

Contents

1	Anise Manual	1
1.1	License	1
1.2	About	1
1.3	Maturity	2
1.4	User Documentation	2
1.4.1	How to use anise	2
1.4.1.1	What is a 'project description' file?	2
1.4.1.2	Executing tasks	3
1.4.1.3	Technical details about the anatomy of anise	3
1.4.1.4	Technical details	5
1.4.1.5	The command line options	8
1.5	Packages	8
1.6	Class Hierarchy	10
1.7	Class List	11
1.8	anise Namespace Reference	13
1.9	anise.data Namespace Reference	14
1.9.1	Detailed Description	14
1.10	anise.data.icons Namespace Reference	14
1.10.1	Detailed Description	14
1.11	anise.data.licenses Namespace Reference	14
1.11.1	Detailed Description	14
1.12	anise.features Namespace Reference	14
1.12.1	Detailed Description	15
1.13	anise.features.build Namespace Reference	15
1.13.1	Detailed Description	16
1.14	anise.features.build.qmake Namespace Reference	16
1.14.1	Detailed Description	16
1.14.2	Function Documentation	16
1.14.2.1	basedependencies	16
1.15	anise.features.datainjections Namespace Reference	16
1.15.1	Detailed Description	17

1.15.2	Function Documentation	17
1.15.2.1	<code>_initproject</code>	17
1.15.2.2	<code>_sync</code>	17
1.15.3	Variable Documentation	17
1.15.3.1	<code>name</code>	17
1.16	<code>anise.features.dependencies</code> Namespace Reference	17
1.16.1	Detailed Description	17
1.16.2	Function Documentation	18
1.16.2.1	<code>_add_homepage_section</code>	18
1.16.2.2	<code>_initproject</code>	18
1.17	<code>anise.features.dependencies.python</code> Namespace Reference	18
1.17.1	Detailed Description	18
1.17.2	Variable Documentation	18
1.17.2.1	<code>_dependenciesfeature</code>	18
1.18	<code>anise.features.distributables</code> Namespace Reference	18
1.18.1	Detailed Description	19
1.18.2	Function Documentation	19
1.18.2.1	<code>_add_homepage_section</code>	19
1.18.2.2	<code>_initproject</code>	19
1.19	<code>anise.features.documentation</code> Namespace Reference	19
1.19.1	Detailed Description	20
1.19.2	Function Documentation	20
1.19.2.1	<code>_add_homepage_section</code>	20
1.19.2.2	<code>_add_release_task_for_export_to_source</code>	20
1.19.2.3	<code>_compose_readmes</code>	20
1.19.2.4	<code>_initproject</code>	20
1.19.2.5	<code>add_anisedoc_source</code>	20
1.19.2.6	<code>has_doxygen_developerdocumentation</code>	20
1.19.3	Variable Documentation	20
1.19.3.1	<code>markdown_downgrade_headings</code>	20
1.20	<code>anise.features.filesystem</code> Namespace Reference	20
1.20.1	Detailed Description	20
1.21	<code>anise.features.homepage</code> Namespace Reference	21
1.21.1	Detailed Description	21
1.21.2	Function Documentation	21
1.21.2.1	<code>_basic_homepage_sections</code>	21
1.21.2.2	<code>_initproject</code>	21
1.21.3	Variable Documentation	21
1.21.3.1	<code>HOOK_UPLOAD_HOMEPAGE_COLLECTLINKS</code>	21
1.22	<code>anise.features.imagegalleries</code> Namespace Reference	21

1.22.1	Detailed Description	22
1.22.2	Function Documentation	22
1.22.2.1	<code>_add_homepage_section</code>	22
1.22.2.2	<code>_initproject</code>	22
1.23	<code>anise.features.licensing</code> Namespace Reference	22
1.23.1	Detailed Description	23
1.23.2	Function Documentation	23
1.23.2.1	<code>_add_homepage_section</code>	23
1.23.2.2	<code>_instantiatelicense</code>	23
1.23.2.3	<code>_writelicense</code>	23
1.23.3	Variable Documentation	23
1.23.3.1	<code>_packagesfeature</code>	23
1.24	<code>anise.features.packages</code> Namespace Reference	23
1.24.1	Detailed Description	24
1.24.2	Function Documentation	24
1.24.2.1	<code>_initproject</code>	24
1.24.3	Variable Documentation	24
1.24.3.1	<code>HOOK_ENRICHRAWPACKAGE</code>	24
1.25	<code>anise.features.packages.android</code> Namespace Reference	24
1.25.1	Detailed Description	24
1.26	<code>anise.features.packages.debian</code> Namespace Reference	24
1.26.1	Detailed Description	25
1.27	<code>anise.features.packages.firefox</code> Namespace Reference	25
1.27.1	Detailed Description	25
1.28	<code>anise.features.packages.python</code> Namespace Reference	25
1.28.1	Detailed Description	25
1.28.2	Function Documentation	25
1.28.2.1	<code>_nsis_get_global</code>	25
1.29	<code>anise.features.packages.win32</code> Namespace Reference	25
1.29.1	Detailed Description	26
1.29.2	Variable Documentation	26
1.29.2.1	<code>HOOK_WINDOWS_PACKAGE_NSIS_GLOBAL</code>	26
1.29.2.2	<code>HOOK_WINDOWS_PACKAGE_NSIS_ONINIT</code>	26
1.29.2.3	<code>HOOK_WINDOWS_PACKAGE_NSIS_ONINSTALL</code>	26
1.30	<code>anise.features.projectdata</code> Namespace Reference	26
1.30.1	Detailed Description	26
1.30.2	Function Documentation	27
1.30.2.1	<code>_beautify_longdescription</code>	27
1.30.2.2	<code>_initproject</code>	27
1.31	<code>anise.features.releasing</code> Namespace Reference	27

1.31.1 Detailed Description	27
1.31.2 Function Documentation	27
1.31.2.1 _initproject	27
1.31.3 Variable Documentation	27
1.31.3.1 HOOK_RELEASE	27
1.32 anise.features.sshfiletransfer Namespace Reference	27
1.32.1 Detailed Description	28
1.32.2 Variable Documentation	28
1.32.2.1 _homepagefeature	28
1.33 anise.features.versioncontrol Namespace Reference	28
1.33.1 Detailed Description	28
1.34 anise.features.versioncontrol.svn Namespace Reference	28
1.34.1 Detailed Description	28
1.34.2 Function Documentation	29
1.34.2.1 is_svn_controlled	29
1.35 anise.framework Namespace Reference	29
1.35.1 Detailed Description	29
1.36 anise.framework.base Namespace Reference	29
1.36.1 Detailed Description	29
1.37 anise.framework.engine Namespace Reference	30
1.37.1 Detailed Description	30
1.37.2 Function Documentation	30
1.37.2.1 _cmdline	30
1.37.2.2 execute	30
1.37.2.3 extractfilesfromretval	30
1.37.2.4 main	30
1.38 anise.framework.features Namespace Reference	31
1.38.1 Detailed Description	31
1.38.2 Function Documentation	31
1.38.2.1 hook	31
1.38.2.2 loadfeature	31
1.38.3 Variable Documentation	31
1.38.3.1 _features	31
1.38.3.2 _featuresources	31
1.38.3.3 _hooks	31
1.38.3.4 HOOK_BEFORE_DEFINITION	31
1.38.3.5 HOOK_BEFORE_EXECUTION	32
1.39 anise.framework.globalvars Namespace Reference	32
1.39.1 Detailed Description	32
1.39.2 Variable Documentation	32

1.39.2.1	logdebug	32
1.39.2.2	removetempdirs	32
1.40	anise.framework.imports Namespace Reference	32
1.40.1	Detailed Description	32
1.41	anise.framework.projectinformations Namespace Reference	32
1.41.1	Variable Documentation	32
1.41.1.1	homepage	32
1.41.1.2	version	32
1.42	anise.framework.projects Namespace Reference	32
1.42.1	Detailed Description	33
1.42.2	Function Documentation	33
1.42.2.1	currentproject	33
1.42.2.2	setcurrentproject	33
1.42.3	Variable Documentation	33
1.42.3.1	_currentproject	33
1.43	anise.framework.results Namespace Reference	33
1.43.1	Detailed Description	34
1.43.2	Function Documentation	34
1.43.2.1	source_to_filestructure	34
1.44	anise.utils Namespace Reference	34
1.44.1	Detailed Description	34
1.45	anise.utils.basic Namespace Reference	34
1.45.1	Detailed Description	35
1.45.2	Function Documentation	35
1.45.2.1	call	35
1.45.2.2	createobjectstructure	35
1.45.2.3	dict2object	35
1.45.2.4	getuniqueidentifier	35
1.45.2.5	linewrap	35
1.45.2.6	maketarball	35
1.45.2.7	makedirs	35
1.45.2.8	params2dict	36
1.45.2.9	readfromfile	36
1.45.2.10	stripname	36
1.45.2.11	writetofile	36
1.46	anise.utils.documentation Namespace Reference	36
1.46.1	Detailed Description	37
1.46.2	Function Documentation	37
1.46.2.1	_derivecolors	37
1.46.2.2	_scalecolor	37

1.46.2.3	generate_doxygen_developerdocumentation_ashtml	37
1.46.2.4	generate_doxygen_documentation	37
1.46.2.5	generate_doxygen_homepage	38
1.46.2.6	generate_doxygen_userdocumentation_ashtml	38
1.46.2.7	generate_doxygen_userdocumentation_asmanpage	39
1.46.2.8	generate_doxygen_userdocumentation_aspdf	39
1.46.2.9	generate_doxygen_userdocumentation_asplaintext	40
1.46.2.10	markdown_downgrade_headings	41
1.46.3	Variable Documentation	41
1.46.3.1	_doxygen_additionalcss	41
1.46.3.2	_doxygen_additionalcss_homepage	41
1.46.3.3	_doxygen_additionalcss_userdocumentation	41
1.46.3.4	_doxygen_baseconfig	41
1.46.3.5	_doxygen_htmlfooter	41
1.46.3.6	_doxygen_htmlheader	42
1.47	anise.utils.exceptions Namespace Reference	42
1.47.1	Detailed Description	42
1.48	anise.utils.logging Namespace Reference	42
1.48.1	Detailed Description	43
1.48.2	Function Documentation	43
1.48.2.1	log_start_stop	43
1.48.2.2	logdebug	43
1.48.2.3	logerror	43
1.48.2.4	logexception	43
1.48.2.5	loginfo	43
1.48.2.6	logmessage	44
1.48.2.7	logwarning	44
1.49	anise.utils.packaging Namespace Reference	44
1.49.1	Detailed Description	45
1.49.2	Function Documentation	45
1.49.2.1	_jsdatafile	45
1.49.2.2	buildandroidpackage_with_ant	45
1.49.2.3	builddebpackage	45
1.49.2.4	buildfirefoxaddonpackage	46
1.49.2.5	buildpythonwheelpackage	46
1.49.2.6	buildwindowsinstallerpackage	46
1.50	anise.utils.ssh Namespace Reference	47
1.50.1	Detailed Description	47
1.50.2	Function Documentation	47
1.50.2.1	call	47

1.50.2.2	copy	47
1.51	anise.utils.svn Namespace Reference	47
1.51.1	Detailed Description	48
1.51.2	Function Documentation	48
1.51.2.1	getsvnrevision	48
1.51.2.2	listsvnrepo	48
1.52	anise.features.documentation._Exports Class Reference	48
1.52.1	Constructor & Destructor Documentation	48
1.52.1.1	__init__	48
1.52.2	Member Function Documentation	48
1.52.2.1	add	48
1.52.3	Member Data Documentation	49
1.52.3.1	list	49
1.53	anise.features.datainjections._Pool Class Reference	49
1.53.1	Member Function Documentation	49
1.53.1.1	add	49
1.54	anise.features.dependencies._Pool Class Reference	49
1.54.1	Constructor & Destructor Documentation	50
1.54.1.1	__init__	50
1.54.2	Member Function Documentation	50
1.54.2.1	add	50
1.54.3	Member Data Documentation	50
1.54.3.1	list	50
1.55	anise.features.documentation._Pool Class Reference	50
1.55.1	Constructor & Destructor Documentation	50
1.55.1.1	__init__	50
1.55.2	Member Function Documentation	50
1.55.2.1	add	50
1.55.3	Member Data Documentation	50
1.55.3.1	list	50
1.56	anise.features.distributables._Pool Class Reference	51
1.56.1	Constructor & Destructor Documentation	51
1.56.1.1	__init__	51
1.56.2	Member Function Documentation	51
1.56.2.1	addgroup	51
1.56.3	Member Data Documentation	51
1.56.3.1	list	51
1.57	anise.features.imagegalleries._Pool Class Reference	51
1.57.1	Constructor & Destructor Documentation	51
1.57.1.1	__init__	51

1.57.2	Member Function Documentation	51
1.57.2.1	add	51
1.57.3	Member Data Documentation	51
1.57.3.1	list	51
1.58	anise.features.releasing._Releasetasks Class Reference	52
1.58.1	Member Function Documentation	52
1.58.1.1	add	52
1.59	anise.features.homepage._Sections Class Reference	52
1.59.1	Constructor & Destructor Documentation	52
1.59.1.1	__init__	52
1.59.2	Member Function Documentation	52
1.59.2.1	add	52
1.59.3	Member Data Documentation	52
1.59.3.1	list	52
1.60	anise.utils.svn._Version Class Reference	53
1.60.1	Constructor & Destructor Documentation	53
1.60.1.1	__init__	53
1.60.2	Member Function Documentation	53
1.60.2.1	format	53
1.60.2.2	isvalid	53
1.60.3	Member Data Documentation	53
1.60.3.1	build	53
1.60.3.2	tag	53
1.61	anise.features.licensing.AFLv3 Class Reference	53
1.61.1	Constructor & Destructor Documentation	54
1.61.1.1	__init__	54
1.61.2	Member Function Documentation	54
1.61.2.1	getfilename	54
1.61.2.2	getlicensetext	54
1.61.3	Member Data Documentation	54
1.61.3.1	filename	54
1.61.3.2	name	54
1.61.3.3	namedebian	54
1.61.3.4	namefirefox	54
1.61.3.5	namepythonsetuputils	54
1.62	anise.features.licensing.AGPLv3 Class Reference	54
1.62.1	Constructor & Destructor Documentation	55
1.62.1.1	__init__	55
1.62.2	Member Function Documentation	55
1.62.2.1	getfilename	55

1.62.2.2	getlicensetext	55
1.62.3	Member Data Documentation	55
1.62.3.1	filename	55
1.62.3.2	name	55
1.62.3.3	namedebian	55
1.62.3.4	namefirefox	55
1.62.3.5	namepythonsetuputils	55
1.63	anise.utils.exceptions.AniseError Class Reference	55
1.63.1	Detailed Description	56
1.63.2	Member Function Documentation	56
1.63.2.1	__call__	56
1.64	anise.features.licensing.Apache2 Class Reference	56
1.64.1	Constructor & Destructor Documentation	56
1.64.1.1	__init__	56
1.64.2	Member Function Documentation	56
1.64.2.1	getfilename	56
1.64.2.2	getlicensetext	57
1.64.3	Member Data Documentation	57
1.64.3.1	filename	57
1.64.3.2	name	57
1.64.3.3	namedebian	57
1.64.3.4	namefirefox	57
1.64.3.5	namepythonsetuputils	57
1.65	anise.features.licensing.Artistic1 Class Reference	57
1.65.1	Constructor & Destructor Documentation	57
1.65.1.1	__init__	57
1.65.2	Member Function Documentation	58
1.65.2.1	getfilename	58
1.65.2.2	getlicensetext	58
1.65.3	Member Data Documentation	58
1.65.3.1	filename	58
1.65.3.2	name	58
1.65.3.3	namedebian	58
1.65.3.4	namefirefox	58
1.65.3.5	namepythonsetuputils	58
1.66	anise.utils.exceptions.BadCommunicationError Class Reference	58
1.66.1	Detailed Description	58
1.66.2	Member Function Documentation	58
1.66.2.1	__call__	59
1.67	anise.utils.exceptions.BadFormatError Class Reference	59

1.67.1 Detailed Description	59
1.67.2 Member Function Documentation	59
1.67.2.1 __call__	59
1.68 anise.utils.exceptions.BadInputError Class Reference	59
1.68.1 Detailed Description	60
1.68.2 Member Function Documentation	60
1.68.2.1 __call__	60
1.69 anise.features.licensing.BaseLicense Class Reference	60
1.69.1 Detailed Description	61
1.69.2 Constructor & Destructor Documentation	61
1.69.2.1 __init__	61
1.69.3 Member Function Documentation	61
1.69.3.1 getfilename	61
1.69.3.2 getlicensetext	61
1.69.4 Member Data Documentation	61
1.69.4.1 filename	61
1.69.4.2 name	61
1.69.4.3 namedebian	61
1.69.4.4 namefirefox	61
1.69.4.5 namepythonsetuputils	61
1.70 anise.features.licensing.BSD2clause Class Reference	62
1.70.1 Constructor & Destructor Documentation	62
1.70.1.1 __init__	62
1.70.2 Member Function Documentation	62
1.70.2.1 getfilename	62
1.70.2.2 getlicensetext	62
1.70.3 Member Data Documentation	62
1.70.3.1 filename	62
1.70.3.2 name	62
1.70.3.3 namedebian	62
1.70.3.4 namefirefox	62
1.70.3.5 namepythonsetuputils	63
1.71 anise.features.licensing.BSD3clause Class Reference	63
1.71.1 Constructor & Destructor Documentation	63
1.71.1.1 __init__	63
1.71.2 Member Function Documentation	63
1.71.2.1 getfilename	63
1.71.2.2 getlicensetext	63
1.71.3 Member Data Documentation	64
1.71.3.1 filename	64

1.71.3.2	name	64
1.71.3.3	namedebian	64
1.71.3.4	namefirefox	64
1.71.3.5	namepythonsetuputils	64
1.72	anise.features.packages.debian.Category Class Reference	64
1.72.1	Detailed Description	65
1.72.2	Member Data Documentation	65
1.72.2.1	ApplicationsAccessibility	65
1.72.2.2	ApplicationsAmateurradio	65
1.72.2.3	ApplicationsDatamanagement	65
1.72.2.4	ApplicationsEditors	65
1.72.2.5	ApplicationsEducation	65
1.72.2.6	ApplicationsEmulators	65
1.72.2.7	ApplicationsFilemanagement	65
1.72.2.8	ApplicationsGraphics	66
1.72.2.9	ApplicationsMobiledevices	66
1.72.2.10	ApplicationsNetwork	66
1.72.2.11	ApplicationsNetworkCommunication	66
1.72.2.12	ApplicationsNetworkFiletransfer	66
1.72.2.13	ApplicationsNetworkMonitoring	66
1.72.2.14	ApplicationsNetworkWebbrowsing	66
1.72.2.15	ApplicationsNetworkWebnews	66
1.72.2.16	ApplicationsOffice	66
1.72.2.17	ApplicationsProgramming	66
1.72.2.18	ApplicationsProjectmanagement	66
1.72.2.19	ApplicationsScience	66
1.72.2.20	ApplicationsScienceAstronomy	66
1.72.2.21	ApplicationsScienceBiology	66
1.72.2.22	ApplicationsScienceChemistry	66
1.72.2.23	ApplicationsScienceDataanalysis	66
1.72.2.24	ApplicationsScienceElectronics	66
1.72.2.25	ApplicationsScienceEngineering	66
1.72.2.26	ApplicationsScienceGeoscience	66
1.72.2.27	ApplicationsScienceMathematics	67
1.72.2.28	ApplicationsScienceMedicine	67
1.72.2.29	ApplicationsSciencePhysics	67
1.72.2.30	ApplicationsScienceSocial	67
1.72.2.31	ApplicationsShells	67
1.72.2.32	ApplicationsSounds	67
1.72.2.33	ApplicationsSystem	67

1.72.2.34 ApplicationsSystemAdministration	67
1.72.2.35 ApplicationsSystemHardware	67
1.72.2.36 ApplicationsSystemLanguageenvironment	67
1.72.2.37 ApplicationsSystemMonitoring	67
1.72.2.38 ApplicationsSystemPackagemanagement	67
1.72.2.39 ApplicationsSystemSecurity	67
1.72.2.40 ApplicationsTerminalemulators	67
1.72.2.41 ApplicationsText	67
1.72.2.42 ApplicationsTvandrado	67
1.72.2.43 ApplicationsVideo	67
1.72.2.44 ApplicationsViewers	67
1.72.2.45 ApplicationsWebdevelopment	67
1.72.2.46 GamesAction	67
1.72.2.47 GamesAdventure	68
1.72.2.48 GamesBlocks	68
1.72.2.49 GamesBoard	68
1.72.2.50 GamesCard	68
1.72.2.51 GamesPuzzles	68
1.72.2.52 GamesSimulation	68
1.72.2.53 GamesStrategy	68
1.72.2.54 GamesTools	68
1.72.2.55 GamesToys	68
1.72.2.56 Help	68
1.72.2.57 ScreenLocking	68
1.72.2.58 ScreenSaving	68
1.73 anise.features.licensing.Cc0v1 Class Reference	68
1.73.1 Constructor & Destructor Documentation	69
1.73.1.1 __init__	69
1.73.2 Member Function Documentation	69
1.73.2.1 getfilename	69
1.73.2.2 getlicensetext	69
1.73.3 Member Data Documentation	69
1.73.3.1 filename	69
1.73.3.2 name	69
1.73.3.3 namedebian	69
1.73.3.4 namefirefox	69
1.73.3.5 namepythonsetuputils	69
1.74 anise.features.licensing.CcBy3 Class Reference	69
1.74.1 Constructor & Destructor Documentation	70
1.74.1.1 __init__	70

1.74.2	Member Function Documentation	70
1.74.2.1	getfilename	70
1.74.2.2	getlicensetext	70
1.74.3	Member Data Documentation	70
1.74.3.1	filename	70
1.74.3.2	name	70
1.74.3.3	namedebian	70
1.74.3.4	namefirefox	70
1.74.3.5	namepythonsetuputils	70
1.75	anise.features.licensing.CcByNc3 Class Reference	70
1.75.1	Constructor & Destructor Documentation	71
1.75.1.1	__init__	71
1.75.2	Member Function Documentation	71
1.75.2.1	getfilename	71
1.75.2.2	getlicensetext	71
1.75.3	Member Data Documentation	71
1.75.3.1	filename	71
1.75.3.2	name	71
1.75.3.3	namedebian	71
1.75.3.4	namefirefox	71
1.75.3.5	namepythonsetuputils	71
1.76	anise.features.licensing.CcByNcNd3 Class Reference	71
1.76.1	Constructor & Destructor Documentation	72
1.76.1.1	__init__	72
1.76.2	Member Function Documentation	72
1.76.2.1	getfilename	72
1.76.2.2	getlicensetext	72
1.76.3	Member Data Documentation	72
1.76.3.1	filename	72
1.76.3.2	name	72
1.76.3.3	namedebian	72
1.76.3.4	namefirefox	72
1.76.3.5	namepythonsetuputils	72
1.77	anise.features.licensing.CcByNcSa3 Class Reference	72
1.77.1	Constructor & Destructor Documentation	73
1.77.1.1	__init__	73
1.77.2	Member Function Documentation	73
1.77.2.1	getfilename	73
1.77.2.2	getlicensetext	73
1.77.3	Member Data Documentation	73

1.77.3.1	filename	73
1.77.3.2	name	73
1.77.3.3	namedebian	73
1.77.3.4	namefirefox	73
1.77.3.5	namepythonsetuputils	73
1.78	anise.features.licensing.CcByNd3 Class Reference	74
1.78.1	Constructor & Destructor Documentation	74
1.78.1.1	__init__	74
1.78.2	Member Function Documentation	74
1.78.2.1	getfilename	74
1.78.2.2	getlicensetext	74
1.78.3	Member Data Documentation	74
1.78.3.1	filename	74
1.78.3.2	name	74
1.78.3.3	namedebian	74
1.78.3.4	namefirefox	74
1.78.3.5	namepythonsetuputils	75
1.79	anise.features.licensing.CcBySa3 Class Reference	75
1.79.1	Constructor & Destructor Documentation	75
1.79.1.1	__init__	75
1.79.2	Member Function Documentation	75
1.79.2.1	getfilename	75
1.79.2.2	getlicensetext	75
1.79.3	Member Data Documentation	76
1.79.3.1	filename	76
1.79.3.2	name	76
1.79.3.3	namedebian	76
1.79.3.4	namefirefox	76
1.79.3.5	namepythonsetuputils	76
1.80	anise.utils.basic.ChDir Class Reference	76
1.80.1	Detailed Description	76
1.80.2	Constructor & Destructor Documentation	76
1.80.2.1	__init__	76
1.80.3	Member Function Documentation	76
1.80.3.1	__enter__	76
1.80.3.2	__exit__	76
1.80.4	Member Data Documentation	76
1.80.4.1	newpath	76
1.80.4.2	oldpath	76
1.81	anise.utils.packaging.DebianPackageMenuEntry Class Reference	77

1.81.1 Detailed Description	77
1.81.2 Constructor & Destructor Documentation	77
1.81.2.1 <code>__init__</code>	77
1.81.3 Member Data Documentation	77
1.81.3.1 <code>category</code>	77
1.81.3.2 <code>command</code>	77
1.81.3.3 <code>gui</code>	77
1.81.3.4 <code>icon</code>	78
1.81.3.5 <code>name</code>	78
1.81.3.6 <code>title</code>	78
1.82 <code>anise.features.dependencies.Dependency Class Reference</code>	78
1.82.1 Detailed Description	78
1.82.2 Constructor & Destructor Documentation	78
1.82.2.1 <code>__init__</code>	78
1.82.3 Member Data Documentation	78
1.82.3.1 <code>comment</code>	78
1.82.3.2 <code>displayname</code>	78
1.82.3.3 <code>icon</code>	78
1.82.3.4 <code>objectname</code>	78
1.82.3.5 <code>type</code>	78
1.82.3.6 <code>visible</code>	78
1.83 <code>anise.features.dependencies.python.DjangoDependency Class Reference</code>	79
1.83.1 Constructor & Destructor Documentation	79
1.83.1.1 <code>__init__</code>	79
1.84 <code>anise.features.documentation.Export Class Reference</code>	79
1.84.1 Detailed Description	79
1.84.2 Constructor & Destructor Documentation	79
1.84.2.1 <code>__init__</code>	79
1.84.3 Member Function Documentation	79
1.84.3.1 <code>supports</code>	79
1.85 <code>anise.framework.features.Feature Class Reference</code>	80
1.85.1 Detailed Description	80
1.85.2 Constructor & Destructor Documentation	80
1.85.2.1 <code>__init__</code>	80
1.85.3 Member Data Documentation	80
1.85.3.1 <code>featuremodule</code>	80
1.85.3.2 <code>hooks</code>	80
1.85.3.3 <code>name</code>	80
1.86 <code>anise.features.distributables.FileGroup Class Reference</code>	80
1.86.1 Detailed Description	81

1.86.2	Constructor & Destructor Documentation	81
1.86.2.1	__init__	81
1.86.3	Member Data Documentation	81
1.86.3.1	buildparams	81
1.86.3.2	description	81
1.86.3.3	linkto	81
1.86.3.4	name	81
1.86.3.5	task	81
1.87	anise.framework.results.Filestructure Class Reference	81
1.87.1	Detailed Description	82
1.87.2	Constructor & Destructor Documentation	82
1.87.2.1	__init__	82
1.87.3	Member Function Documentation	82
1.87.3.1	dl	82
1.87.3.2	with_modified_rootname	82
1.87.4	Member Data Documentation	82
1.87.4.1	path	82
1.88	anise.features.licensing.GPLv3 Class Reference	83
1.88.1	Constructor & Destructor Documentation	83
1.88.1.1	__init__	83
1.88.2	Member Function Documentation	83
1.88.2.1	getfilename	83
1.88.2.2	getlicensetext	83
1.88.3	Member Data Documentation	83
1.88.3.1	filename	83
1.88.3.2	name	83
1.88.3.3	namedebian	83
1.88.3.4	namefirefox	83
1.88.3.5	namepythonsetuputils	84
1.89	anise.features.documentation.HomepageExport Class Reference	84
1.89.1	Detailed Description	84
1.89.2	Constructor & Destructor Documentation	84
1.89.2.1	__init__	84
1.89.3	Member Function Documentation	84
1.89.3.1	supports	84
1.89.4	Member Data Documentation	84
1.89.4.1	title	84
1.90	anise.framework.features.Hook Class Reference	85
1.90.1	Detailed Description	85
1.90.2	Constructor & Destructor Documentation	85

1.90.2.1	<code>__init__</code>	85
1.90.3	Member Data Documentation	85
1.90.3.1	<code>eventname</code>	85
1.90.3.2	<code>function</code>	85
1.90.3.3	<code>prepares</code>	85
1.90.3.4	<code>provides</code>	85
1.90.3.5	<code>requires</code>	85
1.91	<code>anise.utils.exceptions.InternalError</code> Class Reference	86
1.91.1	Detailed Description	86
1.91.2	Member Function Documentation	86
1.91.2.1	<code>__call__</code>	86
1.92	<code>anise.framework.features.internals</code> Class Reference	86
1.92.1	Member Function Documentation	86
1.92.1.1	<code>injectdefaultstoproject</code>	86
1.92.1.2	<code>injecthookstoproject</code>	86
1.92.1.3	<code>loadfeatures</code>	86
1.92.1.4	<code>resetfeatures</code>	86
1.93	<code>anise.features.licensing.LGPLv3</code> Class Reference	87
1.93.1	Constructor & Destructor Documentation	87
1.93.1.1	<code>__init__</code>	87
1.93.2	Member Function Documentation	87
1.93.2.1	<code>getfilename</code>	87
1.93.2.2	<code>getlicensetext</code>	87
1.93.3	Member Data Documentation	87
1.93.3.1	<code>filename</code>	87
1.93.3.2	<code>name</code>	87
1.93.3.3	<code>namedebian</code>	87
1.93.3.4	<code>namefirefox</code>	87
1.93.3.5	<code>namepythonsetuputils</code>	88
1.94	<code>anise.features.projectdata.Maturity</code> Class Reference	88
1.94.1	Detailed Description	88
1.94.2	Member Data Documentation	88
1.94.2.1	<code>Alpha</code>	88
1.94.2.2	<code>Beta</code>	88
1.94.2.3	<code>Inactive</code>	88
1.94.2.4	<code>Mature</code>	88
1.94.2.5	<code>Planning</code>	88
1.94.2.6	<code>PreAlpha</code>	88
1.94.2.7	<code>ProductionStable</code>	88
1.95	<code>anise.features.packages.debian.MenuEntry</code> Class Reference	88

1.95.1	Member Data Documentation	89
1.95.1.1	category	89
1.95.1.2	command	89
1.95.1.3	gui	89
1.95.1.4	icon	89
1.95.1.5	name	89
1.95.1.6	title	89
1.96	anise.features.packages.win32.MenuEntry Class Reference	89
1.96.1	Member Data Documentation	90
1.96.1.1	command	90
1.96.1.2	icon	90
1.96.1.3	title	90
1.97	anise.features.licensing.MIT Class Reference	90
1.97.1	Constructor & Destructor Documentation	90
1.97.1.1	__init__	90
1.97.2	Member Function Documentation	90
1.97.2.1	getfilename	90
1.97.2.2	getlicensetext	90
1.97.3	Member Data Documentation	91
1.97.3.1	filename	91
1.97.3.2	name	91
1.97.3.3	namedebian	91
1.97.3.4	namefirefox	91
1.97.3.5	namepythonsetuputils	91
1.98	anise.utils.basic.Mount Class Reference	91
1.98.1	Detailed Description	91
1.98.2	Constructor & Destructor Documentation	92
1.98.2.1	__init__	92
1.98.3	Member Function Documentation	92
1.98.3.1	__enter__	92
1.98.3.2	__exit__	92
1.98.3.3	mount	92
1.98.3.4	unmount	92
1.98.4	Member Data Documentation	92
1.98.4.1	dst	92
1.98.4.2	mounted	92
1.98.4.3	mounter	92
1.98.4.4	needsroot	92
1.98.4.5	options	92
1.98.4.6	src	92

1.98.4.7	su	92
1.98.4.8	unmounter	92
1.99	anise.utils.ssh.Mount Class Reference	92
1.99.1	Detailed Description	93
1.99.2	Constructor & Destructor Documentation	93
1.99.2.1	__init__	93
1.99.3	Member Function Documentation	93
1.99.3.1	__enter__	93
1.99.3.2	__exit__	93
1.99.3.3	mount	94
1.99.3.4	unmount	94
1.99.4	Member Data Documentation	94
1.99.4.1	dst	94
1.99.4.2	mounted	94
1.99.4.3	mounter	94
1.99.4.4	needsroot	94
1.99.4.5	options	94
1.99.4.6	src	94
1.99.4.7	su	94
1.99.4.8	unmounter	94
1.100	anise.features.licensing.MPLv11 Class Reference	94
1.100.1	Constructor & Destructor Documentation	95
1.100.1.1	__init__	95
1.100.2	Member Function Documentation	95
1.100.2.1	getfilename	95
1.100.2.2	getlicensetext	95
1.100.3	Member Data Documentation	95
1.100.3.1	filename	95
1.100.3.2	name	95
1.100.3.3	namedebian	95
1.100.3.4	namefirefox	95
1.100.3.5	namepythonsetuputils	95
1.101	anise.features.licensing.MPLv2 Class Reference	95
1.101.1	Constructor & Destructor Documentation	96
1.101.1.1	__init__	96
1.101.2	Member Function Documentation	96
1.101.2.1	getfilename	96
1.101.2.2	getlicensetext	96
1.101.3	Member Data Documentation	96
1.101.3.1	filename	96

1.101.3.2 name	96
1.101.3.3 namedebian	96
1.101.3.4 namefirefox	96
1.101.3.5 namepythonsetuputils	96
1.102anise.features.documentation.PackageExport Class Reference	96
1.102.1 Detailed Description	97
1.102.2 Constructor & Destructor Documentation	97
1.102.2.1 __init__	97
1.102.3 Member Function Documentation	97
1.102.3.1 supports	97
1.102.4 Member Data Documentation	97
1.102.4.1 heading	97
1.102.4.2 includedeveloperdoc	97
1.102.4.3 ispackageexport	97
1.102.4.4 ofile	97
1.102.4.5 omode	97
1.103anise.utils.exceptions.ProcessExecutionFailedError Class Reference	98
1.103.1 Detailed Description	98
1.103.2 Member Function Documentation	98
1.103.2.1 __call__	98
1.104anise.framework.projects.Project Class Reference	98
1.104.1 Detailed Description	98
1.104.2 Constructor & Destructor Documentation	99
1.104.2.1 __init__	99
1.104.3 Member Function Documentation	99
1.104.3.1 addhook	99
1.104.3.2 append	99
1.104.3.3 gethooks	99
1.104.4 Member Data Documentation	99
1.104.4.1 hooks	99
1.105anise.features.licensing.PublicDomain Class Reference	99
1.105.1 Constructor & Destructor Documentation	99
1.105.1.1 __init__	99
1.105.2 Member Function Documentation	100
1.105.2.1 getfilename	100
1.105.2.2 getlicensetext	100
1.105.3 Member Data Documentation	100
1.105.3.1 filename	100
1.105.3.2 name	100
1.105.3.3 namedebian	100

1.105.3.4 namefirefox	100
1.105.3.5 namepythonsetuputils	100
1.106anise.features.dependencies.python.PythonDependency Class Reference	100
1.106.1 Constructor & Destructor Documentation	100
1.106.1.1 __init__	100
1.107anise.utils.packaging.PythonWheelPackageApplicationLink Class Reference	101
1.107.1 Detailed Description	101
1.107.2 Constructor & Destructor Documentation	101
1.107.2.1 __init__	101
1.107.3 Member Data Documentation	101
1.107.3.1 gui	101
1.107.3.2 linkname	101
1.107.3.3 method	101
1.107.3.4 module	101
1.108anise.features.build.qmake.QtConfiguration Class Reference	102
1.108.1 Constructor & Destructor Documentation	102
1.108.1.1 __init__	102
1.108.2 Member Data Documentation	102
1.108.2.1 bindir	102
1.108.2.2 makecmd	102
1.108.2.3 pluginsdir	102
1.108.2.4 qmakecmd	102
1.108.2.5 qtdir	102
1.109anise.utils.exceptions.RequirementsMissingError Class Reference	102
1.109.1 Detailed Description	103
1.109.2 Member Function Documentation	103
1.109.2.1 __call__	103
1.110anise.features.packages.debian.ServiceDescription Class Reference	103
1.110.1 Detailed Description	103
1.110.2 Constructor & Destructor Documentation	103
1.110.2.1 __init__	103
1.110.3 Member Data Documentation	103
1.110.3.1 command	103
1.110.3.2 name	103
1.111anise.utils.logging.Severity Class Reference	104
1.111.1 Detailed Description	104
1.111.2 Member Data Documentation	104
1.111.2.1 Debug	104
1.111.2.2 Error	104
1.111.2.3 Info	104

1.111.2.4 Warning	104
1.112anise.features.documentation.SourceExport Class Reference	104
1.112.1 Detailed Description	105
1.112.2 Constructor & Destructor Documentation	105
1.112.2.1 __init__	105
1.112.3 Member Function Documentation	105
1.112.3.1 supports	105
1.112.4 Member Data Documentation	105
1.112.4.1 heading	105
1.112.4.2 includedeveloperdoc	105
1.112.4.3 issourceexport	105
1.112.4.4 ofile	105
1.112.4.5 omode	105
1.113anise.framework.base.TaskExecution Class Reference	105
1.113.1 Detailed Description	106
1.113.2 Constructor & Destructor Documentation	106
1.113.2.1 __init__	106
1.113.3 Member Function Documentation	106
1.113.3.1 execute	106
1.113.4 Member Data Documentation	106
1.113.4.1 params	106
1.113.4.2 task	106
1.114anise.features.imagegalleries.tasks Class Reference	106
1.114.1 Member Function Documentation	106
1.114.1.1 getimagegallery	106
1.115anise.features.packages.tasks Class Reference	107
1.115.1 Member Function Documentation	107
1.115.1.1 enrichrawpackage	107
1.115.1.2 getversion	107
1.115.1.3 getwebsiteurl	107
1.115.1.4 makerawpackage	107
1.115.1.5 tarpackage	107
1.115.2 Member Data Documentation	107
1.115.2.1 _makerawpackage_cache	107
1.116anise.features.homepage.tasks Class Reference	108
1.116.1 Member Function Documentation	108
1.116.1.1 getbannerimage	108
1.116.1.2 makehomepage	108
1.117anise.features.packages.win32.tasks Class Reference	108
1.117.1 Member Function Documentation	108

1.117.1.1 deployqt5dlls	108
1.117.1.2 win32exepackage	108
1.118anise.features.dependencies.tasks Class Reference	109
1.118.1 Member Function Documentation	109
1.118.1.1 getdependencydocument	109
1.119anise.features.packages.python.tasks Class Reference	109
1.119.1 Member Function Documentation	109
1.119.1.1 wheelpackage	109
1.120anise.features.packages.debian.tasks Class Reference	109
1.120.1 Member Function Documentation	110
1.120.1.1 debpackage	110
1.120.1.2 only_programfiles	110
1.121anise.features.projectdata.tasks Class Reference	110
1.121.1 Member Function Documentation	110
1.121.1.1 getprojectstatusdescription	110
1.122anise.features.filesystem.tasks Class Reference	110
1.122.1 Member Function Documentation	111
1.122.1.1 filterbyextension	111
1.122.1.2 filterbyfunction	111
1.122.1.3 filterbyname	111
1.122.1.4 filterbynamepattern	111
1.123anise.features.build.qmake.tasks Class Reference	111
1.123.1 Member Function Documentation	111
1.123.1.1 build	111
1.124anise.features.datainjections.tasks Class Reference	111
1.124.1 Member Function Documentation	112
1.124.1.1 getessentials	112
1.124.1.2 inject_c	112
1.124.1.3 inject_python	112
1.125anise.features.versioncontrol.svn.tasks Class Reference	112
1.125.1 Member Function Documentation	112
1.125.1.1 createrawpackage	112
1.125.1.2 getfullversion	112
1.125.1.3 syncversioncontrolsystem	113
1.126anise.features.releasing.tasks Class Reference	113
1.126.1 Member Function Documentation	113
1.126.1.1 release	113
1.127anise.features.documentation.tasks Class Reference	113
1.127.1 Member Function Documentation	113
1.127.1.1 makedeveloperdoc_doxygen	113

1.127.1.2 writedocumentations	113
1.128anise.features.packages.firefox.tasks Class Reference	114
1.128.1 Member Function Documentation	114
1.128.1.1 addonpackage	114
1.128.1.2 inject_projectdata_to_packagejson	114
1.129anise.features.packages.android.tasks Class Reference	114
1.129.1 Member Function Documentation	115
1.129.1.1 inject_projectdata_to_androidmanifest	115
1.129.1.2 package_with_ant	116
1.130anise.features.sshfiletransfer.tasks Class Reference	116
1.130.1 Member Function Documentation	116
1.130.1.1 upload	116
1.131anise.framework.results.TempDir Class Reference	116
1.131.1 Detailed Description	117
1.131.2 Constructor & Destructor Documentation	117
1.131.2.1 __init__	117
1.131.3 Member Function Documentation	117
1.131.3.1 __enter__	117
1.131.3.2 __exit__	117
1.131.3.3 delete	117
1.131.3.4 dl	117
1.131.3.5 with_modified_rootname	118
1.131.4 Member Data Documentation	118
1.131.4.1 origpath	118
1.131.4.2 path	118
1.132anise.features.dependencies.Type Class Reference	118
1.132.1 Detailed Description	118
1.132.2 Member Data Documentation	118
1.132.2.1 Included	118
1.132.2.2 Optional	118
1.132.2.3 Recommended_HasAlternatives	118
1.132.2.4 Required	118
1.132.2.5 Required_HasAlternatives	119
1.133anise.utils.exceptions.UnexpectedSituationError Class Reference	119
1.133.1 Detailed Description	119
1.133.2 Member Function Documentation	119
1.133.2.1 __call__	119
1.134anise.features.packages.python.WheelApplicationLink Class Reference	119
1.134.1 Member Data Documentation	120
1.134.1.1 gui	120

1.134.1.2 linkname	120
1.134.1.3 method	120
1.134.1.4 module	120
1.135anise.utils.packaging.WindowsPackageMenuEntry Class Reference	120
1.135.1 Detailed Description	120
1.135.2 Constructor & Destructor Documentation	120
1.135.2.1 __init__	120
1.135.3 Member Data Documentation	121
1.135.3.1 command	121
1.135.3.2 icon	121
1.135.3.3 title	121
Index	122

Chapter 1

Anise Manual

1.1 License

anise is written by Josef Hahn under the terms of the GPLv3.

Please read the `LICENSE` file from the package and the [Dependencies](#) section for included third-party stuff.

1.2 About

Anise helps you to implement a common mechanism for executing all automated tasks for all your software projects. The usage scenario is a developer (or a team) working on many different projects with different frameworks and different tools. Anise can streamline such a chaos of tools. Typical tasks you would add to an anise-aware project can be:

- generating documentation
- creating packages
- handling version information
 - e.g. print it in the manual
- creating a homepage automatically built from the available version information, the packages, the documentation and so on
- deploying this homepage to a web server
- unit testing
- ... or whatever your project needs

Anise provides a Python based infrastructure, while you will need to implement some parts in order to adapt the behavior to your project's needs. Anise contains some ready-to-use implementations for some tasks (like building binaries via qmake, debian and some other packages, svn interaction, doxygen, ...), so you can directly use it without non-trivial additional coding to do all kinds of stuff, including generating documentations, packages and a homepage. However, chances are good that you will have to help out with some own implementations.

Your part as a project developer is to create a *project description* file, which contains all project metadata and custom implementations of some tasks, and add this file to your project.

Are you currently reading from another source than the homepage? Are you in doubt if that place is up-to-date? If yes, you should visit <https://ginger.ws/wiki/pino/projs/anise> and check that. You are currently reading the manual for version 1.0.1054.

1.3 Maturity

In this version, the state of anise is considered as production-stable.

Dependencies

There are external parts which are used by anise. Many thanks to the projects and all participants.

Python 3.4 *required*

Typical GNU/Linux Desktop *recommended (maybe has alternatives)*

subversion *optional*

banner image *included*: `_meta/homepage_bannerimage.png`; license [CC BY-SA 3.0](#); copied from [here](#).

1.4 User Documentation

The anise command line tool understands this syntax: The anise command line tool understands this syntax:

1.4.1 How to use anise

Each anise-aware project needs to have a *project description* file, which contains metadata about the project as well as some task implementations, e.g. for interaction with your other tools (requesting a certain piece of information from some version control system, building an executable with some compiler, ...).

1.4.1.1 What is a 'project description' file?

The project description file must be either in the root directory of your project or in the `_meta/` subdirectory and must be named `_projectdesc`. All task definitions and every relevant piece of information about your project are noted in this file. The complete anise operation will run just on top of it (this is not entirely correct, but correct enough for the moment :-P).

This file will be evaluated as Python code at runtime. It contains mostly this kinds of expressions:

- Simple assignments for simple scalar metadata, like the project name, the license of your project or the email address for contacting the developer team. Example: `name = "Anise"`
- Some more complex kinds of project metadata, like software dependencies or package creation information are calls to functions. Example: `distributables.pool.addgroup(packaging.tasks.-tarpackage, name="source tarball")`
- Task definitions. They specify how to actually finish a certain task like generating packages or the documentation. This definitions will be executed either directly (if this is the task you call the engine with), indirectly, or not at all. There are two different ways to define tasks (and an additional one):
 - Write your own task implementation in very much the same way as you would write functions directly in Python code. Example: `def makefoo(...):` //and some lines of Python code
 - Just do an assignment from something from the `tasks` object, which contains the tasks currently provided by anise itself or some of the loaded plugins (a.k.a. features). Example: `makedeveloperdoc = documentation.tasks.makedeveloperdoc_doxygen`

1.4.1.2 Executing tasks

What the user of anise does, after creating a `_projectdesc` file, is executing the tasks defined in that file. You do so by calling anise **from the project root directory** on the command line. In the easiest case, you just need to give the task's name as the only parameter. Assuming to have defined a `dorelease` task, you can simply call

```
anise dorelease
```

from the project directory.

1.4.1.3 Technical details about the anatomy of anise

The 'project object'

Conceptually, there is one global data structure, which contains all the information for executing tasks. This includes all the metadata, like the project's name, its license and so on, as well as all task implementations. This data structure is called 'the *project* object'. A `project` in source code, nearly always refers to it. Each piece of metadata and each task implementation is stored in a property of this object. For example, `project.name` maybe contains the name of the project.

Technically, this project object can be accessed in different ways:

- Each assignment in the `_projectdesc` file directly sets a value for one property. This makes it syntactically easy and short to set metadata there. Each line in the following example sets the value for one property of the project object:

```
name = "anise"
license = licensing.GPLv3
makehomepage = homepage.tasks.makehomepage # a task definition
```

Please note that this does only work for direct assignments, not for indirect ones, e.g. within a function:

```
def sometask():
    name = "anise" # this will never assign anything to the project
                  # object! We will see a workaround some lines below.
```

- From all places in Python code, it is possible to add the line `from anise.framework import projects` to the imports and get the current project with `projects.currentproject()` like this-:

```
projects.currentproject().email = "jen@reynholm-industries.co.uk"
```

or

```
projects.currentproject().email = foo_getusername() + "@" + foo_getmaildomain()
```

Please note that `projects` is automatically imported into the execution context of the `_projectdesc` file, so you can directly access the project object from within functions this way:

```
def sometask():
    projects.currentproject().name = "anise"
```

On the command line, anise can be executed with additional parameters in order to assign values to properties of the project object (this is limited to textual values), like this:

```
anise dorelease --setvalue 'license' 'MS-EULA' # just for this call, i'm a fan of a redmond license
```

A larger example

The following will give a simple, but more overall example for how the value assignments on the project object can be used to actually do useful things (the example itself - surprise, surprise - will again just print some nonsense).

Assume the following `_projectdesc` file in your project:

```
targetname = "peace in the world"
author = "john"
end_of_sentence = "."

def get_formatted_authorname():
    return "<" + projects.currentproject().author + ">"

def get_formatted_targetname():
    return "<" + projects.currentproject().targetname + ">"

def do_something(project):
    part1 = project.get_formatted_authorname(project)
    part2 = project.get_formatted_targetname(project)
    part3 = project.end_of_sentence
    print ("The major goal of " + part1 + " is " + part2 + part3)
```

And the following command line:

```
anise do_something --setvalue 'end_of_sentence' ' :D'
```

The execution of that call will print the following text:

```
The major goal of <john> is <peace in the world> :D
```

Task implementations

A task implementation is just a definition of a Python function. There are two different kinds of task implementations:

- Task implementations without function parameters. They can directly be called from the command line by the name of the function in your Python code.
- Task implementations which have parameters. They can't be called from the command line, but can be used as helpers for others tasks.

Please note that directly calling a task from the command line only makes sense if it returns reasonable types of values. The anise engine understands returning the primitive data types and many more:

- Everything which has a meaningful implementation of `__str__`.
- An instance of [anise.framework.results.Filestructure](#). This is used for returning pieces of the filesystem (i.e. single files or directories with all its contents).

This is an example for a task, which can directly be used by an end user via a command line call:

```
def some_directly_usable_task(): # needs 'thumb' to be set to a number in the project object
    return math.pi * projects.currentproject().thumb # a german phrase equivalent to the
                                                    # english 'rule of thumb approximation'
```

The parts of anise

- Anise is built as a very generic infrastructure. In basic parts, it was used in all the example above. It provides some more parts, though it is rather stupid. It mostly executes Python code and provides some convenience.
- There is a plugin mechanism, called 'features'. Each feature brings some encapsulated additional functionality to anise. Read below more about features. All high-level functionality of anise is encapsuled in such 'features'.
- And there is a place for plain functions with simple arguments and output, not bound to any part of the framework or any feature. Those are utility functions used at some other parts of anise.

Now some details follow about those three parts.

The anise infrastructure

The infrastructure (the active part is [anise.framework.engine](#) btw) executes Python code in a `_projectdesc` file and makes some syntactical tricks leading to shorter (and better readable) files. This tricks include the direct access to the project object and the automatic imports it makes before executing `_projectdesc`.

It also helps tasks creating file structures at temporary places and return them completely or partially to the caller and it provides also much more convenience functionality. Read about [anise.framework](#) if you are interested.

However, as mentioned, the high-level functionality is not implemented here, but in the 'features'.

Features

All high-level functionality provided by anise is implemented in a bunch of 'features', which are something like plugins for anise. They can either provide additional tasks, which your `_projectdesc` file can use somehow, or it can bind some functions to hooks (more details about this in the developer section; it is something like an event) in order to actively execute additional services.

One part of the generic infrastructure is the features handler [anise.framework.features](#). It loads additional features at the beginning of execution.

The developer section below explains, what a feature is in detail (and some hints how to develop one).

Utilities

The utility functionality is put into the [anise.utils](#) package. It can be small convenience things and also rather complex routines. All of them are uncoupled from anise infrastructure, so they could be used in other programs.

1.4.1.4 Technical details

The hook system

An important part for the philosophy of anise is the hook system. It is also the basis of the working of features.

A hook is something like an event. Interesting parties can bind functions to this event. Other parties can iterate over the bound functions at certain moments - mostly in order to execute them.

A hook has a unique name in anise. The name is the identifier used to bind to it or iterate over it. Both can be done anywhere in a task implementation:

```
def bartask(): # we will bind this function to a hook below
    print("hello world")

def footask():
    projects.currentproject().addhook("MYHOOK", bartask) # binding a function to "MYHOOK"

def sometask():
    for hook in projects.currentproject().gethooks("MYHOOK"):
        hook() # iterating and executing the hook "MYHOOK"
```

Providing such a hook (which is not a defined technical process, since a hook is just a name) and iterating/executing it at certain moments (also called events) is a way to provide functionality which other parties can use by binding stuff to that hook.

Binding stuff to a hook, vice versa, is a way to participate on some external functionality.

Note: Using string literals as hook keys this way is unsafe. You should avoid doing this! There should be a constant defined somewhere for each hook.

There are some hooks defined by the engine itself, which have such defined constants. They are defined in [anise.-framework.features](#):

- `HOOK_BEFORE_DEFINITION`: executed by the engine just before the `_projectdesc` file itself is read/executed

- `HOOK_BEFORE_EXECUTION`: executed by the engine just before the chosen task itself is executed

You would correctly use them this way:

```
def bartask(): # we will bind this function to a hook below
    print("hello world")

def footask():
    projects.currentproject().addhook(anise.framework.features.HOOK_BEFORE_DEFINITION, bartask)
```

This was about hooks, which will later be used in the sections about features.

What features are and how they are implemented

Anise features are what elsewhere probably would be called 'plugins'. Each feature encapsulates a certain set of functionality in one Python module file with some class and function definitions. All these items are imported in a new Python module. This module is automatically available in the project object, so this way you can use a function of a feature directly in `__projectdesc` (as with everything in the project object):

```
name = "anise"
...
somefeature.somefunction(42) # calls 'somefunction' from feature 'somefeature'
...
```

The items are available in a full-qualified manner, prepended with the full feature name. The full feature name is given by the file path of the feature source file. By the way: Full feature names can also be longer, like `name.space.some.feature`.

Providing functions directly

With the way above a feature can provide an interface which is callable from `__projectdesc` for controlling the behavior of the feature. This means setting configuration values or adding several items into some lists, which are processed later on by the feature logic later on, for instance.

Example: The `homepage` feature provides `homepage.sections.add` as interface, so other parties can add sections of content to the project homepage in the building process.

Providing classes

A feature can also export classes instead of functions in order to provide a class-based interface, so `__projectdesc` can instantiate them:

```
name = "anise"
...
somefeature.somefunction(somefeature.SomeClass(foo=42, 1, 2, 3))
...
```

Binding to infrastructure-hooks

For actively providing services, a function can be bound to a hook from the infrastructure (for other hooks, we need another trick). Those are `HOOK_BEFORE_DEFINITION` and more; introduced above. Use a function decorator for binding a function to such a hook:

```
@features.hook(features.HOOK_BEFORE_DEFINITION)
def __initproject():
    projects.currentproject().myunittests = [] # initialize, so later usage won't fail
```

This will automatically be executed at a very early stage of anise execution and can prepare some stuff.

Binding to non-infrastructure-hooks (and exporting own ones)

Other hooks, which are not defined in the infrastructure (but by another feature), are trickier. Technically they can be used by a string literal, as mentioned before. This method does not recognize broken references, however. For solving this issue, there is a mechanism for features to export hooks which it actively iterates at times, and for other features to safely refer to them.

Exporting is easy, just prepend a string variable with `HOOK_` in some of your lines in the features file. It looks like this:

```
def foo():
    ...
...
HOOK_DOSOMETHING = "DOSOMETHING"
```

This will automatically be detected by the feature loader and exported for external usage. The own code in this feature should of course use the same variable.

Binding to a non-infrastructure hook can't be done via function decorators but must be done in Python code. It must look up the feature by name, which provides a hook. Then it can use the returned data structures this way:

```
@features.hook(features.HOOK_BEFORE_DEFINITION) # this is NOT the interesting hook here!
def _initproject():

    def foo():
        do_something(1)
        do_something(2)
        do_something(3)

    projects.currentproject().addhook(features.loadfeature("myfeature").featuremodule.HOOK_DOSOMETHING, foo)
```

The large line adds a hook to project object. For the hook name, it searches the feature `myfeature` and accesses the `HOOK_DOSOMETHING` hook.

Exporting task implementations

Last but not least is exporting task implementation one of the common deals of a feature. Task implementations are simple functions, which are supposed to be called as task from command line (or indirectly somewhere from the called task). This differs from the functions mentioned above, which are called directly at the root level of `_projectdesc`.

A task implementation can be used by binding it to a nice name ...

```
makehomepage = homepage.tasks.makehomepage
```

... and call that from command line:

```
$ anise makehomepage
```

One can also call tasks from within other parts. This implementation of `makehomepage` would work nice as well:

```
def makehomepage():
    print("i start making the homepage")
    projects.currentproject().homepage.tasks.makehomepage()
    print("i finished making the homepage")
```

A feature provides tasks (by convention) by providing the class `tasks` with only static methods, like here:

```
def foo(): # not so interesting here
    ...
...
class tasks:

    @staticmethod
    def sometask():
        ...
```

If this feature is named `somefeature` (i.e. its file is `somefeature.py` directly in an anise feature loading path) a `_projectdesc` can use it this way

```
dosomething = somefeature.tasks.sometask
```

so one can call it from command line this way:

```
$ anise dosomething
```

This are the basic recipes for implementing features.

1.4.1.5 The command line options

The anise command line tool understands this syntax:

```
anise [taskname] [options]*
```

The taskname refers to anything callable defined in the project description file. Options are all optional and can be the following:

- `--setvalue [key] [value]` : Sets a value to the project object before execution
- `--projectfile [file]` : Use this project file
- `--loadfeaturesfrom [path1:...:pathn]` : Sets the paths for loading modules (separated by ":", builtin features by "builtin")
- `--debug` : Enabled debug log output
- `--keeptemp` : Don't remove the temporary stuff (for post-execution diagnostic)

1.5 Packages

Here are the packages with brief descriptions (if available):

anise	13
anise.data	
Only data files - no Python code here	14
anise.data.icons	
Png files	14
anise.data.licenses	
License texts	14
anise.features	
Anise Features - everything it does on top of the bare platform	14
anise.features.build	
Compiling binaries	15
anise.features.build.qmake	
Compiling binaries with qmake	16
anise.features.datainjections	
Helps injecting some kinds of data at anise runtime into some file structures, in order to make it available to other build processes (Makefiles, ...) or to the project's runtime	16
anise.features.dependencies	
Dependencies	17
anise.features.dependencies.python	
Dependencies from the Python world	18
anise.features.distributables	
Defining distributables (installation packages for target systems, ...)	18

anise.features.documentation	
Project documentation	19
anise.features.filesystem	
Basic Filesystem tasks	20
anise.features.homepage	
Project homepage generation	21
anise.features.imagegalleries	
Image galleries	21
anise.features.licensing	
The project's license	22
anise.features.packages	
Creation of distribution packages	23
anise.features.packages.android	
Creation of Android distribution packages	24
anise.features.packages.debian	
Creation of Debian distribution packages	24
anise.features.packages.firefox	
Creation of Firefox addon packages	25
anise.features.packages.python	
Creation of Python distribution packages	25
anise.features.packages.win32	
Creation of Win32 distribution packages	25
anise.features.projectdata	
Often used data structures and some smaller services	26
anise.features.releasing	
Generic releasing task	27
anise.features.sshfiletransfer	
File transfers via ssh	27
anise.features.versioncontrol	
Interoperability with version control systems	28
anise.features.versioncontrol.svn	
Interoperability with the Subversion version control system	28
anise.framework	
The anise infrastructure	29
anise.framework.base	
Some things intended to be used from the project description file	29
anise.framework.engine	
The anise engine initializes all the anise platform stuff and executes a task from a project file	30
anise.framework.features	
Anise Features	31
anise.framework.globalvars	
Global variables (often influenced by command line parameters)	32
anise.framework.imports	
Imports some modules for availability within <code>_projectdesc</code>	32
anise.framework.projectinformations	
Implementation for the anise project object	32
anise.framework.projects	
Typical anise result structures	33
anise.framework.results	
Utility functions from several categories which make life easier	34
anise.framework.util	
Basic helpers for filesystem and other operating system related stuff	34
anise.framework.documentation	
Documentation generation	36
anise.framework.exceptions	
Exception subclasses	42

anise.utils.logging	
Logging	42
anise.utils.packaging	
Package generation	44
anise.utils.ssh	
Utilities for ssh usage	47
anise.utils.svn	
Utilities for Subversion usage	47

1.6 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

anise.features.documentation._Exports	48
anise.features.datainjections._Pool	49
anise.features.dependencies._Pool	49
anise.features.documentation._Pool	50
anise.features.distributables._Pool	51
anise.features.imagegalleries._Pool	51
anise.features.releasing._Releasetasks	52
anise.features.homepage._Sections	52
anise.utils.svn._Version	53
anise.features.licensing.BaseLicense	60
anise.features.licensing.AFLv3	53
anise.features.licensing.AGPLv3	54
anise.features.licensing.Apache2	56
anise.features.licensing.Artistic1	57
anise.features.licensing.BSD2clause	62
anise.features.licensing.BSD3clause	63
anise.features.licensing.Cc0v1	68
anise.features.licensing.CcBy3	69
anise.features.licensing.CcByNc3	70
anise.features.licensing.CcByNcNd3	71
anise.features.licensing.CcByNcSa3	72
anise.features.licensing.CcByNd3	74
anise.features.licensing.CcBySa3	75
anise.features.licensing.GPLv3	83
anise.features.licensing.LGPLv3	87
anise.features.licensing.MIT	90
anise.features.licensing.MPLv11	94
anise.features.licensing.MPLv2	95
anise.features.licensing.PublicDomain	99
anise.features.packages.debian.Category	64
anise.utils.basic.ChDir	76
anise.utils.packaging.DebianPackageMenuEntry	77
anise.features.packages.debian.MenuEntry	88
anise.features.dependencies.Dependency	78
Exception	
anise.features.dependencies.python.DjangoDependency	79
anise.features.dependencies.python.PythonDependency	100
anise.utils.exceptions.AniseError	55
anise.utils.exceptions.BadCommunicationError	58
anise.utils.exceptions.BadFormatError	59
anise.utils.exceptions.BadInputError	59
anise.utils.exceptions.InternalError	86
anise.utils.exceptions.ProcessExecutionFailedError	98

anise.utils.exceptions.RequirementsMissingError	102
anise.utils.exceptions.UnexpectedSituationError	119
anise.features.documentation.Export	79
anise.features.documentation.HomepageExport	84
anise.features.documentation.PackageExport	96
anise.features.documentation.SourceExport	104
anise.framework.features.Feature	80
anise.features.distributables.FileGroup	80
anise.framework.results.Filestructure	81
anise.framework.results.TempDir	116
anise.framework.features.Hook	85
anise.framework.features.internals	86
anise.features.projectdata.Maturity	88
anise.utils.basic.Mount	91
anise.utils.ssh.Mount	92
anise.framework.projects.Project	98
anise.utils.packaging.PythonWheelPackageApplicationLink	101
anise.features.packages.python.WheelApplicationLink	119
anise.features.build.qmake.QtConfiguration	102
anise.features.packages.debian.ServiceDescription	103
anise.utils.logging.Severity	104
anise.framework.base.TaskExecution	105
anise.features.imagegalleries.tasks	106
anise.features.packages.tasks	107
anise.features.homepage.tasks	108
anise.features.packages.win32.tasks	108
anise.features.dependencies.tasks	109
anise.features.packages.python.tasks	109
anise.features.packages.debian.tasks	109
anise.features.projectdata.tasks	110
anise.features.filesystem.tasks	110
anise.features.build.qmake.tasks	111
anise.features.datainjections.tasks	111
anise.features.versioncontrol.svn.tasks	112
anise.features.releasing.tasks	113
anise.features.documentation.tasks	113
anise.features.packages.firefox.tasks	114
anise.features.packages.android.tasks	114
anise.features.sshfiletransfer.tasks	116
anise.features.dependencies.Type	118
anise.utils.packaging.WindowsPackageMenuEntry	120
anise.features.packages.win32.MenuEntry	89

1.7 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

anise.features.documentation._Exports	48
anise.features.datainjections._Pool	49
anise.features.dependencies._Pool	49
anise.features.documentation._Pool	50
anise.features.distributables._Pool	51
anise.features.imagegalleries._Pool	51
anise.features.releasing._Releasetasks	52
anise.features.homepage._Sections	52
anise.utils.svn._Version	53

anise.features.licensing.AFLv3	53
anise.features.licensing.AGPLv3	54
anise.utils.exceptions.AniseError	
An error in the execution of anise	55
anise.features.licensing.Apache2	56
anise.features.licensing.Artistic1	57
anise.utils.exceptions.BadCommunicationError	
The communication with some other component failed	58
anise.utils.exceptions.BadFormatError	
Some input is wrongly formatted	59
anise.utils.exceptions.BadInputError	
A bad value was given as some input	59
anise.features.licensing.BaseLicense	
Base class for licenses	60
anise.features.licensing.BSD2clause	62
anise.features.licensing.BSD3clause	63
anise.features.packages.debian.Category	
Enumeration for Debian's categories for software packages	64
anise.features.licensing.Cc0v1	68
anise.features.licensing.CcBy3	69
anise.features.licensing.CcByNc3	70
anise.features.licensing.CcByNcNd3	71
anise.features.licensing.CcByNcSa3	72
anise.features.licensing.CcByNd3	74
anise.features.licensing.CcBySa3	75
anise.utils.basic.ChDir	
Class for temporarily changing the current working directory in a scope of the <code>with</code> keyword	76
anise.utils.packaging.DebianPackageMenuEntry	
Data structure for one menu entry which comes in a Debian package	77
anise.features.dependencies.Dependency	
A dependency	78
anise.features.dependencies.python.DjangoDependency	79
anise.features.documentation.Export	
Describes a documentation export	79
anise.framework.features.Feature	
A feature is some encapsuled anise functionality	80
anise.features.distributables.FileGroup	
A file group	80
anise.framework.results.Filestructure	
This class is a special anise return type which is a file/folder (to be fetched to the current dir)	81
anise.features.licensing.GPLv3	83
anise.features.documentation.HomepageExport	
Describes an export of an anise documentation directly into the homepage	84
anise.framework.features.Hook	
An hook handler bound to some hook	85
anise.utils.exceptions.InternalError	
An internal error in the anise code occurred	86
anise.framework.features.internals	86
anise.features.licensing.LGPLv3	87
anise.features.projectdata.Maturity	
This class helps describing the project state	88
anise.features.packages.debian.MenuEntry	88
anise.features.packages.win32.MenuEntry	89
anise.features.licensing.MIT	90
anise.utils.basic.Mount	
Can be used to mount a volume	91
anise.utils.ssh.Mount	
Mounts a volume via ssh	92

anise.features.licensing.MPLv11	94
anise.features.licensing.MPLv2	95
anise.features.documentation.PackageExport	
Describes an export of an anise documentation into the distribution package(s)	96
anise.utils.exceptions.ProcessExecutionFailedError	
The execution of an external process failed	98
anise.framework.projects.Project	
Implementation for project objects	98
anise.features.licensing.PublicDomain	99
anise.features.dependencies.python.PythonDependency	100
anise.utils.packaging.PythonWheelPackageApplicationLink	
Data structure for one menu entry which comes in a Python wheel package	101
anise.features.build.qmake.QtConfiguration	102
anise.utils.exceptions.RequirementsMissingError	
Something required is not in place	102
anise.features.packages.debian.ServiceDescription	
Description for Debian services to be included in a package	103
anise.utils.logging.Severity	
Log message severities	104
anise.features.documentation.SourceExport	
Describes an export of an anise documentation back into the source tree	104
anise.framework.base.TaskExecution	
Holds a function and some parameters for calling later on (like a delegate)	105
anise.features.imagegalleries.tasks	106
anise.features.packages.tasks	107
anise.features.homepage.tasks	108
anise.features.packages.win32.tasks	108
anise.features.dependencies.tasks	109
anise.features.packages.python.tasks	109
anise.features.packages.debian.tasks	109
anise.features.projectdata.tasks	110
anise.features.filesystem.tasks	110
anise.features.build.qmake.tasks	111
anise.features.datainjections.tasks	111
anise.features.versioncontrol.svn.tasks	112
anise.features.releasing.tasks	113
anise.features.documentation.tasks	113
anise.features.packages.firefox.tasks	114
anise.features.packages.android.tasks	114
anise.features.sshfiletransfer.tasks	116
anise.framework.results.TempDir	
A special anise.framework.results.Filestructure which automatically creates a temporary directory in background	116
anise.features.dependencies.Type	
Enumeration of different types of dependencies	118
anise.utils.exceptions.UnexpectedSituationError	
An unexpected situation led to an error	119
anise.features.packages.python.WheelApplicationLink	119
anise.utils.packaging.WindowsPackageMenuEntry	
Data structure for one menu entry which comes in a Windows package	120

1.8 anise Namespace Reference

Namespaces

- [data](#)

Only data files - no Python code here.

- [features](#)

anise Features - everything it does on top of the bare platform.

- [framework](#)

The anise infrastructure.

- [utils](#)

Utility functions from several categories which make life easier.

1.9 anise.data Namespace Reference

Only data files - no Python code here.

Namespaces

- [icons](#)

png files.

- [licenses](#)

License texts.

1.9.1 Detailed Description

Only data files - no Python code here.

1.10 anise.data.icons Namespace Reference

png files.

1.10.1 Detailed Description

png files.

1.11 anise.data.licenses Namespace Reference

License texts.

1.11.1 Detailed Description

License texts.

1.12 anise.features Namespace Reference

anise Features - everything it does on top of the bare platform.

Namespaces

- [build](#)
Compiling binaries.
- [datainjections](#)
Helps injecting some kinds of data at anise runtime into some file structures, in order to make it available to other build processes (Makefiles, ...) or to the project's runtime.
- [dependencies](#)
Dependencies.
- [distributables](#)
Defining distributables (installation packages for target systems, ...).
- [documentation](#)
Project documentation.
- [filesystem](#)
Basic Filesystem tasks.
- [homepage](#)
Project homepage generation.
- [imagegalleries](#)
Image galleries.
- [licensing](#)
The project's license.
- [packages](#)
Creation of distribution packages.
- [projectdata](#)
Often used data structures and some smaller services.
- [releasing](#)
Generic releasing task.
- [sshfiletransfer](#)
File transfers via ssh.
- [versioncontrol](#)
Interoperability with version control systems.

1.12.1 Detailed Description

anise Features - everything it does on top of the bare platform. Anise features are a plugin system. All high-level functionality of anise is implemented in one of the features which are bundled with anise. They are loaded by the feature loader [anise.framework.features](#) at runtime.

Read more general information in [Features](#) and more about the specific features here.

Note: In order to actually understand what the features do and how to use them, it will be needed to study anise `_projectdesc` file and those of my other projects as examples as well as the anise sources.

1.13 anise.features.build Namespace Reference

Compiling binaries.

Namespaces

- [qmake](#)
Compiling binaries with qmake.

1.13.1 Detailed Description

Compiling binaries. This package bundles some features which implement building binaries, mostly by providing task implementations.

1.14 anise.features.build.qmake Namespace Reference

Compiling binaries with qmake.

Classes

- class [QtConfiguration](#)
- class [tasks](#)

Functions

- def [basedependencies](#)

1.14.1 Detailed Description

Compiling binaries with qmake. Makes the following new functionality available:

- task for compiling a binary with qmake

1.14.2 Function Documentation

1.14.2.1 `def anise.features.build.qmake.basedependencies ()`

1.15 anise.features.datainjections Namespace Reference

Helps injecting some kinds of data at anise runtime into some file structures, in order to make it available to other build processes (Makefiles, ...) or to the project's runtime.

Classes

- class [_Pool](#)
- class [tasks](#)

Functions

- def [_sync](#)
Sync if not marked as `omitsyncvcs`.
- def [_initproject](#)

Variables

- string [name](#) = "datainjections"

1.15.1 Detailed Description

Helps injecting some kinds of data at anise runtime into some file structures, in order to make it available to other build processes (Makefiles, ...) or to the project's runtime. Makes the following new functionality available:

- provides a method for adding a data injection
- getting pieces of project metadata prepared from an injection
- task for injecting some project metadata to a python source file

1.15.2 Function Documentation

1.15.2.1 `def anise.features.datainjections._initproject() [private]`

1.15.2.2 `def anise.features.datainjections._sync() [private]`

Sync if not marked as `omitsyncvcs`.

1.15.3 Variable Documentation

1.15.3.1 `string anise.features.datainjections.name = "datainjections"`

1.16 anise.features.dependencies Namespace Reference

Dependencies.

Namespaces

- [python](#)
Dependencies from the Python world.

Classes

- class [Type](#)
Enumeration of different types of dependencies.
- class [Dependency](#)
A dependency.
- class [_Pool](#)
- class [tasks](#)

Functions

- `def _initproject`
- `def _add_homepage_section`

1.16.1 Detailed Description

Dependencies. Specifying third-party packages which are required by the project in some way.

Makes the following new functionality available:

- provides a method for adding project dependencies
- task for getting a piece of documentation source describing the dependencies
- some enumerations
- automatically adding a documentation section to the homepage

1.16.2 Function Documentation

1.16.2.1 `def anise.features.dependencies._add_homepage_section() [private]`

1.16.2.2 `def anise.features.dependencies._initproject() [private]`

1.17 anise.features.dependencies.python Namespace Reference

Dependencies from the Python world.

Classes

- class [PythonDependency](#)
- class [DjangoDependency](#)

Variables

- tuple [_dependenciesfeature](#) = `features.loadfeature("dependencies")`

1.17.1 Detailed Description

Dependencies from the Python world. Makes the following new functionality available:

- some classes containing dependency info for typical Python projects.

1.17.2 Variable Documentation

1.17.2.1 `tuple anise.features.dependencies.python._dependenciesfeature = features.loadfeature("dependencies")`

1.18 anise.features.distributables Namespace Reference

Defining distributables (installation packages for target systems, ...).

Classes

- class [_Pool](#)
- class [FileGroup](#)
A file group.

Functions

- `def _initproject`
- `def _add_homepage_section`

1.18.1 Detailed Description

Defining distributables (installation packages for target systems, ...). Makes the following new functionality available:

- provides a method for adding distributables groups to a pool
- automatically adding a download section to the homepage

1.18.2 Function Documentation

1.18.2.1 `def anise.features.distributables._add_homepage_section () [private]`

1.18.2.2 `def anise.features.distributables._initproject () [private]`

1.19 anise.features.documentation Namespace Reference

Project documentation.

Classes

- class [Export](#)
Describes a documentation export.
- class [HomepageExport](#)
Describes an export of an anise documentation directly into the homepage.
- class [PackageExport](#)
Describes an export of an anise documentation into the distribution package(s).
- class [SourceExport](#)
Describes an export of an anise documentation back into the source tree.
- class [_Pool](#)
- class [_Exports](#)
- class [tasks](#)

Functions

- def [has_doxygen_developerdocumentation](#)
Configures a project for creating a Doxygen developer documentation.
- def [_initproject](#)
- def [_compose_readmes](#)
- def [_add_homepage_section](#)
- def [_add_release_task_for_export_to_source](#)
- def [add_anisedoc_source](#)
Loads the content of anisedoc sources (which contain ANISEDOC tags) into the documentation store.

Variables

- [markdown_downgrade_headings](#) = `utils.documentation.markdown_downgrade_headings`

1.19.1 Detailed Description

Project documentation. Makes the following new functionality available:

- provides a method for adding a documentation source assigned to a given key to a pool of documentation
- provides a method for adding an export for documentation to some destinations
- automatically generation of a README and FULLREADME documentation
- automatically adding a documentation section to the homepage

1.19.2 Function Documentation

1.19.2.1 `def anise.features.documentation._add_homepage_section () [private]`

1.19.2.2 `def anise.features.documentation._add_release_task_for_export_to_source () [private]`

1.19.2.3 `def anise.features.documentation._compose_readmes () [private]`

1.19.2.4 `def anise.features.documentation._initproject () [private]`

1.19.2.5 `def anise.features.documentation.add_anisedoc_source (src)`

Loads the content of anisedoc sources (which contain ANISEDOC tags) into the documentation store.

Parameters

<code>src</code>	file or directory with anisedoc sources
------------------	---

1.19.2.6 `def anise.features.documentation.has_doxygen_developerdocumentation ()`

Configures a project for creating a Doxygen developer documentation.

1.19.3 Variable Documentation

1.19.3.1 `anise.features.documentation.markdown_downgrade_headings = utils.documentation.markdown_downgrade_headings`

1.20 anise.features.filesystem Namespace Reference

Basic Filesystem tasks.

Classes

- class [tasks](#)

1.20.1 Detailed Description

Basic Filesystem tasks. Makes the following new functionality available:

- tasks for doing stuff on filesystem level

1.21 anise.features.homepage Namespace Reference

Project homepage generation.

Classes

- class [_Sections](#)
- class [tasks](#)

Functions

- def [_initproject](#)
- def [_basic_homepage_sections](#)

Variables

- string [HOOK_UPLOAD_HOMEPAGE_COLLECTLINKS](#) = "UPLOAD_HOMEPAGE_COLLECTLINKS"

1.21.1 Detailed Description

Project homepage generation. Makes the following new functionality available:

- provides a method for adding sections with content to the homepage
- task for creating a project homepage

1.21.2 Function Documentation

1.21.2.1 `def anise.features.homepage._basic_homepage_sections () [private]`

1.21.2.2 `def anise.features.homepage._initproject () [private]`

1.21.3 Variable Documentation

1.21.3.1 `string anise.features.homepage.HOOK_UPLOAD_HOMEPAGE_COLLECTLINKS = "UPLOAD_HOMEPAGE_COLLECTLINKS"`

1.22 anise.features.imagegalleries Namespace Reference

Image galleries.

Classes

- class [_Pool](#)
- class [tasks](#)

Functions

- def [_initproject](#)
- def [_add_homepage_section](#)

1.22.1 Detailed Description

Image galleries. Makes the following new functionality available:

- provides a method for adding image galleries to the pool of image galleries
- automatically adding a image gallery section to the homepage

1.22.2 Function Documentation

1.22.2.1 `def anise.features.imagegalleries._add_homepage_section () [private]`

1.22.2.2 `def anise.features.imagegalleries._initproject () [private]`

1.23 anise.features.licensing Namespace Reference

The project's license.

Classes

- class [BaseLicense](#)
Base class for licenses.
- class [AGPLv3](#)
- class [GPLv3](#)
- class [AFLv3](#)
- class [Apache2](#)
- class [BSD3clause](#)
- class [BSD2clause](#)
- class [LGPLv3](#)
- class [MPLv11](#)
- class [MPLv2](#)
- class [CcBy3](#)
- class [CcBySa3](#)
- class [CcByNd3](#)
- class [CcByNc3](#)
- class [CcByNcSa3](#)
- class [CcByNcNd3](#)
- class [Cc0v1](#)
- class [Artistic1](#)
- class [MIT](#)
- class [PublicDomain](#)

Functions

- `def _writelicense`
- `def _instantiatelicense`
- `def _add_homepage_section`

Variables

- tuple `_packagesfeature` = `features.loadfeature("packages")`

1.23.1 Detailed Description

The project's license. Makes the following new functionality available:

- automatically adds a `LICENSE` file to the raw package
- automatically instantiate the `license` if needed, so you are allowed to only write the class name
- automatically adding a license section to the homepage if `license` is set

1.23.2 Function Documentation

1.23.2.1 `def anise.features.licensing._add_homepage_section () [private]`

1.23.2.2 `def anise.features.licensing._instantiatelicense () [private]`

1.23.2.3 `def anise.features.licensing._writelicense (destdir) [private]`

1.23.3 Variable Documentation

1.23.3.1 `tuple anise.features.licensing._packagesfeature = features.loadfeature("packages")`

1.24 anise.features.packages Namespace Reference

Creation of distribution packages.

Namespaces

- [android](#)
Creation of Android distribution packages.
- [debian](#)
Creation of Debian distribution packages.
- [firefox](#)
Creation of Firefox addon packages.
- [python](#)
Creation of Python distribution packages.
- [win32](#)
Creation of Win32 distribution packages.

Classes

- class [tasks](#)

Functions

- `def _initproject`

Variables

- string [HOOK_ENRICHRAWPACKAGE](#) = "ENRICHRAWPACKAGE"

1.24.1 Detailed Description

Creation of distribution packages. Creating distribution packages.

Makes the following new functionality available:

- framework for enriching a raw package (and hook for adding custom functionality)
- getting the project version number
- creating tar source packages

1.24.2 Function Documentation

1.24.2.1 `def anise.features.packages._initproject () [private]`

1.24.3 Variable Documentation

1.24.3.1 `string anise.features.packages.HOOK_ENRICHRAWPACKAGE = "ENRICHRAWPACKAGE"`

1.25 `anise.features.packages.android` Namespace Reference

Creation of Android distribution packages.

Classes

- class [tasks](#)

1.25.1 Detailed Description

Creation of Android distribution packages. Makes the following new functionality available:

- task for creating Android packages.
- task for keeping the Android project file up-to-date

1.26 `anise.features.packages.debian` Namespace Reference

Creation of Debian distribution packages.

Classes

- class [Category](#)
Enumeration for Debian's categories for software packages.
- class [ServiceDescription](#)
Description for Debian services to be included in a package.
- class [MenuEntry](#)
- class [tasks](#)

1.26.1 Detailed Description

Creation of Debian distribution packages. Makes the following new functionality available:

- task for creating a Debian package

1.27 anise.features.packages.firefox Namespace Reference

Creation of Firefox addon packages.

Classes

- class [tasks](#)

1.27.1 Detailed Description

Creation of Firefox addon packages. Makes the following new functionality available:

- task for creating a Firefox addon package
- task for keeping the Firefox addon project file up-to-date

1.28 anise.features.packages.python Namespace Reference

Creation of Python distribution packages.

Classes

- class [WheelApplicationLink](#)
- class [tasks](#)

Functions

- def [_nsis_get_global](#)

1.28.1 Detailed Description

Creation of Python distribution packages. Makes the following new functionality available:

- task for creating python wheel packages

1.28.2 Function Documentation

1.28.2.1 `def anise.features.packages.python._nsis_get_global () [private]`

1.29 anise.features.packages.win32 Namespace Reference

Creation of Win32 distribution packages.

Classes

- class [MenuEntry](#)
- class [tasks](#)

Variables

- string [HOOK_WINDOWS_PACKAGE_NSIS_ONINIT](#) = "HOOK_WINDOWS_PACKAGE_NSIS_ONINIT"
- string [HOOK_WINDOWS_PACKAGE_NSIS_ONINSTALL](#) = "HOOK_WINDOWS_PACKAGE_NSIS_ONINSTALL"
- string [HOOK_WINDOWS_PACKAGE_NSIS_GLOBAL](#) = "HOOK_WINDOWS_PACKAGE_NSIS_GLOBAL"

1.29.1 Detailed Description

Creation of Win32 distribution packages. Makes the following new functionality available:

- task for creating windows installers

1.29.2 Variable Documentation

1.29.2.1 string `anise.features.packages.win32.HOOK_WINDOWS_PACKAGE_NSIS_GLOBAL` = "HOOK_WINDOWS_PACKAGE_NSIS_GLOBAL"

1.29.2.2 string `anise.features.packages.win32.HOOK_WINDOWS_PACKAGE_NSIS_ONINIT` = "HOOK_WINDOWS_PACKAGE_NSIS_ONINIT"

1.29.2.3 string `anise.features.packages.win32.HOOK_WINDOWS_PACKAGE_NSIS_ONINSTALL` = "HOOK_WINDOWS_PACKAGE_NSIS_ONINSTALL"

1.30 anise.features.projectdata Namespace Reference

Often used data structures and some smaller services.

Classes

- class [Maturity](#)
This class helps describing the project state.
- class [tasks](#)

Functions

- def [_initproject](#)
- def [_beautify_longdescription](#)

1.30.1 Detailed Description

Often used data structures and some smaller services. Makes the following new functionality available:

- automatically beautifies `project.longdescription`
- determining the specified project maturity as string
- enumerations for some project data

1.30.2 Function Documentation

1.30.2.1 `def anise.features.projectdata._beautify_longdescription () [private]`

1.30.2.2 `def anise.features.projectdata._initproject () [private]`

1.31 anise.features.releasing Namespace Reference

Generic releasing task.

Classes

- class [_Releasetasks](#)
- class [tasks](#)

Functions

- `def _initproject`

Variables

- string [HOOK_RELEASE](#) = "RELEASE"

1.31.1 Detailed Description

Generic releasing task. The main usage is providing a hook for other features to mount something there.

Makes the following new functionality available:

- release process framework via [tasks.release](#)
- adding custom release subtasks into the process

1.31.2 Function Documentation

1.31.2.1 `def anise.features.releasing._initproject () [private]`

1.31.3 Variable Documentation

1.31.3.1 `string anise.features.releasing.HOOK_RELEASE = "RELEASE"`

1.32 anise.features.sshfiletransfer Namespace Reference

File transfers via ssh.

Classes

- class [tasks](#)

Variables

- tuple `_homepagefeature` = `features.loadfeature("homepage")`

1.32.1 Detailed Description

File transfers via ssh. Makes the following new functionality available:

- uploading directories via ssh

1.32.2 Variable Documentation

1.32.2.1 tuple `anise.features.sshfiletransfer._homepagefeature` = `features.loadfeature("homepage")`

1.33 `anise.features.versioncontrol` Namespace Reference

Interoperability with version control systems.

Namespaces

- `svn`
Interoperability with the Subversion version control system.

1.33.1 Detailed Description

Interoperability with version control systems.

1.34 `anise.features.versioncontrol.svn` Namespace Reference

Interoperability with the Subversion version control system.

Classes

- class `tasks`

Functions

- def `is_svn_controlled`
Configures a project for Subversion usage (binds some tasks so they get called).

1.34.1 Detailed Description

Interoperability with the Subversion version control system. Makes the following new functionality available for Subversion controlled projects:

- provides a method for getting a complete version number including svn revision
- provides a method for creation of the raw package
- provides syncing functionality between the repository and the working copy
- provides a method for automatically enabling all this stuff

1.34.2 Function Documentation

1.34.2.1 `def anise.features.versioncontrol.svn.is_svn_controlled ()`

Configures a project for Subversion usage (binds some tasks so they get called).

1.35 anise.framework Namespace Reference

The anise infrastructure.

Namespaces

- [base](#)
Some things intended to be used from the project description file.
- [engine](#)
The anise engine initializes all the anise platform stuff and executes a task from a project file.
- [features](#)
anise Features
- [globalvars](#)
Global variables (often influenced by command line parameters)
- [imports](#)
Imports some modules for availability within `_projectdesc`.
- [projectinformations](#)
- [projects](#)
Implementation for the anise project object.
- [results](#)
Typical anise result structures.

1.35.1 Detailed Description

The anise infrastructure.

1.36 anise.framework.base Namespace Reference

Some things intended to be used from the project description file.

Classes

- class [TaskExecution](#)
Holds a function and some parameters for calling later on (like a delegate).

1.36.1 Detailed Description

Some things intended to be used from the project description file. The engine automatically import everything from here into the execution context of the project description file, so they are automatically available there.

1.37 anise.framework.engine Namespace Reference

The anise engine initializes all the anise platform stuff and executes a task from a project file.

Functions

- def [execute](#)
Executes task 'taskname' from the project in 'projectfile'.
- def [_cmdline](#)
Parses the command line.
- def [extractfilesfromretval](#)
Can be used after executing a task in the anise engine for making a result file available in the target directory.
- def [main](#)
The main method when called from command line.

1.37.1 Detailed Description

The anise engine initializes all the anise platform stuff and executes a task from a project file.

1.37.2 Function Documentation

1.37.2.1 `def anise.framework.engine._cmdline (args) [private]`

Parses the command line.

1.37.2.2 `def anise.framework.engine.execute (taskname, projectfile, projectdir, values, loadfeaturesfrom)`

Executes task 'taskname' from the project in 'projectfile'.

Parameters

<i>taskname</i>	The name of the task to be executed.
<i>projectfile</i>	The path to the anise project description file. Their name is usually <code>_projectdesc</code> .
<i>projectdir</i>	The path to the project root directory.
<i>values</i>	Additional values to be set to the project object.
<i>loadfeaturesfrom</i>	List of paths where anise features shall be loaded from.

1.37.2.3 `def anise.framework.engine.extractfilesfromretval (r)`

Can be used after executing a task in the anise engine for making a result file available in the target directory.

Before you do so, all files created as result are in a temporary area, which will be flushed when the process terminates.

1.37.2.4 `def anise.framework.engine.main ()`

The main method when called from command line.

Parses command line arguments and executes the chosen task.

1.38 anise.framework.features Namespace Reference

anise Features

Classes

- class [internals](#)
- class [Hook](#)
An hook handler bound to some hook.
- class [Feature](#)
A featue is some encapsuled anise functionality.

Functions

- def [loadfeature](#)
Load an [anise.framework.features.Feature](#) by name from the predefined sources.
- def [hook](#)
Used by a feature module to register a hook (mostly used as decorator).

Variables

- string [HOOK_BEFORE_DEFINITION](#) = "BEFORE_DEFINITION"
- string [HOOK_BEFORE_EXECUTION](#) = "BEFORE_EXECUTION"
- list [_features](#) = []
- list [_featuresources](#) = []
- list [_hooks](#) = []

1.38.1 Detailed Description

anise Features

1.38.2 Function Documentation

1.38.2.1 `def anise.framework.features.hook (name, provides = None, requires = None, prepares = None)`

Used by a feature module to register a hook (mostly used as decorator).

1.38.2.2 `def anise.framework.features.loadfeature (name)`

Load an [anise.framework.features.Feature](#) by name from the predefined sources.

1.38.3 Variable Documentation

1.38.3.1 `list anise.framework.features._features = []`

1.38.3.2 `list anise.framework.features._featuresources = []`

1.38.3.3 `list anise.framework.features._hooks = []`

1.38.3.4 `string anise.framework.features.HOOK_BEFORE_DEFINITION = "BEFORE_DEFINITION"`

1.38.3.5 `string anise.framework.features.HOOK_BEFORE_EXECUTION = "BEFORE_EXECUTION"`

1.39 `anise.framework.globalvars` Namespace Reference

Global variables (often influenced by command line parameters)

Variables

- `logdebug` = False
- `removetempdirs` = True

1.39.1 Detailed Description

Global variables (often influenced by command line parameters)

1.39.2 Variable Documentation

1.39.2.1 `anise.framework.globalvars.logdebug` = False

1.39.2.2 `anise.framework.globalvars.removetempdirs` = True

1.40 `anise.framework.imports` Namespace Reference

Imports some modules for availability within `_projectdesc`.

1.40.1 Detailed Description

Imports some modules for availability within `_projectdesc`. used internally.

1.41 `anise.framework.projectinformations` Namespace Reference

Variables

- `string version` = '1.0.1054'
- `string homepage` = 'https://ginger.ws/wiki/pino/projs/anise'

1.41.1 Variable Documentation

1.41.1.1 `string anise.framework.projectinformations.homepage` = 'https://ginger.ws/wiki/pino/projs/anise'

1.41.1.2 `string anise.framework.projectinformations.version` = '1.0.1054'

1.42 `anise.framework.projects` Namespace Reference

Implementation for the anise project object.

Classes

- class [Project](#)
Implementation for project objects.

Functions

- def [setcurrentproject](#)
Sets the current project object (used internally by anise).
- def [currentproject](#)
Returns the current project object.

Variables

- [_currentproject](#) = None

1.42.1 Detailed Description

Implementation for the anise project object.

1.42.2 Function Documentation

1.42.2.1 `def anise.framework.projects.currentproject ()`

Returns the current project object.

1.42.2.2 `def anise.framework.projects.setcurrentproject (project)`

Sets the current project object (used internally by anise).

1.42.3 Variable Documentation

1.42.3.1 `anise.framework.projects._currentproject = None`

1.43 anise.framework.results Namespace Reference

Typical anise result structures.

Classes

- class [Filestructure](#)
This class is a special anise return type which is a file/folder (to be fetched to the current dir).
- class [TempDir](#)
A special [anise.framework.results.Filestructure](#) which automatically creates a temporary directory in background.

Functions

- def [source_to_filestructure](#)
Provide a given file source description as an [anise.framework.results.Filestructure](#) object.

1.43.1 Detailed Description

Typical anise result structures.

1.43.2 Function Documentation

1.43.2.1 `def anise.framework.results.source_to_filestructure (source)`

Provide a given file source description as an [anise.framework.results.Filestructure](#) object.

Parameters

<i>source</i>	A file source description.
---------------	----------------------------

1.44 anise.utils Namespace Reference

Utility functions from several categories which make life easier.

Namespaces

- [basic](#)
Basic helpers for filesystem and other operating system related stuff.
- [documentation](#)
Documentation generation.
- [exceptions](#)
Exception subclasses.
- [logging](#)
Logging.
- [packaging](#)
Package generation.
- [ssh](#)
Utilities for ssh usage.
- [svn](#)
Utilities for Subversion usage.

1.44.1 Detailed Description

Utility functions from several categories which make life easier. All stuff here is not tightly bound to anise infrastructure but can easily imported and used from other programs as well.

1.45 anise.utils.basic Namespace Reference

Basic helpers for filesystem and other operating system related stuff.

Classes

- class [Mount](#)
Can be used to mount a volume.
- class [ChDir](#)
Class for temporarily changing the current working directory in a scope of the `with` keyword.

Functions

- def [writetofile](#)
writes some content to a file
- def [readfromfile](#)
reads the content from a file
- def [mkdirs](#)
creates a directory path (recursively)
- def [dict2object](#)
- def [params2dict](#)
- def [maketarball](#)
makes a tarball from a given directory structure
- def [call](#)
executes a command line
- def [getuniqueidentifier](#)
returns a random and (statistically nearly) unique string
- def [linewrap](#)
Wraps a string, so all lines are not larger than 80 characters.
- def [stripname](#)
Strips a name to something very bare with only alphabetic letters.
- def [createobjectstructure](#)

1.45.1 Detailed Description

Basic helpers for filesystem and other operating system related stuff.

1.45.2 Function Documentation

1.45.2.1 `def anise.utils.basic.call (cmdline, shell=False, decode=True, raise_on_errors=None)`

executes a command line

1.45.2.2 `def anise.utils.basic.createobjectstructure (root, newnames)`

1.45.2.3 `def anise.utils.basic.dict2object (d)`

1.45.2.4 `def anise.utils.basic.getuniqueidentifier ()`

returns a random and (statistically nearly) unique string

1.45.2.5 `def anise.utils.basic.linewrap (txt)`

Wraps a string, so all lines are not larger than 80 characters.

1.45.2.6 `def anise.utils.basic.maketarball (rootdir, tarname)`

makes a tarball from a given directory structure

1.45.2.7 `def anise.utils.basic.mkdirs (dname)`

creates a directory path (recursively)

1.45.2.8 `def anise.utils.basic.params2dict (d)`

1.45.2.9 `def anise.utils.basic.readfromfile (filename, onestring = True)`

reads the content from a file

1.45.2.10 `def anise.utils.basic.stripname (name, lowercase = True)`

Strips a name to something very bare with only alphabetic letters.

Parameters

<i>name</i>	The input name.
<i>lowercase</i>	If only lowercase characters are allowed.

1.45.2.11 `def anise.utils.basic.writetofile (filename, content)`

writes some content to a file

1.46 anise.utils.documentation Namespace Reference

Documentation generation.

Functions

- `def _scalecolor`
Internally used for scaling a color tuple to a new one with the same hue but a given brightness.
- `def _derivecolors`
Internally used for computing some css colors for the given basecolor (with different brightnesses)
- `def markdown_downgrade_headings`
Decreases the heading level for all headings in a Markdown document by a certain degree.
- `def generate_doxygen_developerdocumentation_ashtml`
Writes a developer documentation as html (powered by Doxygen).
- `def generate_doxygen_userdocumentation_ashtml`
Writes all user documentations as html (powered by Doxygen).
- `def generate_doxygen_userdocumentation_asmanpage`
Writes one user documentation page as manpage (powered by Doxygen) and returns the path to the result file.
- `def generate_doxygen_userdocumentation_asplaintext`
Writes one user documentation page as plain text (powered by Doxygen).
- `def generate_doxygen_userdocumentation_aspdf`
Writes one user documentation page as pdf (powered by Doxygen).
- `def generate_doxygen_homepage`
Writes the project homepage (powered by Doxygen).
- `def generate_doxygen_documentation`
Writes some output with Doxygen.

Variables

- string [_doxygen_baseconfig](#)
- string [_doxygen_htmlheader](#)
- string [_doxygen_htmlfooter](#)
- string [_doxygen_additionalcss_userdocumentation](#)
- string [_doxygen_additionalcss_homepage](#)
- string [_doxygen_additionalcss](#)

1.46.1 Detailed Description

Documentation generation.

1.46.2 Function Documentation

1.46.2.1 `def anise.utils.documentation._derivecolors(c) [private]`

Internally used for computing some css colors for the given basecolor (with different brightnesses)

1.46.2.2 `def anise.utils.documentation._scalecolor(c, avg) [private]`

Internally used for scaling a color tuple to a new one with the same hue but a given brightness.

1.46.2.3 `def anise.utils.documentation.generate_doxygen_developerdocumentation_ashtml(head1, head2 = "Developer Documentation", head3 = "", additionaltext = "", sourcedir, targetdir, basecolor, bannerimage = None, documentations = [], doxygensettings = "")`

Writes a developer documentation as html (powered by Doxygen).

You find it afterwards in `index.html` (and tons of other files). This will contain all user documentations as well, so references into it will work.

Parameters

<i>head1</i>	The first heading (typically the project name)
<i>head2</i>	The second heading
<i>head3</i>	The third heading (typically used for a special flavour of developer documentation)
<i>additionaltext</i>	Auxiliary source code passed to Doxygen for injecting additional content (e.g. mainpage text)
<i>sourcedir</i>	The directory where the sourcefiles are
<i>targetdir</i>	The directory for the html output
<i>basecolor</i>	The basecolor for colouring the output
<i>bannerimage</i>	Absolute path to the banner image (will be copied)
<i>documentations</i>	An anise documentations tuple for further pages, which do not exist in the sourcecode (e.g. the readme sources) as added by <code>anise.features.documentation.add</code>
<i>doxygensettings</i>	Additional Doxygen configuration for finetuning.

1.46.2.4 `def anise.utils.documentation.generate_doxygen_documentation(head1, head2, head3, head4, sourcedir, targetdir, doxygensettings, outputformat, additionalsources, basecolor, bannerimage, headlinks, showdoxygennavbar, css, documentations, anisedatadir = None, htmlheadertranslation = None)`

Writes some output with Doxygen.

Mostly used by some higher-level functions in this module.

Parameters

<i>head1</i>	The first heading
<i>head2</i>	The second heading
<i>head3</i>	The third heading
<i>head4</i>	The fourth heading
<i>sourcedir</i>	The directory where the sourcefiles are
<i>targetdir</i>	The directory for the output
<i>doxygensettings</i>	Additional Doxygen configuration for finetuning.
<i>outputformat</i>	The desired output format (as in doxygen config's GENERATE_XXX keys)
<i>additional-sources</i>	Auxiliary source code passed to Doxygen for injecting additional content
<i>basecolor</i>	The basecolor for colouring the output
<i>bannerimage</i>	Absolute path to the banner image (will be copied)
<i>headlinks</i>	Links in the heading panel (html output only)
<i>showdoxygen-navbar</i>	If the doxygen navigation bar should be visible
<i>css</i>	Additional css (html output only)
<i>documentations</i>	An anise documentations tuple for further pages, which do not exist in the sourcecode (e.g. the readme sources) as added by <code>anise.features.documentation.add</code>
<i>anisedatadir</i>	Absolute path to the anise data dir (needed for finding icons, ...; typically somewhere in the anise installation path)
<i>htmlheader-translation</i>	A function which can replace stuff in the default html header (html output only)

1.46.2.5 `def anise.utils.documentation.generate_doxygen_homepage(head1, head2, head3, head4, sourcedir, targetdir, basecolor, bannerimage, sections, anisedatadir, doxygensettings = "")`

Writes the project homepage (powered by Doxygen).

You find it afterwards in `index.html` (and tons of other files).

Parameters

<i>head1</i>	The first heading (typically the project name)
<i>head2</i>	The second heading (typically a one-liner description)
<i>head3</i>	The third heading (typically version information)
<i>sourcedir</i>	The directory where the sourcefiles are
<i>targetdir</i>	The directory for the html output
<i>basecolor</i>	The basecolor for colouring the output
<i>bannerimage</i>	Absolute path to the banner image (will be copied)
<i>sections</i>	A list of (sectiontitle,sectionsource) tuples for the actual content.
<i>anisedatadir</i>	Absolute path to the anise data dir (needed for finding icons, ...; typically somewhere in the anise installation path)
<i>doxygensettings</i>	Additional Doxygen configuration for finetuning.

1.46.2.6 `def anise.utils.documentation.generate_doxygen_userdocumentation_ashtml(head1, head2, head3 = "", head4 = "", additionaltext = "", sourcedir, targetdir, basecolor, bannerimage = None, documentations, anisedatadir, doxygensettings = "")`

Writes all user documentations as html (powered by Doxygen).

You find it afterwards in the `(documentkey).html` files (and tons of other files). This will contain the developer documentation as well, so references into it will work.

Parameters

<i>head1</i>	The first heading (typically the project name)
<i>head2</i>	The second heading
<i>head3</i>	The third heading
<i>additionaltext</i>	Auxiliary source code passed to Doxygen for injecting additional content
<i>sourcedir</i>	The directory where the sourcefiles are
<i>targetdir</i>	The directory for the html output
<i>basecolor</i>	The basecolor for colouring the output
<i>bannerimage</i>	Absolute path to the banner image (will be copied)
<i>documentations</i>	An anise documentations tuple for further pages, which do not exist in the sourcecode (e.g. the readme sources) as added by <code>anise.features.documentation.add</code>
<i>doxygensettings</i>	Additional Doxygen configuration for finetuning.

1.46.2.7 `def anise.utils.documentation.generate_doxygen_userdocumentation_asmanpage (projectname, entityname, section, additionaltext = " ", sourcedir, targetdir, documentations, targetdocumentation, doxygensettings = " ")`

Writes one user documentation page as manpage (powered by Doxygen) and returns the path to the result file.

Parameters

<i>projectname</i>	The project name
<i>entityname</i>	The <code>NAME</code> for the manpage (typically the name of the command, function or whatever)
<i>section</i>	The section number (see <code>man manpage</code>)
<i>additionaltext</i>	Auxiliary source code passed to Doxygen for injecting additional content
<i>sourcedir</i>	The directory where the sourcefiles are
<i>targetdir</i>	The directory for the manpage output
<i>documentations</i>	An anise documentations tuple for further pages, which do not exist in the sourcecode (e.g. the readme sources) as added by <code>anise.features.documentation.add</code>
<i>targetdocumentation</i>	The key for the documentation which must be written
<i>doxygensettings</i>	Additional Doxygen configuration for finetuning.

1.46.2.8 `def anise.utils.documentation.generate_doxygen_userdocumentation_aspdf (head1, head2, additionaltext = " ", sourcedir, targetfile, documentations, targetdocumentation, doxygensettings = " ", includedeveloperdoc = False)`

Writes one user documentation page as pdf (powered by Doxygen).

Parameters

<i>head1</i>	The first heading (typically the project name)
<i>head2</i>	The second heading
<i>additionaltext</i>	Auxiliary source code passed to Doxygen for injecting additional content
<i>sourcedir</i>	The directory where the sourcefiles are
<i>targetfile</i>	The target file for the pdf output
<i>documentations</i>	An anise documentations tuple for further pages, which do not exist in the sourcecode (e.g. the readme sources) as added by <code>anise.features.documentation.add</code>
<i>targetdocumentation</i>	The key for the documentation which must be written
<i>doxygensettings</i>	Additional Doxygen configuration for finetuning.
<i>includedeveloperdoc</i>	If the pdf shall include the developer documentation.

1.46.2.9 `def anise.utils.documentation.generate_doxygen_userdocumentation_asplaintext (projectname, head,
additionaltext = " ", sourcedir, targetfile, documentations, targetdocumentation, doxygensettings = " ")`

Writes one user documentation page as plain text (powered by Doxygen).

Parameters

<i>projectname</i>	The project name
<i>head</i>	The heading
<i>additionaltext</i>	Auxiliary source code passed to Doxygen for injecting additional content
<i>sourcedir</i>	The directory where the sourcefiles are
<i>targetfile</i>	The target file for the plaintext output
<i>documentations</i>	An anise documentations tuple for further pages, which do not exist in the sourcecode (e.g. the readme sources) as added by <code>anise.features.documentation.add</code>
<i>targetdocumen- tation</i>	The key for the documentation which must be written
<i>doxygensettings</i>	Additional Doxygen configuration for finetuning.

1.46.2.10 `def anise.utils.documentation.markdown_downgrade_headings (docsrc, levels)`

Decreases the heading level for all headings in a Markdown document by a certain degree.

This is needed for embedding one Markdown document in another one without breaking the section hierarchy.

1.46.3 Variable Documentation

1.46.3.1 `string anise.utils.documentation._doxygen_additionalcss`1.46.3.2 `string anise.utils.documentation._doxygen_additionalcss_homepage`1.46.3.3 `string anise.utils.documentation._doxygen_additionalcss_userdocumentation`

Initial value:

```

1 = """
2 div.header {
3     visibility: collapse;
4     height: 0;
5 }
6 div.contents {
7     margin-left: 40pt;
8     margin-right: 40pt;
9 }
10 body, table, div, p, dl {
11     font: 400 1em/1.2em sans-serif;
12 }
13 """

```

1.46.3.4 `string anise.utils.documentation._doxygen_baseconfig`1.46.3.5 `string anise.utils.documentation._doxygen_htmlfooter`

Initial value:

```

1 = """
2             <!-- </div> -->
3             <div id="elevator">
4                 <a href="#top" title="jump to top">&#x2912;</a>
5             </div>
6             <div>&nbsp;</div>
7         </div>
8     </div>
9     <!--BEGIN GENERATE_TREEVIEW-->
10    <div id="nav-path" class="navpath" style="visibility:collapse; height: 0;" >
11    </div>
12    <!--END GENERATE_TREEVIEW-->
13    <!--BEGIN !GENERATE_TREEVIEW-->
14    <!--END !GENERATE_TREEVIEW-->
15    </body>

```

```
16 </html>
17 """
```

1.46.3.6 string anise.utils.documentation._doxygen_htmlheader

1.47 anise.utils.exceptions Namespace Reference

Exception subclasses.

Classes

- class [AniseError](#)
An error in the execution of anise.
- class [BadInputError](#)
A bad value was given as some input.
- class [BadFormatError](#)
Some input is wrongly formatted.
- class [BadCommunicationError](#)
The communication with some other component failed.
- class [ProcessExecutionFailedError](#)
The execution of an external process failed.
- class [InternalError](#)
An internal error in the anise code occurred.
- class [RequirementsMissingError](#)
Something required is not in place.
- class [UnexpectedSituationError](#)
An unexpected situation led to an error.

1.47.1 Detailed Description

Exception subclasses.

1.48 anise.utils.logging Namespace Reference

Logging.

Classes

- class [Severity](#)
Log message severities.

Functions

- def [logmessage](#)
Logs a text message.
- def [logdebug](#)
Logs a text message with debug severity.
- def [loginfo](#)

- Logs a text message with info severity.*
- def [logwarning](#)
Logs a text message with warning severity.
- def [logerror](#)
Logs a text message with error severity.
- def [logexception](#)
Logs an exception.
- def [log_start_stop](#)
Function decorator for logging start and end of a function call.

1.48.1 Detailed Description

Logging.

1.48.2 Function Documentation

1.48.2.1 `def anise.utils.logging.log_start_stop (description)`

Function decorator for logging start and end of a function call.

1.48.2.2 `def anise.utils.logging.logdebug (text, printcaller = True, skiplevels = 1)`

Logs a text message with debug severity.

Parameters

<i>text</i>	The log message.
<i>printcaller</i>	Also logs the caller name.
<i>skiplevels</i>	If given, it controls the position on the call stack which is used for logging the caller name.

1.48.2.3 `def anise.utils.logging.logerror (text, printcaller = True, skiplevels = 1)`

Logs a text message with error severity.

Parameters

<i>text</i>	The log message.
<i>printcaller</i>	Also logs the caller name.
<i>skiplevels</i>	If given, it controls the position on the call stack which is used for logging the caller name.

1.48.2.4 `def anise.utils.logging.logexception (exception)`

Logs an exception.

Parameters

<i>exception</i>	The occurred exception. This only works from within the <code>except</code> -handler of this exception.
------------------	---

1.48.2.5 `def anise.utils.logging.loginfo (text, printcaller = True, skiplevels = 1)`

Logs a text message with info severity.

Parameters

<i>text</i>	The log message.
<i>printcaller</i>	Also logs the caller name.
<i>skiplevels</i>	If given, it controls the position on the call stack which is used for logging the caller name.

1.48.2.6 `def anise.utils.logging.logmessage (text, severity, printcaller = True, skiplevels = 1)`

Logs a text message.

Parameters

<i>text</i>	The log message.
<i>severity</i>	The message severity. One of the values from anise.utils.logging.Severity .
<i>printcaller</i>	Also logs the caller name.
<i>skiplevels</i>	If given, it controls the position on the call stack which is used for logging the caller name.

1.48.2.7 `def anise.utils.logging.logwarning (text, printcaller = True, skiplevels = 1)`

Logs a text message with warning severity.

Parameters

<i>text</i>	The log message.
<i>printcaller</i>	Also logs the caller name.
<i>skiplevels</i>	If given, it controls the position on the call stack which is used for logging the caller name.

1.49 anise.utils.packaging Namespace Reference

Package generation.

Classes

- class [PythonWheelPackageApplicationLink](#)
Data structure for one menu entry which comes in a Python wheel package.
- class [DebianPackageMenuEntry](#)
Data structure for one menu entry which comes in a Debian package.
- class [WindowsPackageMenuEntry](#)
Data structure for one menu entry which comes in a Windows package.

Functions

- def [_isdatafile](#)
- def [buildpythonwheelpackage](#)
Builds a Python wheel package.
- def [builddebpackage](#)
Builds a Debian package (.deb).
- def [buildfirefoxaddonpackage](#)
Builds a Firefox addon.
- def [buildandroidpackage_with_ant](#)
Builds an Android package with Ant.

- def [buildwindowsinstallerpackage](#)
Builds a Windows installer executable (.exe).

1.49.1 Detailed Description

Package generation.

1.49.2 Function Documentation

1.49.2.1 def `anise.utils.packaging._isdatafile (fullpath)` [private]

1.49.2.2 def `anise.utils.packaging.buildandroidpackage_with_ant (projectroot, name, version, targetname, androidpath)`

Builds an Android package with Ant.

Parameters

<i>projectroot</i>	The path to the Android package.
<i>name</i>	The project name.
<i>version</i>	The project version.
<i>targetname</i>	The Ant target name which shall be built.
<i>androidpath</i>	Path to the Android sdk's <code>android</code> tool.

1.49.2.3 def `anise.utils.packaging.builddebpackage (name, version, image, licensename, authors, email, summary, description, section, architecture, websiteurl, menuentries, executablelinks, dependencies, services, rawpkgpath, projectfsroot, prerm = " ", postinst = " ")`

Builds a Debian package (.deb).

Parameters

<i>name</i>	The package name.
<i>version</i>	The package version string.
<i>image</i>	A project screenshot
<i>licensename</i>	The name of the project license.
<i>authors</i>	Project authors as list of strings.
<i>email</i>	Project email adress.
<i>summary</i>	Short project description.
<i>description</i>	Longer project description.
<i>section</i>	The Debian section name.
<i>architecture</i>	The Debian architecture name.
<i>websiteurl</i>	The project homepage url.
<i>menuentries</i>	Menu entries to register.
<i>executablelinks</i>	Symlinks to executables to be created.
<i>dependencies</i>	List of dependencies.
<i>services</i>	List of services to be registered.
<i>rawpkgpath</i>	The raw source directory; can be damaged.
<i>projectfsroot</i>	The project source directory path.
<i>prerm</i>	Bash code snippet executed before the package is removed on a target machine.

<i>postinst</i>	Bash code snippet executed after the package is installed on a target machine.
-----------------	--

1.49.2.4 `def anise.utils.packaging.buildfirefoxaddonpackage (addonsdkpath, projectroot, name, version)`

Builds a Firefox addon.

Parameters

<i>addonsdkpath</i>	The path to the Firefox development sdk.
<i>projectroot</i>	Path to the Firefox addon project directory.
<i>name</i>	Name of the package.
<i>version</i>	Version of the package.

1.49.2.5 `def anise.utils.packaging.buildpythonwheelpackage (name, version, homepage, description, long_description, license, maturity, authors, email, keywords, fsroot, applicationlinks, datafileacceptor = _isdatafile)`

Builds a Python wheel package.

Parameters

<i>name</i>	The package name.
<i>version</i>	The package version string.
<i>homepage</i>	The homepage url.
<i>description</i>	A short project description.
<i>long_description</i>	A longer project description.
<i>license</i>	The name of the project license.
<i>maturity</i>	The project maturity string (may be <code>None</code>).
<i>authors</i>	The project authors as list of string.
<i>email</i>	The contact email address.
<i>keywords</i>	Project keywords for search services.
<i>fsroot</i>	The root directory for the package content (may be damaged).
<i>applicationlinks</i>	List of packaging.PythonWheelPackageApplicationLink which this package provide.
<i>datafileacceptor</i>	An optional function which helps deciding if a file in the source folder is a data file.

1.49.2.6 `def anise.utils.packaging.buildwindowsinstallerpackage (name, version, licensefile, authors, email, summary, architecture, websiteurl, menuentries, rawpkgpath, projectfsroot, projecticon = None, globalscript = "", oninitscript = "", oninstallscript = "")`

Builds a Windows installer executable (.exe).

Parameters

<i>name</i>	The package name.
<i>version</i>	The package version string.
<i>licensefile</i>	The path to the full text of the project's license.
<i>authors</i>	Project authors as list of strings.
<i>email</i>	Project email address.
<i>summary</i>	Short project description.
<i>architecture</i>	The Debian architecture name.

<i>websiteurl</i>	The project homepage url.
<i>menuentries</i>	Menu entries to register.
<i>rawpkgpath</i>	The raw source directory; can be damaged.
<i>projectfsroot</i>	The project source directory path.
<i>globalscript</i>	NSIS script global part.
<i>oninitscript</i>	NSIS script which runs at initialization of the installer.
<i>oninstallscript</i>	NSIS script which runs after file copying of the installer.

1.50 anise.utils.ssh Namespace Reference

Utilities for ssh usage.

Classes

- class [Mount](#)
Mounts a volume via ssh.

Functions

- def [call](#)
Executes a command via ssh.
- def [copy](#)
Copies a local directory to a remote place via ssh.

1.50.1 Detailed Description

Utilities for ssh usage.

1.50.2 Function Documentation

1.50.2.1 `def anise.utils.ssh.call (dst, cmd, options = [], port = 22, identityfile = None)`

Executes a command via ssh.

1.50.2.2 `def anise.utils.ssh.copy (src, dst, options = [], port = 22, identityfile = None)`

Copies a local directory to a remote place via ssh.

Parameters

<i>src</i>	The source.
<i>dst</i>	The destination.
<i>options</i>	Optional additional mount options.
<i>port</i>	Optional tcp port.
<i>identityfile</i>	Optional identity file for authentication.

1.51 anise.utils.svn Namespace Reference

Utilities for Subversion usage.

Classes

- class [_Version](#)

Functions

- def [getsvnrevision](#)
Determines the revision number of the Subversion working copy.
- def [listsvnrepo](#)
Determines the file tree of an Subversion repository.

1.51.1 Detailed Description

Utilities for Subversion usage.

1.51.2 Function Documentation

1.51.2.1 `def anise.utils.svn.getsvnrevision (d)`

Determines the revision number of the Subversion working copy.

1.51.2.2 `def anise.utils.svn.listsvnrepo (directory)`

Determines the file tree of an Subversion repository.

Parameters

<i>directory</i>	The directory you want to list.
------------------	---------------------------------

1.52 anise.features.documentation._Exports Class Reference

Public Member Functions

- def [__init__](#)
- def [add](#)
Exports an anise documentation to some targets.

Public Attributes

- [list](#)

1.52.1 Constructor & Destructor Documentation

1.52.1.1 `def anise.features.documentation._Exports.__init__ (self)`

1.52.2 Member Function Documentation

1.52.2.1 `def anise.features.documentation._Exports.add (self, key, outchannels)`

Exports an anise documentation to some targets.

Parameters

<i>key</i>	The documentation key name you want to export.
<i>outchannels</i>	A list of output channels which specify an documentation export target.

1.52.3 Member Data Documentation

1.52.3.1 anise.features.documentation._Exports.list

The documentation for this class was generated from the following file:

- anise/features/[documentation.py](#)

1.53 anise.features.datainjections._Pool Class Reference

Public Member Functions

- def [add](#)
Add a data injection.

1.53.1 Member Function Documentation

1.53.1.1 def anise.features.datainjections._Pool.add (*self*, *method*, *outfile*, *propnames* = None, *do_in_source* = False, *params*)

Add a data injection.

Parameters

<i>method</i>	The implementation for the injection.
<i>outfile</i>	Path to the target file for the injections (relative to anise project root path).
<i>propnames</i>	List of entry names you want to inject (beyond the few default ones).
<i>do_in_source</i>	If the source itself should be updated as well (instead of only the on-the-fly copies for packages).
<i>params</i>	Additional parameters for calling the injection implementation.

The documentation for this class was generated from the following file:

- anise/features/[datainjections.py](#)

1.54 anise.features.dependencies._Pool Class Reference

Public Member Functions

- def [__init__](#)
- def [add](#)

Public Attributes

- [list](#)

1.54.1 Constructor & Destructor Documentation

1.54.1.1 `def anise.features.dependencies._Pool.__init__(self)`

1.54.2 Member Function Documentation

1.54.2.1 `def anise.features.dependencies._Pool.add(self, dependency)`

1.54.3 Member Data Documentation

1.54.3.1 `anise.features.dependencies._Pool.list`

The documentation for this class was generated from the following file:

- [anise/features/dependencies.py](#)

1.55 anise.features.documentation._Pool Class Reference

Public Member Functions

- `def __init__`
- `def add`

Adds an anise documentation.

Public Attributes

- `list`

1.55.1 Constructor & Destructor Documentation

1.55.1.1 `def anise.features.documentation._Pool.__init__(self)`

1.55.2 Member Function Documentation

1.55.2.1 `def anise.features.documentation._Pool.add(self, key, src)`

Adds an anise documentation.

Parameters

<i>key</i>	The documentation key for storage (later used for referencing this text).
<i>src</i>	The documentation source text.

1.55.3 Member Data Documentation

1.55.3.1 `anise.features.documentation._Pool.list`

The documentation for this class was generated from the following file:

- [anise/features/documentation.py](#)

1.56 anise.features.distributables._Pool Class Reference

Public Member Functions

- def [__init__](#)
- def [addgroup](#)

Public Attributes

- [list](#)

1.56.1 Constructor & Destructor Documentation

1.56.1.1 def anise.features.distributables._Pool.__init__(*self*)

1.56.2 Member Function Documentation

1.56.2.1 def anise.features.distributables._Pool.addgroup(*self*, *task*, *params*)

1.56.3 Member Data Documentation

1.56.3.1 anise.features.distributables._Pool.list

The documentation for this class was generated from the following file:

- anise/features/[distributables.py](#)

1.57 anise.features.imagegalleries._Pool Class Reference

Public Member Functions

- def [__init__](#)
- def [add](#)

Public Attributes

- [list](#)

1.57.1 Constructor & Destructor Documentation

1.57.1.1 def anise.features.imagegalleries._Pool.__init__(*self*)

1.57.2 Member Function Documentation

1.57.2.1 def anise.features.imagegalleries._Pool.add(*self*, *name*, *gallerydir*)

1.57.3 Member Data Documentation

1.57.3.1 anise.features.imagegalleries._Pool.list

The documentation for this class was generated from the following file:

- [anise/features/imagegalleries.py](#)

1.58 anise.features.releasing._Releasetasks Class Reference

Public Member Functions

- def [add](#)
Specifies a new task and adds it into the releasing chain.

1.58.1 Member Function Documentation

1.58.1.1 def anise.features.releasing._Releasetasks.add (self, task, params)

Specifies a new task and adds it into the releasing chain.

Parameters

<i>The</i>	task implementation to add.
<i>Additional</i>	parameters for calling that task.

The documentation for this class was generated from the following file:

- [anise/features/releasing.py](#)

1.59 anise.features.homepage._Sections Class Reference

Public Member Functions

- def [__init__](#)
- def [add](#)

Public Attributes

- [list](#)

1.59.1 Constructor & Destructor Documentation

1.59.1.1 def anise.features.homepage._Sections.__init__ (self)

1.59.2 Member Function Documentation

1.59.2.1 def anise.features.homepage._Sections.add (self, index, key, text)

1.59.3 Member Data Documentation

1.59.3.1 anise.features.homepage._Sections.list

The documentation for this class was generated from the following file:

- [anise/features/homepage.py](#)

1.60 anise.utils.svn._Version Class Reference

Public Member Functions

- def [__init__](#)
- def [isvalid](#)
- def [format](#)

Public Attributes

- [build](#)
- [tag](#)

1.60.1 Constructor & Destructor Documentation

1.60.1.1 `def anise.utils.svn._Version.__init__(self, build, tag)`

1.60.2 Member Function Documentation

1.60.2.1 `def anise.utils.svn._Version.format(self)`

1.60.2.2 `def anise.utils.svn._Version.isvalid(self)`

1.60.3 Member Data Documentation

1.60.3.1 `anise.utils.svn._Version.build`

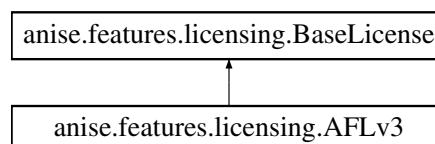
1.60.3.2 `anise.utils.svn._Version.tag`

The documentation for this class was generated from the following file:

- `anise/utils/svn.py`

1.61 anise.features.licensing.AFLv3 Class Reference

Inheritance diagram for `anise.features.licensing.AFLv3`:



Public Member Functions

- def [__init__](#)
- def [getfilename](#)
Returns the full path of the license text file.
- def [getlicensetext](#)
Returns the full license text.

Public Attributes

- [namedebian](#)
- [namepythonsetuputils](#)
- [namefirefox](#)
- [name](#)
- [filename](#)

1.61.1 Constructor & Destructor Documentation

1.61.1.1 `def anise.features.licensing.AFLv3.__init__(self)`

1.61.2 Member Function Documentation

1.61.2.1 `def anise.features.licensing.BaseLicense.getfilename(self)` [inherited]

Returns the full path of the license text file.

1.61.2.2 `def anise.features.licensing.BaseLicense.getlicensetext(self)` [inherited]

Returns the full license text.

1.61.3 Member Data Documentation

1.61.3.1 `anise.features.licensing.BaseLicense.filename` [inherited]

1.61.3.2 `anise.features.licensing.BaseLicense.name` [inherited]

1.61.3.3 `anise.features.licensing.BaseLicense.namedebian` [inherited]

1.61.3.4 `anise.features.licensing.BaseLicense.namefirefox` [inherited]

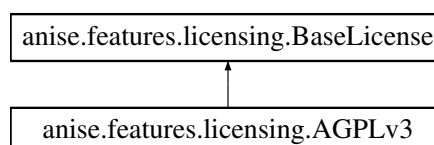
1.61.3.5 `anise.features.licensing.BaseLicense.namepythonsetuputils` [inherited]

The documentation for this class was generated from the following file:

- [anise/features/licensing.py](#)

1.62 anise.features.licensing.AGPLv3 Class Reference

Inheritance diagram for `anise.features.licensing.AGPLv3`:



Public Member Functions

- `def __init__`

- def [getfilename](#)
Returns the full path of the license text file.
- def [getlicensetext](#)
Returns the full license text.

Public Attributes

- [namedebian](#)
- [namepythonsetuputils](#)
- [namefirefox](#)
- [name](#)
- [filename](#)

1.62.1 Constructor & Destructor Documentation

1.62.1.1 `def anise.features.licensing.AGPLv3.__init__(self)`

1.62.2 Member Function Documentation

1.62.2.1 `def anise.features.licensing.BaseLicense.getfilename(self)` [inherited]

Returns the full path of the license text file.

1.62.2.2 `def anise.features.licensing.BaseLicense.getlicensetext(self)` [inherited]

Returns the full license text.

1.62.3 Member Data Documentation

1.62.3.1 `anise.features.licensing.BaseLicense.filename` [inherited]

1.62.3.2 `anise.features.licensing.BaseLicense.name` [inherited]

1.62.3.3 `anise.features.licensing.BaseLicense.namedebian` [inherited]

1.62.3.4 `anise.features.licensing.BaseLicense.namefirefox` [inherited]

1.62.3.5 `anise.features.licensing.BaseLicense.namepythonsetuputils` [inherited]

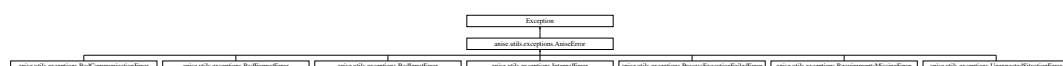
The documentation for this class was generated from the following file:

- `anise/features/licensing.py`

1.63 anise.utils.exceptions.AniseError Class Reference

An error in the execution of anise.

Inheritance diagram for `anise.utils.exceptions.AniseError`:



Public Member Functions

- `def __call__`

1.63.1 Detailed Description

An error in the execution of anise.

1.63.2 Member Function Documentation

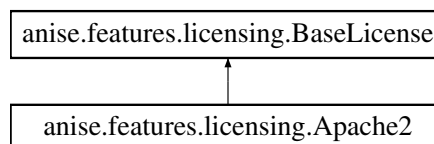
1.63.2.1 `def anise.utils.exceptions.AniseError.__call__(self, args)`

The documentation for this class was generated from the following file:

- `anise/utlils/exceptions.py`

1.64 anise.features.licensing.Apache2 Class Reference

Inheritance diagram for `anise.features.licensing.Apache2`:



Public Member Functions

- `def __init__`
- `def getfilename`
Returns the full path of the license text file.
- `def getlicensetext`
Returns the full license text.

Public Attributes

- `namedebian`
- `namepythonsetuputils`
- `namefirefox`
- `name`
- `filename`

1.64.1 Constructor & Destructor Documentation

1.64.1.1 `def anise.features.licensing.Apache2.__init__(self)`

1.64.2 Member Function Documentation

1.64.2.1 `def anise.features.licensing.BaseLicense.getfilename(self)` *[inherited]*

Returns the full path of the license text file.

1.64.2.2 `def anise.features.licensing.BaseLicense.getlicensetext (self)` [inherited]

Returns the full license text.

1.64.3 Member Data Documentation

1.64.3.1 `anise.features.licensing.BaseLicense.filename` [inherited]

1.64.3.2 `anise.features.licensing.BaseLicense.name` [inherited]

1.64.3.3 `anise.features.licensing.BaseLicense.namedebian` [inherited]

1.64.3.4 `anise.features.licensing.BaseLicense.namefirefox` [inherited]

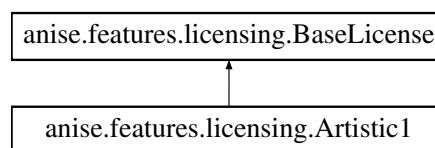
1.64.3.5 `anise.features.licensing.BaseLicense.namepythonsetuputils` [inherited]

The documentation for this class was generated from the following file:

- [anise/features/licensing.py](#)

1.65 anise.features.licensing.Artistic1 Class Reference

Inheritance diagram for `anise.features.licensing.Artistic1`:



Public Member Functions

- `def __init__`
- `def getfilename`
Returns the full path of the license text file.
- `def getlicensetext`
Returns the full license text.

Public Attributes

- `namedebian`
- `namepythonsetuputils`
- `namefirefox`
- `name`
- `filename`

1.65.1 Constructor & Destructor Documentation

1.65.1.1 `def anise.features.licensing.Artistic1.__init__ (self)`

1.65.2 Member Function Documentation

1.65.2.1 `def anise.features.licensing.BaseLicense.getfilename (self)` [inherited]

Returns the full path of the license text file.

1.65.2.2 `def anise.features.licensing.BaseLicense.getlicensetext (self)` [inherited]

Returns the full license text.

1.65.3 Member Data Documentation

1.65.3.1 `anise.features.licensing.BaseLicense.filename` [inherited]

1.65.3.2 `anise.features.licensing.BaseLicense.name` [inherited]

1.65.3.3 `anise.features.licensing.BaseLicense.namedebian` [inherited]

1.65.3.4 `anise.features.licensing.BaseLicense.namefirefox` [inherited]

1.65.3.5 `anise.features.licensing.BaseLicense.namepythonsetuputils` [inherited]

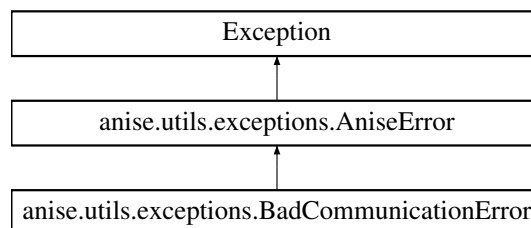
The documentation for this class was generated from the following file:

- [anise/features/licensing.py](#)

1.66 anise.utils.exceptions.BadCommunicationError Class Reference

The communication with some other component failed.

Inheritance diagram for `anise.utils.exceptions.BadCommunicationError`:



Public Member Functions

- `def __call__`

1.66.1 Detailed Description

The communication with some other component failed.

1.66.2 Member Function Documentation

1.66.2.1 `def anise.utils.exceptions.AniseError.__call__(self, args)` [inherited]

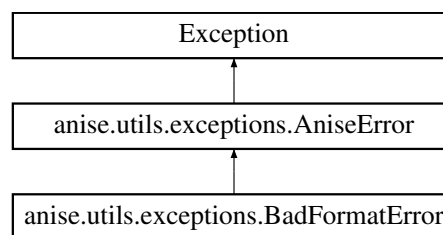
The documentation for this class was generated from the following file:

- [anise/utils/exceptions.py](#)

1.67 anise.utils.exceptions.BadFormatError Class Reference

Some input is wrongly formatted.

Inheritance diagram for `anise.utils.exceptions.BadFormatError`:



Public Member Functions

- `def __call__`

1.67.1 Detailed Description

Some input is wrongly formatted.

1.67.2 Member Function Documentation

1.67.2.1 `def anise.utils.exceptions.AniseError.__call__(self, args)` [inherited]

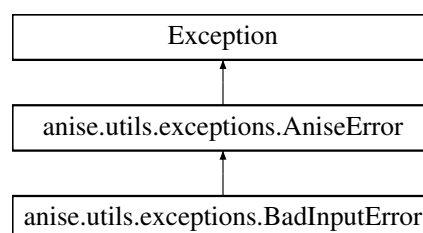
The documentation for this class was generated from the following file:

- [anise/utils/exceptions.py](#)

1.68 anise.utils.exceptions.BadInputError Class Reference

A bad value was given as some input.

Inheritance diagram for `anise.utils.exceptions.BadInputError`:



Public Member Functions

- [def __call__](#)

1.68.1 Detailed Description

A bad value was given as some input.

1.68.2 Member Function Documentation

1.68.2.1 `def anise.utils.exceptions.AniseError.__call__(self, args) [inherited]`

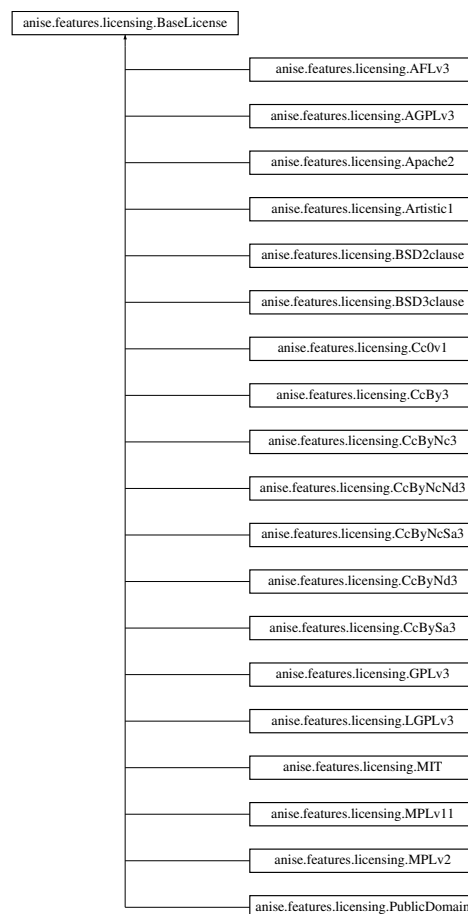
The documentation for this class was generated from the following file:

- [anise/utils/exceptions.py](#)

1.69 anise.features.licensing.BaseLicense Class Reference

Base class for licenses.

Inheritance diagram for `anise.features.licensing.BaseLicense`:



Public Member Functions

- [def __init__](#)

- def [getfilename](#)
Returns the full path of the license text file.
- def [getlicensetext](#)
Returns the full license text.

Public Attributes

- [namedebian](#)
- [namepythonsetuputils](#)
- [namefirefox](#)
- [name](#)
- [filename](#)

1.69.1 Detailed Description

Base class for licenses.

This interface is probably used by many features.

1.69.2 Constructor & Destructor Documentation

- 1.69.2.1 `def anise.features.licensing.BaseLicense.__init__(self, namedebian = None, namepythonsetuputils = None, namefirefox = None, name = None, filename = None)`

1.69.3 Member Function Documentation

- 1.69.3.1 `def anise.features.licensing.BaseLicense.getfilename(self)`

Returns the full path of the license text file.

- 1.69.3.2 `def anise.features.licensing.BaseLicense.getlicensetext(self)`

Returns the full license text.

1.69.4 Member Data Documentation

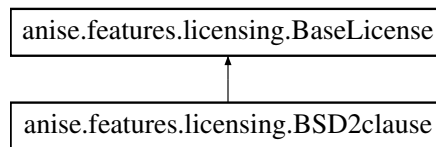
- 1.69.4.1 `anise.features.licensing.BaseLicense.filename`
- 1.69.4.2 `anise.features.licensing.BaseLicense.name`
- 1.69.4.3 `anise.features.licensing.BaseLicense.namedebian`
- 1.69.4.4 `anise.features.licensing.BaseLicense.namefirefox`
- 1.69.4.5 `anise.features.licensing.BaseLicense.namepythonsetuputils`

The documentation for this class was generated from the following file:

- [anise/features/licensing.py](#)

1.70 anise.features.licensing.BSD2clause Class Reference

Inheritance diagram for anise.features.licensing.BSD2clause:



Public Member Functions

- def `__init__`
- def `getfilename`
Returns the full path of the license text file.
- def `getlicensetext`
Returns the full license text.

Public Attributes

- `namedebian`
- `namepythonsetuputils`
- `namefirefox`
- `name`
- `filename`

1.70.1 Constructor & Destructor Documentation

1.70.1.1 `def anise.features.licensing.BSD2clause.__init__(self)`

1.70.2 Member Function Documentation

1.70.2.1 `def anise.features.licensing.BaseLicense.getfilename(self)` [inherited]

Returns the full path of the license text file.

1.70.2.2 `def anise.features.licensing.BaseLicense.getlicensetext(self)` [inherited]

Returns the full license text.

1.70.3 Member Data Documentation

1.70.3.1 `anise.features.licensing.BaseLicense.filename` [inherited]

1.70.3.2 `anise.features.licensing.BaseLicense.name` [inherited]

1.70.3.3 `anise.features.licensing.BaseLicense.namedebian` [inherited]

1.70.3.4 `anise.features.licensing.BaseLicense.namefirefox` [inherited]

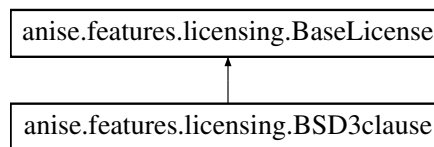
1.70.3.5 anise.features.licensing.BaseLicense.namepythonsetuputils [inherited]

The documentation for this class was generated from the following file:

- anise/features/[licensing.py](#)

1.71 anise.features.licensing.BSD3clause Class Reference

Inheritance diagram for anise.features.licensing.BSD3clause:



Public Member Functions

- def [__init__](#)
- def [getfilename](#)
Returns the full path of the license text file.
- def [getlicensetext](#)
Returns the full license text.

Public Attributes

- [namedebian](#)
- [namepythonsetuputils](#)
- [namefirefox](#)
- [name](#)
- [filename](#)

1.71.1 Constructor & Destructor Documentation

1.71.1.1 def anise.features.licensing.BSD3clause.__init__(self)

1.71.2 Member Function Documentation

1.71.2.1 def anise.features.licensing.BaseLicense.getfilename(self) [inherited]

Returns the full path of the license text file.

1.71.2.2 def anise.features.licensing.BaseLicense.getlicensetext(self) [inherited]

Returns the full license text.

1.71.3 Member Data Documentation

1.71.3.1 `anise.features.licensing.BaseLicense.filename` `[inherited]`

1.71.3.2 `anise.features.licensing.BaseLicense.name` `[inherited]`

1.71.3.3 `anise.features.licensing.BaseLicense.namedebian` `[inherited]`

1.71.3.4 `anise.features.licensing.BaseLicense.namefirefox` `[inherited]`

1.71.3.5 `anise.features.licensing.BaseLicense.namepythonsetuputils` `[inherited]`

The documentation for this class was generated from the following file:

- [anise/features/licensing.py](#)

1.72 `anise.features.packages.debian.Category` Class Reference

Enumeration for Debian's categories for software packages.

Static Public Attributes

- tuple `ApplicationsAccessibility` = ("Applications/Accessibility", "System")
- tuple `ApplicationsAmateurradio` = ("Applications/Amateur Radio", "Utility")
- tuple `ApplicationsDatamanagement` = ("Applications/Data Management", "System")
- tuple `ApplicationsEditors` = ("Applications/Editors", "Utility")
- tuple `ApplicationsEducation` = ("Applications/Education", "Education")
- tuple `ApplicationsEmulators` = ("Applications/Emulators", "System")
- tuple `ApplicationsFilemanagement` = ("Applications/File Management", "System")
- tuple `ApplicationsGraphics` = ("Applications/Graphics", "Graphics")
- tuple `ApplicationsMobiledevices` = ("Applications/Mobile Devices", "Utility")
- tuple `ApplicationsNetwork` = ("Applications/Network", "Network")
- tuple `ApplicationsNetworkCommunication` = ("Applications/Network/Communication", "Network")
- tuple `ApplicationsNetworkFiletransfer` = ("Applications/Network/File Transfer", "Network")
- tuple `ApplicationsNetworkMonitoring` = ("Applications/Network/Monitoring", "Network")
- tuple `ApplicationsNetworkWebbrowsing` = ("Applications/Network/Web Browsing", "Network")
- tuple `ApplicationsNetworkWebnews` = ("Applications/Network/Web News", "Network")
- tuple `ApplicationsOffice` = ("Applications/Office", "Office")
- tuple `ApplicationsProgramming` = ("Applications/Programming", "Development")
- tuple `ApplicationsProjectmanagement` = ("Applications/Project Management", "Development")
- tuple `ApplicationsScience` = ("Applications/Science", "Education")
- tuple `ApplicationsScienceAstronomy` = ("Applications/Science/Astronomy", "Education")
- tuple `ApplicationsScienceBiology` = ("Applications/Science/Biology", "Education")
- tuple `ApplicationsScienceChemistry` = ("Applications/Science/Chemistry", "Education")
- tuple `ApplicationsScienceDataanalysis` = ("Applications/Science/Data Analysis", "Education")
- tuple `ApplicationsScienceElectronics` = ("Applications/Science/Electronics", "Education")
- tuple `ApplicationsScienceEngineering` = ("Applications/Science/Engineering", "Education")
- tuple `ApplicationsScienceGeoscience` = ("Applications/Science/Geoscience", "Education")
- tuple `ApplicationsScienceMathematics` = ("Applications/Science/Mathematics", "Education")
- tuple `ApplicationsScienceMedicine` = ("Applications/Science/Medicine", "Education")
- tuple `ApplicationsSciencePhysics` = ("Applications/Science/Physics", "Education")
- tuple `ApplicationsScienceSocial` = ("Applications/Science/Social", "Education")
- tuple `ApplicationsShells` = ("Applications/Shells", "System")

- tuple `ApplicationsSounds` = ("Applications/Sound", "AudioVideo")
- tuple `ApplicationsSystem` = ("Applications/System", "System")
- tuple `ApplicationsSystemAdministration` = ("Applications/System/Administration", "System")
- tuple `ApplicationsSystemHardware` = ("Applications/System/Hardware", "System")
- tuple `ApplicationsSystemLanguageenvironment` = ("Applications/System/Language Environment", "System")
- tuple `ApplicationsSystemMonitoring` = ("Applications/System/Monitoring", "System")
- tuple `ApplicationsSystemPackagemanagement` = ("Applications/System/Package Management", "System")
- tuple `ApplicationsSystemSecurity` = ("Applications/System/Security", "System")
- tuple `ApplicationsTerminalemulators` = ("Applications/Terminal Emulators", "System")
- tuple `ApplicationsText` = ("Applications/Text", "Utility")
- tuple `ApplicationsTvandrado` = ("Applications/TV and Radio", "AudioVideo")
- tuple `ApplicationsViewers` = ("Applications/Viewers", "Utility")
- tuple `ApplicationsVideo` = ("Applications/Video", "AudioVideo")
- tuple `ApplicationsWebdevelopment` = ("Applications/Web Development", "Development")
- tuple `GamesAction` = ("Games/Action", "Game")
- tuple `GamesAdventure` = ("Games/Adventure", "Game")
- tuple `GamesBlocks` = ("Games/Blocks", "Game")
- tuple `GamesBoard` = ("Games/Board", "Game")
- tuple `GamesCard` = ("Games/Card", "Game")
- tuple `GamesPuzzles` = ("Games/Puzzles", "Game")
- tuple `GamesSimulation` = ("Games/Simulation", "Game")
- tuple `GamesStrategy` = ("Games/Strategy", "Game")
- tuple `GamesTools` = ("Games/Tools", "Game")
- tuple `GamesToys` = ("Games/Toys", "Game")
- tuple `Help` = ("Help", "System")
- tuple `ScreenSaving` = ("Screen/Saving", "System")
- tuple `ScreenLocking` = ("Screen/Locking", "System")

1.72.1 Detailed Description

Enumeration for Debian's categories for software packages.

1.72.2 Member Data Documentation

- 1.72.2.1 tuple `anise.features.packages.debian.Category.ApplicationsAccessibility` = ("Applications/Accessibility", "System")
[static]
- 1.72.2.2 tuple `anise.features.packages.debian.Category.ApplicationsAmateurradio` = ("Applications/Amateur Radio", "Utility")
[static]
- 1.72.2.3 tuple `anise.features.packages.debian.Category.ApplicationsDatamanagement` = ("Applications/Data Management", "System") [static]
- 1.72.2.4 tuple `anise.features.packages.debian.Category.ApplicationsEditors` = ("Applications/Editors", "Utility") [static]
- 1.72.2.5 tuple `anise.features.packages.debian.Category.ApplicationsEducation` = ("Applications/Education", "Education")
[static]
- 1.72.2.6 tuple `anise.features.packages.debian.Category.ApplicationsEmulators` = ("Applications/Emulators", "System")
[static]
- 1.72.2.7 tuple `anise.features.packages.debian.Category.ApplicationsFilemanagement` = ("Applications/File Management", "System") [static]

- 1.72.2.8 `tuple anise.features.packages.debian.Category.ApplicationsGraphics = ("Applications/Graphics", "Graphics")`
`[static]`
- 1.72.2.9 `tuple anise.features.packages.debian.Category.ApplicationsMobiledevices = ("Applications/Mobile Devices", "Utility")`
`[static]`
- 1.72.2.10 `tuple anise.features.packages.debian.Category.ApplicationsNetwork = ("Applications/Network", "Network")`
`[static]`
- 1.72.2.11 `tuple anise.features.packages.debian.Category.ApplicationsNetworkCommunication =`
`("Applications/Network/Communication", "Network") [static]`
- 1.72.2.12 `tuple anise.features.packages.debian.Category.ApplicationsNetworkFiletransfer = ("Applications/Network/File`
`Transfer", "Network") [static]`
- 1.72.2.13 `tuple anise.features.packages.debian.Category.ApplicationsNetworkMonitoring = ("Applications/Network/Monitoring",`
`"Network") [static]`
- 1.72.2.14 `tuple anise.features.packages.debian.Category.ApplicationsNetworkWebbrowsing = ("Applications/Network/Web`
`Browsing", "Network") [static]`
- 1.72.2.15 `tuple anise.features.packages.debian.Category.ApplicationsNetworkWebnews = ("Applications/Network/Web News",`
`"Network") [static]`
- 1.72.2.16 `tuple anise.features.packages.debian.Category.ApplicationsOffice = ("Applications/Office", "Office") [static]`
- 1.72.2.17 `tuple anise.features.packages.debian.Category.ApplicationsProgramming = ("Applications/Programming",`
`"Development") [static]`
- 1.72.2.18 `tuple anise.features.packages.debian.Category.ApplicationsProjectmanagement = ("Applications/Project`
`Management", "Development") [static]`
- 1.72.2.19 `tuple anise.features.packages.debian.Category.ApplicationsScience = ("Applications/Science", "Education")`
`[static]`
- 1.72.2.20 `tuple anise.features.packages.debian.Category.ApplicationsScienceAstronomy = ("Applications/Science/Astronomy",`
`"Education") [static]`
- 1.72.2.21 `tuple anise.features.packages.debian.Category.ApplicationsScienceBiology = ("Applications/Science/Biology",`
`"Education") [static]`
- 1.72.2.22 `tuple anise.features.packages.debian.Category.ApplicationsScienceChemistry = ("Applications/Science/Chemistry",`
`"Education") [static]`
- 1.72.2.23 `tuple anise.features.packages.debian.Category.ApplicationsScienceDataanalysis = ("Applications/Science/Data`
`Analysis", "Education") [static]`
- 1.72.2.24 `tuple anise.features.packages.debian.Category.ApplicationsScienceElectronics = ("Applications/Science/Electronics",`
`"Education") [static]`
- 1.72.2.25 `tuple anise.features.packages.debian.Category.ApplicationsScienceEngineering =`
`("Applications/Science/Engineering", "Education") [static]`
- 1.72.2.26 `tuple anise.features.packages.debian.Category.ApplicationsScienceGeoscience =`
`("Applications/Science/Geoscience", "Education") [static]`

- 1.72.2.27 `tuple anise.features.packages.debian.Category.ApplicationsScienceMathematics = ("Applications/Science/Mathematics", "Education") [static]`
- 1.72.2.28 `tuple anise.features.packages.debian.Category.ApplicationsScienceMedicine = ("Applications/Science/Medicine", "Education") [static]`
- 1.72.2.29 `tuple anise.features.packages.debian.Category.ApplicationsSciencePhysics = ("Applications/Science/Physics", "Education") [static]`
- 1.72.2.30 `tuple anise.features.packages.debian.Category.ApplicationsScienceSocial = ("Applications/Science/Social", "Education") [static]`
- 1.72.2.31 `tuple anise.features.packages.debian.Category.ApplicationsShells = ("Applications/Shells", "System") [static]`
- 1.72.2.32 `tuple anise.features.packages.debian.Category.ApplicationsSounds = ("Applications/Sound", "AudioVideo") [static]`
- 1.72.2.33 `tuple anise.features.packages.debian.Category.ApplicationsSystem = ("Applications/System", "System") [static]`
- 1.72.2.34 `tuple anise.features.packages.debian.Category.ApplicationsSystemAdministration = ("Applications/System/Administration", "System") [static]`
- 1.72.2.35 `tuple anise.features.packages.debian.Category.ApplicationsSystemHardware = ("Applications/System/Hardware", "System") [static]`
- 1.72.2.36 `tuple anise.features.packages.debian.Category.ApplicationsSystemLanguageenvironment = ("Applications/System/Language Environment", "System") [static]`
- 1.72.2.37 `tuple anise.features.packages.debian.Category.ApplicationsSystemMonitoring = ("Applications/System/Monitoring", "System") [static]`
- 1.72.2.38 `tuple anise.features.packages.debian.Category.ApplicationsSystemPackagemanagement = ("Applications/System/Package Management", "System") [static]`
- 1.72.2.39 `tuple anise.features.packages.debian.Category.ApplicationsSystemSecurity = ("Applications/System/Security", "System") [static]`
- 1.72.2.40 `tuple anise.features.packages.debian.Category.ApplicationsTerminalemulators = ("Applications/Terminal Emulators", "System") [static]`
- 1.72.2.41 `tuple anise.features.packages.debian.Category.ApplicationsText = ("Applications/Text", "Utility") [static]`
- 1.72.2.42 `tuple anise.features.packages.debian.Category.ApplicationsTvandrado = ("Applications/TV and Radio", "AudioVideo") [static]`
- 1.72.2.43 `tuple anise.features.packages.debian.Category.ApplicationsVideo = ("Applications/Video", "AudioVideo") [static]`
- 1.72.2.44 `tuple anise.features.packages.debian.Category.ApplicationsViewers = ("Applications/Viewers", "Utility") [static]`
- 1.72.2.45 `tuple anise.features.packages.debian.Category.ApplicationsWebdevelopment = ("Applications/Web Development", "Development") [static]`
- 1.72.2.46 `tuple anise.features.packages.debian.Category.GamesAction = ("Games/Action", "Game") [static]`

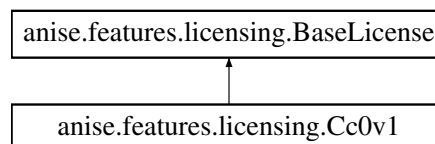
- 1.72.2.47 `tuple anise.features.packages.debian.Category.GamesAdventure = ("Games/Adventure", "Game") [static]`
- 1.72.2.48 `tuple anise.features.packages.debian.Category.GamesBlocks = ("Games/Blocks", "Game") [static]`
- 1.72.2.49 `tuple anise.features.packages.debian.Category.GamesBoard = ("Games/Board", "Game") [static]`
- 1.72.2.50 `tuple anise.features.packages.debian.Category.GamesCard = ("Games/Card", "Game") [static]`
- 1.72.2.51 `tuple anise.features.packages.debian.Category.GamesPuzzles = ("Games/Puzzles", "Game") [static]`
- 1.72.2.52 `tuple anise.features.packages.debian.Category.GamesSimulation = ("Games/Simulation", "Game") [static]`
- 1.72.2.53 `tuple anise.features.packages.debian.Category.GamesStrategy = ("Games/Strategy", "Game") [static]`
- 1.72.2.54 `tuple anise.features.packages.debian.Category.GamesTools = ("Games/Tools", "Game") [static]`
- 1.72.2.55 `tuple anise.features.packages.debian.Category.GamesToys = ("Games/Toys", "Game") [static]`
- 1.72.2.56 `tuple anise.features.packages.debian.Category.Help = ("Help", "System") [static]`
- 1.72.2.57 `tuple anise.features.packages.debian.Category.ScreenLocking = ("Screen/Locking", "System") [static]`
- 1.72.2.58 `tuple anise.features.packages.debian.Category.ScreenSaving = ("Screen/Saving", "System") [static]`

The documentation for this class was generated from the following file:

- [anise/features/packages/debian.py](#)

1.73 anise.features.licensing.Cc0v1 Class Reference

Inheritance diagram for `anise.features.licensing.Cc0v1`:



Public Member Functions

- `def __init__`
- `def getfilename`
Returns the full path of the license text file.
- `def getlicensetext`
Returns the full license text.

Public Attributes

- `namedebian`
- `namepythonsetuputils`
- `namefirefox`
- `name`
- `filename`

1.73.1 Constructor & Destructor Documentation

1.73.1.1 `def anise.features.licensing.Cc0v1.__init__(self)`

1.73.2 Member Function Documentation

1.73.2.1 `def anise.features.licensing.BaseLicense.getfilename(self)` [inherited]

Returns the full path of the license text file.

1.73.2.2 `def anise.features.licensing.BaseLicense.getlicensetext(self)` [inherited]

Returns the full license text.

1.73.3 Member Data Documentation

1.73.3.1 `anise.features.licensing.BaseLicense.filename` [inherited]

1.73.3.2 `anise.features.licensing.BaseLicense.name` [inherited]

1.73.3.3 `anise.features.licensing.BaseLicense.namedebian` [inherited]

1.73.3.4 `anise.features.licensing.BaseLicense.namefirefox` [inherited]

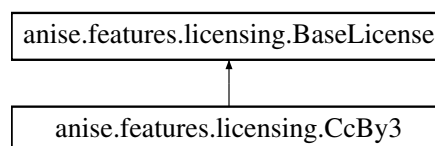
1.73.3.5 `anise.features.licensing.BaseLicense.namepythonsetuputils` [inherited]

The documentation for this class was generated from the following file:

- [anise/features/licensing.py](#)

1.74 anise.features.licensing.CcBy3 Class Reference

Inheritance diagram for `anise.features.licensing.CcBy3`:



Public Member Functions

- `def __init__`
- `def getfilename`
Returns the full path of the license text file.
- `def getlicensetext`
Returns the full license text.

Public Attributes

- [namedebian](#)
- [namepythonsetuputils](#)
- [namefirefox](#)
- [name](#)
- [filename](#)

1.74.1 Constructor & Destructor Documentation

1.74.1.1 `def anise.features.licensing.CcBy3.__init__(self)`

1.74.2 Member Function Documentation

1.74.2.1 `def anise.features.licensing.BaseLicense.getfilename(self)` [inherited]

Returns the full path of the license text file.

1.74.2.2 `def anise.features.licensing.BaseLicense.getlicensetext(self)` [inherited]

Returns the full license text.

1.74.3 Member Data Documentation

1.74.3.1 `anise.features.licensing.BaseLicense.filename` [inherited]

1.74.3.2 `anise.features.licensing.BaseLicense.name` [inherited]

1.74.3.3 `anise.features.licensing.BaseLicense.namedebian` [inherited]

1.74.3.4 `anise.features.licensing.BaseLicense.namefirefox` [inherited]

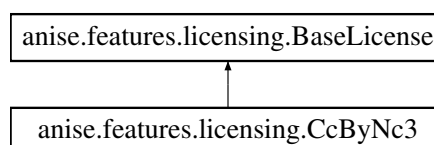
1.74.3.5 `anise.features.licensing.BaseLicense.namepythonsetuputils` [inherited]

The documentation for this class was generated from the following file:

- [anise/features/licensing.py](#)

1.75 anise.features.licensing.CcByNc3 Class Reference

Inheritance diagram for `anise.features.licensing.CcByNc3`:



Public Member Functions

- `def __init__`

- def [getfilename](#)
Returns the full path of the license text file.
- def [getlicensetext](#)
Returns the full license text.

Public Attributes

- [namedebian](#)
- [namepythonsetuputils](#)
- [namefirefox](#)
- [name](#)
- [filename](#)

1.75.1 Constructor & Destructor Documentation

1.75.1.1 `def anise.features.licensing.CcByNc3.__init__(self)`

1.75.2 Member Function Documentation

1.75.2.1 `def anise.features.licensing.BaseLicense.getfilename(self)` [inherited]

Returns the full path of the license text file.

1.75.2.2 `def anise.features.licensing.BaseLicense.getlicensetext(self)` [inherited]

Returns the full license text.

1.75.3 Member Data Documentation

1.75.3.1 `anise.features.licensing.BaseLicense.filename` [inherited]

1.75.3.2 `anise.features.licensing.BaseLicense.name` [inherited]

1.75.3.3 `anise.features.licensing.BaseLicense.namedebian` [inherited]

1.75.3.4 `anise.features.licensing.BaseLicense.namefirefox` [inherited]

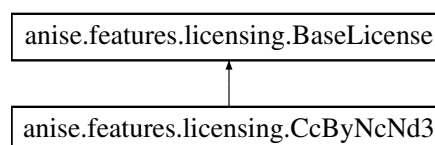
1.75.3.5 `anise.features.licensing.BaseLicense.namepythonsetuputils` [inherited]

The documentation for this class was generated from the following file:

- [anise/features/licensing.py](#)

1.76 anise.features.licensing.CcByNcNd3 Class Reference

Inheritance diagram for `anise.features.licensing.CcByNcNd3`:



Public Member Functions

- `def __init__`
- `def getfilename`
Returns the full path of the license text file.
- `def getlicensetext`
Returns the full license text.

Public Attributes

- `namedebian`
- `namepythonsetuputils`
- `namefirefox`
- `name`
- `filename`

1.76.1 Constructor & Destructor Documentation

1.76.1.1 `def anise.features.licensing.CcByNcNd3.__init__(self)`

1.76.2 Member Function Documentation

1.76.2.1 `def anise.features.licensing.BaseLicense.getfilename(self)` [inherited]

Returns the full path of the license text file.

1.76.2.2 `def anise.features.licensing.BaseLicense.getlicensetext(self)` [inherited]

Returns the full license text.

1.76.3 Member Data Documentation

1.76.3.1 `anise.features.licensing.BaseLicense.filename` [inherited]

1.76.3.2 `anise.features.licensing.BaseLicense.name` [inherited]

1.76.3.3 `anise.features.licensing.BaseLicense.namedebian` [inherited]

1.76.3.4 `anise.features.licensing.BaseLicense.namefirefox` [inherited]

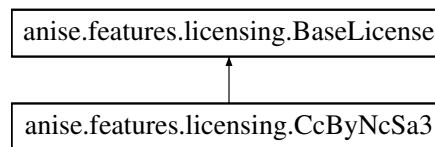
1.76.3.5 `anise.features.licensing.BaseLicense.namepythonsetuputils` [inherited]

The documentation for this class was generated from the following file:

- `anise/features/licensing.py`

1.77 anise.features.licensing.CcByNcSa3 Class Reference

Inheritance diagram for `anise.features.licensing.CcByNcSa3`:



Public Member Functions

- def [__init__](#)
- def [getfilename](#)
Returns the full path of the license text file.
- def [getlicensetext](#)
Returns the full license text.

Public Attributes

- [namedebian](#)
- [namepythonsetuputils](#)
- [namefirefox](#)
- [name](#)
- [filename](#)

1.77.1 Constructor & Destructor Documentation

1.77.1.1 `def anise.features.licensing.CcByNcSa3.__init__(self)`

1.77.2 Member Function Documentation

1.77.2.1 `def anise.features.licensing.BaseLicense.getfilename(self)` [inherited]

Returns the full path of the license text file.

1.77.2.2 `def anise.features.licensing.BaseLicense.getlicensetext(self)` [inherited]

Returns the full license text.

1.77.3 Member Data Documentation

1.77.3.1 `anise.features.licensing.BaseLicense.filename` [inherited]

1.77.3.2 `anise.features.licensing.BaseLicense.name` [inherited]

1.77.3.3 `anise.features.licensing.BaseLicense.namedebian` [inherited]

1.77.3.4 `anise.features.licensing.BaseLicense.namefirefox` [inherited]

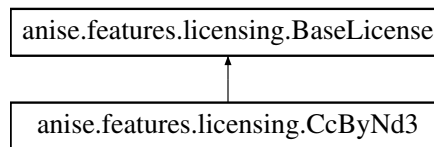
1.77.3.5 `anise.features.licensing.BaseLicense.namepythonsetuputils` [inherited]

The documentation for this class was generated from the following file:

- [anise/features/licensing.py](#)

1.78 anise.features.licensing.CcByNd3 Class Reference

Inheritance diagram for anise.features.licensing.CcByNd3:



Public Member Functions

- def `__init__`
- def `getfilename`
Returns the full path of the license text file.
- def `getlicensetext`
Returns the full license text.

Public Attributes

- `namedebian`
- `namepythonsetuputils`
- `namefirefox`
- `name`
- `filename`

1.78.1 Constructor & Destructor Documentation

1.78.1.1 `def anise.features.licensing.CcByNd3.__init__(self)`

1.78.2 Member Function Documentation

1.78.2.1 `def anise.features.licensing.BaseLicense.getfilename(self)` [inherited]

Returns the full path of the license text file.

1.78.2.2 `def anise.features.licensing.BaseLicense.getlicensetext(self)` [inherited]

Returns the full license text.

1.78.3 Member Data Documentation

1.78.3.1 `anise.features.licensing.BaseLicense.filename` [inherited]

1.78.3.2 `anise.features.licensing.BaseLicense.name` [inherited]

1.78.3.3 `anise.features.licensing.BaseLicense.namedebian` [inherited]

1.78.3.4 `anise.features.licensing.BaseLicense.namefirefox` [inherited]

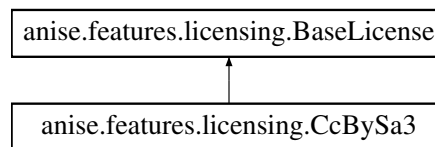
1.78.3.5 anise.features.licensing.BaseLicense.namepythonsetuputils [inherited]

The documentation for this class was generated from the following file:

- anise/features/[licensing.py](#)

1.79 anise.features.licensing.CcBySa3 Class Reference

Inheritance diagram for anise.features.licensing.CcBySa3:



Public Member Functions

- def [__init__](#)
- def [getfilename](#)
Returns the full path of the license text file.
- def [getlicensetext](#)
Returns the full license text.

Public Attributes

- [namedebian](#)
- [namepythonsetuputils](#)
- [namefirefox](#)
- [name](#)
- [filename](#)

1.79.1 Constructor & Destructor Documentation

1.79.1.1 def anise.features.licensing.CcBySa3.__init__(self)

1.79.2 Member Function Documentation

1.79.2.1 def anise.features.licensing.BaseLicense.getfilename(self) [inherited]

Returns the full path of the license text file.

1.79.2.2 def anise.features.licensing.BaseLicense.getlicensetext(self) [inherited]

Returns the full license text.

1.79.3 Member Data Documentation

1.79.3.1 `anise.features.licensing.BaseLicense.filename` `[inherited]`

1.79.3.2 `anise.features.licensing.BaseLicense.name` `[inherited]`

1.79.3.3 `anise.features.licensing.BaseLicense.namedebian` `[inherited]`

1.79.3.4 `anise.features.licensing.BaseLicense.namefirefox` `[inherited]`

1.79.3.5 `anise.features.licensing.BaseLicense.namepythonsetuputils` `[inherited]`

The documentation for this class was generated from the following file:

- `anise/features/licensing.py`

1.80 `anise.utils.basic.ChDir` Class Reference

Class for temporarily changing the current working directory in a scope of the `with` keyword.

Public Member Functions

- `def __init__`
- `def __enter__`
- `def __exit__`

Public Attributes

- `newpath`
- `oldpath`

1.80.1 Detailed Description

Class for temporarily changing the current working directory in a scope of the `with` keyword.

1.80.2 Constructor & Destructor Documentation

1.80.2.1 `def anise.utils.basic.ChDir.__init__(self, path)`

1.80.3 Member Function Documentation

1.80.3.1 `def anise.utils.basic.ChDir.__enter__(self)`

1.80.3.2 `def anise.utils.basic.ChDir.__exit__(self, type, value, traceback)`

1.80.4 Member Data Documentation

1.80.4.1 `anise.utils.basic.ChDir.newpath`

1.80.4.2 `anise.utils.basic.ChDir.oldpath`

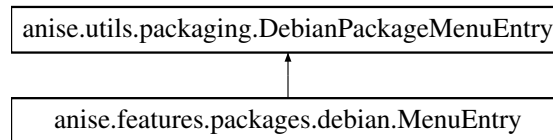
The documentation for this class was generated from the following file:

- [anise/utils/basic.py](#)

1.81 anise.utils.packaging.DebianPackageMenuEntry Class Reference

Data structure for one menu entry which comes in a Debian package.

Inheritance diagram for anise.utils.packaging.DebianPackageMenuEntry:



Public Member Functions

- `def __init__`

Public Attributes

- [name](#)
- [title](#)
- [category](#)
- [command](#)
- [gui](#)
- [icon](#)

1.81.1 Detailed Description

Data structure for one menu entry which comes in a Debian package.

1.81.2 Constructor & Destructor Documentation

1.81.2.1 `def anise.utils.packaging.DebianPackageMenuEntry.__init__(self, name, title, category, command, gui, icon)`

Parameters

<i>name</i>	Internal menu entry name.
<i>title</i>	Menu entry Title.
<i>category</i>	Menu entry category.
<i>command</i>	Menu entry command.
<i>gui</i>	If the command opens a gui (instead of a terminal application).
<i>icon</i>	The path to a menu entry icon.

1.81.3 Member Data Documentation

1.81.3.1 `anise.utils.packaging.DebianPackageMenuEntry.category`

1.81.3.2 `anise.utils.packaging.DebianPackageMenuEntry.command`

1.81.3.3 `anise.utils.packaging.DebianPackageMenuEntry.gui`

1.81.3.4 `anise.utils.packaging.DebianPackageMenuEntry.icon`

1.81.3.5 `anise.utils.packaging.DebianPackageMenuEntry.name`

1.81.3.6 `anise.utils.packaging.DebianPackageMenuEntry.title`

The documentation for this class was generated from the following file:

- [anise/utils/packaging.py](#)

1.82 `anise.features.dependencies.Dependency` Class Reference

A dependency.

Public Member Functions

- `def __init__`

Public Attributes

- `type`
- `objectname`
- `comment`
- `icon`
- `visible`
- `displayname`

1.82.1 Detailed Description

A dependency.

1.82.2 Constructor & Destructor Documentation

1.82.2.1 `def anise.features.dependencies.Dependency.__init__(self, type, objectname=None, comment="", icon=None, visible=True, displayname=None, kwargs)`

1.82.3 Member Data Documentation

1.82.3.1 `anise.features.dependencies.Dependency.comment`

1.82.3.2 `anise.features.dependencies.Dependency.displayname`

1.82.3.3 `anise.features.dependencies.Dependency.icon`

1.82.3.4 `anise.features.dependencies.Dependency.objectname`

1.82.3.5 `anise.features.dependencies.Dependency.type`

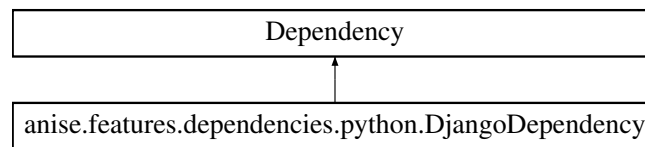
1.82.3.6 `anise.features.dependencies.Dependency.visible`

The documentation for this class was generated from the following file:

- [anise/features/dependencies.py](#)

1.83 anise.features.dependencies.python.DjangoDependency Class Reference

Inheritance diagram for anise.features.dependencies.python.DjangoDependency:



Public Member Functions

- def [__init__](#)

1.83.1 Constructor & Destructor Documentation

1.83.1.1 `def anise.features.dependencies.python.DjangoDependency.__init__(self, type, version, python2=False, kwargs)`

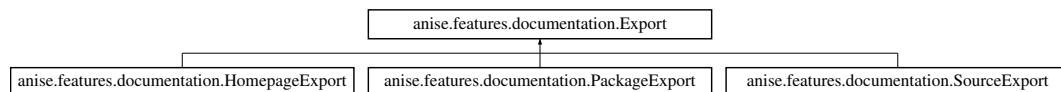
The documentation for this class was generated from the following file:

- anise/features/dependencies/[python.py](#)

1.84 anise.features.documentation.Export Class Reference

Describes a documentation export.

Inheritance diagram for anise.features.documentation.Export:



Public Member Functions

- def [__init__](#)
- def [supports](#)

1.84.1 Detailed Description

Describes a documentation export.

1.84.2 Constructor & Destructor Documentation

1.84.2.1 `def anise.features.documentation.Export.__init__(self)`

1.84.3 Member Function Documentation

1.84.3.1 `def anise.features.documentation.Export.supports(self, name)`

The documentation for this class was generated from the following file:

- [anise/features/documentation.py](#)

1.85 anise.framework.features.Feature Class Reference

A featue is some encapsuled anise functionality.

Public Member Functions

- `def __init__`

Public Attributes

- [name](#)
- [hooks](#)
- [featuremodule](#)

1.85.1 Detailed Description

A featue is some encapsuled anise functionality.

1.85.2 Constructor & Destructor Documentation

1.85.2.1 `def anise.framework.features.Feature.__init__(self, name, hooks, featuremodule)`

Parameters

<i>name</i>	The optional name of the feature.
<i>hooks</i>	All hook handlers this feature registers.

1.85.3 Member Data Documentation

1.85.3.1 `anise.framework.features.Feature.featuremodule`

1.85.3.2 `anise.framework.features.Feature.hooks`

1.85.3.3 `anise.framework.features.Feature.name`

The documentation for this class was generated from the following file:

- [anise/framework/features.py](#)

1.86 anise.features.distributables.FileGroup Class Reference

A file group.

Public Member Functions

- `def __init__`

Public Attributes

- [buildparams](#)
- [task](#)
- [description](#)
- [linkto](#)
- [name](#)

1.86.1 Detailed Description

A file group.

1.86.2 Constructor & Destructor Documentation

1.86.2.1 `def anise.features.distributables.FileGroup.__init__(self, task, prms)`

Parameters

<i>task</i>	The task implementation which generates the download group content.
<i>prms</i>	Additional parameters for calling <code>task</code> .

1.86.3 Member Data Documentation

1.86.3.1 `anise.features.distributables.FileGroup.buildparams`

1.86.3.2 `anise.features.distributables.FileGroup.description`

1.86.3.3 `anise.features.distributables.FileGroup.linkto`

1.86.3.4 `anise.features.distributables.FileGroup.name`

1.86.3.5 `anise.features.distributables.FileGroup.task`

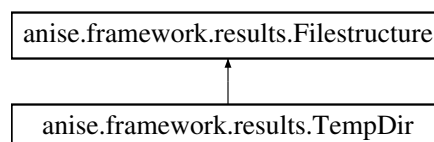
The documentation for this class was generated from the following file:

- [anise/features/distributables.py](#)

1.87 anise.framework.results.Filestructure Class Reference

This class is a special anise return type which is a file/folder (to be fetched to the current dir).

Inheritance diagram for `anise.framework.results.Filestructure`:



Public Member Functions

- `def __init__`

- `def dl`
Copies the complete filestructure or a subdirectory to a new destination.
- `def with_modified_rootname`
Returns a new (temporary) [anise.framework.results.Filestructure](#) clone with a new directory name.

Public Attributes

- `path`

1.87.1 Detailed Description

This class is a special anise return type which is a file/folder (to be fetched to the current dir).

Example:

```
def sometask():
    somedir = create_some_directory_somewhat()
    return results.Filestructure(somedir)
```

1.87.2 Constructor & Destructor Documentation

1.87.2.1 `def anise.framework.results.Filestructure.__init__(self, path)`

1.87.3 Member Function Documentation

1.87.3.1 `def anise.framework.results.Filestructure.dl(self, subpath = "", to = None)`

Copies the complete filestructure or a subdirectory to a new destination.

Parameters

<i>subpath</i>	if given, only this subdirectory will be copied.
<i>to</i>	if given, it specifies the destination path; if not given, a new directory in the current working directory will be used.

1.87.3.2 `def anise.framework.results.Filestructure.with_modified_rootname(self, newname)`

Returns a new (temporary) [anise.framework.results.Filestructure](#) clone with a new directory name.

Parameters

<i>newname</i>	The name which the new root directory should get.
----------------	---

1.87.4 Member Data Documentation

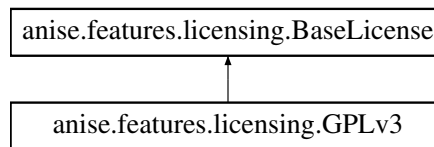
1.87.4.1 `anise.framework.results.Filestructure.path`

The documentation for this class was generated from the following file:

- [anise/framework/results.py](#)

1.88 anise.features.licensing.GPLv3 Class Reference

Inheritance diagram for anise.features.licensing.GPLv3:



Public Member Functions

- def `__init__`
- def `getfilename`
Returns the full path of the license text file.
- def `getlicensetext`
Returns the full license text.

Public Attributes

- `namedebian`
- `namepythonsetuputils`
- `namefirefox`
- `name`
- `filename`

1.88.1 Constructor & Destructor Documentation

1.88.1.1 `def anise.features.licensing.GPLv3.__init__(self)`

1.88.2 Member Function Documentation

1.88.2.1 `def anise.features.licensing.BaseLicense.getfilename(self)` [inherited]

Returns the full path of the license text file.

1.88.2.2 `def anise.features.licensing.BaseLicense.getlicensetext(self)` [inherited]

Returns the full license text.

1.88.3 Member Data Documentation

1.88.3.1 `anise.features.licensing.BaseLicense.filename` [inherited]

1.88.3.2 `anise.features.licensing.BaseLicense.name` [inherited]

1.88.3.3 `anise.features.licensing.BaseLicense.namedebian` [inherited]

1.88.3.4 `anise.features.licensing.BaseLicense.namefirefox` [inherited]

1.88.3.5 `anise.features.licensing.BaseLicense.namepythonsetuputils` [inherited]

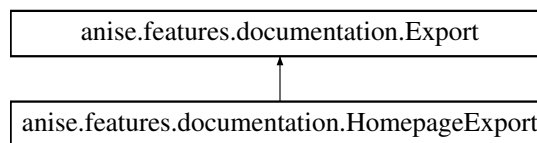
The documentation for this class was generated from the following file:

- [anise/features/licensing.py](#)

1.89 `anise.features.documentation.HomepageExport` Class Reference

Describes an export of an anise documentation directly into the homepage.

Inheritance diagram for `anise.features.documentation.HomepageExport`:



Public Member Functions

- `def __init__`
- `def supports`

Public Attributes

- `title`

1.89.1 Detailed Description

Describes an export of an anise documentation directly into the homepage.

1.89.2 Constructor & Destructor Documentation

1.89.2.1 `def anise.features.documentation.HomepageExport.__init__(self, title)`

Parameters

<i>title</i>	The title of the export, also used as link label in the documentation section of the homepage.
--------------	--

1.89.3 Member Function Documentation

1.89.3.1 `def anise.features.documentation.Export.supports(self, name)` [inherited]

1.89.4 Member Data Documentation

1.89.4.1 `anise.features.documentation.HomepageExport.title`

The documentation for this class was generated from the following file:

- [anise/features/documentation.py](#)

1.90 anise.framework.features.Hook Class Reference

An hook handler bound to some hook.

Public Member Functions

- [def __init__](#)

Public Attributes

- [eventname](#)
- [function](#)
- [provides](#)
- [requires](#)
- [prepares](#)

1.90.1 Detailed Description

An hook handler bound to some hook.

1.90.2 Constructor & Destructor Documentation

1.90.2.1 `def anise.framework.features.Hook.__init__(self, eventname, function, provides, requires, prepares)`

Parameters

<i>eventname</i>	The hook symbol.
<i>function</i>	The handler function.
<i>provides</i>	A space-separated list of symbol names this hook participates to provide (dependency provider).
<i>requires</i>	A space-separated list of symbol names this hook requires to be executed before.
<i>prepares</i>	A space-separated list of symbol names this hook must prepend.

1.90.3 Member Data Documentation

1.90.3.1 `anise.framework.features.Hook.eventname`

1.90.3.2 `anise.framework.features.Hook.function`

1.90.3.3 `anise.framework.features.Hook.prepares`

1.90.3.4 `anise.framework.features.Hook.provides`

1.90.3.5 `anise.framework.features.Hook.requires`

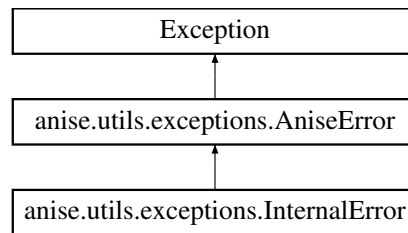
The documentation for this class was generated from the following file:

- [anise/framework/features.py](#)

1.91 anise.utils.exceptions.InternalError Class Reference

An internal error in the anise code occurred.

Inheritance diagram for anise.utils.exceptions.InternalError:



Public Member Functions

- [def `__call__`](#)

1.91.1 Detailed Description

An internal error in the anise code occurred.

1.91.2 Member Function Documentation

1.91.2.1 `def anise.utils.exceptions.AniseError.__call__(self, args)` [inherited]

The documentation for this class was generated from the following file:

- [anise/utils/exceptions.py](#)

1.92 anise.framework.features.internals Class Reference

Static Public Member Functions

- [def `injectdefaultstopproject`](#)
- [def `injecthookstopproject`](#)
- [def `loadfeatures`](#)
- [def `resetfeatures`](#)

1.92.1 Member Function Documentation

1.92.1.1 `def anise.framework.features.internals.injectdefaultstopproject(project)` [static]

1.92.1.2 `def anise.framework.features.internals.injecthookstopproject(project)` [static]

1.92.1.3 `def anise.framework.features.internals.loadfeatures(featurespath, name = " ")` [static]

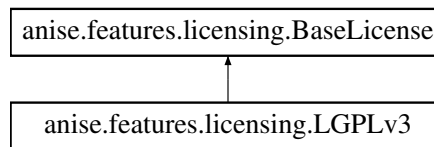
1.92.1.4 `def anise.framework.features.internals.resetfeatures()` [static]

The documentation for this class was generated from the following file:

- [anise/framework/features.py](#)

1.93 anise.features.licensing.LGPLv3 Class Reference

Inheritance diagram for anise.features.licensing.LGPLv3:



Public Member Functions

- def `__init__`
- def `getfilename`
Returns the full path of the license text file.
- def `getlicensetext`
Returns the full license text.

Public Attributes

- `namedebian`
- `namepythonsetuputils`
- `namefirefox`
- `name`
- `filename`

1.93.1 Constructor & Destructor Documentation

1.93.1.1 `def anise.features.licensing.LGPLv3.__init__(self)`

1.93.2 Member Function Documentation

1.93.2.1 `def anise.features.licensing.BaseLicense.getfilename(self)` [inherited]

Returns the full path of the license text file.

1.93.2.2 `def anise.features.licensing.BaseLicense.getlicensetext(self)` [inherited]

Returns the full license text.

1.93.3 Member Data Documentation

1.93.3.1 `anise.features.licensing.BaseLicense.filename` [inherited]

1.93.3.2 `anise.features.licensing.BaseLicense.name` [inherited]

1.93.3.3 `anise.features.licensing.BaseLicense.namedebian` [inherited]

1.93.3.4 `anise.features.licensing.BaseLicense.namefirefox` [inherited]

1.93.3.5 `anise.features.licensing.BaseLicense.namepythonsetuputils` `[inherited]`

The documentation for this class was generated from the following file:

- [anise/features/licensing.py](#)

1.94 `anise.features.projectdata.Maturity` Class Reference

This class helps describing the project state.

Static Public Attributes

- string `Planning` = 'Development Status :: 1 - Planning'
- string `PreAlpha` = 'Development Status :: 2 - Pre-Alpha'
- string `Alpha` = 'Development Status :: 3 - Alpha'
- string `Beta` = 'Development Status :: 4 - Beta'
- string `ProductionStable` = 'Development Status :: 5 - Production/Stable'
- string `Mature` = 'Development Status :: 6 - Mature'
- string `Inactive` = 'Development Status :: 7 - Inactive'

1.94.1 Detailed Description

This class helps describing the project state.

1.94.2 Member Data Documentation

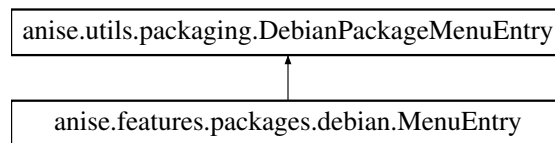
- 1.94.2.1 string `anise.features.projectdata.Maturity.Alpha` = 'Development Status :: 3 - Alpha' `[static]`
- 1.94.2.2 string `anise.features.projectdata.Maturity.Beta` = 'Development Status :: 4 - Beta' `[static]`
- 1.94.2.3 string `anise.features.projectdata.Maturity.Inactive` = 'Development Status :: 7 - Inactive' `[static]`
- 1.94.2.4 string `anise.features.projectdata.Maturity.Mature` = 'Development Status :: 6 - Mature' `[static]`
- 1.94.2.5 string `anise.features.projectdata.Maturity.Planning` = 'Development Status :: 1 - Planning' `[static]`
- 1.94.2.6 string `anise.features.projectdata.Maturity.PreAlpha` = 'Development Status :: 2 - Pre-Alpha' `[static]`
- 1.94.2.7 string `anise.features.projectdata.Maturity.ProductionStable` = 'Development Status :: 5 - Production/Stable' `[static]`

The documentation for this class was generated from the following file:

- [anise/features/projectdata.py](#)

1.95 `anise.features.packages.debian.MenuEntry` Class Reference

Inheritance diagram for `anise.features.packages.debian.MenuEntry`:



Public Attributes

- [name](#)
- [title](#)
- [category](#)
- [command](#)
- [gui](#)
- [icon](#)

1.95.1 Member Data Documentation

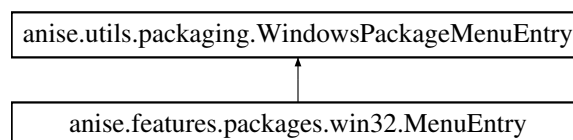
- 1.95.1.1 `anise.utils.packaging.DebianPackageMenuEntry.category` [inherited]
- 1.95.1.2 `anise.utils.packaging.DebianPackageMenuEntry.command` [inherited]
- 1.95.1.3 `anise.utils.packaging.DebianPackageMenuEntry.gui` [inherited]
- 1.95.1.4 `anise.utils.packaging.DebianPackageMenuEntry.icon` [inherited]
- 1.95.1.5 `anise.utils.packaging.DebianPackageMenuEntry.name` [inherited]
- 1.95.1.6 `anise.utils.packaging.DebianPackageMenuEntry.title` [inherited]

The documentation for this class was generated from the following file:

- `anise/features/packages/debian.py`

1.96 anise.features.packages.win32.MenuEntry Class Reference

Inheritance diagram for `anise.features.packages.win32.MenuEntry`:



Public Attributes

- [title](#)
- [command](#)
- [icon](#)

1.96.1 Member Data Documentation

1.96.1.1 `anise.utils.packaging.WindowsPackageMenuEntry.command` `[inherited]`

1.96.1.2 `anise.utils.packaging.WindowsPackageMenuEntry.icon` `[inherited]`

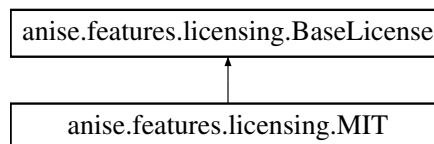
1.96.1.3 `anise.utils.packaging.WindowsPackageMenuEntry.title` `[inherited]`

The documentation for this class was generated from the following file:

- `anise/features/packages/win32.py`

1.97 `anise.features.licensing.MIT` Class Reference

Inheritance diagram for `anise.features.licensing.MIT`:



Public Member Functions

- `def __init__`
- `def getfilename`
Returns the full path of the license text file.
- `def getlicensetext`
Returns the full license text.

Public Attributes

- `namedebian`
- `namepythonsetuputils`
- `namefirefox`
- `name`
- `filename`

1.97.1 Constructor & Destructor Documentation

1.97.1.1 `def anise.features.licensing.MIT.__init__(self)`

1.97.2 Member Function Documentation

1.97.2.1 `def anise.features.licensing.BaseLicense.getfilename(self)` `[inherited]`

Returns the full path of the license text file.

1.97.2.2 `def anise.features.licensing.BaseLicense.getlicensetext(self)` `[inherited]`

Returns the full license text.

1.97.3 Member Data Documentation

1.97.3.1 `anise.features.licensing.BaseLicense.filename` [inherited]

1.97.3.2 `anise.features.licensing.BaseLicense.name` [inherited]

1.97.3.3 `anise.features.licensing.BaseLicense.namedebian` [inherited]

1.97.3.4 `anise.features.licensing.BaseLicense.namefirefox` [inherited]

1.97.3.5 `anise.features.licensing.BaseLicense.namepythonsetuputils` [inherited]

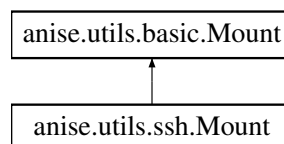
The documentation for this class was generated from the following file:

- [anise/features/licensing.py](#)

1.98 anise.utils.basic.Mount Class Reference

Can be used to mount a volume.

Inheritance diagram for `anise.utils.basic.Mount`:



Public Member Functions

- `def __init__`
- `def mount`
Manually mount the volume.
- `def __enter__`
- `def unmount`
Manually unmount the volume.
- `def __exit__`

Public Attributes

- `su`
- `mounted`
- `src`
- `dst`
- `options`
- `mounter`
- `unmounter`
- `needsroot`

1.98.1 Detailed Description

Can be used to mount a volume.

Can either be used automatically with the `with` keyword or manually.

1.98.2 Constructor & Destructor Documentation

1.98.2.1 `def anise.utils.basic.Mount.__init__(self, src, dst, options = [], mounter = ["mount"], unmounter = ["umount"], needsroot = True)`

Parameters

<i>src</i>	Mount source.
<i>dst</i>	Mount destination.
<i>options</i>	Optional additional mount options.
<i>mounter</i>	Optional string list overriding the default mount call.
<i>unmounter</i>	Optional string list overriding the default unmount call.
<i>needsroot</i>	If this mount action will need root permissions.

1.98.3 Member Function Documentation

1.98.3.1 `def anise.utils.basic.Mount.__enter__(self)`

1.98.3.2 `def anise.utils.basic.Mount.__exit__(self, type, value, traceback)`

1.98.3.3 `def anise.utils.basic.Mount.mount(self)`

Manually mount the volume.

1.98.3.4 `def anise.utils.basic.Mount.unmount(self)`

Manually unmount the volume.

1.98.4 Member Data Documentation

1.98.4.1 `anise.utils.basic.Mount.dst`

1.98.4.2 `anise.utils.basic.Mount.mounted`

1.98.4.3 `anise.utils.basic.Mount.mounter`

1.98.4.4 `anise.utils.basic.Mount.needsroot`

1.98.4.5 `anise.utils.basic.Mount.options`

1.98.4.6 `anise.utils.basic.Mount.src`

1.98.4.7 `anise.utils.basic.Mount.su`

1.98.4.8 `anise.utils.basic.Mount.unmounter`

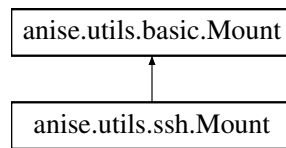
The documentation for this class was generated from the following file:

- [anise/utils/basic.py](#)

1.99 anise.utils.ssh.Mount Class Reference

Mounts a volume via ssh.

Inheritance diagram for anise.utils.ssh.Mount:



Public Member Functions

- def `__init__`
- def `mount`
Manually mount the volume.
- def `__enter__`
- def `unmount`
Manually unmount the volume.
- def `__exit__`

Public Attributes

- `su`
- `mounted`
- `src`
- `dst`
- `options`
- `mounter`
- `unmounter`
- `needsroot`

1.99.1 Detailed Description

Mounts a volume via ssh.

1.99.2 Constructor & Destructor Documentation

1.99.2.1 `def anise.utils.ssh.Mount.__init__(self, src, dst, options = [], port = 22, identityfile = None)`

Parameters

<i>src</i>	The remote mount source (an ssh address like user).
<i>dst</i>	The local mount destination.
<i>options</i>	Optional additional mount options.
<i>port</i>	Optional tcp port.
<i>identityfile</i>	Optional identity file for authentication.

1.99.3 Member Function Documentation

1.99.3.1 `def anise.utils.basic.Mount.__enter__(self)` [inherited]

1.99.3.2 `def anise.utils.basic.Mount.__exit__(self, type, value, traceback)` [inherited]

1.99.3.3 `def anise.utils.basic.Mount.mount (self)` [inherited]

Manually mount the volume.

1.99.3.4 `def anise.utils.basic.Mount.unmount (self)` [inherited]

Manually unmount the volume.

1.99.4 Member Data Documentation

1.99.4.1 `anise.utils.basic.Mount.dst` [inherited]

1.99.4.2 `anise.utils.basic.Mount.mounted` [inherited]

1.99.4.3 `anise.utils.basic.Mount.mounter` [inherited]

1.99.4.4 `anise.utils.basic.Mount.needsroot` [inherited]

1.99.4.5 `anise.utils.basic.Mount.options` [inherited]

1.99.4.6 `anise.utils.basic.Mount.src` [inherited]

1.99.4.7 `anise.utils.basic.Mount.su` [inherited]

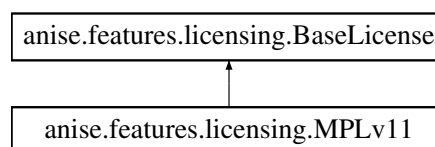
1.99.4.8 `anise.utils.basic.Mount.unmounter` [inherited]

The documentation for this class was generated from the following file:

- [anise/utils/ssh.py](#)

1.100 anise.features.licensing.MPLv11 Class Reference

Inheritance diagram for `anise.features.licensing.MPLv11`:



Public Member Functions

- `def __init__`
- `def getfilename`
Returns the full path of the license text file.
- `def getlicensetext`
Returns the full license text.

Public Attributes

- `namedebian`

- [namepythonsetuputils](#)
- [namefirefox](#)
- [name](#)
- [filename](#)

1.100.1 Constructor & Destructor Documentation

1.100.1.1 `def anise.features.licensing.MPLv11.__init__(self)`

1.100.2 Member Function Documentation

1.100.2.1 `def anise.features.licensing.BaseLicense.getfilename(self)` [inherited]

Returns the full path of the license text file.

1.100.2.2 `def anise.features.licensing.BaseLicense.getlicensetext(self)` [inherited]

Returns the full license text.

1.100.3 Member Data Documentation

1.100.3.1 `anise.features.licensing.BaseLicense.filename` [inherited]

1.100.3.2 `anise.features.licensing.BaseLicense.name` [inherited]

1.100.3.3 `anise.features.licensing.BaseLicense.namedebian` [inherited]

1.100.3.4 `anise.features.licensing.BaseLicense.namefirefox` [inherited]

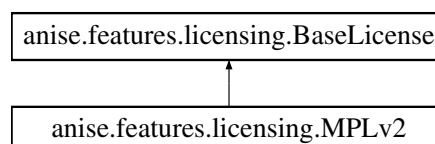
1.100.3.5 `anise.features.licensing.BaseLicense.namepythonsetuputils` [inherited]

The documentation for this class was generated from the following file:

- [anise/features/licensing.py](#)

1.101 anise.features.licensing.MPLv2 Class Reference

Inheritance diagram for `anise.features.licensing.MPLv2`:



Public Member Functions

- `def __init__`
- `def getfilename`
Returns the full path of the license text file.
- `def getlicensetext`
Returns the full license text.

Public Attributes

- [namedebian](#)
- [namepythonsetuputils](#)
- [namefirefox](#)
- [name](#)
- [filename](#)

1.101.1 Constructor & Destructor Documentation

1.101.1.1 `def anise.features.licensing.MPLv2.__init__(self)`

1.101.2 Member Function Documentation

1.101.2.1 `def anise.features.licensing.BaseLicense.getfilename(self)` [inherited]

Returns the full path of the license text file.

1.101.2.2 `def anise.features.licensing.BaseLicense.getlicensetext(self)` [inherited]

Returns the full license text.

1.101.3 Member Data Documentation

1.101.3.1 `anise.features.licensing.BaseLicense.filename` [inherited]

1.101.3.2 `anise.features.licensing.BaseLicense.name` [inherited]

1.101.3.3 `anise.features.licensing.BaseLicense.namedebian` [inherited]

1.101.3.4 `anise.features.licensing.BaseLicense.namefirefox` [inherited]

1.101.3.5 `anise.features.licensing.BaseLicense.namepythonsetuputils` [inherited]

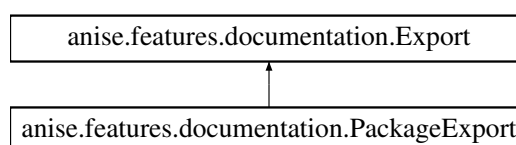
The documentation for this class was generated from the following file:

- [anise/features/licensing.py](#)

1.102 `anise.features.documentation.PackageExport` Class Reference

Describes an export of an anise documentation into the distribution package(s).

Inheritance diagram for `anise.features.documentation.PackageExport`:



Public Member Functions

- def [__init__](#)
- def [supports](#)

Public Attributes

- [ispackageexport](#)
- [ofile](#)
- [heading](#)
- [omode](#)
- [includedeveloperdoc](#)

1.102.1 Detailed Description

Describes an export of an anise documentation into the distribution package(s).

1.102.2 Constructor & Destructor Documentation

1.102.2.1 `def anise.features.documentation.PackageExport.__init__(self, ofile, heading, mode, includedeveloperdoc = False)`

Parameters

<i>ofile</i>	Target file path relative to the 'raw package's root.
<i>heading</i>	The title.
<i>mode</i>	Output format ('plain' or 'pdf').
<i>includedeveloperdoc</i>	pdf only: If the output shall include the developer documentation as well.

1.102.3 Member Function Documentation

1.102.3.1 `def anise.features.documentation.Export.supports(self, name)` [inherited]

1.102.4 Member Data Documentation

1.102.4.1 `anise.features.documentation.PackageExport.heading`

1.102.4.2 `anise.features.documentation.PackageExport.includedeveloperdoc`

1.102.4.3 `anise.features.documentation.PackageExport.ispackageexport`

1.102.4.4 `anise.features.documentation.PackageExport.ofile`

1.102.4.5 `anise.features.documentation.PackageExport.omode`

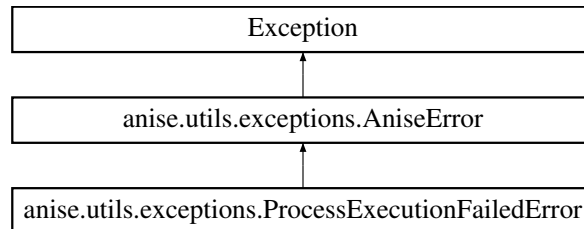
The documentation for this class was generated from the following file:

- anise/features/[documentation.py](#)

1.103 `anise.utils.exceptions.ProcessExecutionFailedError` Class Reference

The execution of an external process failed.

Inheritance diagram for `anise.utils.exceptions.ProcessExecutionFailedError`:



Public Member Functions

- `def __call__`

1.103.1 Detailed Description

The execution of an external process failed.

1.103.2 Member Function Documentation

1.103.2.1 `def anise.utils.exceptions.AniseError.__call__(self, args)` `[inherited]`

The documentation for this class was generated from the following file:

- `anise/utils/exceptions.py`

1.104 `anise.framework.projects.Project` Class Reference

Implementation for project objects.

Public Member Functions

- `def __init__`
- `def append`
- `def addhook`
- `def gethooks`

Public Attributes

- `hooks`

1.104.1 Detailed Description

Implementation for project objects.

1.104.2 Constructor & Destructor Documentation

1.104.2.1 `def anise.framework.projects.Project.__init__(self, d)`

1.104.3 Member Function Documentation

1.104.3.1 `def anise.framework.projects.Project.addhook (self, eventname, fct, params = None, provides = None, requires = None, prepares = None)`

1.104.3.2 `def anise.framework.projects.Project.append (self, d)`

1.104.3.3 `def anise.framework.projects.Project.gethooks (self, eventname)`

1.104.4 Member Data Documentation

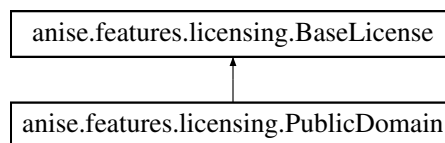
1.104.4.1 `anise.framework.projects.Project.hooks`

The documentation for this class was generated from the following file:

- [anise/framework/projects.py](#)

1.105 anise.features.licensing.PublicDomain Class Reference

Inheritance diagram for `anise.features.licensing.PublicDomain`:



Public Member Functions

- `def __init__`
- `def getfilename`
Returns the full path of the license text file.
- `def getlicensetext`
Returns the full license text.

Public Attributes

- `namedebian`
- `namepythonsetuputils`
- `namefirefox`
- `name`
- `filename`

1.105.1 Constructor & Destructor Documentation

1.105.1.1 `def anise.features.licensing.PublicDomain.__init__(self)`

1.105.2 Member Function Documentation

1.105.2.1 `def anise.features.licensing.BaseLicense.getfilename (self)` [inherited]

Returns the full path of the license text file.

1.105.2.2 `def anise.features.licensing.BaseLicense.getlicensetext (self)` [inherited]

Returns the full license text.

1.105.3 Member Data Documentation

1.105.3.1 `anise.features.licensing.BaseLicense.filename` [inherited]

1.105.3.2 `anise.features.licensing.BaseLicense.name` [inherited]

1.105.3.3 `anise.features.licensing.BaseLicense.namedebian` [inherited]

1.105.3.4 `anise.features.licensing.BaseLicense.namefirefox` [inherited]

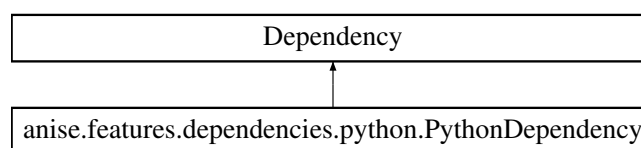
1.105.3.5 `anise.features.licensing.BaseLicense.namepythonsetuputils` [inherited]

The documentation for this class was generated from the following file:

- [anise/features/licensing.py](#)

1.106 `anise.features.dependencies.python.PythonDependency` Class Reference

Inheritance diagram for `anise.features.dependencies.python.PythonDependency`:



Public Member Functions

- `def __init__`

1.106.1 Constructor & Destructor Documentation

1.106.1.1 `def anise.features.dependencies.python.PythonDependency.__init__ (self, type, version, kwargs)`

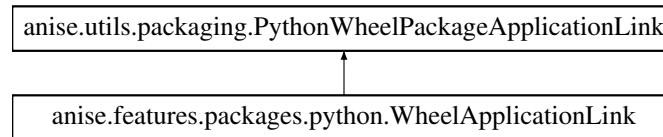
The documentation for this class was generated from the following file:

- [anise/features/dependencies/python.py](#)

1.107 anise.utils.packaging.PythonWheelPackageApplicationLink Class Reference

Data structure for one menu entry which comes in a Python wheel package.

Inheritance diagram for anise.utils.packaging.PythonWheelPackageApplicationLink:



Public Member Functions

- `def __init__`

Public Attributes

- `linkname`
- `module`
- `method`
- `gui`

1.107.1 Detailed Description

Data structure for one menu entry which comes in a Python wheel package.

1.107.2 Constructor & Destructor Documentation

1.107.2.1 `def anise.utils.packaging.PythonWheelPackageApplicationLink.__init__(self, linkname, module, method, gui)`

Parameters

<i>linkname</i>	The name of the link as the user will be able to call it.
<i>module</i>	The name of the module which contains the entry method.
<i>method</i>	The name of the entry method.
<i>gui</i>	If the command opens a gui (instead of a terminal application).

1.107.3 Member Data Documentation

1.107.3.1 `anise.utils.packaging.PythonWheelPackageApplicationLink.gui`

1.107.3.2 `anise.utils.packaging.PythonWheelPackageApplicationLink.linkname`

1.107.3.3 `anise.utils.packaging.PythonWheelPackageApplicationLink.method`

1.107.3.4 `anise.utils.packaging.PythonWheelPackageApplicationLink.module`

The documentation for this class was generated from the following file:

- `anise/utils/packaging.py`

1.108 anise.features.build.qmake.QtConfiguration Class Reference

Public Member Functions

- [def __init__](#)

Public Attributes

- [qtdir](#)
- [bindir](#)
- [pluginsdir](#)
- [makecmd](#)
- [qmakecmd](#)

1.108.1 Constructor & Destructor Documentation

- 1.108.1.1 `def anise.features.build.qmake.QtConfiguration.__init__(self, qtdir, makecmd = "make", qmakecmd = "bin/qmake")`

1.108.2 Member Data Documentation

- 1.108.2.1 `anise.features.build.qmake.QtConfiguration.bindir`
- 1.108.2.2 `anise.features.build.qmake.QtConfiguration.makecmd`
- 1.108.2.3 `anise.features.build.qmake.QtConfiguration.pluginsdir`
- 1.108.2.4 `anise.features.build.qmake.QtConfiguration.qmakecmd`
- 1.108.2.5 `anise.features.build.qmake.QtConfiguration.qtdir`

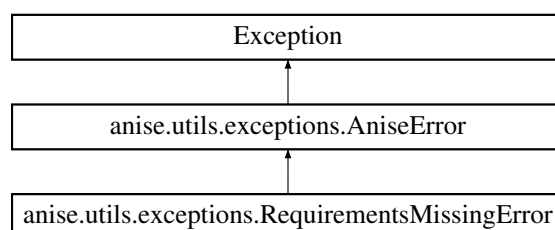
The documentation for this class was generated from the following file:

- `anise/features/build/qmake.py`

1.109 anise.utils.exceptions.RequirementsMissingError Class Reference

Something required is not in place.

Inheritance diagram for `anise.utils.exceptions.RequirementsMissingError`:



Public Member Functions

- [def __call__](#)

1.109.1 Detailed Description

Something required is not in place.

1.109.2 Member Function Documentation

1.109.2.1 `def anise.utils.exceptions.AniseError.__call__(self, args)` [inherited]

The documentation for this class was generated from the following file:

- [anise/utils/exceptions.py](#)

1.110 anise.features.packages.debian.ServiceDescription Class Reference

Description for Debian services to be included in a package.

Public Member Functions

- `def __init__`

Public Attributes

- [name](#)
- [command](#)

1.110.1 Detailed Description

Description for Debian services to be included in a package.

1.110.2 Constructor & Destructor Documentation

1.110.2.1 `def anise.features.packages.debian.ServiceDescription.__init__(self, name, command)`

Parameters

<i>name</i>	The display name.
<i>command</i>	The command to be executed.

1.110.3 Member Data Documentation

1.110.3.1 `anise.features.packages.debian.ServiceDescription.command`

1.110.3.2 `anise.features.packages.debian.ServiceDescription.name`

The documentation for this class was generated from the following file:

- [anise/features/packages/debian.py](#)

1.111 anise.utils.logging.Severity Class Reference

Log message severities.

Static Public Attributes

- int [Debug](#) = 0
- int [Info](#) = 100
- int [Warning](#) = 200
- int [Error](#) = 300

1.111.1 Detailed Description

Log message severities.

1.111.2 Member Data Documentation

1.111.2.1 int `anise.utils.logging.Severity.Debug` = 0 `[static]`

1.111.2.2 int `anise.utils.logging.Severity.Error` = 300 `[static]`

1.111.2.3 int `anise.utils.logging.Severity.Info` = 100 `[static]`

1.111.2.4 int `anise.utils.logging.Severity.Warning` = 200 `[static]`

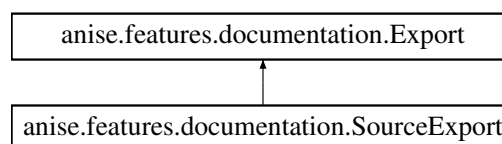
The documentation for this class was generated from the following file:

- [anise/utils/logging.py](#)

1.112 anise.features.documentation.SourceExport Class Reference

Describes an export of an anise documentation back into the source tree.

Inheritance diagram for `anise.features.documentation.SourceExport`:



Public Member Functions

- def `__init__`
- def `supports`

Public Attributes

- `issourceexport`
- `ofile`
- `heading`

- [omode](#)
- [includedeveloperdoc](#)

1.112.1 Detailed Description

Describes an export of an anise documentation back into the source tree.

1.112.2 Constructor & Destructor Documentation

1.112.2.1 `def anise.features.documentation.SourceExport.__init__(self, ofile, heading, mode, includedeveloperdoc = False)`

Parameters

<i>ofile</i>	Target file path relative to the source root.
<i>heading</i>	The title.
<i>mode</i>	Output format ('plain' or 'pdf').
<i>includedeveloperdoc</i>	pdf only: If the output shall include the developer documentation as well.

1.112.3 Member Function Documentation

1.112.3.1 `def anise.features.documentation.Export.supports(self, name)` [inherited]

1.112.4 Member Data Documentation

1.112.4.1 `anise.features.documentation.SourceExport.heading`

1.112.4.2 `anise.features.documentation.SourceExport.includedeveloperdoc`

1.112.4.3 `anise.features.documentation.SourceExport.issourceexport`

1.112.4.4 `anise.features.documentation.SourceExport.ofile`

1.112.4.5 `anise.features.documentation.SourceExport.omode`

The documentation for this class was generated from the following file:

- `anise/features/documentation.py`

1.113 anise.framework.base.TaskExecution Class Reference

Holds a function and some parameters for calling later on (like a delegate).

Public Member Functions

- `def __init__`
- `def execute`

Public Attributes

- [task](#)
- [params](#)

1.113.1 Detailed Description

Holds a function and some parameters for calling later on (like a delegate).

1.113.2 Constructor & Destructor Documentation

1.113.2.1 `def anise.framework.base.TaskExecution.__init__(self, task, kwargs)`

Parameters

<i>task</i>	The task implementation for execution.
-------------	--

1.113.3 Member Function Documentation

1.113.3.1 `def anise.framework.base.TaskExecution.execute(self, kwargs)`

1.113.4 Member Data Documentation

1.113.4.1 `anise.framework.base.TaskExecution.params`

1.113.4.2 `anise.framework.base.TaskExecution.task`

The documentation for this class was generated from the following file:

- `anise/framework/base.py`

1.114 anise.features.imagegalleries.tasks Class Reference

Static Public Member Functions

- `def getimagegallery`
Returns a data structure which contains all information for building one image gallery.

1.114.1 Member Function Documentation

1.114.1.1 `def anise.features.imagegalleries.tasks.getimagegallery(imgdir) [static]`

Returns a data structure which contains all information for building one image gallery.

Parameters

<i>imgdir</i>	The absolute path to the directory which contains the gallery.
---------------	--

The documentation for this class was generated from the following file:

- `anise/features/imagegalleries.py`

1.115 anise.features.packages.tasks Class Reference

Static Public Member Functions

- def [getwebsiteurl](#)
Returns the url from the project website.
- def [enrichrawpackage](#)
Adds certain stuff to a raw packages (as created by makerawpackage), like the license file.
- def [getversion](#)
Determines the version string of the project.
- def [makerawpackage](#)
Builds a raw package in some flavour.
- def [tarpkg](#)
Builds a tarball which can be distributed.

Static Private Attributes

- [_makerawpackage_cache](#) = None

1.115.1 Member Function Documentation

1.115.1.1 `def anise.features.packages.tasks.enrichrawpackage (packagedir) [static]`

Adds certain stuff to a raw packages (as created by makerawpackage), like the license file.

1.115.1.2 `def anise.features.packages.tasks.getversion () [static]`

Determines the version string of the project.

1.115.1.3 `def anise.features.packages.tasks.getwebsiteurl () [static]`

Returns the url from the project website.

1.115.1.4 `def anise.features.packages.tasks.makerawpackage (dirname) [static]`

Builds a raw package in some flavour.

This is used by other components (e.g. packaging) for creating distributable binary packages.

1.115.1.5 `def anise.features.packages.tasks.tarpkg (source = None, namepostfix = None, pkgdescription = " ") [static]`

Builds a tarball which can be distributed.

This is mostly a source package and is possibly not the most convenient package type for an end-user.

1.115.2 Member Data Documentation

1.115.2.1 `anise.features.packages.tasks._makerawpackage_cache = None [static], [private]`

The documentation for this class was generated from the following file:

- [anise/features/packages.py](#)

1.116 anise.features.homepage.tasks Class Reference

Static Public Member Functions

- def [makehomepage](#)
Writes the developer documentation, as specified in the project description, and returns it as Filestructure.
- def [getbannerimage](#)
Determines the location of the banner image.

1.116.1 Member Function Documentation

1.116.1.1 def anise.features.homepage.tasks.getbannerimage () [static]

Determines the location of the banner image.

1.116.1.2 def anise.features.homepage.tasks.makehomepage () [static]

Writes the developer documentation, as specified in the project description, and returns it as Filestructure.

The documentation for this class was generated from the following file:

- [anise/features/homepage.py](#)

1.117 anise.features.packages.win32.tasks Class Reference

Static Public Member Functions

- def [win32exepackage](#)
Builds a Win32 installation executable (as .exe file).
- def [deployqt5dlls](#)
Returns the DLLs needed for Qt deployment on Windows.

1.117.1 Member Function Documentation

1.117.1.1 def anise.features.packages.win32.tasks.deployqt5dlls (*binsource* = None, *pluginssource* = None) [static]

Returns the DLLs needed for Qt deployment on Windows.

1.117.1.2 def anise.features.packages.win32.tasks.win32exepackage (*source* = None, *architecture* = "all", *menuentries* = [], *projecticon* = None, *oninitscript* = "", *oninstallscript* = "") [static]

Builds a Win32 installation executable (as .exe file).

Parameters

<i>architecture</i>	The target architecture name.
<i>menuentries</i>	A list of menu entries for creation on target machines.

The documentation for this class was generated from the following file:

- [anise/features/packages/win32.py](#)

1.118 anise.features.dependencies.tasks Class Reference

Static Public Member Functions

- def [getdependencydocument](#)

Creates a Markdown documentation string which describes the project dependencies in a human-readable form.

1.118.1 Member Function Documentation

1.118.1.1 def anise.features.dependencies.tasks.getdependencydocument () [static]

Creates a Markdown documentation string which describes the project dependencies in a human-readable form.

It is mostly used for including it into Doxygen documentation source.

The documentation for this class was generated from the following file:

- [anise/features/dependencies.py](#)

1.119 anise.features.packages.python.tasks Class Reference

Static Public Member Functions

- def [wheelpackage](#)

Builds a python wheel for distribution of Python applications.

1.119.1 Member Function Documentation

1.119.1.1 def anise.features.packages.python.tasks.wheelpackage (source = None, applicationlinks = None) [static]

Builds a python wheel for distribution of Python applications.

Parameters

<i>consolescripts</i>	List of wheel entry points for console scripts.
<i>guiscripts</i>	List of wheel entry points for gui scripts.

The documentation for this class was generated from the following file:

- [anise/features/packages/python.py](#)

1.120 anise.features.packages.debian.tasks Class Reference

Static Public Member Functions

- def [debpackage](#)

Builds a Debian package.

- def [only_programfiles](#)

Returns a filtered set of files for a Debian program directory (no documentation, license, tests, ...).

1.120.1 Member Function Documentation

1.120.1.1 `def anise.features.packages.debian.tasks.debpackage (source=None, dependencies=[], executablelinks={}, menuentries=[], services=[], prerm="", postinst="", architecture="all") [static]`

Builds a Debian package.

@ param dependencies A list of Debian dependencies for installation on target machines. @ param executablelinks A dict for links to executables for deployment on target machines. @ param menuentries A list of menu entries for creation on target machines. @ param services A list of services for deployment on target machines. @ param prerm Piece of bash script executed before removal of the package on target machines. @ param postinst Piece of bash script executed after installation of the package on the target machines. @ param architecture The target architecture name.

1.120.1.2 `def anise.features.packages.debian.tasks.only_programfiles (source=None) [static]`

Returns a filtered set of files for a Debian program directory (no documentation, license, tests, ...).

The documentation for this class was generated from the following file:

- [anise/features/packages/debian.py](#)

1.121 anise.features.projectdata.tasks Class Reference

Static Public Member Functions

- `def getprojectstatusdescription`

Returns a short plaintext describing the specified project maturity (`project.maturity`).

1.121.1 Member Function Documentation

1.121.1.1 `def anise.features.projectdata.tasks.getprojectstatusdescription (short=False) [static]`

Returns a short plaintext describing the specified project maturity (`project.maturity`).

Parameters

<i>short</i>	If The output shall be a very reduced one.
--------------	--

The documentation for this class was generated from the following file:

- [anise/features/projectdata.py](#)

1.122 anise.features.filesystem.tasks Class Reference

Static Public Member Functions

- `def filterbyfunction`
- `def filterbyextension`
- `def filterbyname`
- `def filterbynamepattern`

1.122.1 Member Function Documentation

1.122.1.1 `def anise.features.filesystem.tasks.filterbyextension (source = None, exclude = None, include = None, casesensitive = False) [static]`

1.122.1.2 `def anise.features.filesystem.tasks.filterbyfunction (acceptorfunction, source = None) [static]`

1.122.1.3 `def anise.features.filesystem.tasks.filterbyname (source = None, exclude = None, include = None, casesensitive = False) [static]`

1.122.1.4 `def anise.features.filesystem.tasks.filterbynamepattern (source = None, exclude = None, include = None, casesensitive = False) [static]`

The documentation for this class was generated from the following file:

- [anise/features/filesystem.py](#)

1.123 anise.features.build.qmake.tasks Class Reference

Static Public Member Functions

- `def build`
Builds a binary with qmake.

1.123.1 Member Function Documentation

1.123.1.1 `def anise.features.build.qmake.tasks.build (qmakeproject, qt, targetname = None, targetfile = None, source = None, qmakeoptions = []) [static]`

Builds a binary with qmake.

Parameters

<i>qmakeproject</i>	The qmake project file path relative to <code>sources</code> root.
<i>targetname</i>	The name of the make target for creating.
<i>qt</i>	The qmake.QtConfiguration to use.

The documentation for this class was generated from the following file:

- [anise/features/build/qmake.py](#)

1.124 anise.features.datainjections.tasks Class Reference

Public Member Functions

- `def inject_python`
Injects some project data into a Python file, so it is available at program's runtime.
- `def inject_c`

Static Public Member Functions

- `def getessentials`
Returns a list of name/value tuples taken from the project object, in order to provide some project metadata like the version number.

1.124.1 Member Function Documentation

1.124.1.1 `def anise.features.datainjections.tasks.getessentials (propnames = None) [static]`

Returns a list of name/value tuples taken from the project object, in order to provide some project metadata like the version number.

Parameters

<i>propnames</i>	List of entry names you want to fetch (beyond the few default ones).
------------------	--

1.124.1.2 `def anise.features.datainjections.tasks.inject_c (outfile = None, data = None)`

1.124.1.3 `def anise.features.datainjections.tasks.inject_python (outfile = None, data = None, asclass = None)`

Injects some project data into a Python file, so it is available at program's runtime.

Parameters

<i>outfile</i>	path to the target file, either absolute or relative to the project's root.
<i>asclass</i>	if the output shall be wrapped into a class definition, give a class name here.

The documentation for this class was generated from the following file:

- [anise/features/datainjections.py](#)

1.125 anise.features.versioncontrol.svn.tasks Class Reference

Static Public Member Functions

- `def getfullversion`
Returns a complete version string, including the Subversion build number.
- `def createrawpackage`
Generates a raw package by using the svn repository as a blueprint.
- `def syncversioncontrolsystem`
Synchronizes the local working copy with the project's svn repository.

1.125.1 Member Function Documentation

1.125.1.1 `def anise.features.versioncontrol.svn.tasks.createrawpackage (dirname) [static]`

Generates a raw package by using the svn repository as a blueprint.

This package is the basis for most (or all) of the other package types.

Parameters

<i>dirname</i>	Specify which name the root directory shall get.
----------------	--

1.125.1.2 `def anise.features.versioncontrol.svn.tasks.getfullversion () [static]`

Returns a complete version string, including the Subversion build number.

1.125.1.3 `def anise.features.versioncontrol.svn.tasks.syncversioncontrolsystem (commitmessage = " ", force = True)`
`[static]`

Synchronizes the local working copy with the project's svn repository.

The documentation for this class was generated from the following file:

- [anise/features/versioncontrol/svn.py](#)

1.126 anise.features.releasing.tasks Class Reference

Static Public Member Functions

- `def release`
Releases the project.

1.126.1 Member Function Documentation

1.126.1.1 `def anise.features.releasing.tasks.release ()` `[static]`

Releases the project.

Mostly this is executing all registered hooks for this event.

The documentation for this class was generated from the following file:

- [anise/features/releasing.py](#)

1.127 anise.features.documentation.tasks Class Reference

Static Public Member Functions

- `def writedocumentations`
Writes the anise documentation, as specified in the project description, to the filesystem.
- `def makedeveloperdoc_doxygen`
Writes the developer documentation, as specified in the project description, and returns it as Filestructure.

1.127.1 Member Function Documentation

1.127.1.1 `def anise.features.documentation.tasks.makedeveloperdoc_doxygen ()` `[static]`

Writes the developer documentation, as specified in the project description, and returns it as Filestructure.

1.127.1.2 `def anise.features.documentation.tasks.writedocumentations (destdir, throw_on_existing = True, fortype = "packageexport")` `[static]`

Writes the anise documentation, as specified in the project description, to the filesystem.

Parameters

<i>destdir</i>	The destination path for writing the output to.
<i>throw_on_existing</i>	If already existing files are critical errors.

The documentation for this class was generated from the following file:

- [anise/features/documentation.py](#)

1.128 anise.features.packages.firefox.tasks Class Reference

Static Public Member Functions

- def [addonpackage](#)
Builds a Firefox addon.
- def [inject_projectdata_to_packagejson](#)
Updates the `package.json` project file.

1.128.1 Member Function Documentation

1.128.1.1 `def anise.features.packages.firefox.tasks.addonpackage (source = None, relpath = " ") [static]`

Builds a Firefox addon.

Parameters

<i>relpath</i>	Path of the firefox addon project relative to 'source's root.
----------------	---

1.128.1.2 `def anise.features.packages.firefox.tasks.inject_projectdata_to_packagejson (outfile, data) [static]`

Updates the `package.json` project file.

Parameters

<i>outfile</i>	The directory which holds <code>package.json</code> , either absolute or relative to the anise project directory.
<i>data</i>	unused.

The documentation for this class was generated from the following file:

- [anise/features/packages/firefox.py](#)

1.129 anise.features.packages.android.tasks Class Reference

Static Public Member Functions

- def [package_with_ant](#)
Builds an Android package with Ant.
- def [inject_projectdata_to_androidmanifest](#)
Updates some project data (like version number) in the `AndroidManifest.xml`.

1.129.1 Member Function Documentation

1.129.1.1 `def anise.features.packages.android.tasks.inject_projectdata_to_androidmanifest (outfile, data) [static]`

Updates some project data (like version number) in the `AndroidManifest.xml`.

Parameters

<i>outfile</i>	The folder which holds the <code>AndroidManifest.xml</code> . Can be absolute or relative to the anise project directory.
<i>data</i>	unused.

1.129.1.2 `def anise.features.packages.android.tasks.package_with_ant (androidpath, source = None, relpath = "", targetname = "debug") [static]`

Builds an Android package with Ant.

Parameters

<i>androidpath</i>	Full path to the <code>android</code> tool (something like <code>../android-sdk/tools/android</code>).
<i>relpath</i>	The root path of the android project, relative to <code>sources</code> root.
<i>targetname</i>	Name of the Ant target to build.

The documentation for this class was generated from the following file:

- [anise/features/packages/android.py](#)

1.130 anise.features.sshfiletransfer.tasks Class Reference

Static Public Member Functions

- `def upload`
Deploys the homepage to a web server via ssh.

1.130.1 Member Function Documentation

1.130.1.1 `def anise.features.sshfiletransfer.tasks.upload (source = None, server = "homepage", collectlinkshook = _homepagefeature.HOOK_UPLOAD_HOMEPAGE_COLLECTLINKS) [static]`

Deploys the homepage to a web server via ssh.

Parameters

<i>server</i>	The server description name (a prefix for some property names).
---------------	---

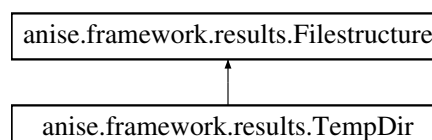
The documentation for this class was generated from the following file:

- [anise/features/sshfiletransfer.py](#)

1.131 anise.framework.results.TempDir Class Reference

A special [anise.framework.results.Filestructure](#) which automatically creates a temporary directory in background.

Inheritance diagram for `anise.framework.results.TempDir`:



Public Member Functions

- def [__init__](#)
- def [__enter__](#)
- def [delete](#)
Deletes the content from the hard drive.
- def [__exit__](#)
- def [dl](#)
Copies the complete filestructure or a subdirectory to a new destination.
- def [with_modified_rootname](#)
Returns a new (temporary) [anise.framework.results.Filestructure](#) clone with a new directory name.

Public Attributes

- [origpath](#)
- [path](#)

1.131.1 Detailed Description

A special [anise.framework.results.Filestructure](#) which automatically creates a temporary directory in background.

Example:

```
def sometask():
    mytmp = results.TempDir() # creates a temporary directory somewhere in no-man's-land
    myfile = mytmp.path + "/somefile"
    with open(myfile, "w") as f:
        f.write("some content for our new file")
    return mytmp
```

1.131.2 Constructor & Destructor Documentation

1.131.2.1 `def anise.framework.results.TempDir.__init__(self, dirname=None)`

Parameters

<i>dirname</i>	An optional name which the root directory should get; if not given, it gets a random name.
----------------	--

1.131.3 Member Function Documentation

1.131.3.1 `def anise.framework.results.TempDir.__enter__(self)`

1.131.3.2 `def anise.framework.results.TempDir.__exit__(self, type, value, traceback)`

1.131.3.3 `def anise.framework.results.TempDir.delete(self)`

Deletes the content from the hard drive.

1.131.3.4 `def anise.framework.results.Filestructure.dl(self, subpath=" ", to=None)` [inherited]

Copies the complete filestructure or a subdirectory to a new destination.

Parameters

<i>subpath</i>	if given, only this subdirectory will be copied.
<i>to</i>	if given, it specifies the destination path; if not given, a new directory in the current working directory will be used.

1.131.3.5 `def anise.framework.results.Filestructure.with_modified_rootname (self, newname)` `[inherited]`

Returns a new (temporary) [anise.framework.results.Filestructure](#) clone with a new directory name.

Parameters

<i>newname</i>	The name which the new root directory should get.
----------------	---

1.131.4 Member Data Documentation

1.131.4.1 `anise.framework.results.TempDir.origpath`

1.131.4.2 `anise.framework.results.Filestructure.path` `[inherited]`

The documentation for this class was generated from the following file:

- [anise/framework/results.py](#)

1.132 anise.features.dependencies.Type Class Reference

Enumeration of different types of dependencies.

Static Public Attributes

- string [Included](#) = "included"
- string [Required](#) = "required"
- string [Required_HasAlternatives](#) = "required (has alternatives)"
- string [Recommended_HasAlternatives](#) = "recommended (maybe has alternatives)"
- string [Optional](#) = "optional"

1.132.1 Detailed Description

Enumeration of different types of dependencies.

1.132.2 Member Data Documentation

1.132.2.1 `string anise.features.dependencies.Type.Included = "included"` `[static]`

1.132.2.2 `string anise.features.dependencies.Type.Optional = "optional"` `[static]`

1.132.2.3 `string anise.features.dependencies.Type.Recommended_HasAlternatives = "recommended (maybe has alternatives)"` `[static]`

1.132.2.4 `string anise.features.dependencies.Type.Required = "required"` `[static]`

1.132.2.5 string anise.features.dependencies.Type.Required_HasAlternatives = "required (has alternatives)" [static]

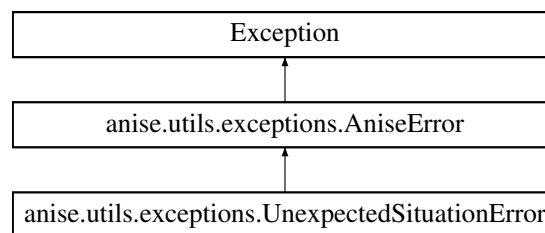
The documentation for this class was generated from the following file:

- anise/features/[dependencies.py](#)

1.133 anise.utils.exceptions.UnexpectedSituationError Class Reference

An unexpected situation led to an error.

Inheritance diagram for anise.utils.exceptions.UnexpectedSituationError:



Public Member Functions

- def [__call__](#)

1.133.1 Detailed Description

An unexpected situation led to an error.

1.133.2 Member Function Documentation

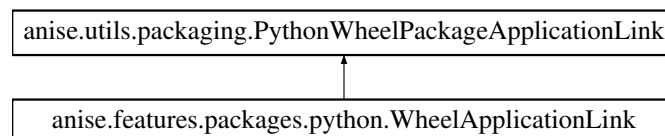
1.133.2.1 def anise.utils.exceptions.AniseError.__call__(self, args) [inherited]

The documentation for this class was generated from the following file:

- anise/utils/[exceptions.py](#)

1.134 anise.features.packages.python.WheelApplicationLink Class Reference

Inheritance diagram for anise.features.packages.python.WheelApplicationLink:



Public Attributes

- [linkname](#)
- [module](#)

- [method](#)
- [gui](#)

1.134.1 Member Data Documentation

- 1.134.1.1 `anise.utils.packaging.PythonWheelPackageApplicationLink.gui` [*inherited*]
- 1.134.1.2 `anise.utils.packaging.PythonWheelPackageApplicationLink.linkname` [*inherited*]
- 1.134.1.3 `anise.utils.packaging.PythonWheelPackageApplicationLink.method` [*inherited*]
- 1.134.1.4 `anise.utils.packaging.PythonWheelPackageApplicationLink.module` [*inherited*]

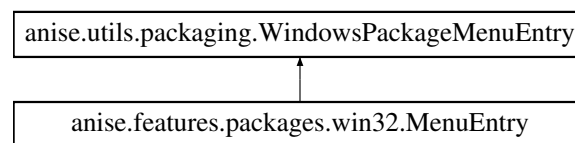
The documentation for this class was generated from the following file:

- `anise/features/packages/python.py`

1.135 `anise.utils.packaging.WindowsPackageMenuEntry` Class Reference

Data structure for one menu entry which comes in a Windows package.

Inheritance diagram for `anise.utils.packaging.WindowsPackageMenuEntry`:



Public Member Functions

- `def __init__`

Public Attributes

- [title](#)
- [command](#)
- [icon](#)

1.135.1 Detailed Description

Data structure for one menu entry which comes in a Windows package.

1.135.2 Constructor & Destructor Documentation

- 1.135.2.1 `def anise.utils.packaging.WindowsPackageMenuEntry.__init__(self, title, command, icon=None)`

Parameters

<i>title</i>	Menu entry Title.
<i>command</i>	Menu entry command.
<i>icon</i>	The path to a menu entry icon.

1.135.3 Member Data Documentation

1.135.3.1 anise.utils.packaging.WindowsPackageMenuEntry.command

1.135.3.2 anise.utils.packaging.WindowsPackageMenuEntry.icon

1.135.3.3 anise.utils.packaging.WindowsPackageMenuEntry.title

The documentation for this class was generated from the following file:

- anise/utils/[packaging.py](#)

Index

- `__call__`
 - `anise::utils::exceptions::AniseError`, 56
 - `anise::utils::exceptions::BadCommunicationError`, 58
 - `anise::utils::exceptions::BadFormatError`, 59
 - `anise::utils::exceptions::BadInputError`, 60
 - `anise::utils::exceptions::InternalError`, 86
 - `anise::utils::exceptions::ProcessExecutionFailedError`, 98
 - `anise::utils::exceptions::RequirementsMissingError`, 103
 - `anise::utils::exceptions::UnexpectedSituationError`, 119
- `__enter__`
 - `anise::framework::results::TempDir`, 117
 - `anise::utils::basic::ChDir`, 76
 - `anise::utils::basic::Mount`, 92
 - `anise::utils::ssh::Mount`, 93
- `__exit__`
 - `anise::framework::results::TempDir`, 117
 - `anise::utils::basic::ChDir`, 76
 - `anise::utils::basic::Mount`, 92
 - `anise::utils::ssh::Mount`, 93
- `__init__`
 - `anise::features::build::qmake::QtConfiguration`, 102
 - `anise::features::dependencies::_Pool`, 50
 - `anise::features::dependencies::Dependency`, 78
 - `anise::features::dependencies::python::DjangoDependency`, 79
 - `anise::features::dependencies::python::PythonDependency`, 100
 - `anise::features::distributables::_Pool`, 51
 - `anise::features::distributables::FileGroup`, 81
 - `anise::features::documentation::_Exports`, 48
 - `anise::features::documentation::_Pool`, 50
 - `anise::features::documentation::Export`, 79
 - `anise::features::documentation::HomepageExport`, 84
 - `anise::features::documentation::PackageExport`, 97
 - `anise::features::documentation::SourceExport`, 105
 - `anise::features::homepage::_Sections`, 52
 - `anise::features::imagegalleries::_Pool`, 51
 - `anise::features::licensing::AFLv3`, 54
 - `anise::features::licensing::AGPLv3`, 55
 - `anise::features::licensing::Apache2`, 56
 - `anise::features::licensing::Artistic1`, 57
 - `anise::features::licensing::BaseLicense`, 61
 - `anise::features::licensing::BSD2clause`, 62
 - `anise::features::licensing::BSD3clause`, 63
 - `anise::features::licensing::Cc0v1`, 69
 - `anise::features::licensing::CcBy3`, 70
 - `anise::features::licensing::CcByNc3`, 71
 - `anise::features::licensing::CcByNcNd3`, 72
 - `anise::features::licensing::CcByNcSa3`, 73
 - `anise::features::licensing::CcByNd3`, 74
 - `anise::features::licensing::CcBySa3`, 75
 - `anise::features::licensing::GPLv3`, 83
 - `anise::features::licensing::LGPLv3`, 87
 - `anise::features::licensing::MIT`, 90
 - `anise::features::licensing::MPLv11`, 95
 - `anise::features::licensing::MPLv2`, 96
 - `anise::features::licensing::PublicDomain`, 99
 - `anise::features::packages::debian::ServiceDescription`, 103
 - `anise::framework::base::TaskExecution`, 106
 - `anise::framework::features::Feature`, 80
 - `anise::framework::features::Hook`, 85
 - `anise::framework::projects::Project`, 99
 - `anise::framework::results::Filestructure`, 82
 - `anise::framework::results::TempDir`, 117
 - `anise::utils::basic::ChDir`, 76
 - `anise::utils::basic::Mount`, 92
 - `anise::utils::packaging::DebianPackageMenuEntry`, 77
 - `anise::utils::packaging::PythonWheelPackageApplicationLink`, 101
 - `anise::utils::packaging::WindowsPackageMenuEntry`, 120
 - `anise::utils::ssh::Mount`, 93
 - `anise::utils::svn::_Version`, 53
- `_add_homepage_section`
 - `anise::features::dependencies`, 18
 - `anise::features::distributables`, 19
 - `anise::features::documentation`, 20
 - `anise::features::imagegalleries`, 22
 - `anise::features::licensing`, 23
- `_add_release_task_for_export_to_source`
 - `anise::features::documentation`, 20
- `_basic_homepage_sections`
 - `anise::features::homepage`, 21
- `_beautify_longdescription`
 - `anise::features::projectdata`, 27
- `_cmdline`
 - `anise::framework::engine`, 30
- `_compose_readmes`
 - `anise::features::documentation`, 20
- `_currentproject`

- anise::framework::projects, 33
- _dependenciesfeature
 - anise::features::dependencies::python, 18
- _derivecolors
 - anise::utils::documentation, 37
- _doxygen_additionalcss
 - anise::utils::documentation, 41
- _doxygen_additionalcss_homepage
 - anise::utils::documentation, 41
- _doxygen_additionalcss_userdocumentation
 - anise::utils::documentation, 41
- _doxygen_baseconfig
 - anise::utils::documentation, 41
- _doxygen_htmlfooter
 - anise::utils::documentation, 41
- _doxygen_htmlheader
 - anise::utils::documentation, 42
- _features
 - anise::framework::features, 31
- _featuresources
 - anise::framework::features, 31
- _homepagefeature
 - anise::features::sshfiletransfer, 28
- _hooks
 - anise::framework::features, 31
- _initproject
 - anise::features::datainjections, 17
 - anise::features::dependencies, 18
 - anise::features::distributables, 19
 - anise::features::documentation, 20
 - anise::features::homepage, 21
 - anise::features::imagegalleries, 22
 - anise::features::packages, 24
 - anise::features::projectdata, 27
 - anise::features::releasing, 27
- _instantiatelicense
 - anise::features::licensing, 23
- _isdatafile
 - anise::utils::packaging, 45
- _makerawpackage_cache
 - anise::features::packages::tasks, 107
- _nsis_get_global
 - anise::features::packages::python, 25
- _packagesfeature
 - anise::features::licensing, 23
- _scalecolor
 - anise::utils::documentation, 37
- _sync
 - anise::features::datainjections, 17
- _writelicense
 - anise::features::licensing, 23
- add
 - anise::features::datainjections::_Pool, 49
 - anise::features::dependencies::_Pool, 50
 - anise::features::documentation::_Exports, 48
 - anise::features::documentation::_Pool, 50
 - anise::features::homepage::_Sections, 52
 - anise::features::imagegalleries::_Pool, 51
- anise::features::releasing::_Releasetasks, 52
- add_anisedoc_source
 - anise::features::documentation, 20
- addgroup
 - anise::features::distributables::_Pool, 51
- addhook
 - anise::framework::projects::Project, 99
- addonpackage
 - anise::features::packages::firefox::tasks, 114
- Alpha
 - anise::features::projectdata::Maturity, 88
- anise, 13
- anise.data, 14
- anise.data.icons, 14
- anise.data.licenses, 14
- anise.features, 14
- anise.features.build, 15
- anise.features.build.qmake, 16
- anise.features.build.qmake.QtConfiguration, 102
- anise.features.build.qmake.tasks, 111
- anise.features.datainjections, 16
- anise.features.datainjections._Pool, 49
- anise.features.datainjections.tasks, 111
- anise.features.dependencies, 17
- anise.features.dependencies._Pool, 49
- anise.features.dependencies.Dependency, 78
- anise.features.dependencies.python, 18
- anise.features.dependencies.python.DjangoDependency, 79
- anise.features.dependencies.python.PythonDependency, 100
- anise.features.dependencies.tasks, 109
- anise.features.dependencies.Type, 118
- anise.features.distributables, 18
- anise.features.distributables._Pool, 51
- anise.features.distributables.FileGroup, 80
- anise.features.documentation, 19
- anise.features.documentation._Exports, 48
- anise.features.documentation._Pool, 50
- anise.features.documentation.Export, 79
- anise.features.documentation.HomepageExport, 84
- anise.features.documentation.PackageExport, 96
- anise.features.documentation.SourceExport, 104
- anise.features.documentation.tasks, 113
- anise.features.filesystem, 20
- anise.features.filesystem.tasks, 110
- anise.features.homepage, 21
- anise.features.homepage._Sections, 52
- anise.features.homepage.tasks, 108
- anise.features.imagegalleries, 21
- anise.features.imagegalleries._Pool, 51
- anise.features.imagegalleries.tasks, 106
- anise.features.licensing, 22
- anise.features.licensing.AFLv3, 53
- anise.features.licensing.AGPLv3, 54
- anise.features.licensing.Apache2, 56
- anise.features.licensing.Artistic1, 57
- anise.features.licensing.BSD2clause, 62

- anise.features.licensing.BSD3clause, 63
- anise.features.licensing.BaseLicense, 60
- anise.features.licensing.Cc0v1, 68
- anise.features.licensing.CcBy3, 69
- anise.features.licensing.CcByNc3, 70
- anise.features.licensing.CcByNcNd3, 71
- anise.features.licensing.CcByNcSa3, 72
- anise.features.licensing.CcByNd3, 74
- anise.features.licensing.CcBySa3, 75
- anise.features.licensing.GPLv3, 83
- anise.features.licensing.LGPLv3, 87
- anise.features.licensing.MIT, 90
- anise.features.licensing.MPLv11, 94
- anise.features.licensing.MPLv2, 95
- anise.features.licensing.PublicDomain, 99
- anise.features.packages, 23
- anise.features.packages.android, 24
- anise.features.packages.android.tasks, 114
- anise.features.packages.debian, 24
- anise.features.packages.debian.Category, 64
- anise.features.packages.debian.MenuEntry, 88
- anise.features.packages.debian.ServiceDescription, 103
- anise.features.packages.debian.tasks, 109
- anise.features.packages.firefox, 25
- anise.features.packages.firefox.tasks, 114
- anise.features.packages.python, 25
- anise.features.packages.python.tasks, 109
- anise.features.packages.python.WheelApplicationLink, 119
- anise.features.packages.tasks, 107
- anise.features.packages.win32, 25
- anise.features.packages.win32.MenuEntry, 89
- anise.features.packages.win32.tasks, 108
- anise.features.projectdata, 26
- anise.features.projectdata.Maturity, 88
- anise.features.projectdata.tasks, 110
- anise.features.releasing, 27
- anise.features.releasing._Releasetasks, 52
- anise.features.releasing.tasks, 113
- anise.features.sshfiletransfer, 27
- anise.features.sshfiletransfer.tasks, 116
- anise.features.versioncontrol, 28
- anise.features.versioncontrol.svn, 28
- anise.features.versioncontrol.svn.tasks, 112
- anise.framework, 29
- anise.framework.base, 29
- anise.framework.base.TaskExecution, 105
- anise.framework.engine, 30
- anise.framework.features, 31
- anise.framework.features.Feature, 80
- anise.framework.features.Hook, 85
- anise.framework.features.internals, 86
- anise.framework.globalvars, 32
- anise.framework.imports, 32
- anise.framework.projectinformations, 32
- anise.framework.projects, 32
- anise.framework.projects.Project, 98
- anise.framework.results, 33
- anise.framework.results.Filestructure, 81
- anise.framework.results.TempDir, 116
- anise.utils, 34
- anise.utils.basic, 34
- anise.utils.basic.ChDir, 76
- anise.utils.basic.Mount, 91
- anise.utils.documentation, 36
- anise.utils.exceptions, 42
- anise.utils.exceptions.AniseError, 55
- anise.utils.exceptions.BadCommunicationError, 58
- anise.utils.exceptions.BadFormatError, 59
- anise.utils.exceptions.BadInputError, 59
- anise.utils.exceptions.InternalError, 86
- anise.utils.exceptions.ProcessExecutionFailedError, 98
- anise.utils.exceptions.RequirementsMissingError, 102
- anise.utils.exceptions.UnexpectedSituationError, 119
- anise.utils.logging, 42
- anise.utils.logging.Severity, 104
- anise.utils.packaging, 44
- anise.utils.packaging.DebianPackageMenuEntry, 77
- anise.utils.packaging.PythonWheelPackageApplicationLink, 101
- anise.utils.packaging.WindowsPackageMenuEntry, 120
- anise.utils.ssh, 47
- anise.utils.ssh.Mount, 92
- anise.utils.svn, 47
- anise.utils.svn._Version, 53
- anise::features::build::qmake
 - basedependencies, 16
- anise::features::build::qmake::QtConfiguration
 - __init__, 102
 - bindir, 102
 - makecmd, 102
 - pluginsdir, 102
 - qmakecmd, 102
 - qtdir, 102
- anise::features::build::qmake::tasks
 - build, 111
- anise::features::datainjections
 - _initproject, 17
 - _sync, 17
 - name, 17
- anise::features::datainjections::_Pool
 - add, 49
- anise::features::datainjections::tasks
 - getessentials, 112
 - inject_c, 112
 - inject_python, 112
- anise::features::dependencies
 - _add_homepage_section, 18
 - _initproject, 18
- anise::features::dependencies::_Pool
 - __init__, 50
 - add, 50
 - list, 50
- anise::features::dependencies::Dependency
 - __init__, 78

- comment, 78
- displayname, 78
- icon, 78
- objectname, 78
- type, 78
- visible, 78
- anise::features::dependencies::Type
 - Included, 118
 - Optional, 118
 - Recommended_HasAlternatives, 118
 - Required, 118
 - Required_HasAlternatives, 118
- anise::features::dependencies::python
 - _dependenciesfeature, 18
- anise::features::dependencies::python::DjangoDependency
 - __init__, 79
- anise::features::dependencies::python::PythonDependency
 - __init__, 100
- anise::features::dependencies::tasks
 - getdependencydocument, 109
- anise::features::distributables
 - _add_homepage_section, 19
 - _initproject, 19
- anise::features::distributables::_Pool
 - __init__, 51
 - addgroup, 51
 - list, 51
- anise::features::distributables::FileGroup
 - __init__, 81
 - buildparams, 81
 - description, 81
 - linkto, 81
 - name, 81
 - task, 81
- anise::features::documentation
 - _add_homepage_section, 20
 - _add_release_task_for_export_to_source, 20
 - _compose_readmes, 20
 - _initproject, 20
 - add_anisedoc_source, 20
 - has_doxygen_developerdocumentation, 20
 - markdown_downgrade_headings, 20
- anise::features::documentation::_Exports
 - __init__, 48
 - add, 48
 - list, 49
- anise::features::documentation::_Pool
 - __init__, 50
 - add, 50
 - list, 50
- anise::features::documentation::Export
 - __init__, 79
 - supports, 79
- anise::features::documentation::HomepageExport
 - __init__, 84
 - supports, 84
 - title, 84
- anise::features::documentation::PackageExport
 - __init__, 97
 - heading, 97
 - includedeveloperdoc, 97
 - ispackageexport, 97
 - ofile, 97
 - omode, 97
 - supports, 97
- anise::features::documentation::SourceExport
 - __init__, 105
 - heading, 105
 - includedeveloperdoc, 105
 - isourceexport, 105
 - ofile, 105
 - omode, 105
 - supports, 105
- anise::features::documentation::tasks
 - madeveloperdoc_doxygen, 113
 - writedocumentations, 113
- anise::features::filesystem::tasks
 - filterbyextension, 111
 - filterbyfunction, 111
 - filterbyname, 111
 - filterbynamepattern, 111
- anise::features::homepage
 - _basic_homepage_sections, 21
 - _initproject, 21
- anise::features::homepage::_Sections
 - __init__, 52
 - add, 52
 - list, 52
- anise::features::homepage::tasks
 - getbannerimage, 108
 - makehomepage, 108
- anise::features::imagegalleries
 - _add_homepage_section, 22
 - _initproject, 22
- anise::features::imagegalleries::_Pool
 - __init__, 51
 - add, 51
 - list, 51
- anise::features::imagegalleries::tasks
 - getimagegallery, 106
- anise::features::licensing
 - _add_homepage_section, 23
 - _instantiatelicense, 23
 - _packagesfeature, 23
 - _writelicense, 23
- anise::features::licensing::AFLv3
 - __init__, 54
 - filename, 54
 - getfilename, 54
 - getlicensetext, 54
 - name, 54
 - namedebian, 54
 - namefirefox, 54
 - namepythonsetuputils, 54
- anise::features::licensing::AGPLv3
 - __init__, 55

- filename, [55](#)
- getfilename, [55](#)
- getlicensetext, [55](#)
- name, [55](#)
- namedebian, [55](#)
- namefirefox, [55](#)
- namepythonsetuputils, [55](#)
- anise::features::licensing::Apache2
 - __init__, [56](#)
 - filename, [57](#)
 - getfilename, [56](#)
 - getlicensetext, [56](#)
 - name, [57](#)
 - namedebian, [57](#)
 - namefirefox, [57](#)
 - namepythonsetuputils, [57](#)
- anise::features::licensing::Artistic1
 - __init__, [57](#)
 - filename, [58](#)
 - getfilename, [58](#)
 - getlicensetext, [58](#)
 - name, [58](#)
 - namedebian, [58](#)
 - namefirefox, [58](#)
 - namepythonsetuputils, [58](#)
- anise::features::licensing::BSD2clause
 - __init__, [62](#)
 - filename, [62](#)
 - getfilename, [62](#)
 - getlicensetext, [62](#)
 - name, [62](#)
 - namedebian, [62](#)
 - namefirefox, [62](#)
 - namepythonsetuputils, [62](#)
- anise::features::licensing::BSD3clause
 - __init__, [63](#)
 - filename, [64](#)
 - getfilename, [63](#)
 - getlicensetext, [63](#)
 - name, [64](#)
 - namedebian, [64](#)
 - namefirefox, [64](#)
 - namepythonsetuputils, [64](#)
- anise::features::licensing::BaseLicense
 - __init__, [61](#)
 - filename, [61](#)
 - getfilename, [61](#)
 - getlicensetext, [61](#)
 - name, [61](#)
 - namedebian, [61](#)
 - namefirefox, [61](#)
 - namepythonsetuputils, [61](#)
- anise::features::licensing::Cc0v1
 - __init__, [69](#)
 - filename, [69](#)
 - getfilename, [69](#)
 - getlicensetext, [69](#)
 - name, [69](#)
 - namedebian, [69](#)
 - namefirefox, [69](#)
 - namepythonsetuputils, [69](#)
- anise::features::licensing::CcBy3
 - __init__, [70](#)
 - filename, [70](#)
 - getfilename, [70](#)
 - getlicensetext, [70](#)
 - name, [70](#)
 - namedebian, [70](#)
 - namefirefox, [70](#)
 - namepythonsetuputils, [70](#)
- anise::features::licensing::CcByNc3
 - __init__, [71](#)
 - filename, [71](#)
 - getfilename, [71](#)
 - getlicensetext, [71](#)
 - name, [71](#)
 - namedebian, [71](#)
 - namefirefox, [71](#)
 - namepythonsetuputils, [71](#)
- anise::features::licensing::CcByNcNd3
 - __init__, [72](#)
 - filename, [72](#)
 - getfilename, [72](#)
 - getlicensetext, [72](#)
 - name, [72](#)
 - namedebian, [72](#)
 - namefirefox, [72](#)
 - namepythonsetuputils, [72](#)
- anise::features::licensing::CcByNcSa3
 - __init__, [73](#)
 - filename, [73](#)
 - getfilename, [73](#)
 - getlicensetext, [73](#)
 - name, [73](#)
 - namedebian, [73](#)
 - namefirefox, [73](#)
 - namepythonsetuputils, [73](#)
- anise::features::licensing::CcByNd3
 - __init__, [74](#)
 - filename, [74](#)
 - getfilename, [74](#)
 - getlicensetext, [74](#)
 - name, [74](#)
 - namedebian, [74](#)
 - namefirefox, [74](#)
 - namepythonsetuputils, [74](#)
- anise::features::licensing::CcBySa3
 - __init__, [75](#)
 - filename, [76](#)
 - getfilename, [75](#)
 - getlicensetext, [75](#)
 - name, [76](#)
 - namedebian, [76](#)
 - namefirefox, [76](#)
 - namepythonsetuputils, [76](#)
- anise::features::licensing::GPLv3

- [__init__](#), [83](#)
 - [filename](#), [83](#)
 - [getfilename](#), [83](#)
 - [getlicensetext](#), [83](#)
 - [name](#), [83](#)
 - [namedebian](#), [83](#)
 - [namefirefox](#), [83](#)
 - [namepythonsetuputils](#), [83](#)
- [anise::features::licensing::LGPLv3](#)
 - [__init__](#), [87](#)
 - [filename](#), [87](#)
 - [getfilename](#), [87](#)
 - [getlicensetext](#), [87](#)
 - [name](#), [87](#)
 - [namedebian](#), [87](#)
 - [namefirefox](#), [87](#)
 - [namepythonsetuputils](#), [87](#)
- [anise::features::licensing::MIT](#)
 - [__init__](#), [90](#)
 - [filename](#), [91](#)
 - [getfilename](#), [90](#)
 - [getlicensetext](#), [90](#)
 - [name](#), [91](#)
 - [namedebian](#), [91](#)
 - [namefirefox](#), [91](#)
 - [namepythonsetuputils](#), [91](#)
- [anise::features::licensing::MPLv11](#)
 - [__init__](#), [95](#)
 - [filename](#), [95](#)
 - [getfilename](#), [95](#)
 - [getlicensetext](#), [95](#)
 - [name](#), [95](#)
 - [namedebian](#), [95](#)
 - [namefirefox](#), [95](#)
 - [namepythonsetuputils](#), [95](#)
- [anise::features::licensing::MPLv2](#)
 - [__init__](#), [96](#)
 - [filename](#), [96](#)
 - [getfilename](#), [96](#)
 - [getlicensetext](#), [96](#)
 - [name](#), [96](#)
 - [namedebian](#), [96](#)
 - [namefirefox](#), [96](#)
 - [namepythonsetuputils](#), [96](#)
- [anise::features::licensing::PublicDomain](#)
 - [__init__](#), [99](#)
 - [filename](#), [100](#)
 - [getfilename](#), [100](#)
 - [getlicensetext](#), [100](#)
 - [name](#), [100](#)
 - [namedebian](#), [100](#)
 - [namefirefox](#), [100](#)
 - [namepythonsetuputils](#), [100](#)
- [anise::features::packages](#)
 - [__initproject](#), [24](#)
- [anise::features::packages::android::tasks](#)
 - [inject_projectdata_to_androidmanifest](#), [115](#)
 - [package_with_ant](#), [116](#)
- [anise::features::packages::debian::Category](#)
 - [ApplicationsAccessibility](#), [65](#)
 - [ApplicationsAmateurradio](#), [65](#)
 - [ApplicationsDatamanagement](#), [65](#)
 - [ApplicationsEditors](#), [65](#)
 - [ApplicationsEducation](#), [65](#)
 - [ApplicationsEmulators](#), [65](#)
 - [ApplicationsFilemanagement](#), [65](#)
 - [ApplicationsGraphics](#), [65](#)
 - [ApplicationsMobiledevices](#), [66](#)
 - [ApplicationsNetwork](#), [66](#)
 - [ApplicationsNetworkCommunication](#), [66](#)
 - [ApplicationsNetworkFiletransfer](#), [66](#)
 - [ApplicationsNetworkMonitoring](#), [66](#)
 - [ApplicationsNetworkWebbrowsing](#), [66](#)
 - [ApplicationsNetworkWebnews](#), [66](#)
 - [ApplicationsOffice](#), [66](#)
 - [ApplicationsProgramming](#), [66](#)
 - [ApplicationsProjectmanagement](#), [66](#)
 - [ApplicationsScience](#), [66](#)
 - [ApplicationsScienceAstronomy](#), [66](#)
 - [ApplicationsScienceBiology](#), [66](#)
 - [ApplicationsScienceChemistry](#), [66](#)
 - [ApplicationsScienceDataanalysis](#), [66](#)
 - [ApplicationsScienceElectronics](#), [66](#)
 - [ApplicationsScienceEngineering](#), [66](#)
 - [ApplicationsScienceGeoscience](#), [66](#)
 - [ApplicationsScienceMathematics](#), [66](#)
 - [ApplicationsScienceMedicine](#), [67](#)
 - [ApplicationsSciencePhysics](#), [67](#)
 - [ApplicationsScienceSocial](#), [67](#)
 - [ApplicationsShells](#), [67](#)
 - [ApplicationsSounds](#), [67](#)
 - [ApplicationsSystem](#), [67](#)
 - [ApplicationsSystemAdministration](#), [67](#)
 - [ApplicationsSystemHardware](#), [67](#)
 - [ApplicationsSystemLanguageenvironment](#), [67](#)
 - [ApplicationsSystemMonitoring](#), [67](#)
 - [ApplicationsSystemPackagemanagement](#), [67](#)
 - [ApplicationsSystemSecurity](#), [67](#)
 - [ApplicationsTerminalemulators](#), [67](#)
 - [ApplicationsText](#), [67](#)
 - [ApplicationsTvandrado](#), [67](#)
 - [ApplicationsVideo](#), [67](#)
 - [ApplicationsViewers](#), [67](#)
 - [ApplicationsWebdevelopment](#), [67](#)
 - [GamesAction](#), [67](#)
 - [GamesAdventure](#), [67](#)
 - [GamesBlocks](#), [68](#)
 - [GamesBoard](#), [68](#)
 - [GamesCard](#), [68](#)
 - [GamesPuzzles](#), [68](#)
 - [GamesSimulation](#), [68](#)
 - [GamesStrategy](#), [68](#)
 - [GamesTools](#), [68](#)
 - [GamesToys](#), [68](#)
 - [Help](#), [68](#)
 - [ScreenLocking](#), [68](#)

- ScreenSaving, 68
- anise::features::packages::debian::MenuEntry
 - category, 89
 - command, 89
 - gui, 89
 - icon, 89
 - name, 89
 - title, 89
- anise::features::packages::debian::ServiceDescription
 - __init__, 103
 - command, 103
 - name, 103
- anise::features::packages::debian::tasks
 - debpackage, 110
 - only_programfiles, 110
- anise::features::packages::firefox::tasks
 - addonpackage, 114
 - inject_projectdata_to_packagejson, 114
- anise::features::packages::python
 - _nsis_get_global, 25
- anise::features::packages::python::WheelApplication-Link
 - gui, 120
 - linkname, 120
 - method, 120
 - module, 120
- anise::features::packages::python::tasks
 - wheelpackage, 109
- anise::features::packages::tasks
 - _makerawpackage_cache, 107
 - enrichrawpackage, 107
 - getversion, 107
 - getwebsiteurl, 107
 - makerawpackage, 107
 - tarpackage, 107
- anise::features::packages::win32::MenuEntry
 - command, 90
 - icon, 90
 - title, 90
- anise::features::packages::win32::tasks
 - deployqt5dlls, 108
 - win32exepackage, 108
- anise::features::projectdata
 - _beautify_longdescription, 27
 - _initproject, 27
- anise::features::projectdata::Maturity
 - Alpha, 88
 - Beta, 88
 - Inactive, 88
 - Mature, 88
 - Planning, 88
 - PreAlpha, 88
 - ProductionStable, 88
- anise::features::projectdata::tasks
 - getprojectstatusdescription, 110
- anise::features::releasing
 - _initproject, 27
 - HOOK_RELEASE, 27
- anise::features::releasing::_Releasetasks
 - add, 52
- anise::features::releasing::tasks
 - release, 113
- anise::features::sshfiletransfer
 - _homepagefeature, 28
- anise::features::sshfiletransfer::tasks
 - upload, 116
- anise::features::versioncontrol::svn
 - is_svn_controlled, 29
- anise::features::versioncontrol::svn::tasks
 - createrawpackage, 112
 - getfullversion, 112
 - syncversioncontrolsystem, 112
- anise::framework::base::TaskExecution
 - __init__, 106
 - execute, 106
 - params, 106
 - task, 106
- anise::framework::engine
 - _cmdline, 30
 - execute, 30
 - extractfilesfromretval, 30
 - main, 30
- anise::framework::features
 - _features, 31
 - _featuresources, 31
 - _hooks, 31
 - hook, 31
 - loadfeature, 31
- anise::framework::features::Feature
 - __init__, 80
 - featuremodule, 80
 - hooks, 80
 - name, 80
- anise::framework::features::Hook
 - __init__, 85
 - eventname, 85
 - function, 85
 - prepares, 85
 - provides, 85
 - requires, 85
- anise::framework::features::internals
 - injectdefaultstopproject, 86
 - injecthookstopproject, 86
 - loadfeatures, 86
 - resetfeatures, 86
- anise::framework::globalvars
 - logdebug, 32
 - removetempdirs, 32
- anise::framework::projectinformations
 - homepage, 32
 - version, 32
- anise::framework::projects
 - _currentproject, 33
 - currentproject, 33
 - setcurrentproject, 33
- anise::framework::projects::Project

- `__init__`, 99
- `addhook`, 99
- `append`, 99
- `gethooks`, 99
- `hooks`, 99
- `anise::framework::results`
 - `source_to_filestructure`, 34
- `anise::framework::results::Filestructure`
 - `__init__`, 82
 - `dl`, 82
 - `path`, 82
 - `with_modified_rootname`, 82
- `anise::framework::results::TempDir`
 - `__enter__`, 117
 - `__exit__`, 117
 - `__init__`, 117
 - `delete`, 117
 - `dl`, 117
 - `origpath`, 118
 - `path`, 118
 - `with_modified_rootname`, 118
- `anise::utils::basic`
 - `call`, 35
 - `createobjectstructure`, 35
 - `dict2object`, 35
 - `getuniqueidentifier`, 35
 - `linewrap`, 35
 - `maketarball`, 35
 - `makedirs`, 35
 - `params2dict`, 35
 - `readfromfile`, 36
 - `stripname`, 36
 - `writetofile`, 36
- `anise::utils::basic::ChDir`
 - `__enter__`, 76
 - `__exit__`, 76
 - `__init__`, 76
 - `newpath`, 76
 - `oldpath`, 76
- `anise::utils::basic::Mount`
 - `__enter__`, 92
 - `__exit__`, 92
 - `__init__`, 92
 - `dst`, 92
 - `mount`, 92
 - `mounted`, 92
 - `mounter`, 92
 - `needsroot`, 92
 - `options`, 92
 - `src`, 92
 - `su`, 92
 - `unmount`, 92
 - `unmounter`, 92
- `anise::utils::documentation`
 - `_derivecolors`, 37
 - `_doxygen_additionalcss`, 41
 - `_doxygen_additionalcss_homepage`, 41
 - `_doxygen_additionalcss_userdocumentation`, 41
 - `_doxygen_baseconfig`, 41
 - `_doxygen_htmlfooter`, 41
 - `_doxygen_htmlheader`, 42
 - `_scalecolor`, 37
 - `generate_doxygen_developerdocumentation_`
`ashtml`, 37
 - `generate_doxygen_documentation`, 37
 - `generate_doxygen_homepage`, 38
 - `generate_doxygen_userdocumentation_ashtml`, 38
 - `generate_doxygen_userdocumentation_asmanpage`, 39
 - `generate_doxygen_userdocumentation_aspdf`, 39
 - `generate_doxygen_userdocumentation_asplaintext`, 39
 - `markdown_downgrade_headings`, 41
- `anise::utils::exceptions::AniseError`
 - `__call__`, 56
- `anise::utils::exceptions::BadCommunicationError`
 - `__call__`, 58
- `anise::utils::exceptions::BadFormatError`
 - `__call__`, 59
- `anise::utils::exceptions::BadInputError`
 - `__call__`, 60
- `anise::utils::exceptions::InternalError`
 - `__call__`, 86
- `anise::utils::exceptions::ProcessExecutionFailedError`
 - `__call__`, 98
- `anise::utils::exceptions::RequirementsMissingError`
 - `__call__`, 103
- `anise::utils::exceptions::UnexpectedSituationError`
 - `__call__`, 119
- `anise::utils::logging`
 - `log_start_stop`, 43
 - `logdebug`, 43
 - `logerror`, 43
 - `logexception`, 43
 - `loginfo`, 43
 - `logmessage`, 44
 - `logwarning`, 44
- `anise::utils::logging::Severity`
 - `Debug`, 104
 - `Error`, 104
 - `Info`, 104
 - `Warning`, 104
- `anise::utils::packaging`
 - `_isdatafile`, 45
 - `buildandroidpackage_with_ant`, 45
 - `builddebpackage`, 45
 - `buildfirefoxaddonpackage`, 46
 - `buildpythonwheelpackage`, 46
 - `buildwindowsinstallerpackage`, 46
- `anise::utils::packaging::DebianPackageMenuEntry`
 - `__init__`, 77
 - `category`, 77
 - `command`, 77
 - `gui`, 77
 - `icon`, 77
 - `name`, 78

- title, [78](#)
- anise::utils::packaging::PythonWheelPackageApplication-Link
 - __init__, [101](#)
 - gui, [101](#)
 - linkname, [101](#)
 - method, [101](#)
 - module, [101](#)
- anise::utils::packaging::WindowsPackageMenuEntry
 - __init__, [120](#)
 - command, [121](#)
 - icon, [121](#)
 - title, [121](#)
- anise::utils::ssh
 - call, [47](#)
 - copy, [47](#)
- anise::utils::ssh::Mount
 - __enter__, [93](#)
 - __exit__, [93](#)
 - __init__, [93](#)
 - dst, [94](#)
 - mount, [93](#)
 - mounted, [94](#)
 - mounter, [94](#)
 - needsroot, [94](#)
 - options, [94](#)
 - src, [94](#)
 - su, [94](#)
 - unmount, [94](#)
 - unmounter, [94](#)
- anise::utils::svn
 - getsvnrevision, [48](#)
 - listsvnrepo, [48](#)
- anise::utils::svn::_Version
 - __init__, [53](#)
 - build, [53](#)
 - format, [53](#)
 - isvalid, [53](#)
 - tag, [53](#)
- append
 - anise::framework::projects::Project, [99](#)
- ApplicationsAccessibility
 - anise::features::packages::debian::Category, [65](#)
- ApplicationsAmateurradio
 - anise::features::packages::debian::Category, [65](#)
- ApplicationsDatamanagement
 - anise::features::packages::debian::Category, [65](#)
- ApplicationsEditors
 - anise::features::packages::debian::Category, [65](#)
- ApplicationsEducation
 - anise::features::packages::debian::Category, [65](#)
- ApplicationsEmulators
 - anise::features::packages::debian::Category, [65](#)
- ApplicationsFilemanagement
 - anise::features::packages::debian::Category, [65](#)
- ApplicationsGraphics
 - anise::features::packages::debian::Category, [65](#)
- ApplicationsMobiledevices
 - anise::features::packages::debian::Category, [66](#)
- ApplicationsNetwork
 - anise::features::packages::debian::Category, [66](#)
- ApplicationsNetworkCommunication
 - anise::features::packages::debian::Category, [66](#)
- ApplicationsNetworkFiletransfer
 - anise::features::packages::debian::Category, [66](#)
- ApplicationsNetworkMonitoring
 - anise::features::packages::debian::Category, [66](#)
- ApplicationsNetworkWebbrowsing
 - anise::features::packages::debian::Category, [66](#)
- ApplicationsNetworkWebnews
 - anise::features::packages::debian::Category, [66](#)
- ApplicationsOffice
 - anise::features::packages::debian::Category, [66](#)
- ApplicationsProgramming
 - anise::features::packages::debian::Category, [66](#)
- ApplicationsProjectmanagement
 - anise::features::packages::debian::Category, [66](#)
- ApplicationsScience
 - anise::features::packages::debian::Category, [66](#)
- ApplicationsScienceAstronomy
 - anise::features::packages::debian::Category, [66](#)
- ApplicationsScienceBiology
 - anise::features::packages::debian::Category, [66](#)
- ApplicationsScienceChemistry
 - anise::features::packages::debian::Category, [66](#)
- ApplicationsScienceDataanalysis
 - anise::features::packages::debian::Category, [66](#)
- ApplicationsScienceElectronics
 - anise::features::packages::debian::Category, [66](#)
- ApplicationsScienceEngineering
 - anise::features::packages::debian::Category, [66](#)
- ApplicationsScienceGeoscience
 - anise::features::packages::debian::Category, [66](#)
- ApplicationsScienceMathematics
 - anise::features::packages::debian::Category, [66](#)
- ApplicationsScienceMedicine
 - anise::features::packages::debian::Category, [67](#)
- ApplicationsSciencePhysics
 - anise::features::packages::debian::Category, [67](#)
- ApplicationsScienceSocial
 - anise::features::packages::debian::Category, [67](#)
- ApplicationsShells
 - anise::features::packages::debian::Category, [67](#)
- ApplicationsSounds
 - anise::features::packages::debian::Category, [67](#)
- ApplicationsSystem
 - anise::features::packages::debian::Category, [67](#)
- ApplicationsSystemAdministration
 - anise::features::packages::debian::Category, [67](#)
- ApplicationsSystemHardware
 - anise::features::packages::debian::Category, [67](#)
- ApplicationsSystemLanguageenvironment
 - anise::features::packages::debian::Category, [67](#)
- ApplicationsSystemMonitoring
 - anise::features::packages::debian::Category, [67](#)
- ApplicationsSystemPackagemanagement
 - anise::features::packages::debian::Category, [67](#)

- anise::features::packages::debian::Category, 67
- ApplicationsSystemSecurity
 - anise::features::packages::debian::Category, 67
- ApplicationsTerminalemulators
 - anise::features::packages::debian::Category, 67
- ApplicationsText
 - anise::features::packages::debian::Category, 67
- ApplicationsTvandrado
 - anise::features::packages::debian::Category, 67
- ApplicationsVideo
 - anise::features::packages::debian::Category, 67
- ApplicationsViewers
 - anise::features::packages::debian::Category, 67
- ApplicationsWebdevelopment
 - anise::features::packages::debian::Category, 67
- basedependencies
 - anise::features::build::qmake, 16
- Beta
 - anise::features::projectdata::Maturity, 88
- bindir
 - anise::features::build::qmake::QtConfiguration, 102
- build
 - anise::features::build::qmake::tasks, 111
 - anise::utils::svn::_Version, 53
- buildandroidpackage_with_ant
 - anise::utils::packaging, 45
- builddebpackage
 - anise::utils::packaging, 45
- buildfirefoxaddonpackage
 - anise::utils::packaging, 46
- buildparams
 - anise::features::distributables::FileGroup, 81
- buildpythonwheelpackage
 - anise::utils::packaging, 46
- buildwindowsinstallerpackage
 - anise::utils::packaging, 46
- call
 - anise::utils::basic, 35
 - anise::utils::ssh, 47
- category
 - anise::features::packages::debian::MenuEntry, 89
 - anise::utils::packaging::DebianPackageMenuEntry, 77
- command
 - anise::features::packages::debian::MenuEntry, 89
 - anise::features::packages::debian::ServiceDescription, 103
 - anise::features::packages::win32::MenuEntry, 90
 - anise::utils::packaging::DebianPackageMenuEntry, 77
 - anise::utils::packaging::WindowsPackageMenuEntry, 121
- comment
 - anise::features::dependencies::Dependency, 78
- copy
 - anise::utils::ssh, 47
- createobjectstructure
 - anise::utils::basic, 35
- createrawpackage
 - anise::features::versioncontrol::svn::tasks, 112
- currentproject
 - anise::framework::projects, 33
- debpackage
 - anise::features::packages::debian::tasks, 110
- Debug
 - anise::utils::logging::Severity, 104
- delete
 - anise::framework::results::TempDir, 117
- deployqt5dlls
 - anise::features::packages::win32::tasks, 108
- description
 - anise::features::distributables::FileGroup, 81
- dict2object
 - anise::utils::basic, 35
- displayname
 - anise::features::dependencies::Dependency, 78
- dl
 - anise::framework::results::Filestructure, 82
 - anise::framework::results::TempDir, 117
- dst
 - anise::utils::basic::Mount, 92
 - anise::utils::ssh::Mount, 94
- enrichrawpackage
 - anise::features::packages::tasks, 107
- Error
 - anise::utils::logging::Severity, 104
- eventname
 - anise::framework::features::Hook, 85
- execute
 - anise::framework::base::TaskExecution, 106
 - anise::framework::engine, 30
- extractfilesfromretval
 - anise::framework::engine, 30
- featuremodule
 - anise::framework::features::Feature, 80
- filename
 - anise::features::licensing::AFLv3, 54
 - anise::features::licensing::AGPLv3, 55
 - anise::features::licensing::Apache2, 57
 - anise::features::licensing::Artistic1, 58
 - anise::features::licensing::BaseLicense, 61
 - anise::features::licensing::BSD2clause, 62
 - anise::features::licensing::BSD3clause, 64
 - anise::features::licensing::Cc0v1, 69
 - anise::features::licensing::CcBy3, 70
 - anise::features::licensing::CcByNc3, 71
 - anise::features::licensing::CcByNcNd3, 72
 - anise::features::licensing::CcByNcSa3, 73
 - anise::features::licensing::CcByNd3, 74
 - anise::features::licensing::CcBySa3, 76
 - anise::features::licensing::GPLv3, 83
 - anise::features::licensing::LGPLv3, 87
 - anise::features::licensing::MIT, 91

- anise::features::licensing::MPLv11, 95
 - anise::features::licensing::MPLv2, 96
 - anise::features::licensing::PublicDomain, 100
- filterbyextension
 - anise::features::filesystem::tasks, 111
- filterbyfunction
 - anise::features::filesystem::tasks, 111
- filterbyname
 - anise::features::filesystem::tasks, 111
- filterbynamepattern
 - anise::features::filesystem::tasks, 111
- format
 - anise::utils::svn::_Version, 53
- function
 - anise::framework::features::Hook, 85
- GamesAction
 - anise::features::packages::debian::Category, 67
- GamesAdventure
 - anise::features::packages::debian::Category, 67
- GamesBlocks
 - anise::features::packages::debian::Category, 68
- GamesBoard
 - anise::features::packages::debian::Category, 68
- GamesCard
 - anise::features::packages::debian::Category, 68
- GamesPuzzles
 - anise::features::packages::debian::Category, 68
- GamesSimulation
 - anise::features::packages::debian::Category, 68
- GamesStrategy
 - anise::features::packages::debian::Category, 68
- GamesTools
 - anise::features::packages::debian::Category, 68
- GamesToys
 - anise::features::packages::debian::Category, 68
- generate_doxxygen_developerdocumentation_ashtml
 - anise::utils::documentation, 37
- generate_doxxygen_documentation
 - anise::utils::documentation, 37
- generate_doxxygen_homepage
 - anise::utils::documentation, 38
- generate_doxxygen_userdocumentation_ashtml
 - anise::utils::documentation, 38
- generate_doxxygen_userdocumentation_asmanpage
 - anise::utils::documentation, 39
- generate_doxxygen_userdocumentation_aspdf
 - anise::utils::documentation, 39
- generate_doxxygen_userdocumentation_asplaintext
 - anise::utils::documentation, 39
- getbannerimage
 - anise::features::homepage::tasks, 108
- getdependencydocument
 - anise::features::dependencies::tasks, 109
- getessentials
 - anise::features::datainjections::tasks, 112
- getfilename
 - anise::features::licensing::AFLv3, 54
 - anise::features::licensing::AGPLv3, 55
 - anise::features::licensing::Apache2, 56
 - anise::features::licensing::Artistic1, 58
 - anise::features::licensing::BaseLicense, 61
 - anise::features::licensing::BSD2clause, 62
 - anise::features::licensing::BSD3clause, 63
 - anise::features::licensing::Cc0v1, 69
 - anise::features::licensing::CcBy3, 70
 - anise::features::licensing::CcByNc3, 71
 - anise::features::licensing::CcByNcNd3, 72
 - anise::features::licensing::CcByNcSa3, 73
 - anise::features::licensing::CcByNd3, 74
 - anise::features::licensing::CcBySa3, 75
 - anise::features::licensing::GPLv3, 83
 - anise::features::licensing::LGPLv3, 87
 - anise::features::licensing::MIT, 90
 - anise::features::licensing::MPLv11, 95
 - anise::features::licensing::MPLv2, 96
 - anise::features::licensing::PublicDomain, 100
- getfullversion
 - anise::features::versioncontrol::svn::tasks, 112
- gethooks
 - anise::framework::projects::Project, 99
- getimagegallery
 - anise::features::imagegalleries::tasks, 106
- getlicensetext
 - anise::features::licensing::AFLv3, 54
 - anise::features::licensing::AGPLv3, 55
 - anise::features::licensing::Apache2, 56
 - anise::features::licensing::Artistic1, 58
 - anise::features::licensing::BaseLicense, 61
 - anise::features::licensing::BSD2clause, 62
 - anise::features::licensing::BSD3clause, 63
 - anise::features::licensing::Cc0v1, 69
 - anise::features::licensing::CcBy3, 70
 - anise::features::licensing::CcByNc3, 71
 - anise::features::licensing::CcByNcNd3, 72
 - anise::features::licensing::CcByNcSa3, 73
 - anise::features::licensing::CcByNd3, 74
 - anise::features::licensing::CcBySa3, 75
 - anise::features::licensing::GPLv3, 83
 - anise::features::licensing::LGPLv3, 87
 - anise::features::licensing::MIT, 90
 - anise::features::licensing::MPLv11, 95
 - anise::features::licensing::MPLv2, 96
 - anise::features::licensing::PublicDomain, 100
- getprojectstatusdescription
 - anise::features::projectdata::tasks, 110
- getsvnrevision
 - anise::utils::svn, 48
- getuniqueidentifier
 - anise::utils::basic, 35
- getversion
 - anise::features::packages::tasks, 107
- getwebsiteurl
 - anise::features::packages::tasks, 107
- gui
 - anise::features::packages::debian::MenuEntry, 89

- anise::features::packages::python::WheelApplication-Link, 120
 - anise::utils::packaging::DebianPackageMenuEntry, 77
 - anise::utils::packaging::PythonWheelPackage-ApplicationLink, 101
- HOOK_RELEASE
 - anise::features::releasing, 27
- has_doxygen_developerdocumentation
 - anise::features::documentation, 20
- heading
 - anise::features::documentation::PackageExport, 97
 - anise::features::documentation::SourceExport, 105
- Help
 - anise::features::packages::debian::Category, 68
- homepage
 - anise::framework::projectinformations, 32
- hook
 - anise::framework::features, 31
- hooks
 - anise::framework::features::Feature, 80
 - anise::framework::projects::Project, 99
- icon
 - anise::features::dependencies::Dependency, 78
 - anise::features::packages::debian::MenuEntry, 89
 - anise::features::packages::win32::MenuEntry, 90
 - anise::utils::packaging::DebianPackageMenuEntry, 77
 - anise::utils::packaging::WindowsPackageMenu-Entry, 121
- Inactive
 - anise::features::projectdata::Maturity, 88
- Included
 - anise::features::dependencies::Type, 118
- includedeveloperdoc
 - anise::features::documentation::PackageExport, 97
 - anise::features::documentation::SourceExport, 105
- Info
 - anise::utils::logging::Severity, 104
- inject_c
 - anise::features::datainjections::tasks, 112
- inject_projectdata_to_androidmanifest
 - anise::features::packages::android::tasks, 115
- inject_projectdata_to_packagejson
 - anise::features::packages::firefox::tasks, 114
- inject_python
 - anise::features::datainjections::tasks, 112
- injectdefaultstopproject
 - anise::framework::features::internals, 86
- injecthookstopproject
 - anise::framework::features::internals, 86
- is_svn_controlled
 - anise::features::versioncontrol::svn, 29
- ispackageexport
 - anise::features::documentation::PackageExport, 97
- issourceexport
 - anise::features::documentation::SourceExport, 105
- isvalid
 - anise::utils::svn::_Version, 53
- linewrap
 - anise::utils::basic, 35
- linkname
 - anise::features::packages::python::WheelApplication-Link, 120
 - anise::utils::packaging::PythonWheelPackage-ApplicationLink, 101
- linkto
 - anise::features::distributables::FileGroup, 81
- list
 - anise::features::dependencies::_Pool, 50
 - anise::features::distributables::_Pool, 51
 - anise::features::documentation::_Exports, 49
 - anise::features::documentation::_Pool, 50
 - anise::features::homepage::_Sections, 52
 - anise::features::imagegalleries::_Pool, 51
- listsvnrepo
 - anise::utils::svn, 48
- loadfeature
 - anise::framework::features, 31
- loadfeatures
 - anise::framework::features::internals, 86
- log_start_stop
 - anise::utils::logging, 43
- logdebug
 - anise::framework::globalvars, 32
 - anise::utils::logging, 43
- logerror
 - anise::utils::logging, 43
- logexception
 - anise::utils::logging, 43
- loginfo
 - anise::utils::logging, 43
- logmessage
 - anise::utils::logging, 44
- logwarning
 - anise::utils::logging, 44
- main
 - anise::framework::engine, 30
- makecmd
 - anise::features::build::qmake::QtConfiguration, 102
- makedeveloperdoc_doxygen
 - anise::features::documentation::tasks, 113
- makehomepage
 - anise::features::homepage::tasks, 108
- makerawpackage
 - anise::features::packages::tasks, 107
- maketarball
 - anise::utils::basic, 35
- markdown_downgrade_headings
 - anise::features::documentation, 20

- anise::utils::documentation, 41
- Mature
 - anise::features::projectdata::Maturity, 88
- method
 - anise::features::packages::python::WheelApplication-Link, 120
 - anise::utils::packaging::PythonWheelPackage-ApplicationLink, 101
- makedirs
 - anise::utils::basic, 35
- module
 - anise::features::packages::python::WheelApplication-Link, 120
 - anise::utils::packaging::PythonWheelPackage-ApplicationLink, 101
- mount
 - anise::utils::basic::Mount, 92
 - anise::utils::ssh::Mount, 93
- mounted
 - anise::utils::basic::Mount, 92
 - anise::utils::ssh::Mount, 94
- mounter
 - anise::utils::basic::Mount, 92
 - anise::utils::ssh::Mount, 94
- name
 - anise::features::datainjections, 17
 - anise::features::distributables::FileGroup, 81
 - anise::features::licensing::AFLv3, 54
 - anise::features::licensing::AGPLv3, 55
 - anise::features::licensing::Apache2, 57
 - anise::features::licensing::Artistic1, 58
 - anise::features::licensing::BaseLicense, 61
 - anise::features::licensing::BSD2clause, 62
 - anise::features::licensing::BSD3clause, 64
 - anise::features::licensing::Cc0v1, 69
 - anise::features::licensing::CcBy3, 70
 - anise::features::licensing::CcByNc3, 71
 - anise::features::licensing::CcByNcNd3, 72
 - anise::features::licensing::CcByNcSa3, 73
 - anise::features::licensing::CcByNd3, 74
 - anise::features::licensing::CcBySa3, 76
 - anise::features::licensing::GPLv3, 83
 - anise::features::licensing::LGPLv3, 87
 - anise::features::licensing::MIT, 91
 - anise::features::licensing::MPLv11, 95
 - anise::features::licensing::MPLv2, 96
 - anise::features::licensing::PublicDomain, 100
 - anise::features::packages::debian::MenuEntry, 89
 - anise::features::packages::debian::ServiceDescription, 103
 - anise::framework::features::Feature, 80
 - anise::utils::packaging::DebianPackageMenuEntry, 78
- namedebian
 - anise::features::licensing::AFLv3, 54
 - anise::features::licensing::AGPLv3, 55
 - anise::features::licensing::Apache2, 57
 - anise::features::licensing::Artistic1, 58
- anise::features::licensing::BaseLicense, 61
- anise::features::licensing::BSD2clause, 62
- anise::features::licensing::BSD3clause, 64
- anise::features::licensing::Cc0v1, 69
- anise::features::licensing::CcBy3, 70
- anise::features::licensing::CcByNc3, 71
- anise::features::licensing::CcByNcNd3, 72
- anise::features::licensing::CcByNcSa3, 73
- anise::features::licensing::CcByNd3, 74
- anise::features::licensing::CcBySa3, 76
- anise::features::licensing::GPLv3, 83
- anise::features::licensing::LGPLv3, 87
- anise::features::licensing::MIT, 91
- anise::features::licensing::MPLv11, 95
- anise::features::licensing::MPLv2, 96
- anise::features::licensing::PublicDomain, 100
- namefirefox
 - anise::features::licensing::AFLv3, 54
 - anise::features::licensing::AGPLv3, 55
 - anise::features::licensing::Apache2, 57
 - anise::features::licensing::Artistic1, 58
 - anise::features::licensing::BaseLicense, 61
 - anise::features::licensing::BSD2clause, 62
 - anise::features::licensing::BSD3clause, 64
 - anise::features::licensing::Cc0v1, 69
 - anise::features::licensing::CcBy3, 70
 - anise::features::licensing::CcByNc3, 71
 - anise::features::licensing::CcByNcNd3, 72
 - anise::features::licensing::CcByNcSa3, 73
 - anise::features::licensing::CcByNd3, 74
 - anise::features::licensing::CcBySa3, 76
 - anise::features::licensing::GPLv3, 83
 - anise::features::licensing::LGPLv3, 87
 - anise::features::licensing::MIT, 91
 - anise::features::licensing::MPLv11, 95
 - anise::features::licensing::MPLv2, 96
 - anise::features::licensing::PublicDomain, 100
- namepythonsetuputils
 - anise::features::licensing::AFLv3, 54
 - anise::features::licensing::AGPLv3, 55
 - anise::features::licensing::Apache2, 57
 - anise::features::licensing::Artistic1, 58
 - anise::features::licensing::BaseLicense, 61
 - anise::features::licensing::BSD2clause, 62
 - anise::features::licensing::BSD3clause, 64
 - anise::features::licensing::Cc0v1, 69
 - anise::features::licensing::CcBy3, 70
 - anise::features::licensing::CcByNc3, 71
 - anise::features::licensing::CcByNcNd3, 72
 - anise::features::licensing::CcByNcSa3, 73
 - anise::features::licensing::CcByNd3, 74
 - anise::features::licensing::CcBySa3, 76
 - anise::features::licensing::GPLv3, 83
 - anise::features::licensing::LGPLv3, 87
 - anise::features::licensing::MIT, 91
 - anise::features::licensing::MPLv11, 95
 - anise::features::licensing::MPLv2, 96
 - anise::features::licensing::PublicDomain, 100

- needsroot
 - anise::utils::basic::Mount, [92](#)
 - anise::utils::ssh::Mount, [94](#)
- newpath
 - anise::utils::basic::ChDir, [76](#)
- objectname
 - anise::features::dependencies::Dependency, [78](#)
- ofile
 - anise::features::documentation::PackageExport, [97](#)
 - anise::features::documentation::SourceExport, [105](#)
- oldpath
 - anise::utils::basic::ChDir, [76](#)
- omode
 - anise::features::documentation::PackageExport, [97](#)
 - anise::features::documentation::SourceExport, [105](#)
- only_programfiles
 - anise::features::packages::debian::tasks, [110](#)
- Optional
 - anise::features::dependencies::Type, [118](#)
- options
 - anise::utils::basic::Mount, [92](#)
 - anise::utils::ssh::Mount, [94](#)
- origpath
 - anise::framework::results::TempDir, [118](#)
- package_with_ant
 - anise::features::packages::android::tasks, [116](#)
- params
 - anise::framework::base::TaskExecution, [106](#)
- params2dict
 - anise::utils::basic, [35](#)
- path
 - anise::framework::results::Filestructure, [82](#)
 - anise::framework::results::TempDir, [118](#)
- Planning
 - anise::features::projectdata::Maturity, [88](#)
- pluginsdir
 - anise::features::build::qmake::QtConfiguration, [102](#)
- PreAlpha
 - anise::features::projectdata::Maturity, [88](#)
- prepares
 - anise::framework::features::Hook, [85](#)
- ProductionStable
 - anise::features::projectdata::Maturity, [88](#)
- provides
 - anise::framework::features::Hook, [85](#)
- qmakecmd
 - anise::features::build::qmake::QtConfiguration, [102](#)
- qtdir
 - anise::features::build::qmake::QtConfiguration, [102](#)
- readfromfile
 - anise::utils::basic, [36](#)
- Recommended_HasAlternatives
 - anise::features::dependencies::Type, [118](#)
- release
 - anise::features::releasing::tasks, [113](#)
- removetempdirs
 - anise::framework::globalvars, [32](#)
- Required
 - anise::features::dependencies::Type, [118](#)
- Required_HasAlternatives
 - anise::features::dependencies::Type, [118](#)
- requires
 - anise::framework::features::Hook, [85](#)
- resetfeatures
 - anise::framework::features::internals, [86](#)
- ScreenLocking
 - anise::features::packages::debian::Category, [68](#)
- ScreenSaving
 - anise::features::packages::debian::Category, [68](#)
- setcurrentproject
 - anise::framework::projects, [33](#)
- source_to_filestructure
 - anise::framework::results, [34](#)
- src
 - anise::utils::basic::Mount, [92](#)
 - anise::utils::ssh::Mount, [94](#)
- stripname
 - anise::utils::basic, [36](#)
- su
 - anise::utils::basic::Mount, [92](#)
 - anise::utils::ssh::Mount, [94](#)
- supports
 - anise::features::documentation::Export, [79](#)
 - anise::features::documentation::HomepageExport, [84](#)
 - anise::features::documentation::PackageExport, [97](#)
 - anise::features::documentation::SourceExport, [105](#)
- syncversioncontrolsystem
 - anise::features::versioncontrol::svn::tasks, [112](#)
- tag
 - anise::utils::svn::_Version, [53](#)
- tarpackage
 - anise::features::packages::tasks, [107](#)
- task
 - anise::features::distributables::FileGroup, [81](#)
 - anise::framework::base::TaskExecution, [106](#)
- title
 - anise::features::documentation::HomepageExport, [84](#)
 - anise::features::packages::debian::MenuEntry, [89](#)
 - anise::features::packages::win32::MenuEntry, [90](#)
 - anise::utils::packaging::DebianPackageMenuEntry, [78](#)
 - anise::utils::packaging::WindowsPackageMenuEntry, [121](#)
- type
 - anise::features::dependencies::Dependency, [78](#)
- unmount

- anise::utils::basic::Mount, [92](#)
 - anise::utils::ssh::Mount, [94](#)
- unmounter
 - anise::utils::basic::Mount, [92](#)
 - anise::utils::ssh::Mount, [94](#)
- upload
 - anise::features::sshfiletransfer::tasks, [116](#)
- version
 - anise::framework::projectinformations, [32](#)
- visible
 - anise::features::dependencies::Dependency, [78](#)
- Warning
 - anise::utils::logging::Severity, [104](#)
- wheelpackage
 - anise::features::packages::python::tasks, [109](#)
- win32exepackage
 - anise::features::packages::win32::tasks, [108](#)
- with_modified_rootname
 - anise::framework::results::Filestructure, [82](#)
 - anise::framework::results::TempDir, [118](#)
- writedocumentations
 - anise::features::documentation::tasks, [113](#)
- writetofile
 - anise::utils::basic, [36](#)