpec* property), 50
 time() (*annize.features.changelog.common.Entry* property), 22
 title() (*annize.features.distributables.common.Group* property), 26
 title() (*annize.features.distributables.debian.MenuEntry* property), 33
 title() (*annize.features.documentation.common.HtmlOutputSpec* property), 60
 title() (*annize.features.documentation.sphinx.common.ReadmeDocument* property), 55
 title() (*annize.features.homepage.common.Homepage* property), 68
 title() (*annize.features.mediagalleries.Gallery* property), 77
 TODO (*Klasse in annize.i18n*), 84
 token() (*annize.features.distributables.store.pypi.Connection* property), 25
 tr() (*im Modul annize.i18n*), 85
 tr() (*statische Methode von annize.i18n.TrStr*), 84
 tr_if_trstr() (*im Modul annize.i18n*), 85
 transfer_action_copy() (*Methode von annize.fs._TransferHelper*), 82
 transfer_action_move() (*Methode von annize.fs._TransferHelper*), 82
 transfer_to() (*Methode von annize.fs._TransferHelper*), 82
 transformed() (*Methode von annize.data.color.Color*), 19
 translate() (*Methode von annize.features.i18n.common.ProjectDefinedTranslationProvider*), 71
 translate() (*Methode von annize.i18n.GettextTranslationProvider*), 83
 translate() (*Methode von annize.i18n.IndependentTrStr*), 84
 translate() (*Methode von annize.i18n.ProvidedTrStr*), 84
 translate() (*Methode von annize.i18n.TODO*), 84
 translate() (*Methode von annize.i18n.TranslationProvider*), 84
 translate() (*Methode von annize.i18n.TrStr*), 84
 translation_providers() (*im Modul annize.i18n*), 85
 TranslationProvider (*Klasse in annize.i18n*), 84
 TranslationUnavailableError, 85
 TrStr (*Klasse in annize.i18n*), 84
 TrStrOrStr (*in Modul annize.i18n*), 84
 try_get_materialization_for_node() (*Methode von annize.project.materializer.core.NodeMaterialization*), 91
 TryAgain, 92
 UniqueId (*Klasse in annize.data.unique*), 19
 UnresolvableReferenceError, 89
 UnsatisfiableUserFeedbackAttemptError, 94
 UpdatePOs (*Klasse in annize.features.i18n.gettext*), 71
 Upload (*Klasse in annize.features.distributables.store.pypi*), 25
 Upload (*Klasse in annize.features.files.transfer.common*), 62
 upload() (*Methode von annize.features.distributables.flatpak.LocalRepository*), 43
 upload() (*Methode von annize.features.distributables.flatpak.Repository*), 44
 UserFeedbackController (*Klasse in annize.userfeedback*), 94
 username() (*annize.features.files.transfer.ssh.Endpoint* property), 62
 UtilitiesSection (*in Modul annize.features.distributables.debian*), 39
 value() (*annize.project.ScalarValueNode* property), 88
 version (*Attribut von annize.features.distributables.debian.Package._BuildInfo*), 46
 version (*Attribut von annize.features.distributables.pythonwheel.Package._BuildInfo*), 46
 Version (*Klasse in annize.data.version*), 21
 Version (*Klasse in annize.features.version*), 78
 version() (*annize.features.changelog.common.Entry* property), 22
 version() (*annize.features.version.Line* property), 78
 VersionControlSystem (*Klasse in annize.features.versioncontrol.common*), 72
 VersionControlSystem (*Klasse in annize.features.versioncontrol.git*), 73
 VersionControlSystemsSection (*in Modul annize.features.distributables.debian*), 39
 VersionPattern (*Klasse in annize.data.version*), 21
 VideoSection (*in Modul annize.features.distributables.debian*), 39
 VirtualPackagesSection (*in Modul annize.features.distributables.debian*), 39
 WebServersSection (*in Modul annize*), 39

[ze.features.distributables.debian](#)), 39
WebSoftwareSection (in Modul [anni-](#)
[ze.features.distributables.debian](#)), 39
write_file() (Methode von [annize.fs.FsEntry](#)), 80

X

XfceSection (in Modul [anni-](#)
[ze.features.distributables.debian](#)), 39
XWindowSystemSoftwareSection (in Modul [an-](#)
[nize.features.distributables.debian](#)), 39

Z

ZopePloneFrameworkSection (in Modul [anni-](#)
[ze.features.distributables.debian](#)), 39