

Ecodena

1.0

Generated by Doxygen 1.7.4

Sun Mar 25 2012 11:04:32



# Contents

<b>1</b>	<b>Namespace Index</b>	<b>1</b>
1.1	Packages . . . . .	1
<b>2</b>	<b>Class Index</b>	<b>3</b>
2.1	Class Hierarchy . . . . .	3
<b>3</b>	<b>Class Index</b>	<b>5</b>
3.1	Class List . . . . .	5
<b>4</b>	<b>File Index</b>	<b>7</b>
4.1	File List . . . . .	7
<b>5</b>	<b>Namespace Documentation</b>	<b>9</b>
5.1	Package Ecodena . . . . .	9
5.2	Package Ecodena.Attempt . . . . .	9
5.3	Package Ecodena.Attempt.admin . . . . .	9
5.4	Package Ecodena.Attempt.models . . . . .	10
5.4.1	Detailed Description . . . . .	10
5.5	Package Ecodena.Attempt.tests . . . . .	10
5.5.1	Detailed Description . . . . .	10
5.6	Package Ecodena.Attempt.views . . . . .	10
5.6.1	Function Documentation . . . . .	11
5.6.1.1	compileSolution . . . . .	11
5.6.1.2	listSolutions . . . . .	11
5.6.1.3	submitSolution . . . . .	11
5.6.1.4	viewSolution . . . . .	11
5.7	Package Ecodena.Comment . . . . .	11

5.8	Package Ecodena.Comment.admin	11
5.9	Package Ecodena.Comment.models	12
5.9.1	Detailed Description	12
5.10	Package Ecodena.Comment.tests	12
5.10.1	Detailed Description	12
5.11	Package Ecodena.Comment.views	12
5.12	Package Ecodena.Compiler	12
5.13	Package Ecodena.Compiler.admin	12
5.14	Package Ecodena.Compiler.dictionary	13
5.14.1	Variable Documentation	13
5.14.1.1	compilers	13
5.15	Package Ecodena.Compiler.models	13
5.16	Package Ecodena.Compiler.tests	13
5.16.1	Detailed Description	13
5.17	Package Ecodena.Compiler.views	14
5.17.1	Function Documentation	14
5.17.1.1	CCompile110	14
5.17.1.2	JavaCompile110	14
5.18	Package Ecodena.Main	14
5.19	Package Ecodena.Main.models	14
5.20	Package Ecodena.Main.tests	14
5.20.1	Detailed Description	14
5.21	Package Ecodena.Main.views	15
5.21.1	Function Documentation	15
5.21.1.1	home	15
5.22	Package Ecodena.manage	15
5.23	Package Ecodena.Query	15
5.24	Package Ecodena.Query.admin	15
5.25	Package Ecodena.Query.models	15
5.25.1	Detailed Description	15
5.26	Package Ecodena.Query.tests	16
5.26.1	Detailed Description	16
5.27	Package Ecodena.Query.views	16
5.27.1	Detailed Description	16

5.27.2	Function Documentation	16
5.27.2.1	postQuery	16
5.28	Package Ecodena.Question	16
5.29	Package Ecodena.Question.admin	17
5.30	Package Ecodena.Question.models	17
5.31	Package Ecodena.Question.tests	17
5.31.1	Detailed Description	17
5.32	Package Ecodena.Question.views	17
5.32.1	Function Documentation	18
5.32.1.1	listQuestions	18
5.32.1.2	test	18
5.32.1.3	viewQuestionByID	18
5.33	Package Ecodena.settings	18
5.33.1	Variable Documentation	19
5.33.1.1	ADMIN_MEDIA_PREFIX	19
5.33.1.2	ADMINS	19
5.33.1.3	AUTH_PROFILE_MODULE	19
5.33.1.4	DATABASES	19
5.33.1.5	DEBUG	20
5.33.1.6	INSTALLED_APPS	20
5.33.1.7	LANGUAGE_CODE	20
5.33.1.8	LOGGING	20
5.33.1.9	MANAGERS	21
5.33.1.10	MEDIA_ROOT	21
5.33.1.11	MEDIA_URL	21
5.33.1.12	MIDDLEWARE_CLASSES	21
5.33.1.13	ROOT_URLCONF	21
5.33.1.14	SECRET_KEY	21
5.33.1.15	SITE_ID	22
5.33.1.16	STATIC_ROOT	22
5.33.1.17	STATIC_URL	22
5.33.1.18	STATICFILES_DIRS	22
5.33.1.19	STATICFILES_FINDERS	22
5.33.1.20	TEMPLATE_DEBUG	22

5.33.1.21	TEMPLATE_DIRS	22
5.33.1.22	TEMPLATE_LOADERS	23
5.33.1.23	TIME_ZONE	23
5.33.1.24	USE_I18N	23
5.33.1.25	USE_L10N	23
5.34	Package Ecodena.urls	23
5.34.1	Variable Documentation	23
5.34.1.1	urlpatterns	23
5.35	Package Ecodena.User	24
5.36	Package Ecodena.User.admin	24
5.37	Package Ecodena.User.models	24
5.37.1	Detailed Description	25
5.38	Package Ecodena.User.tests	25
5.38.1	Detailed Description	25
5.39	Package Ecodena.User.views	25
5.39.1	Function Documentation	25
5.39.1.1	editProfile	25
5.39.1.2	viewProfile	25
<b>6</b>	<b>Class Documentation</b>	<b>27</b>
6.1	Ecodena.Attempt.models.Attempt Class Reference	27
6.1.1	Detailed Description	28
6.1.2	Member Function Documentation	28
6.1.2.1	__unicode__	28
6.1.2.2	getAttemptID	29
6.1.2.3	getCompilerVersionOfAttempt	29
6.1.2.4	getErrorReportID	29
6.1.2.5	getQuestionID	29
6.1.2.6	getSolution	29
6.1.2.7	getTimeOfSubmission	29
6.1.2.8	getUserID	29
6.1.2.9	isCorrect	29
6.1.2.10	setAttemptID	29
6.1.2.11	setCompilerVersionOfAttempt	29

---

6.1.2.12	<a href="#">setErrorReportID</a>	30
6.1.2.13	<a href="#">setQuestionID</a>	30
6.1.2.14	<a href="#">setSolution</a>	30
6.1.2.15	<a href="#">setStatus</a>	30
6.1.2.16	<a href="#">setTimeOfSubmission</a>	30
6.1.2.17	<a href="#">setUserID</a>	30
6.1.3	<a href="#">Member Data Documentation</a>	30
6.1.3.1	<a href="#">attemptID_f</a>	30
6.1.3.2	<a href="#">attemptID_f</a>	30
6.1.3.3	<a href="#">compilerVersion_f</a>	30
6.1.3.4	<a href="#">compilerVersion_f</a>	31
6.1.3.5	<a href="#">errorReportID_f</a>	31
6.1.3.6	<a href="#">errorReportID_f</a>	31
6.1.3.7	<a href="#">questionID_f</a>	31
6.1.3.8	<a href="#">questionID_f</a>	31
6.1.3.9	<a href="#">solution_f</a>	31
6.1.3.10	<a href="#">solution_f</a>	31
6.1.3.11	<a href="#">status_f</a>	31
6.1.3.12	<a href="#">status_f</a>	31
6.1.3.13	<a href="#">timeOfSubmission_f</a>	32
6.1.3.14	<a href="#">timeOfSubmission_f</a>	32
6.1.3.15	<a href="#">userID_f</a>	32
6.1.3.16	<a href="#">userID_f</a>	32
6.1.4	<a href="#">Property Documentation</a>	32
6.1.4.1	<a href="#">attemptID</a>	32
6.1.4.2	<a href="#">compilerVersion</a>	32
6.1.4.3	<a href="#">errorReportID</a>	32
6.1.4.4	<a href="#">questionID</a>	32
6.1.4.5	<a href="#">solutionText</a>	33
6.1.4.6	<a href="#">status</a>	33
6.1.4.7	<a href="#">timeOfSubmission</a>	33
6.1.4.8	<a href="#">userID</a>	33
6.2	<a href="#">Ecodena.Attempt.admin.AttemptAdmin Class Reference</a>	33
6.2.1	<a href="#">Detailed Description</a>	33

---

6.2.2	Member Data Documentation . . . . .	33
6.2.2.1	list_display . . . . .	33
6.3	Ecodena.Question.models.BattleQuestion Class Reference . . . . .	34
6.3.1	Detailed Description . . . . .	34
6.4	Ecodena.Compiler.models.CCompiler111 Class Reference . . . . .	34
6.4.1	Detailed Description . . . . .	34
6.4.2	Member Function Documentation . . . . .	34
6.4.2.1	compile . . . . .	34
6.5	Ecodena.Comment.models.Comment Class Reference . . . . .	34
6.5.1	Detailed Description . . . . .	36
6.5.2	Member Function Documentation . . . . .	36
6.5.2.1	__unicode__ . . . . .	36
6.5.2.2	getCommentText . . . . .	36
6.5.2.3	getQuestionID . . . . .	36
6.5.2.4	getReportingUserID . . . . .	36
6.5.2.5	getTimeStamp . . . . .	36
6.5.2.6	getUserID . . . . .	36
6.5.2.7	isReported . . . . .	37
6.5.2.8	report . . . . .	37
6.5.2.9	setCommentText . . . . .	37
6.5.2.10	setQuestionID . . . . .	37
6.5.2.11	setReportingUserID . . . . .	37
6.5.2.12	setTimeStamp . . . . .	37
6.5.2.13	setUserID . . . . .	37
6.5.3	Member Data Documentation . . . . .	37
6.5.3.1	commentID_f . . . . .	37
6.5.3.2	commentText_f . . . . .	38
6.5.3.3	commentText_f . . . . .	38
6.5.3.4	isReported_f . . . . .	38
6.5.3.5	isReported_f . . . . .	38
6.5.3.6	questionID_f . . . . .	38
6.5.3.7	questionID_f . . . . .	38
6.5.3.8	reportingUserID_f . . . . .	38
6.5.3.9	reportingUserID_f . . . . .	38



6.5.3.10	timeStamp_f . . . . .	38
6.5.3.11	timeStamp_f . . . . .	39
6.5.3.12	userID_f . . . . .	39
6.5.3.13	userID_f . . . . .	39
6.5.4	Property Documentation . . . . .	39
6.5.4.1	commentText . . . . .	39
6.5.4.2	isReported . . . . .	39
6.5.4.3	questionID . . . . .	39
6.5.4.4	reportingUserID . . . . .	39
6.5.4.5	timeStamp . . . . .	39
6.5.4.6	userID . . . . .	40
6.6	Ecodena.Comment.admin.CommentAdmin Class Reference . . . . .	40
6.6.1	Detailed Description . . . . .	40
6.6.2	Member Data Documentation . . . . .	40
6.6.2.1	list_display . . . . .	40
6.7	Ecodena.Question.views.CommentForm Class Reference . . . . .	40
6.7.1	Detailed Description . . . . .	40
6.7.2	Member Data Documentation . . . . .	41
6.7.2.1	text . . . . .	41
6.8	Ecodena.Compiler.models.Compiler Class Reference . . . . .	41
6.8.1	Detailed Description . . . . .	42
6.8.2	Member Function Documentation . . . . .	42
6.8.2.1	compile . . . . .	42
6.8.2.2	generateErrorReport . . . . .	42
6.8.2.3	getAttempt . . . . .	42
6.8.2.4	getCompilerVersion . . . . .	42
6.8.2.5	getLanguage . . . . .	42
6.8.2.6	getQuestion . . . . .	42
6.8.2.7	setAttempt . . . . .	42
6.8.2.8	setCompilerVersion . . . . .	43
6.8.2.9	setLanguage . . . . .	43
6.8.2.10	setQuestion . . . . .	43
6.8.3	Member Data Documentation . . . . .	43
6.8.3.1	attempt . . . . .	43

6.8.3.2	<a href="#">compilerVersion</a>	43
6.8.3.3	<a href="#">errorReport</a>	43
6.8.3.4	<a href="#">language</a>	43
6.8.3.5	<a href="#">question</a>	43
6.8.4	<a href="#">Property Documentation</a>	43
6.8.4.1	<a href="#">CompilerVersion</a>	43
6.8.4.2	<a href="#">language</a>	44
6.9	<a href="#">Ecodena.Compiler.models.CompilerVersion Class Reference</a>	44
6.9.1	<a href="#">Detailed Description</a>	45
6.9.2	<a href="#">Member Function Documentation</a>	45
6.9.2.1	<a href="#">__unicode__</a>	45
6.9.2.2	<a href="#">getLanguage</a>	45
6.9.2.3	<a href="#">getVersionName</a>	45
6.9.2.4	<a href="#">languageName</a>	45
6.9.2.5	<a href="#">setLanguage</a>	45
6.9.2.6	<a href="#">setVersionName</a>	45
6.9.3	<a href="#">Member Data Documentation</a>	45
6.9.3.1	<a href="#">language_f</a>	45
6.9.3.2	<a href="#">language_f</a>	45
6.9.3.3	<a href="#">versionID_f</a>	46
6.9.3.4	<a href="#">versionName_f</a>	46
6.9.3.5	<a href="#">versionName_f</a>	46
6.9.4	<a href="#">Property Documentation</a>	46
6.9.4.1	<a href="#">language</a>	46
6.9.4.2	<a href="#">versionName</a>	46
6.10	<a href="#">Ecodena.Compiler.admin.CompilerVersionAdmin Class Reference</a>	46
6.10.1	<a href="#">Detailed Description</a>	46
6.10.2	<a href="#">Member Data Documentation</a>	47
6.10.2.1	<a href="#">list_display</a>	47
6.11	<a href="#">Ecodena.Attempt.models.ErrorReport Class Reference</a>	47
6.11.1	<a href="#">Detailed Description</a>	48
6.11.2	<a href="#">Member Function Documentation</a>	48
6.11.2.1	<a href="#">__unicode__</a>	48
6.11.2.2	<a href="#">getErrorMessage</a>	48

6.11.2.3	<a href="#">getErrorType</a>	48
6.11.2.4	<a href="#">getMemory</a>	48
6.11.2.5	<a href="#">getTestCaseLevel</a>	48
6.11.2.6	<a href="#">getTimeRequirement</a>	49
6.11.2.7	<a href="#">setErrorMessage</a>	49
6.11.2.8	<a href="#">setErrorType</a>	49
6.11.2.9	<a href="#">setMemory</a>	49
6.11.2.10	<a href="#">setTestCaseLevel</a>	49
6.11.2.11	<a href="#">setTimeRequirement</a>	49
6.11.3	<a href="#">Member Data Documentation</a>	49
6.11.3.1	<a href="#">ERROR_TYPE</a>	49
6.11.3.2	<a href="#">errorMessage_f</a>	49
6.11.3.3	<a href="#">errorMessage_f</a>	49
6.11.3.4	<a href="#">errorReportID_f</a>	50
6.11.3.5	<a href="#">errorType_f</a>	50
6.11.3.6	<a href="#">errorType_f</a>	50
6.11.3.7	<a href="#">memory_f</a>	50
6.11.3.8	<a href="#">memory_f</a>	50
6.11.3.9	<a href="#">testCaseLevel_f</a>	50
6.11.3.10	<a href="#">testCaseLevel_f</a>	50
6.11.3.11	<a href="#">timeRequirement_f</a>	50
6.11.3.12	<a href="#">timeRequirement_f</a>	50
6.11.4	<a href="#">Property Documentation</a>	51
6.11.4.1	<a href="#">errorMessage</a>	51
6.11.4.2	<a href="#">errorType</a>	51
6.11.4.3	<a href="#">memory</a>	51
6.11.4.4	<a href="#">testCaseLevel</a>	51
6.11.4.5	<a href="#">timeRequirement</a>	51
6.12	<a href="#">Ecodena.Attempt.admin.ErrorReportAdmin Class Reference</a>	51
6.12.1	<a href="#">Detailed Description</a>	51
6.12.2	<a href="#">Member Data Documentation</a>	52
6.12.2.1	<a href="#">list_display</a>	52
6.13	<a href="#">Ecodena.Compiler.models.JavaCompiler111 Class Reference</a>	52
6.13.1	<a href="#">Detailed Description</a>	52

6.13.2	Member Function Documentation	52
6.13.2.1	compile	52
6.14	Ecodena.Compiler.models.Language Class Reference	52
6.14.1	Detailed Description	53
6.14.2	Member Function Documentation	53
6.14.2.1	__unicode__	53
6.14.2.2	getLanguageName	53
6.14.2.3	setLanguageName	53
6.14.3	Member Data Documentation	54
6.14.3.1	languageID_f	54
6.14.3.2	languageName_f	54
6.14.3.3	languageName_f	54
6.14.4	Property Documentation	54
6.14.4.1	languageName	54
6.15	Ecodena.Compiler.admin.LanguageAdmin Class Reference	54
6.15.1	Detailed Description	54
6.15.2	Member Data Documentation	54
6.15.2.1	list_display	54
6.16	Ecodena.Question.models.Level Class Reference	55
6.16.1	Detailed Description	55
6.16.2	Member Function Documentation	55
6.16.2.1	__unicode__	55
6.16.2.2	getLevelName	56
6.16.2.3	setLevelName	56
6.16.3	Member Data Documentation	56
6.16.3.1	levelID_f	56
6.16.3.2	levelName_f	56
6.16.3.3	levelName_f	56
6.16.4	Property Documentation	56
6.16.4.1	levelName	56
6.17	Ecodena.Question.admin.LevelInline Class Reference	56
6.17.1	Detailed Description	57
6.17.2	Member Data Documentation	57
6.17.2.1	extra	57

6.17.2.2	model	57
6.18	<a href="#">Ecodena.Question.models.Level.Meta Class Reference</a>	57
6.18.1	Detailed Description	57
6.18.2	Member Data Documentation	57
6.18.2.1	verbose_name	57
6.18.2.2	verbose_name_plural	57
6.19	<a href="#">Ecodena.Compiler.models.Language.Meta Class Reference</a>	58
6.19.1	Detailed Description	58
6.19.2	Member Data Documentation	58
6.19.2.1	verbose_name	58
6.19.2.2	verbose_name_plural	58
6.20	<a href="#">Ecodena.Question.models.Type.Meta Class Reference</a>	58
6.20.1	Detailed Description	58
6.20.2	Member Data Documentation	58
6.20.2.1	verbose_name	58
6.20.2.2	verbose_name_plural	59
6.21	<a href="#">Ecodena.Compiler.models.CompilerVersion.Meta Class Reference</a>	59
6.21.1	Detailed Description	59
6.21.2	Member Data Documentation	59
6.21.2.1	verbose_name	59
6.21.2.2	verbose_name_plural	59
6.22	<a href="#">Ecodena.Question.models.Question.Meta Class Reference</a>	59
6.22.1	Detailed Description	60
6.22.2	Member Data Documentation	60
6.22.2.1	verbose_name	60
6.22.2.2	verbose_name_plural	60
6.23	<a href="#">Ecodena.Attempt.models.ErrorReport.Meta Class Reference</a>	60
6.23.1	Detailed Description	60
6.23.2	Member Data Documentation	60
6.23.2.1	verbose_name	60
6.23.2.2	verbose_name_plural	60
6.24	<a href="#">Ecodena.Question.models.TestCase.Meta Class Reference</a>	61
6.24.1	Detailed Description	61
6.24.2	Member Data Documentation	61

6.24.2.1	verbose_name	61
6.24.2.2	verbose_name_plural	61
6.25	Ecodena.Comment.models.Comment.Meta Class Reference	61
6.25.1	Detailed Description	61
6.25.2	Member Data Documentation	62
6.25.2.1	verbose_name	62
6.25.2.2	verbose_name_plural	62
6.26	Ecodena.Attempt.models.Attempt.Meta Class Reference	62
6.26.1	Detailed Description	62
6.26.2	Member Data Documentation	62
6.26.2.1	verbose_name	62
6.26.2.2	verbose_name_plural	62
6.27	Ecodena.Query.models.Query.Meta Class Reference	63
6.27.1	Detailed Description	63
6.27.2	Member Data Documentation	63
6.27.2.1	verbose_name	63
6.27.2.2	verbose_name_plural	63
6.28	Ecodena.User.models.Profile Class Reference	63
6.28.1	Detailed Description	64
6.28.2	Member Function Documentation	64
6.28.2.1	getAddress	64
6.28.2.2	getDOB	64
6.28.2.3	getGender	65
6.28.2.4	getNumberOfAttempts	65
6.28.2.5	getUserID	65
6.28.2.6	isAdmin	65
6.28.2.7	isProgrammer	65
6.28.2.8	setAddress	65
6.28.2.9	setDOB	65
6.28.2.10	setGender	65
6.28.2.11	setNumberOfAttempts	65
6.28.3	Member Data Documentation	65
6.28.3.1	address_f	65
6.28.3.2	address_f	66

6.28.3.3	dob_f	66
6.28.3.4	dob_f	66
6.28.3.5	gender_f	66
6.28.3.6	gender_f	66
6.28.3.7	GENDER_TYPE	66
6.28.3.8	isAdmin_f	66
6.28.3.9	isProgrammer_f	66
6.28.3.10	numberOfAttempts_f	66
6.28.3.11	numberOfAttempts_f	67
6.28.3.12	profileID_f	67
6.28.3.13	userID_f	67
6.28.4	Property Documentation	67
6.28.4.1	address	67
6.28.4.2	dob	67
6.28.4.3	gender	67
6.28.4.4	numberOfAttempts	67
6.29	Ecodena.User.admin.ProfileAdmin Class Reference	67
6.29.1	Detailed Description	68
6.29.2	Member Data Documentation	68
6.29.2.1	list_display	68
6.30	Ecodena.User.views.ProfileForm Class Reference	68
6.30.1	Detailed Description	68
6.30.2	Member Data Documentation	68
6.30.2.1	address	68
6.30.2.2	dob	69
6.30.2.3	email	69
6.30.2.4	gender	69
6.30.2.5	GENDER_CHOICES	69
6.30.2.6	username	69
6.31	Ecodena.User.models.Programmer Class Reference	69
6.31.1	Detailed Description	70
6.31.2	Member Function Documentation	70
6.31.2.1	getLevel	70
6.31.2.2	setLevel	70

6.31.3	Member Data Documentation	70
6.31.3.1	level_f	70
6.31.3.2	level_f	70
6.31.4	Property Documentation	71
6.31.4.1	level	71
6.32	Ecodena.Query.models.Query Class Reference	71
6.32.1	Detailed Description	72
6.32.2	Member Function Documentation	72
6.32.2.1	__unicode__	72
6.32.2.2	getAdminID	72
6.32.2.3	getQueryID	72
6.32.2.4	getQueryText	73
6.32.2.5	getQueryTime	73
6.32.2.6	getReplyTime	73
6.32.2.7	getSolution	73
6.32.2.8	getUserID	73
6.32.2.9	setAdminID	73
6.32.2.10	setQueryID	73
6.32.2.11	setQueryText	73
6.32.2.12	setQueryTime	73
6.32.2.13	setReplyTime	73
6.32.2.14	setSolution	74
6.32.2.15	setUserID	74
6.32.3	Member Data Documentation	74
6.32.3.1	adminID_f	74
6.32.3.2	adminID_f	74
6.32.3.3	queryID_f	74
6.32.3.4	queryID_f	74
6.32.3.5	queryText_f	74
6.32.3.6	queryText_f	74
6.32.3.7	queryTime_f	74
6.32.3.8	queryTime_f	75
6.32.3.9	replyTime_f	75
6.32.3.10	replyTime_f	75



6.32.3.11 solution_f . . . . .	75
6.32.3.12 solution_f . . . . .	75
6.32.3.13 userID_f . . . . .	75
6.32.3.14 userID_f . . . . .	75
6.32.4 Property Documentation . . . . .	75
6.32.4.1 adminID . . . . .	75
6.32.4.2 queryID . . . . .	76
6.32.4.3 queryText . . . . .	76
6.32.4.4 queryTime . . . . .	76
6.32.4.5 replyTime . . . . .	76
6.32.4.6 solution . . . . .	76
6.32.4.7 userID . . . . .	76
6.33 Ecodena.Query.admin.QueryAdmin Class Reference . . . . .	76
6.33.1 Detailed Description . . . . .	76
6.33.2 Member Data Documentation . . . . .	77
6.33.2.1 list_display . . . . .	77
6.34 Ecodena.Query.views.QueryForm Class Reference . . . . .	77
6.34.1 Detailed Description . . . . .	77
6.34.2 Member Data Documentation . . . . .	77
6.34.2.1 text . . . . .	77
6.35 Ecodena.Question.models.Question Class Reference . . . . .	77
6.35.1 Detailed Description . . . . .	78
6.35.2 Member Function Documentation . . . . .	79
6.35.2.1 __unicode__ . . . . .	79
6.35.2.2 addComments . . . . .	79
6.35.2.3 getComments . . . . .	79
6.35.2.4 getQuestionID . . . . .	79
6.35.2.5 getQuestionText . . . . .	79
6.35.2.6 getQuestionTitle . . . . .	79
6.35.2.7 getTestCasesList . . . . .	79
6.35.2.8 removeTestCase . . . . .	79
6.35.2.9 setCommentList . . . . .	79
6.35.2.10 setQuestionID . . . . .	79
6.35.2.11 setQuestionText . . . . .	80

6.35.2.12	setQuestionTitle	80
6.35.2.13	setTestCasesList	80
6.35.3	Member Data Documentation	80
6.35.3.1	commentList_f	80
6.35.3.2	commentList_f	80
6.35.3.3	level_f	80
6.35.3.4	questionID_f	80
6.35.3.5	questionID_f	80
6.35.3.6	questionText_f	80
6.35.3.7	questionText_f	81
6.35.3.8	questionTitle_f	81
6.35.3.9	questionTitle_f	81
6.35.3.10	testCasesList_f	81
6.35.3.11	testCasesList_f	81
6.35.3.12	timeLimit_f	81
6.35.3.13	type_f	81
6.35.4	Property Documentation	81
6.35.4.1	commentList	81
6.35.4.2	questionID	81
6.35.4.3	questionText	82
6.35.4.4	questionTitle	82
6.35.4.5	testCasesList	82
6.36	Ecodena.Question.admin.QuestionAdmin Class Reference	82
6.36.1	Detailed Description	82
6.36.2	Member Data Documentation	82
6.36.2.1	inlines1	82
6.36.2.2	inlines2	82
6.36.2.3	list_display	83
6.37	Ecodena.Query.tests.SimpleTest Class Reference	83
6.37.1	Detailed Description	83
6.37.2	Member Function Documentation	83
6.37.2.1	test_basic_addition	83
6.38	Ecodena.Compiler.tests.SimpleTest Class Reference	83
6.38.1	Detailed Description	83

6.38.2	Member Function Documentation	84
6.38.2.1	test_basic_addition	84
6.39	Ecodena.Main.tests.SimpleTest Class Reference	84
6.39.1	Detailed Description	84
6.39.2	Member Function Documentation	84
6.39.2.1	test_basic_addition	84
6.40	Ecodena.Question.tests.SimpleTest Class Reference	84
6.40.1	Detailed Description	85
6.40.2	Member Function Documentation	85
6.40.2.1	test_basic_addition	85
6.41	Ecodena.Comment.tests.SimpleTest Class Reference	85
6.41.1	Detailed Description	85
6.41.2	Member Function Documentation	85
6.41.2.1	test_basic_addition	85
6.42	Ecodena.Attempt.tests.SimpleTest Class Reference	85
6.42.1	Detailed Description	86
6.42.2	Member Function Documentation	86
6.42.2.1	test_basic_addition	86
6.43	Ecodena.User.tests.SimpleTest Class Reference	86
6.43.1	Detailed Description	86
6.43.2	Member Function Documentation	86
6.43.2.1	test_basic_addition	86
6.44	Ecodena.Attempt.views.SolutionForm Class Reference	87
6.44.1	Detailed Description	87
6.44.2	Member Data Documentation	87
6.44.2.1	text	87
6.44.2.2	version	87
6.45	Ecodena.Question.models.TestCase Class Reference	87
6.45.1	Detailed Description	88
6.45.2	Member Function Documentation	88
6.45.2.1	__unicode__	88
6.45.2.2	getCaseType	88
6.45.2.3	getOutput	89
6.45.2.4	getQuestion	89

6.45.2.5	<a href="#">getTestID</a>	89
6.45.2.6	<a href="#">setCaseType</a>	89
6.45.2.7	<a href="#">setOutput</a>	89
6.45.2.8	<a href="#">setQuestion</a>	89
6.45.2.9	<a href="#">setTestID</a>	89
6.45.3	<a href="#">Member Data Documentation</a>	89
6.45.3.1	<a href="#">CASE_TYPE</a>	89
6.45.3.2	<a href="#">caseID_f</a>	89
6.45.3.3	<a href="#">caseID_f</a>	89
6.45.3.4	<a href="#">caseType_f</a>	90
6.45.3.5	<a href="#">caseType_f</a>	90
6.45.3.6	<a href="#">input_f</a>	90
6.45.3.7	<a href="#">output_f</a>	90
6.45.3.8	<a href="#">output_f</a>	90
6.45.3.9	<a href="#">questionID_f</a>	90
6.45.3.10	<a href="#">questionID_f</a>	90
6.45.4	<a href="#">Property Documentation</a>	90
6.45.4.1	<a href="#">caseID</a>	90
6.45.4.2	<a href="#">caseType</a>	91
6.45.4.3	<a href="#">output</a>	91
6.45.4.4	<a href="#">questionID</a>	91
6.46	<a href="#">Ecodena.Question.models.Type Class Reference</a>	91
6.46.1	<a href="#">Detailed Description</a>	91
6.46.2	<a href="#">Member Function Documentation</a>	92
6.46.2.1	<a href="#">__unicode__</a>	92
6.46.3	<a href="#">Member Data Documentation</a>	92
6.46.3.1	<a href="#">typeID_f</a>	92
6.46.3.2	<a href="#">typeName_f</a>	92
6.47	<a href="#">Ecodena.Question.admin.TypeInline Class Reference</a>	92
6.47.1	<a href="#">Detailed Description</a>	92
6.47.2	<a href="#">Member Data Documentation</a>	92
6.47.2.1	<a href="#">model</a>	92
<b>7</b>	<b><a href="#">File Documentation</a></b>	<b>93</b>

7.1	/home/panx/ecodena/src/Ecodena/__init__.py File Reference . . . . .	93
7.2	/home/panx/ecodena/src/Ecodena/Attempt/__init__.py File Reference . .	93
7.3	/home/panx/ecodena/src/Ecodena/Comment/__init__.py File Reference	93
7.4	/home/panx/ecodena/src/Ecodena/Compiler/__init__.py File Reference .	93
7.5	/home/panx/ecodena/src/Ecodena/Main/__init__.py File Reference . . .	94
7.6	/home/panx/ecodena/src/Ecodena/Query/__init__.py File Reference . .	94
7.7	/home/panx/ecodena/src/Ecodena/Question/__init__.py File Reference .	94
7.8	/home/panx/ecodena/src/Ecodena/User/__init__.py File Reference . . .	94
7.9	/home/panx/ecodena/src/Ecodena/Attempt/admin.py File Reference . .	94
7.10	/home/panx/ecodena/src/Ecodena/Comment/admin.py File Reference .	95
7.11	/home/panx/ecodena/src/Ecodena/Compiler/admin.py File Reference . .	95
7.12	/home/panx/ecodena/src/Ecodena/Query/admin.py File Reference . . .	95
7.13	/home/panx/ecodena/src/Ecodena/Question/admin.py File Reference . .	95
7.14	/home/panx/ecodena/src/Ecodena/User/admin.py File Reference . . . .	96
7.15	/home/panx/ecodena/src/Ecodena/Attempt/models.py File Reference . .	96
7.16	/home/panx/ecodena/src/Ecodena/Comment/models.py File Reference .	96
7.17	/home/panx/ecodena/src/Ecodena/Compiler/models.py File Reference .	97
7.18	/home/panx/ecodena/src/Ecodena/Main/models.py File Reference . . .	97
7.19	/home/panx/ecodena/src/Ecodena/Query/models.py File Reference . . .	97
7.20	/home/panx/ecodena/src/Ecodena/Question/models.py File Reference .	98
7.21	/home/panx/ecodena/src/Ecodena/User/models.py File Reference . . . .	98
7.22	/home/panx/ecodena/src/Ecodena/Attempt/tests.py File Reference . . .	98
7.23	/home/panx/ecodena/src/Ecodena/Comment/tests.py File Reference . .	99
7.24	/home/panx/ecodena/src/Ecodena/Compiler/tests.py File Reference . . .	99
7.25	/home/panx/ecodena/src/Ecodena/Main/tests.py File Reference . . . . .	99
7.26	/home/panx/ecodena/src/Ecodena/Query/tests.py File Reference . . . .	99
7.27	/home/panx/ecodena/src/Ecodena/Question/tests.py File Reference . . .	100
7.28	/home/panx/ecodena/src/Ecodena/User/tests.py File Reference . . . . .	100
7.29	/home/panx/ecodena/src/Ecodena/Attempt/views.py File Reference . . .	100
7.30	/home/panx/ecodena/src/Ecodena/Comment/views.py File Reference . .	101
7.31	/home/panx/ecodena/src/Ecodena/Compiler/views.py File Reference . .	101
7.32	/home/panx/ecodena/src/Ecodena/Main/views.py File Reference . . . .	101
7.33	/home/panx/ecodena/src/Ecodena/Query/views.py File Reference . . . .	101
7.34	/home/panx/ecodena/src/Ecodena/Question/views.py File Reference . .	102

7.35	<a href="#">/home/panx/ecodena/src/Ecodena/User/views.py File Reference</a>	102
7.36	<a href="#">/home/panx/ecodena/src/Ecodena/Compiler/dictionary.py File Reference</a>	103
7.37	<a href="#">/home/panx/ecodena/src/Ecodena/manage.py File Reference</a>	103
7.38	<a href="#">/home/panx/ecodena/src/Ecodena/settings.py File Reference</a>	103
7.39	<a href="#">/home/panx/ecodena/src/Ecodena/urls.py File Reference</a>	104

# Chapter 1

## Namespace Index

### 1.1 Packages

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

<a href="#">Ecodena</a>	9
<a href="#">Ecodena.Attempt</a>	9
<a href="#">Ecodena.Attempt.admin</a>	9
<a href="#">Ecodena.Attempt.models</a>	10
<a href="#">Ecodena.Attempt.tests</a>	10
<a href="#">Ecodena.Attempt.views</a>	10
<a href="#">Ecodena.Comment</a>	11
<a href="#">Ecodena.Comment.admin</a>	11
<a href="#">Ecodena.Comment.models</a>	12
<a href="#">Ecodena.Comment.tests</a>	12
<a href="#">Ecodena.Comment.views</a>	12
<a href="#">Ecodena.Compiler</a>	12
<a href="#">Ecodena.Compiler.admin</a>	12
<a href="#">Ecodena.Compiler.dictionary</a>	13
<a href="#">Ecodena.Compiler.models</a>	13
<a href="#">Ecodena.Compiler.tests</a>	13
<a href="#">Ecodena.Compiler.views</a>	14
<a href="#">Ecodena.Main</a>	14
<a href="#">Ecodena.Main.models</a>	14
<a href="#">Ecodena.Main.tests</a>	14
<a href="#">Ecodena.Main.views</a>	15
<a href="#">Ecodena.manage</a>	15
<a href="#">Ecodena.Query</a>	15
<a href="#">Ecodena.Query.admin</a>	15
<a href="#">Ecodena.Query.models</a>	15
<a href="#">Ecodena.Query.tests</a>	16
<a href="#">Ecodena.Query.views</a>	16
<a href="#">Ecodena.Question</a>	16
<a href="#">Ecodena.Question.admin</a>	17

<a href="#">Ecodena.Question.models</a>	17
<a href="#">Ecodena.Question.tests</a>	17
<a href="#">Ecodena.Question.views</a>	17
<a href="#">Ecodena.settings</a>	18
<a href="#">Ecodena.urls</a>	23
<a href="#">Ecodena.User</a>	24
<a href="#">Ecodena.User.admin</a>	24
<a href="#">Ecodena.User.models</a>	24
<a href="#">Ecodena.User.tests</a>	25
<a href="#">Ecodena.User.views</a>	25



## Chapter 2

# Class Index

### 2.1 Class Hierarchy

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

Ecodena.Attempt.models.Attempt . . . . .	27
Ecodena.Attempt.admin.AttemptAdmin . . . . .	33
Ecodena.Question.models.BattleQuestion . . . . .	34
Ecodena.Comment.models.Comment . . . . .	34
Ecodena.Comment.admin.CommentAdmin . . . . .	40
Ecodena.Question.views.CommentForm . . . . .	40
Ecodena.Compiler.models.Compiler . . . . .	41
Ecodena.Compiler.models.CCompiler111 . . . . .	34
Ecodena.Compiler.models.JavaCompiler111 . . . . .	52
Ecodena.Compiler.models.CompilerVersion . . . . .	44
Ecodena.Compiler.admin.CompilerVersionAdmin . . . . .	46
Ecodena.Attempt.models.ErrorReport . . . . .	47
Ecodena.Attempt.admin.ErrorReportAdmin . . . . .	51
Ecodena.Compiler.models.Language . . . . .	52
Ecodena.Compiler.admin.LanguageAdmin . . . . .	54
Ecodena.Question.models.Level . . . . .	55
Ecodena.Question.admin.LevelInline . . . . .	56
Ecodena.Question.models.Level.Meta . . . . .	57
Ecodena.Compiler.models.Language.Meta . . . . .	58
Ecodena.Question.models.Type.Meta . . . . .	58
Ecodena.Compiler.models.CompilerVersion.Meta . . . . .	59
Ecodena.Question.models.Question.Meta . . . . .	59
Ecodena.Attempt.models.ErrorReport.Meta . . . . .	60
Ecodena.Question.models.TestCase.Meta . . . . .	61
Ecodena.Comment.models.Comment.Meta . . . . .	61
Ecodena.Attempt.models.Attempt.Meta . . . . .	62
Ecodena.Query.models.Query.Meta . . . . .	63
Ecodena.User.models.Profile . . . . .	63
Ecodena.User.admin.ProfileAdmin . . . . .	67

Ecodena.User.views.ProfileForm . . . . .	68
Ecodena.User.models.Programmer . . . . .	69
Ecodena.Query.models.Query . . . . .	71
Ecodena.Query.admin.QueryAdmin . . . . .	76
Ecodena.Query.views.QueryForm . . . . .	77
Ecodena.Question.models.Question . . . . .	77
Ecodena.Question.admin.QuestionAdmin . . . . .	82
Ecodena.Query.tests.SimpleTest . . . . .	83
Ecodena.Compiler.tests.SimpleTest . . . . .	83
Ecodena.Main.tests.SimpleTest . . . . .	84
Ecodena.Question.tests.SimpleTest . . . . .	84
Ecodena.Comment.tests.SimpleTest . . . . .	85
Ecodena.Attempt.tests.SimpleTest . . . . .	85
Ecodena.User.tests.SimpleTest . . . . .	86
Ecodena.Attempt.views.SolutionForm . . . . .	87
Ecodena.Question.models.TestCase . . . . .	87
Ecodena.Question.models.Type . . . . .	91
Ecodena.Question.admin.TypeInline . . . . .	92

## Chapter 3

# Class Index

### 3.1 Class List

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

<a href="#">Ecodena.Attempt.models.Attempt</a>	27
<a href="#">Ecodena.Attempt.admin.AttemptAdmin</a>	33
<a href="#">Ecodena.Question.models.BattleQuestion</a>	34
<a href="#">Ecodena.Compiler.models.CCompiler111</a>	34
<a href="#">Ecodena.Comment.models.Comment</a>	34
<a href="#">Ecodena.Comment.admin.CommentAdmin</a>	40
<a href="#">Ecodena.Question.views.CommentForm</a>	40
<a href="#">Ecodena.Compiler.models.Compiler</a>	41
<a href="#">Ecodena.Compiler.models.CompilerVersion</a>	44
<a href="#">Ecodena.Compiler.admin.CompilerVersionAdmin</a>	46
<a href="#">Ecodena.Attempt.models.ErrorReport</a>	47
<a href="#">Ecodena.Attempt.admin.ErrorReportAdmin</a>	51
<a href="#">Ecodena.Compiler.models.JavaCompiler111</a>	52
<a href="#">Ecodena.Compiler.models.Language</a>	52
<a href="#">Ecodena.Compiler.admin.LanguageAdmin</a>	54
<a href="#">Ecodena.Question.models.Level</a>	55
<a href="#">Ecodena.Question.admin.LevelInline</a>	56
<a href="#">Ecodena.Question.models.Level.Meta</a>	57
<a href="#">Ecodena.Compiler.models.Language.Meta</a>	58
<a href="#">Ecodena.Question.models.Type.Meta</a>	58
<a href="#">Ecodena.Compiler.models.CompilerVersion.Meta</a>	59
<a href="#">Ecodena.Question.models.Question.Meta</a>	59
<a href="#">Ecodena.Attempt.models.ErrorReport.Meta</a>	60
<a href="#">Ecodena.Question.models.TestCase.Meta</a>	61
<a href="#">Ecodena.Comment.models.Comment.Meta</a>	61
<a href="#">Ecodena.Attempt.models.Attempt.Meta</a>	62
<a href="#">Ecodena.Query.models.Query.Meta</a>	63
<a href="#">Ecodena.User.models.Profile</a>	63
<a href="#">Ecodena.User.admin.ProfileAdmin</a>	67

<a href="#">Ecodena.User.views.ProfileForm</a>	68
<a href="#">Ecodena.User.models.Programmer</a>	69
<a href="#">Ecodena.Query.models.Query</a>	71
<a href="#">Ecodena.Query.admin.QueryAdmin</a>	76
<a href="#">Ecodena.Query.views.QueryForm</a>	77
<a href="#">Ecodena.Question.models.Question</a>	77
<a href="#">Ecodena.Question.admin.QuestionAdmin</a>	82
<a href="#">Ecodena.Query.tests.SimpleTest</a>	83
<a href="#">Ecodena.Compiler.tests.SimpleTest</a>	83
<a href="#">Ecodena.Main.tests.SimpleTest</a>	84
<a href="#">Ecodena.Question.tests.SimpleTest</a>	84
<a href="#">Ecodena.Comment.tests.SimpleTest</a>	85
<a href="#">Ecodena.Attempt.tests.SimpleTest</a>	85
<a href="#">Ecodena.User.tests.SimpleTest</a>	86
<a href="#">Ecodena.Attempt.views.SolutionForm</a>	87
<a href="#">Ecodena.Question.models.TestCase</a>	87
<a href="#">Ecodena.Question.models.Type</a>	91
<a href="#">Ecodena.Question.admin.TypeInline</a>	92

## Chapter 4

# File Index

### 4.1 File List

Here is a list of all files with brief descriptions:

/home/panx/ecodena/src/Ecodena/___init__.py	93
/home/panx/ecodena/src/Ecodena/manage.py	103
/home/panx/ecodena/src/Ecodena/settings.py	103
/home/panx/ecodena/src/Ecodena/urls.py	104
/home/panx/ecodena/src/Ecodena/Attempt/___init__.py	93
/home/panx/ecodena/src/Ecodena/Attempt/admin.py	94
/home/panx/ecodena/src/Ecodena/Attempt/models.py	96
/home/panx/ecodena/src/Ecodena/Attempt/tests.py	98
/home/panx/ecodena/src/Ecodena/Attempt/views.py	100
/home/panx/ecodena/src/Ecodena/Comment/___init__.py	93
/home/panx/ecodena/src/Ecodena/Comment/admin.py	95
/home/panx/ecodena/src/Ecodena/Comment/models.py	96
/home/panx/ecodena/src/Ecodena/Comment/tests.py	99
/home/panx/ecodena/src/Ecodena/Comment/views.py	101
/home/panx/ecodena/src/Ecodena/Compiler/___init__.py	93
/home/panx/ecodena/src/Ecodena/Compiler/admin.py	95
/home/panx/ecodena/src/Ecodena/Compiler/dictionary.py	103
/home/panx/ecodena/src/Ecodena/Compiler/models.py	97
/home/panx/ecodena/src/Ecodena/Compiler/tests.py	99
/home/panx/ecodena/src/Ecodena/Compiler/views.py	101
/home/panx/ecodena/src/Ecodena/Main/___init__.py	94
/home/panx/ecodena/src/Ecodena/Main/models.py	97
/home/panx/ecodena/src/Ecodena/Main/tests.py	99
/home/panx/ecodena/src/Ecodena/Main/views.py	101
/home/panx/ecodena/src/Ecodena/Query/___init__.py	94
/home/panx/ecodena/src/Ecodena/Query/admin.py	95
/home/panx/ecodena/src/Ecodena/Query/models.py	97
/home/panx/ecodena/src/Ecodena/Query/tests.py	99
/home/panx/ecodena/src/Ecodena/Query/views.py	101

/home/panx/ecodena/src/Ecodena/Question/___init___py . . . . .	94
/home/panx/ecodena/src/Ecodena/Question/admin.py . . . . .	95
/home/panx/ecodena/src/Ecodena/Question/models.py . . . . .	98
/home/panx/ecodena/src/Ecodena/Question/tests.py . . . . .	100
/home/panx/ecodena/src/Ecodena/Question/views.py . . . . .	102
/home/panx/ecodena/src/Ecodena/User/___init___py . . . . .	94
/home/panx/ecodena/src/Ecodena/User/admin.py . . . . .	96
/home/panx/ecodena/src/Ecodena/User/models.py . . . . .	98
/home/panx/ecodena/src/Ecodena/User/tests.py . . . . .	100
/home/panx/ecodena/src/Ecodena/User/views.py . . . . .	102

## Chapter 5

# Namespace Documentation

### 5.1 Package Ecodena

#### Packages

- package [Attempt](#)
- package [Comment](#)
- package [Compiler](#)
- package [Main](#)
- package [manage](#)
- package [Query](#)
- package [Question](#)
- package [settings](#)
- package [urls](#)
- package [User](#)

### 5.2 Package Ecodena.Attempt

#### Packages

- package [admin](#)
- package [models](#)
- package [tests](#)
- package [views](#)

### 5.3 Package Ecodena.Attempt.admin

#### Classes

- class [AttemptAdmin](#)

- class [ErrorReportAdmin](#)

## 5.4 Package Ecodena.Attempt.models

### Classes

- class [ErrorReport](#)
- class [Attempt](#)

#### 5.4.1 Detailed Description

`This file creates the model Attempt`

## 5.5 Package Ecodena.Attempt.tests

### Classes

- class [SimpleTest](#)

#### 5.5.1 Detailed Description

`This file demonstrates writing tests using the unittest module. These will pass when you run "manage.py test".`

`Replace this with more appropriate tests for your application.`

## 5.6 Package Ecodena.Attempt.views

### Classes

- class [SolutionForm](#)

### Functions

- def [listSolutions](#)
- def [viewSolution](#)
- def [submitSolution](#)
- def [compileSolution](#)



### 5.6.1 Function Documentation

#### 5.6.1.1 `def Ecodena.Attempt.views.compileSolution ( version, solution, question )`

compiles the solution by calling the appropriate compiler  
Parameter:version-->contains the version of the compiler that the user has selected  
Parameter:solution-->contains the solution object to be compiled  
Parameter:question-->contains the question object so that the appropriate test cases can be fetched.  
returns nothing if the appropriate compiler is available else it gives an error saying compiler not available

Definition at line 86 of file views.py.

#### 5.6.1.2 `def Ecodena.Attempt.views.listSolutions ( request )`

lists all the solutionIDs' that a particular user has submitted  
parameter:request-->All the details associated with URL  
returns the solutions.html page and the list of solution ids

Definition at line 23 of file views.py.

#### 5.6.1.3 `def Ecodena.Attempt.views.submitSolution ( request, questionID, userID )`

Uploading of the solution submitted by the user. The necessary details are added to the database and commit is made.  
Parameter:request-->All the information associated with the url  
returns the submitSolution.html page making the necessary updation of the database.

Definition at line 50 of file views.py.

#### 5.6.1.4 `def Ecodena.Attempt.views.viewSolution ( request, solutionID )`

Definition at line 33 of file views.py.

## 5.7 Package Ecodena.Comment

### Packages

- package [admin](#)
- package [models](#)
- package [tests](#)
- package [views](#)

## 5.8 Package Ecodena.Comment.admin

### Classes

- class [CommentAdmin](#)

## 5.9 Package Ecodena.Comment.models

### Classes

- class [Comment](#)

### 5.9.1 Detailed Description

Its a models.py of Comment app, i.e. it will create a basic model for Comment app.

## 5.10 Package Ecodena.Comment.tests

### Classes

- class [SimpleTest](#)

### 5.10.1 Detailed Description

This file demonstrates writing tests using the unittest module. These will pass when you run "manage.py test".

Replace this with more appropriate tests for your application.

## 5.11 Package Ecodena.Comment.views

## 5.12 Package Ecodena.Compiler

### Packages

- package [admin](#)
- package [dictionary](#)
- package [models](#)
- package [tests](#)
- package [views](#)

## 5.13 Package Ecodena.Compiler.admin

### Classes

- class [LanguageAdmin](#)
- class [CompilerVersionAdmin](#)

## 5.14 Package Ecodena.Compiler.dictionary

### Variables

- dictionary [compilers](#)

#### 5.14.1 Variable Documentation

##### 5.14.1.1 dictionary Ecodena::Compiler::dictionary.compilers

###### Initial value:

```
1 {  
2  
3 'C 1.1.1': Compiler.views.CCompile111,  
4 'Java 1.1.1':Compiler.views.JavaCompile111,  
5  
6 }
```

Definition at line 1 of file dictionary.py.

## 5.15 Package Ecodena.Compiler.models

### Classes

- class [Language](#)
- class [CompilerVersion](#)
- class [Compiler](#)
- class [CCompiler111](#)
- class [JavaCompiler111](#)

## 5.16 Package Ecodena.Compiler.tests

### Classes

- class [SimpleTest](#)

#### 5.16.1 Detailed Description

This file demonstrates writing tests using the unittest module. These will pass when you run "manage.py test".

Replace this with more appropriate tests for your application.

## 5.17 Package Ecodena.Compiler.views

### Functions

- def [CCompile110](#)
- def [JavaCompile110](#)

#### 5.17.1 Function Documentation

5.17.1.1 `def Ecodena.Compiler.views.CCompile110 ( question, attempt )`

Definition at line 3 of file views.py.

5.17.1.2 `def Ecodena.Compiler.views.JavaCompile110 ( question, attempt )`

Definition at line 14 of file views.py.

## 5.18 Package Ecodena.Main

### Packages

- package [models](#)
- package [tests](#)
- package [views](#)

## 5.19 Package Ecodena.Main.models

## 5.20 Package Ecodena.Main.tests

### Classes

- class [SimpleTest](#)

#### 5.20.1 Detailed Description

This file demonstrates writing tests using the unittest module. These will pass when you run "manage.py test".

Replace this with more appropriate tests for your application.

## 5.21 Package Ecodena.Main.views

### Functions

- def [home](#)

### 5.21.1 Function Documentation

5.21.1.1 `def Ecodena.Main.views.home ( request )`

Definition at line 4 of file views.py.

## 5.22 Package Ecodena.manage

## 5.23 Package Ecodena.Query

### Packages

- package [admin](#)
- package [models](#)
- package [tests](#)
- package [views](#)

## 5.24 Package Ecodena.Query.admin

### Classes

- class [QueryAdmin](#)

## 5.25 Package Ecodena.Query.models

### Classes

- class [Query](#)

### 5.25.1 Detailed Description

Its a models.py of post Query, i.e. it creates a basic model for the post Query.

## 5.26 Package Ecodena.Query.tests

### Classes

- class [SimpleTest](#)

### 5.26.1 Detailed Description

This file demonstrates writing tests using the unittest module. These will pass when you run "manage.py test".

Replace this with more appropriate tests for your application.

## 5.27 Package Ecodena.Query.views

### Classes

- class [QueryForm](#)

### Functions

- def [postQuery](#)

### 5.27.1 Detailed Description

Its our Post Query app's views.py, i.e. how Post Query app will shown to user

### 5.27.2 Function Documentation

#### 5.27.2.1 def Ecodena.Query.views.postQuery ( *request* )

Method will provide a form to user, method will note down the time, date and userID of the user before all this method will check whether the user is authenticated or not, if authenticated the user will be able to post query and if not authenticated then a message will pop-up user has to login first.  
Parameter: request --> All the information of the form is stored in request.  
Function will return a Post query html page.

Definition at line 16 of file views.py.

## 5.28 Package Ecodena.Question

### Packages

- package [admin](#)

- package [models](#)
- package [tests](#)
- package [views](#)

## 5.29 Package Ecodena.Question.admin

### Classes

- class [TypeInline](#)
- class [LevelInline](#)
- class [QuestionAdmin](#)

## 5.30 Package Ecodena.Question.models

### Classes

- class [Level](#)
- class [Type](#)
- class [Question](#)
- class [BattleQuestion](#)
- class [TestCase](#)

## 5.31 Package Ecodena.Question.tests

### Classes

- class [SimpleTest](#)

### 5.31.1 Detailed Description

This file demonstrates writing tests using the unittest module. These will pass when you run "manage.py test".

Replace this with more appropriate tests for your application.

## 5.32 Package Ecodena.Question.views

### Classes

- class [CommentForm](#)

## Functions

- def [test](#)
- def [listQuestions](#)
- def [viewQuestionByID](#)

### 5.32.1 Function Documentation

#### 5.32.1.1 def Ecodena.Question.views.listQuestions ( *request* )

list all the questions with its title  
 parameter:request-->All the details associated with URL  
 returns the question.html webpage and the list of questions

Definition at line 19 of file views.py.

#### 5.32.1.2 def Ecodena.Question.views.test ( *request* )

Definition at line 15 of file views.py.

#### 5.32.1.3 def Ecodena.Question.views.viewQuestionByID ( *request*, *questionID* )

A question whose Id is passed as the parameter is displayed with the question title, question  
 Parameter:request-->All the information of the form is stored in request  
 questionID-->the ID of question to be displayed  
 If the user is not authenticated then this returns an Http Response stating that the user is  
 If the user is authenticated and the form filled is valid then it returns the page with the q

Definition at line 26 of file views.py.

## 5.33 Package Ecodena.settings

### Variables

- [DEBUG](#) = True
- [TEMPLATE\\_DEBUG](#) = [DEBUG](#)
- tuple [ADMINS](#)
- [MANAGERS](#) = [ADMINS](#)
- dictionary [DATABASES](#)
- string [TIME\\_ZONE](#) = 'America/Chicago'
- string [LANGUAGE\\_CODE](#) = 'en-us'
- int [SITE\\_ID](#) = 1
- [USE\\_I18N](#) = True
- [USE\\_L10N](#) = True
- string [MEDIA\\_ROOT](#) = "



- string `MEDIA_URL` = ""
- string `STATIC_ROOT` = ""
- string `STATIC_URL` = '/static/'
- string `ADMIN_MEDIA_PREFIX` = '/static/admin/'
- tuple `STATICFILES_DIRS`
- tuple `STATICFILES_FINDERS`
- string `SECRET_KEY` = 'k!\*n&!9\$t9sp61=vq(%t^(k@yz7o^=\*y=i2kmxtpi!4+88gyhl'
- tuple `TEMPLATE_LOADERS`
- tuple `MIDDLEWARE_CLASSES`
- string `ROOT_URLCONF` = 'Ecodena.urls'
- tuple `TEMPLATE_DIRS`
- tuple `INSTALLED_APPS`
- string `AUTH_PROFILE_MODULE` = 'Ecodena.User.Profile'
- dictionary `LOGGING`

### 5.33.1 Variable Documentation

#### 5.33.1.1 string `Ecodena::settings.ADMIN_MEDIA_PREFIX` = '/static/admin/'

Definition at line 68 of file settings.py.

#### 5.33.1.2 tuple `Ecodena::settings.ADMINS`

Initial value:

```
1 (
2     # ('Pankaj Bhambhani', 'pankajb64@gmail.com'),
3 )
```

Definition at line 6 of file settings.py.

#### 5.33.1.3 string `Ecodena::settings.AUTH_PROFILE_MODULE` = 'Ecodena.User.Profile'

Definition at line 139 of file settings.py.

#### 5.33.1.4 dictionary `Ecodena::settings.DATABASES`

Initial value:

```
1 {
2     'default': {
3         'ENGINE': 'django.db.backends.postgresql_psycopg2', # Add 'postgresql_psy
4         'NAME': '200901047', # Or path to database file if u
5         'USER': '200901047', # Not used with sqlite3.
6         'PASSWORD': 'admin', # Not used with sqlite3.
7         'HOST': '10.100.71.21', # Set to empty string for lo
```

```

        calhost. Not used with sqlite3.
8      'PORT': '5432',                                # Set to empty string for default. N
        ot used with sqlite3.
9      }
10 }

```

Definition at line 12 of file settings.py.

#### 5.33.1.5 Ecodena::settings.DEBUG = True

Definition at line 3 of file settings.py.

#### 5.33.1.6 tuple Ecodena::settings.INSTALLED\_APPS

**Initial value:**

```

1 (
2     'django.contrib.auth',
3     'django.contrib.contenttypes',
4     'django.contrib.sessions',
5     'django.contrib.sites',
6     'django.contrib.messages',
7     'django.contrib.staticfiles',
8     'Ecodena.User',
9     'Ecodena.Question',
10    'Ecodena.Attempt',
11    'Ecodena.Comment',
12    'Ecodena.Query',
13    'Ecodena.Compiler',
14    # Uncomment the next line to enable the admin:
15    'django.contrib.admin',
16    # Uncomment the next line to enable admin documentation:
17    'django.contrib.admindocs',
18 )

```

Definition at line 120 of file settings.py.

#### 5.33.1.7 string Ecodena::settings.LANGUAGE\_CODE = 'en-us'

Definition at line 34 of file settings.py.

#### 5.33.1.8 dictionary Ecodena::settings.LOGGING

**Initial value:**

```

1 {
2     'version': 1,
3     'disable_existing_loggers': False,
4     'handlers': {
5         'mail_admins': {
6             'level': 'ERROR',
7             'class': 'django.utils.log.AdminEmailHandler'
8         }
9     }
10 }

```

```
9     },
10     'loggers': {
11         'django.request': {
12             'handlers': ['mail_admins'],
13             'level': 'ERROR',
14             'propagate': True,
15         },
16     }
17 }
```

Definition at line 146 of file settings.py.

#### 5.33.1.9 Ecodena::settings.MANAGERS = ADMINS

Definition at line 10 of file settings.py.

#### 5.33.1.10 string Ecodena::settings.MEDIA\_ROOT = "

Definition at line 48 of file settings.py.

#### 5.33.1.11 string Ecodena::settings.MEDIA\_URL = "

Definition at line 53 of file settings.py.

#### 5.33.1.12 tuple Ecodena::settings.MIDDLEWARE\_CLASSES

Initial value:

```
1 (
2     'django.middleware.csrf.CsrfViewMiddleware',
3     'django.middleware.csrf.CsrfResponseMiddleware',
4     'django.middleware.common.CommonMiddleware',
5     'django.contrib.sessions.middleware.SessionMiddleware',
6     #'django.middleware.csrf.CsrfViewMiddleware',
7     'django.contrib.auth.middleware.AuthenticationMiddleware',
8     'django.contrib.messages.middleware.MessageMiddleware',
9 )
```

Definition at line 95 of file settings.py.

#### 5.33.1.13 string Ecodena::settings.ROOT\_URLCONF = 'Ecodena.urls'

Definition at line 105 of file settings.py.

#### 5.33.1.14 string Ecodena::settings.SECRET\_KEY = 'k!\*n&!9\$t9sp61=vq(0%t^(k@yz7o^=\*y=i2kmxtpi!4+88gyhl'

Definition at line 86 of file settings.py.

**5.33.1.15 int Ecodena::settings.SITE\_ID = 1**

Definition at line 36 of file settings.py.

**5.33.1.16 string Ecodena::settings.STATIC\_ROOT = "**

Definition at line 59 of file settings.py.

**5.33.1.17 string Ecodena::settings.STATIC\_URL = '/static/'**

Definition at line 63 of file settings.py.

**5.33.1.18 tuple Ecodena::settings.STATICFILES\_DIRS**

**Initial value:**

```
1 (
2     # Put strings here, like "/home/html/static" or "C:/www/django/static".
3     # Always use forward slashes, even on Windows.
4     # Don't forget to use absolute paths, not relative paths.
5 )
```

Definition at line 71 of file settings.py.

**5.33.1.19 tuple Ecodena::settings.STATICFILES\_FINDERS**

**Initial value:**

```
1 (
2     'django.contrib.staticfiles.finders.FileSystemFinder',
3     'django.contrib.staticfiles.finders.AppDirectoriesFinder',
4     # 'django.contrib.staticfiles.finders.DefaultStorageFinder',
5 )
```

Definition at line 79 of file settings.py.

**5.33.1.20 Ecodena::settings.TEMPLATE\_DEBUG = DEBUG**

Definition at line 4 of file settings.py.

**5.33.1.21 tuple Ecodena::settings.TEMPLATE\_DIRS**

**Initial value:**

```
1 (
2     # Put strings here, like "/home/html/django_templates" or "C:/www/django/temp
   lates".
3     # Always use forward slashes, even on Windows.
```

```
4     # Don't forget to use absolute paths, not relative paths.
5     'Main/Template',
6     'Question/Template',
7     'Query/Template',
8     'Attempt/Template'
9 )
```

Definition at line 107 of file settings.py.

#### 5.33.1.22 tuple Ecodena::settings.TEMPLATE\_LOADERS

**Initial value:**

```
1 (
2     'django.template.loaders.filesystem.Loader',
3     'django.template.loaders.app_directories.Loader',
4     # 'django.template.loaders.eggs.Loader',
5 )
```

Definition at line 89 of file settings.py.

#### 5.33.1.23 string Ecodena::settings.TIME\_ZONE = 'America/Chicago'

Definition at line 30 of file settings.py.

#### 5.33.1.24 Ecodena::settings.USE\_I18N = True

Definition at line 40 of file settings.py.

#### 5.33.1.25 Ecodena::settings.USE\_L10N = True

Definition at line 44 of file settings.py.

## 5.34 Package Ecodena.urls

### Variables

- tuple [urlpatterns](#)

#### 5.34.1 Variable Documentation

##### 5.34.1.1 tuple Ecodena::urls.urlpatterns

**Initial value:**

```

1 patterns('',
2 # Examples:
3     url(r'^$', 'Ecodena.Main.views.home', name='home'),
4     # url(r'^Ecodena/', include('Ecodena.foo.urls')),
5
6     # Uncomment the admin/doc line below to enable admin documentation:
7     url(r'^admin/doc/', include('django.contrib.admindocs.urls')),
8
9     # Uncomment the next line to enable the admin:
10    url(r'^admin/', include(admin.site.urls)),
11
12    #url(r'^test/$', 'Ecodena.Question.views.test'),
13
14    url(r'^questions/$', 'Ecodena.Question.views.listQuestions'),
15    url(r'^postQuery/$', 'Ecodena.Query.views.postQuery'),
16    url(r'^questions/(?P<questionID>[0-9]+)/$', 'Ecodena.Question.views.viewQ
uestionByID'),
17    url(r'^submitSolution$', 'Ecodena.Attempt.views.submitSolution'),
18    #url(r'^profile/', Ecodena.User.views.viewProfile),
19    #url(r'^profile/?P<profileID>[0-9]+$', Ecodena.User.views.viewProfileByID)
20    ,
21    #url(r'^profile/edit/', Ecodena.User.views.editProfile),
22    url(r'^solutions/$', 'Ecodena.Attempt.views.listSolutions'),
23    url(r'^solutions/(?P<solutionID>[0-9]+)$', 'Ecodena.Attempt.views.viewSolu
tion')
24    )

```

Definition at line 7 of file urls.py.

## 5.35 Package Ecodena.User

### Packages

- package [admin](#)
- package [models](#)
- package [tests](#)
- package [views](#)

## 5.36 Package Ecodena.User.admin

### Classes

- class [ProfileAdmin](#)

## 5.37 Package Ecodena.User.models

### Classes

- class [Programmer](#)
- class [Profile](#)

### 5.37.1 Detailed Description

Its a models.py of User app, i.e. it creates a basic model for User app.

## 5.38 Package Ecodena.User.tests

### Classes

- class [SimpleTest](#)

### 5.38.1 Detailed Description

This file demonstrates writing tests using the unittest module. These will pass when you run "manage.py test".

Replace this with more appropriate tests for your application.

## 5.39 Package Ecodena.User.views

### Classes

- class [ProfileForm](#)

### Functions

- def [viewProfile](#)
- def [editProfile](#)

### 5.39.1 Function Documentation

5.39.1.1 def Ecodena.User.views.editProfile ( *request*, *if*, *request*, *user*, *is\_authenticated* )

Definition at line 43 of file views.py.

5.39.1.2 def Ecodena.User.views.viewProfile ( *request* )

Definition at line 24 of file views.py.





## Chapter 6

# Class Documentation

### 6.1 Ecodena.Attempt.models.Attempt Class Reference

#### Classes

- class [Meta](#)

#### Public Member Functions

- def [getAttemptID](#)
- def [setAttemptID](#)
- def [isCorrect](#)
- def [setStatus](#)
- def [getTimeOfSubmission](#)
- def [setTimeOfSubmission](#)
- def [getUserID](#)
- def [setUserID](#)
- def [getErrorReportID](#)
- def [setErrorReportID](#)
- def [getSolution](#)
- def [setSolution](#)
- def [getQuestionID](#)
- def [setQuestionID](#)
- def [getCompilerVersionOfAttempt](#)
- def [setCompilerVersionOfAttempt](#)
- def [\\_\\_unicode\\_\\_](#)

#### Public Attributes

- [attemptID\\_f](#)
- [status\\_f](#)

- [timeOfSubmission\\_f](#)
- [userID\\_f](#)
- [errorReportID\\_f](#)
- [solution\\_f](#)
- [questionID\\_f](#)
- [compilerVersion\\_f](#)

### Static Public Attributes

- tuple [attemptID\\_f](#) = models.AutoField(primary\_key=True)
- tuple [questionID\\_f](#) = models.ForeignKey([Question](#), verbose\_name="Question", null=False)
- tuple [userID\\_f](#) = models.ForeignKey(User, verbose\_name="User ID", null=False)
- tuple [solution\\_f](#) = models.TextField("Solution uploaded by the User", null=False)
- tuple [errorReportID\\_f](#) = models.ForeignKey([ErrorReport](#), verbose\_name="Error Report for the Solution", null=True)
- tuple [status\\_f](#) = models.BooleanField("Status of attempt - true = right, false = wrong", blank=True, default=False)
- tuple [timeOfSubmission\\_f](#) = models.DateTimeField("Time of Submission", null=False)
- tuple [compilerVersion\\_f](#) = models.ForeignKey([CompilerVersion](#), verbose\_name="Compiler version of the [Attempt](#)", null=False)

### Properties

- [attemptID](#) = property(getAttemptID, setAttemptID)
- [status](#) = property(isCorrect, setStatus)
- [timeOfSubmission](#) = property(getTimeOfSubmission, setTimeOfSubmission)
- [userID](#) = property(getUserID, setUserID)
- [errorReportID](#) = property(getErrorReportID, setErrorReportID)
- [solutionText](#) = property(getSolution, setSolution)
- [questionID](#) = property(getQuestionID, setQuestionID)
- [compilerVersion](#) = property(getCompilerVersionOfAttempt, setCompilerVersionOfAttempt)

#### 6.1.1 Detailed Description

Creates an Attempt Entity which is a subclass of models.Model class

Definition at line 66 of file models.py.

#### 6.1.2 Member Function Documentation

##### 6.1.2.1 def Ecodena.Attempt.models.Attempt.\_\_unicode\_\_( self )

Definition at line 137 of file models.py.

**6.1.2.2** `def Ecodena.Attempt.models.Attempt.getAttemptID ( self )`

Definition at line 77 of file models.py.

**6.1.2.3** `def Ecodena.Attempt.models.Attempt.getCompilerVersionOfAttempt ( self )`

Definition at line 126 of file models.py.

**6.1.2.4** `def Ecodena.Attempt.models.Attempt.getErrorReportID ( self )`

Definition at line 106 of file models.py.

**6.1.2.5** `def Ecodena.Attempt.models.Attempt.getQuestionID ( self )`

Definition at line 119 of file models.py.

**6.1.2.6** `def Ecodena.Attempt.models.Attempt.getSolution ( self )`

Definition at line 112 of file models.py.

**6.1.2.7** `def Ecodena.Attempt.models.Attempt.getTimeOfSubmission ( self )`

Definition at line 92 of file models.py.

**6.1.2.8** `def Ecodena.Attempt.models.Attempt.getUserID ( self )`

Definition at line 99 of file models.py.

**6.1.2.9** `def Ecodena.Attempt.models.Attempt.isCorrect ( self )`

Checks the status and returns its value

Definition at line 84 of file models.py.

**6.1.2.10** `def Ecodena.Attempt.models.Attempt.setAttemptID ( self, aID )`

Definition at line 79 of file models.py.

**6.1.2.11** `def Ecodena.Attempt.models.Attempt.setCompilerVersionOfAttempt ( self,  
compilerVersion )`

Definition at line 128 of file models.py.

6.1.2.12 `def Ecodena.Attempt.models.Attempt.setErrorReportID ( self, reportID )`

Definition at line 108 of file models.py.

6.1.2.13 `def Ecodena.Attempt.models.Attempt.setQuestionID ( self, qID )`

Definition at line 121 of file models.py.

6.1.2.14 `def Ecodena.Attempt.models.Attempt.setSolution ( self, sol )`

Definition at line 114 of file models.py.

6.1.2.15 `def Ecodena.Attempt.models.Attempt.setStatus ( self, status )`

Definition at line 87 of file models.py.

6.1.2.16 `def Ecodena.Attempt.models.Attempt.setTimeOfSubmission ( self, time )`

Definition at line 94 of file models.py.

6.1.2.17 `def Ecodena.Attempt.models.Attempt.setUserID ( self, uID )`

Definition at line 101 of file models.py.

### 6.1.3 Member Data Documentation

6.1.3.1 `tuple Ecodena.Attempt.models.Attempt.attemptID_f =  
models.AutoField(primary_key=True) [static]`

Definition at line 68 of file models.py.

6.1.3.2 `Ecodena.Attempt.models.Attempt.attemptID_f`

Definition at line 79 of file models.py.

6.1.3.3 `tuple Ecodena.Attempt.models.Attempt.compilerVersion_f =  
models.ForeignKey(CompilerVersion, verbose_name = "Compiler version of the  
Attempt",null=False) [static]`

Definition at line 75 of file models.py.

**6.1.3.4 Ecodena.Attempt.models.Attempt.compilerVersion\_f**

Definition at line 128 of file models.py.

```
6.1.3.5 tuple Ecodena.Attempt.models.Attempt.errorReportID_f =  
models.ForeignKey(ErrorReport, verbose_name="Error Report for the Solution",  
null=True) [static]
```

Definition at line 72 of file models.py.

**6.1.3.6 Ecodena.Attempt.models.Attempt.errorReportID\_f**

Definition at line 108 of file models.py.

```
6.1.3.7 tuple Ecodena.Attempt.models.Attempt.questionID_f =  
models.ForeignKey(Question, verbose_name="Question", null=False) [static]
```

Definition at line 69 of file models.py.

**6.1.3.8 Ecodena.Attempt.models.Attempt.questionID\_f**

Definition at line 121 of file models.py.

```
6.1.3.9 tuple Ecodena.Attempt.models.Attempt.solution_f = models.TextField("Solution  
uploaded by the User", null=False) [static]
```

Definition at line 71 of file models.py.

**6.1.3.10 Ecodena.Attempt.models.Attempt.solution\_f**

Definition at line 114 of file models.py.

**6.1.3.11 Ecodena.Attempt.models.Attempt.status\_f**

Definition at line 87 of file models.py.

```
6.1.3.12 tuple Ecodena.Attempt.models.Attempt.status_f =  
models.BooleanField("Status of attempt - true = right, false = wrong", blank=True,  
default=False) [static]
```

Definition at line 73 of file models.py.

**6.1.3.13** tuple `Ecodena.Attempt.models.Attempt.timeOfSubmission_f = models.DateTimeField("Time of Submission", null=False)` `[static]`

Definition at line 74 of file models.py.

**6.1.3.14** `Ecodena.Attempt.models.Attempt.timeOfSubmission_f`

Definition at line 94 of file models.py.

**6.1.3.15** tuple `Ecodena.Attempt.models.Attempt.userID_f = models.ForeignKey(User, verbose_name="User ID", null=False)` `[static]`

Definition at line 70 of file models.py.

**6.1.3.16** `Ecodena.Attempt.models.Attempt.userID_f`

Definition at line 101 of file models.py.

## 6.1.4 Property Documentation

**6.1.4.1** `Ecodena.Attempt.models.Attempt.attemptID = property(getAttemptID,setAttemptID)` `[static]`

Definition at line 81 of file models.py.

**6.1.4.2** `Ecodena.Attempt.models.Attempt.compilerVersion = property(getCompilerVersionOfAttempt,setCompilerVersionOfAttempt)` `[static]`

Definition at line 131 of file models.py.

**6.1.4.3** `Ecodena.Attempt.models.Attempt.errorReportID = property(getErrorReportID,setErrorReportID)` `[static]`

Definition at line 110 of file models.py.

**6.1.4.4** `Ecodena.Attempt.models.Attempt.questionID = property(getQuestionID,setQuestionID)` `[static]`

Definition at line 124 of file models.py.

6.1.4.5 Ecodena.Attempt.models.Attempt.solutionText = property(getSolution,setSolution)  
[static]

Definition at line 117 of file models.py.

6.1.4.6 Ecodena.Attempt.models.Attempt.status = property(isCorrect, setStatus) [static]

Definition at line 89 of file models.py.

6.1.4.7 Ecodena.Attempt.models.Attempt.timeOfSubmission =  
property(getTimeOfSubmission, setTimeOfSubmission) [static]

Definition at line 96 of file models.py.

6.1.4.8 Ecodena.Attempt.models.Attempt.userID = property(getUserID,setUserID)  
[static]

Definition at line 103 of file models.py.

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

- /home/panx/ecodena/src/Ecodena/Attempt/[models.py](#)

## 6.2 Ecodena.Attempt.admin.AttemptAdmin Class Reference

### Static Public Attributes

- tuple [list\\_display](#) = ('attemptID\_f', 'questionID\_f', 'userID', 'solution\_f', 'errorReportID', 'status\_f', 'timeOfSubmission' )

### 6.2.1 Detailed Description

Definition at line 5 of file admin.py.

### 6.2.2 Member Data Documentation

6.2.2.1 tuple Ecodena.Attempt.admin.AttemptAdmin.list\_display = ('attemptID\_f',  
'questionID\_f', 'userID', 'solution\_f', 'errorReportID', 'status\_f', 'timeOfSubmission' )  
[static]

Definition at line 6 of file admin.py.

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

- /home/panx/ecodena/src/Ecodena/Attempt/[admin.py](#)

## 6.3 Ecodena.Question.models.BattleQuestion Class Reference

### 6.3.1 Detailed Description

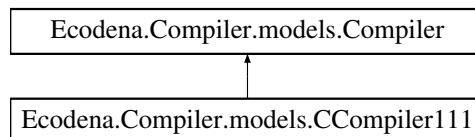
Definition at line 103 of file models.py.

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

- [/home/panx/ecodena/src/Ecodena/Question/models.py](#)

## 6.4 Ecodena.Compiler.models.CCompiler111 Class Reference

Inheritance diagram for Ecodena.Compiler.models.CCompiler111:



### Public Member Functions

- def [compile](#)

### 6.4.1 Detailed Description

Definition at line 99 of file models.py.

### 6.4.2 Member Function Documentation

#### 6.4.2.1 def Ecodena.Compiler.models.CCompiler111.compile ( self )

Reimplemented from [Ecodena.Compiler.models.Compiler](#).

Definition at line 101 of file models.py.

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

- [/home/panx/ecodena/src/Ecodena/Compiler/models.py](#)

## 6.5 Ecodena.Comment.models.Comment Class Reference



## Classes

- class [Meta](#)

## Public Member Functions

- def [getCommentText](#)
- def [setCommentText](#)
- def [getTimeStamp](#)
- def [setTimeStamp](#)
- def [isReported](#)
- def [report](#)
- def [getUserID](#)
- def [setUserID](#)
- def [getReportingUserID](#)
- def [setReportingUserID](#)
- def [getQuestionID](#)
- def [setQuestionID](#)
- def [\\_\\_unicode\\_\\_](#)

## Public Attributes

- [commentText\\_f](#)
- [timeStamp\\_f](#)
- [isReported\\_f](#)
- [userID\\_f](#)
- [reportingUserID\\_f](#)
- [questionID\\_f](#)

## Static Public Attributes

- tuple [commentID\\_f](#) = models.AutoField(primary\_key=True)
- tuple [commentText\\_f](#) = models.TextField("The Body of the [Comment](#)", null=False)
- tuple [isReported\\_f](#) = models.BooleanField("Is the comment Reported ?", default=False)
- tuple [userID\\_f](#) = models.ForeignKey(User, verbose\_name="User ID", null=False, related\_name='+')
- tuple [reportingUserID\\_f](#) = models.ForeignKey(User, verbose\_name="User who reported this comment", blank=True, null=True)
- tuple [questionID\\_f](#) = models.ForeignKey([Question](#), verbose\_name="Question ID", null=False)
- tuple [timeStamp\\_f](#) = models.DateTimeField("Time of Posting", null=False)

## Properties

- `commentText` = property(getCommentText, setCommentText)
- `timeStamp` = property(getTimeStamp, setTimeStamp)
- `isReported` = property(isReported, report)
- `userID` = property(getUserID, setUserID)
- `reportingUserID` = property(getReportingUserID, setReportingUserID)
- `questionID` = property(getQuestionID, setQuestionID)

### 6.5.1 Detailed Description

Here `Comment` is subclass of `models.Model` and Creates an entity named `Comment`.

Definition at line 9 of file `models.py`.

### 6.5.2 Member Function Documentation

6.5.2.1 `def Ecodena.Comment.models.Comment.__unicode__( self )`

Definition at line 75 of file `models.py`.

6.5.2.2 `def Ecodena.Comment.models.Comment.getCommentText( self )`

Definition at line 20 of file `models.py`.

6.5.2.3 `def Ecodena.Comment.models.Comment.getQuestionID( self )`

Definition at line 65 of file `models.py`.

6.5.2.4 `def Ecodena.Comment.models.Comment.getReportingUserID( self )`

Definition at line 56 of file `models.py`.

6.5.2.5 `def Ecodena.Comment.models.Comment.getTimeStamp( self )`

Definition at line 28 of file `models.py`.

6.5.2.6 `def Ecodena.Comment.models.Comment.getUserID( self )`

Definition at line 49 of file `models.py`.

**6.5.2.7 def Ecodena.Comment.models.Comment.isReported ( self )**

If user is abused from any other user's comment then that user can report abuse that comment.  
And if the function will return comment is reported thanthat means someone has reported it.

Definition at line 36 of file models.py.

**6.5.2.8 def Ecodena.Comment.models.Comment.report ( self, status )**

If user is abused from any other user's comment then that user can report abuse that comment.  
So the user will set this status i.e comment isreported now.

Definition at line 42 of file models.py.

**6.5.2.9 def Ecodena.Comment.models.Comment.setCommentText ( self, comment )**

Definition at line 22 of file models.py.

**6.5.2.10 def Ecodena.Comment.models.Comment.setQuestionID ( self, qID )**

Definition at line 67 of file models.py.

**6.5.2.11 def Ecodena.Comment.models.Comment.setReportingUserID ( self, ruID )**

Definition at line 58 of file models.py.

**6.5.2.12 def Ecodena.Comment.models.Comment.setTimeStamp ( self, Timestamp )**

Definition at line 30 of file models.py.

**6.5.2.13 def Ecodena.Comment.models.Comment.setUserID ( self, uID )**

Definition at line 51 of file models.py.

**6.5.3 Member Data Documentation****6.5.3.1 tuple Ecodena.Comment.models.Comment.commentID\_f =  
models.AutoField(primary\_key=True) [static]**

Definition at line 11 of file models.py.

**6.5.3.2** tuple `Ecodena.Comment.models.Comment.commentText_f =`  
`models.TextField("The Body of the Comment", null=False)` `[static]`

Definition at line 12 of file models.py.

**6.5.3.3** `Ecodena.Comment.models.Comment.commentText_f`

Definition at line 22 of file models.py.

**6.5.3.4** `Ecodena.Comment.models.Comment.isReported_f`

Definition at line 43 of file models.py.

**6.5.3.5** tuple `Ecodena.Comment.models.Comment.isReported_f =`  
`models.BooleanField("Is the comment Reported ?", default=False)` `[static]`

Definition at line 13 of file models.py.

**6.5.3.6** `Ecodena.Comment.models.Comment.questionID_f`

Definition at line 67 of file models.py.

**6.5.3.7** tuple `Ecodena.Comment.models.Comment.questionID_f =`  
`models.ForeignKey(Question, verbose_name="Question ID", null=False)`  
`[static]`

Definition at line 16 of file models.py.

**6.5.3.8** `Ecodena.Comment.models.Comment.reportingUserID_f`

Definition at line 58 of file models.py.

**6.5.3.9** tuple `Ecodena.Comment.models.Comment.reportingUserID_f =`  
`models.ForeignKey(User, verbose_name="User who reported this comment", blank=True,`  
`null=True)` `[static]`

Definition at line 15 of file models.py.

**6.5.3.10** tuple `Ecodena.Comment.models.Comment.timeStamp_f =`  
`models.DateTimeField("Time of Posting", null=False)` `[static]`

Definition at line 17 of file models.py.

**6.5.3.11 Ecodena.Comment.models.Comment.timeStamp\_f**

Definition at line 30 of file models.py.

```
6.5.3.12 tuple Ecodena.Comment.models.Comment.userID_f =  
models.ForeignKey(User, verbose_name="User ID", null=False, related_name='+')  
[static]
```

Definition at line 14 of file models.py.

**6.5.3.13 Ecodena.Comment.models.Comment.userID\_f**

Definition at line 51 of file models.py.

**6.5.4 Property Documentation**

```
6.5.4.1 Ecodena.Comment.models.Comment.commentText = property(getCommentText,  
setCommentText) [static]
```

Definition at line 24 of file models.py.

```
6.5.4.2 Ecodena.Comment.models.Comment.isReported = property(isReported, report)  
[static]
```

Definition at line 47 of file models.py.

```
6.5.4.3 Ecodena.Comment.models.Comment.questionID = property(getQuestionID,  
setQuestionID) [static]
```

Definition at line 69 of file models.py.

```
6.5.4.4 Ecodena.Comment.models.Comment.reportingUserID = property(getReportingUserID,  
setReportingUserID) [static]
```

Definition at line 61 of file models.py.

```
6.5.4.5 Ecodena.Comment.models.Comment.timeStamp = property(getTimeStamp,  
setTimeStamp) [static]
```

Definition at line 32 of file models.py.

**6.5.4.6** `Ecodena.Comment.models.Comment.userID = property(getUserID, setUserID)`  
`[static]`

Definition at line 53 of file `models.py`.

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

- `/home/panx/ecodena/src/Ecodena/Comment/models.py`

## 6.6 Ecodena.Comment.admin.CommentAdmin Class Reference

### Static Public Attributes

- tuple `list_display` = ('commentID\_f', 'commentText', 'userID', 'isReported\_f', 'reportingUserID', 'questionID', 'timeStamp' )

#### 6.6.1 Detailed Description

Definition at line 4 of file `admin.py`.

#### 6.6.2 Member Data Documentation

**6.6.2.1** tuple `Ecodena.Comment.admin.CommentAdmin.list_display` =  
( 'commentID\_f', 'commentText', 'userID', 'isReported\_f', 'reportingUserID', 'questionID',  
'timeStamp' ) `[static]`

Definition at line 5 of file `admin.py`.

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

- `/home/panx/ecodena/src/Ecodena/Comment/admin.py`

## 6.7 Ecodena.Question.views.CommentForm Class Reference

### Static Public Attributes

- tuple `text` = `forms.CharField(widget=forms.Textarea)`

#### 6.7.1 Detailed Description

Creates a form for writing a comment

Definition at line 10 of file `views.py`.

### 6.7.2 Member Data Documentation

6.7.2.1 tuple `Ecodena.Question.views.CommentForm.text` =  
`forms.CharField(widget=forms.Textarea)` `[static]`

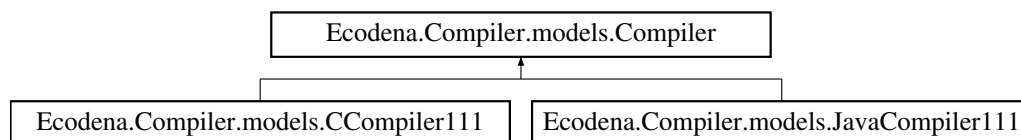
Definition at line 12 of file `views.py`.

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

- `/home/panx/ecodena/src/Ecodena/Question/views.py`

## 6.8 Ecodena.Compiler.models.Compiler Class Reference

Inheritance diagram for `Ecodena.Compiler.models.Compiler`:



### Public Member Functions

- def [getLanguage](#)
- def [setLanguage](#)
- def [getCompilerVersion](#)
- def [setCompilerVersion](#)
- def [getQuestion](#)
- def [setQuestion](#)
- def [getAttempt](#)
- def [setAttempt](#)
- def [compile](#)
- def [generateErrorReport](#)

### Static Public Attributes

- string [language](#) = ""
- string [compilerVersion](#) = ""
- [question](#) = None
- [attempt](#) = None
- [errorReport](#) = None

## Properties

- [language](#) = property(getLanguage,setLanguage)
- [CompilerVersion](#) = property(getCompilerVersion,setCompilerVersion)

### 6.8.1 Detailed Description

Definition at line 60 of file models.py.

### 6.8.2 Member Function Documentation

#### 6.8.2.1 `def Ecodena.Compiler.models.Compiler.compile ( self )`

Reimplemented in [Ecodena.Compiler.models.CCompiler111](#), and [Ecodena.Compiler.models.JavaCompiler111](#).

Definition at line 93 of file models.py.

#### 6.8.2.2 `def Ecodena.Compiler.models.Compiler.generateErrorReport ( self )`

Definition at line 96 of file models.py.

#### 6.8.2.3 `def Ecodena.Compiler.models.Compiler.getAttempt ( self )`

Definition at line 87 of file models.py.

#### 6.8.2.4 `def Ecodena.Compiler.models.Compiler.getCompilerVersion ( self )`

Definition at line 75 of file models.py.

#### 6.8.2.5 `def Ecodena.Compiler.models.Compiler.getLanguage ( self )`

Definition at line 69 of file models.py.

#### 6.8.2.6 `def Ecodena.Compiler.models.Compiler.getQuestion ( self )`

Definition at line 81 of file models.py.

#### 6.8.2.7 `def Ecodena.Compiler.models.Compiler.setAttempt ( self, attempt )`

Definition at line 89 of file models.py.



**6.8.2.8** `def Ecodena.Compiler.models.Compiler.setCompilerVersion ( self, text )`

Definition at line 77 of file models.py.

**6.8.2.9** `def Ecodena.Compiler.models.Compiler.setLanguage ( self, text )`

Definition at line 71 of file models.py.

**6.8.2.10** `def Ecodena.Compiler.models.Compiler.setQuestion ( self, question )`

Definition at line 83 of file models.py.

### 6.8.3 Member Data Documentation

**6.8.3.1** `Ecodena.Compiler.models.Compiler.attempt = None` `[static]`

Definition at line 65 of file models.py.

**6.8.3.2** `string Ecodena.Compiler.models.Compiler.compilerVersion = ""`  
`[static]`

Definition at line 63 of file models.py.

**6.8.3.3** `Ecodena.Compiler.models.Compiler.errorReport = None` `[static]`

Definition at line 66 of file models.py.

**6.8.3.4** `string Ecodena.Compiler.models.Compiler.language = ""` `[static]`

Definition at line 62 of file models.py.

**6.8.3.5** `Ecodena.Compiler.models.Compiler.question = None` `[static]`

Definition at line 64 of file models.py.

### 6.8.4 Property Documentation

**6.8.4.1** `Ecodena.Compiler.models.Compiler.CompilerVersion =`  
`property(getCompilerVersion,setCompilerVersion)` `[static]`

Definition at line 79 of file models.py.

6.8.4.2 **Ecodena.Compiler.models.Compiler.language** =  
property(getLanguage,setLanguage) [static]

Definition at line 73 of file models.py.

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

- /home/panx/ecodena/src/Ecodena/Compiler/[models.py](#)

## 6.9 Ecodena.Compiler.models.CompilerVersion Class Reference

### Classes

- class [Meta](#)

### Public Member Functions

- def [getVersionName](#)
- def [setVersionName](#)
- def [getLanguage](#)
- def [setLanguage](#)
- def [languageName](#)
- def [\\_\\_unicode\\_\\_](#)

### Public Attributes

- [versionName\\_f](#)
- [language\\_f](#)

### Static Public Attributes

- tuple [versionID\\_f](#) = models.AutoField(primary\_key=True)
- tuple [versionName\\_f](#) = models.CharField("Version Name (Code or Number)", max\_length=40, null=False)
- tuple [language\\_f](#) = models.ForeignKey([Language](#), verbose\_name="Language of Compilation", null=False)

### Properties

- [versionName](#) = property(getVersionName, setVersionName)
- [language](#) = property(getLanguage, setLanguage)

### 6.9.1 Detailed Description

Definition at line 28 of file models.py.

### 6.9.2 Member Function Documentation

#### 6.9.2.1 `def Ecodena.Compiler.models.CompilerVersion.__unicode__( self )`

Definition at line 56 of file models.py.

#### 6.9.2.2 `def Ecodena.Compiler.models.CompilerVersion.getLanguage( self )`

Definition at line 41 of file models.py.

#### 6.9.2.3 `def Ecodena.Compiler.models.CompilerVersion.getVersionName( self )`

Definition at line 33 of file models.py.

#### 6.9.2.4 `def Ecodena.Compiler.models.CompilerVersion.languageName( self )`

Definition at line 49 of file models.py.

#### 6.9.2.5 `def Ecodena.Compiler.models.CompilerVersion.setLanguage( self, language )`

Definition at line 44 of file models.py.

#### 6.9.2.6 `def Ecodena.Compiler.models.CompilerVersion.setVersionName( self, versionName )`

Definition at line 36 of file models.py.

### 6.9.3 Member Data Documentation

#### 6.9.3.1 `tuple Ecodena.Compiler.models.CompilerVersion.language_f = models.ForeignKey(Language, verbose_name="Language of Compilation", null=False)` `[static]`

Definition at line 31 of file models.py.

#### 6.9.3.2 `Ecodena.Compiler.models.CompilerVersion.language_f`

Definition at line 44 of file models.py.

6.9.3.3 tuple `Ecodena.Compiler.models.CompilerVersion.versionID_f = models.AutoField(primary_key=True)` `[static]`

Definition at line 29 of file `models.py`.

6.9.3.4 `Ecodena.Compiler.models.CompilerVersion.versionName_f`

Definition at line 36 of file `models.py`.

6.9.3.5 tuple `Ecodena.Compiler.models.CompilerVersion.versionName_f = models.CharField("Version Name (Code or Number)", max_length=40, null=False)` `[static]`

Definition at line 30 of file `models.py`.

## 6.9.4 Property Documentation

6.9.4.1 `Ecodena.Compiler.models.CompilerVersion.language = property(getLanguage, setLanguage)` `[static]`

Definition at line 47 of file `models.py`.

6.9.4.2 `Ecodena.Compiler.models.CompilerVersion.versionName = property(getVersionName, setVersionName)` `[static]`

Definition at line 39 of file `models.py`.

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

- </home/panx/ecodena/src/Ecodena/Compiler/models.py>

## 6.10 Ecodena.Compiler.admin.CompilerVersionAdmin Class Reference

### Static Public Attributes

- tuple `list_display = ('versionName_f', 'languageName')`

### 6.10.1 Detailed Description

Definition at line 8 of file `admin.py`.

### 6.10.2 Member Data Documentation

6.10.2.1 tuple `Ecodena.Compiler.admin.CompilerVersionAdmin.list_display` = (  
    'versionName\_f', 'languageName') `[static]`

Definition at line 10 of file `admin.py`.

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

- `/home/panx/ecodena/src/Ecodena/Compiler/admin.py`

## 6.11 Ecodena.Attempt.models.ErrorReport Class Reference

### Classes

- class `Meta`

### Public Member Functions

- def `getTimeRequirement`
- def `setTimeRequirement`
- def `getMemory`
- def `setMemory`
- def `getErrorType`
- def `setErrorType`
- def `getErrorMessage`
- def `setErrorMessage`
- def `getTestCaseLevel`
- def `setTestCaseLevel`
- def `__unicode__`

### Public Attributes

- `timeRequirement_f`
- `memory_f`
- `errorType_f`
- `errorMessage_f`
- `testCaseLevel_f`

### Static Public Attributes

- tuple `errorReportID_f` = `models.AutoField(primary_key=True)`
- tuple `timeRequirement_f` = `models.FloatField("Time Required for compiling")`
- tuple `memory_f` = `models.FloatField("Memory Requirement")`

- tuple `ERROR_TYPE` = ((0,'Correct Answer'),(1,'Compilation Error'),(2,'Run Time Error'),(3,'Time limit Exceeded'),(4,'Memory Limit Exceeded'),(5,'Wrong Answer'),(6,'Pending'))
- tuple `errorType_f` = models.CharField( max\_length = 40, choices=`ERROR_TYPE`,verbose\_name="Types of Errors")
- tuple `errorMessage_f` = models.TextField("Error Message generated")
- tuple `testCaseLevel_f` = models.CharField(max\_length=40, choices=TestCase.CASE\_TYPE, verbose\_name="Test Case Level where error occurred (Select High if solution is correct)")

## Properties

- `timeRequirement` = property(getTimeRequirement, setTimeRequirement)
- `memory` = property(getMemory, setMemory)
- `errorType` = property(getErrorType, setErrorType)
- `errorMessage` = property(getErrorMessage, setErrorMessage)
- `testCaseLevel` = property(getTestCaseLevel, setTestCaseLevel)

### 6.11.1 Detailed Description

Creates an `ErrorReport` Entity which is a subclass of `models.Model` class

Definition at line 13 of file `models.py`.

### 6.11.2 Member Function Documentation

6.11.2.1 `def Ecodena.Attempt.models.ErrorReport.__unicode__( self )`

Definition at line 62 of file `models.py`.

6.11.2.2 `def Ecodena.Attempt.models.ErrorReport.getErrorMessage ( self )`

Definition at line 45 of file `models.py`.

6.11.2.3 `def Ecodena.Attempt.models.ErrorReport.getErrorType ( self )`

Definition at line 38 of file `models.py`.

6.11.2.4 `def Ecodena.Attempt.models.ErrorReport.getMemory ( self )`

Definition at line 31 of file `models.py`.

6.11.2.5 `def Ecodena.Attempt.models.ErrorReport.getTestCaseLevel ( self )`

Definition at line 52 of file `models.py`.

6.11.2.6 `def Ecodena.Attempt.models.ErrorReport.getTimeRequirement ( self )`

Definition at line 24 of file models.py.

6.11.2.7 `def Ecodena.Attempt.models.ErrorReport.setErrorMessage ( self, message )`

Definition at line 47 of file models.py.

6.11.2.8 `def Ecodena.Attempt.models.ErrorReport.setErrorType ( self, type )`

Definition at line 40 of file models.py.

6.11.2.9 `def Ecodena.Attempt.models.ErrorReport.setMemory ( self, memory )`

Definition at line 33 of file models.py.

6.11.2.10 `def Ecodena.Attempt.models.ErrorReport.setTestCaseLevel ( self, level )`

Definition at line 54 of file models.py.

6.11.2.11 `def Ecodena.Attempt.models.ErrorReport.setTimeRequirement ( self, time )`

Definition at line 26 of file models.py.

### 6.11.3 Member Data Documentation

6.11.3.1 `tuple Ecodena.Attempt.models.ErrorReport.ERROR_TYPE = ((0,'Correct Answer'),(1,'Compilation Error'),(2,'Run Time Error'),(3,'Time limit Exceeded'),(4,'Memory Limit Exceeded'),(5,'Wrong Answer'),(6,'Pending'))` `[static]`

Definition at line 18 of file models.py.

6.11.3.2 `Ecodena.Attempt.models.ErrorReport.errorMessage_f`

Definition at line 47 of file models.py.

6.11.3.3 `tuple Ecodena.Attempt.models.ErrorReport.errorMessage_f = models.TextField("Error Message generated")` `[static]`

Definition at line 20 of file models.py.

**6.11.3.4** `tuple Ecodena.Attempt.models.ErrorReport.errorReportID_f =  
models.AutoField(primary_key=True) [static]`

Definition at line 15 of file models.py.

**6.11.3.5** `Ecodena.Attempt.models.ErrorReport.errorType_f`

Definition at line 40 of file models.py.

**6.11.3.6** `tuple Ecodena.Attempt.models.ErrorReport.errorType_f = models.CharField(  
max_length = 40, choices=ERROR_TYPE,verbose_name="Types of Errors")  
[static]`

Definition at line 19 of file models.py.

**6.11.3.7** `Ecodena.Attempt.models.ErrorReport.memory_f`

Definition at line 33 of file models.py.

**6.11.3.8** `tuple Ecodena.Attempt.models.ErrorReport.memory_f =  
models.FloatField("Memory Requirement") [static]`

Definition at line 17 of file models.py.

**6.11.3.9** `Ecodena.Attempt.models.ErrorReport.testCaseLevel_f`

Definition at line 54 of file models.py.

**6.11.3.10** `tuple Ecodena.Attempt.models.ErrorReport.testCaseLevel_f  
= models.CharField(max_length=40, choices=TestCase.CASE_TYPE,  
verbose_name="Test Case Level where error occurred (Select High if solution is  
correct)") [static]`

Definition at line 21 of file models.py.

**6.11.3.11** `Ecodena.Attempt.models.ErrorReport.timeRequirement_f`

Definition at line 26 of file models.py.

**6.11.3.12** `tuple Ecodena.Attempt.models.ErrorReport.timeRequirement_f =  
models.FloatField("Time Required for compiling") [static]`

Definition at line 16 of file models.py.



### 6.11.4 Property Documentation

6.11.4.1 Ecodena.Attempt.models.ErrorReport.errorMessage =  
property(getErrorMessage,setErrorMessage) [static]

Definition at line 49 of file models.py.

6.11.4.2 Ecodena.Attempt.models.ErrorReport.errorType = property(getErrorType,setErrorType)  
[static]

Definition at line 42 of file models.py.

6.11.4.3 Ecodena.Attempt.models.ErrorReport.memory = property(getMemory,setMemory)  
[static]

Definition at line 35 of file models.py.

6.11.4.4 Ecodena.Attempt.models.ErrorReport.testCaseLevel =  
property(getTestCaseLevel,setTestCaseLevel) [static]

Definition at line 56 of file models.py.

6.11.4.5 Ecodena.Attempt.models.ErrorReport.timeRequirement =  
property(getTimeRequirement,setTimeRequirement) [static]

Definition at line 28 of file models.py.

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

- /home/panx/ecodena/src/Ecodena/Attempt/[models.py](#)

## 6.12 Ecodena.Attempt.admin.ErrorReportAdmin Class Reference

### Static Public Attributes

- tuple [list\\_display](#) = ('errorReportID\_f', 'timeRequirement\_f', 'memory\_f', 'errorType\_f', 'errorMessage\_f', 'testCaseLevel\_f')

### 6.12.1 Detailed Description

Definition at line 10 of file admin.py.

### 6.12.2 Member Data Documentation

6.12.2.1 `tuple Ecodena.Attempt.admin.ErrorReportAdmin.list_display = ('errorReportID_f', 'timeRequirement_f', 'memory_f', 'errorType_f', 'errorMessage_f', 'testCaseLevel_f') [static]`

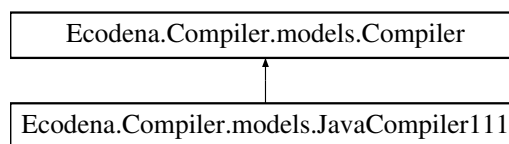
Definition at line 11 of file `admin.py`.

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

- `/home/panx/ecodena/src/Ecodena/Attempt/admin.py`

## 6.13 Ecodena.Compiler.models.JavaCompiler111 Class Reference

Inheritance diagram for `Ecodena.Compiler.models.JavaCompiler111`:



### Public Member Functions

- `def compile`

#### 6.13.1 Detailed Description

Definition at line 104 of file `models.py`.

#### 6.13.2 Member Function Documentation

6.13.2.1 `def Ecodena.Compiler.models.JavaCompiler111.compile ( self )`

Reimplemented from `Ecodena.Compiler.models.Compiler`.

Definition at line 105 of file `models.py`.

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

- `/home/panx/ecodena/src/Ecodena/Compiler/models.py`

## 6.14 Ecodena.Compiler.models.Language Class Reference

## Classes

- class [Meta](#)

## Public Member Functions

- def [getLanguageName](#)
- def [setLanguageName](#)
- def [\\_\\_unicode\\_\\_](#)

## Public Attributes

- [languageName\\_f](#)

## Static Public Attributes

- tuple [languageID\\_f](#) = models.AutoField(primary\_key=True)
- tuple [languageName\\_f](#) = models.CharField("Language Name", max\_length=40, null=False, unique=True)

## Properties

- [languageName](#) = property(getLanguageName, setLanguageName)

### 6.14.1 Detailed Description

Definition at line 5 of file models.py.

### 6.14.2 Member Function Documentation

6.14.2.1 def Ecodena.Compiler.models.Language.\_\_unicode\_\_( self )

Definition at line 22 of file models.py.

6.14.2.2 def Ecodena.Compiler.models.Language.getLanguageName ( self )

Definition at line 10 of file models.py.

6.14.2.3 def Ecodena.Compiler.models.Language.setLanguageName ( self, languageName )

Definition at line 13 of file models.py.

### 6.14.3 Member Data Documentation

6.14.3.1 `tuple Ecodena.Compiler.models.Language.languageID_f =  
models.AutoField(primary_key=True) [static]`

Definition at line 7 of file models.py.

6.14.3.2 `Ecodena.Compiler.models.Language.languageName_f`

Definition at line 13 of file models.py.

6.14.3.3 `tuple Ecodena.Compiler.models.Language.languageName_f =  
models.CharField("Language Name", max_length=40, null=False, unique=True)  
[static]`

Definition at line 8 of file models.py.

### 6.14.4 Property Documentation

6.14.4.1 `Ecodena.Compiler.models.Language.languageName = property(getLanguageName,  
setLanguageName) [static]`

Definition at line 16 of file models.py.

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

- [/home/panx/ecodena/src/Ecodena/Compiler/models.py](#)

## 6.15 Ecodena.Compiler.admin.LanguageAdmin Class Reference

### Static Public Attributes

- list `list_display` = `['languageName_f']`

### 6.15.1 Detailed Description

Definition at line 4 of file admin.py.

### 6.15.2 Member Data Documentation

6.15.2.1 `list Ecodena.Compiler.admin.LanguageAdmin.list_display =  
['languageName_f'] [static]`

Definition at line 5 of file admin.py.

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

- [/home/panx/ecodena/src/Ecodena/Compiler/admin.py](#)

## 6.16 Ecodena.Question.models.Level Class Reference

### Classes

- class [Meta](#)

### Public Member Functions

- def [getLevelName](#)
- def [setLevelName](#)
- def [\\_\\_unicode\\_\\_](#)

### Public Attributes

- [levelName\\_f](#)

### Static Public Attributes

- tuple [levelID\\_f](#) = `models.AutoField(primary_key=True)`
- tuple [levelName\\_f](#) = `models.CharField("Level Name", max_length = 20 , null = False, unique=True)`

### Properties

- [levelName](#) = `property(getLevelName, setLevelName)`

#### 6.16.1 Detailed Description

Creates the entity Level which is a subclass of `models.Model` class

Definition at line 8 of file `models.py`.

#### 6.16.2 Member Function Documentation

##### 6.16.2.1 `def Ecodena.Question.models.Level.__unicode__( self )`

Definition at line 26 of file `models.py`.

6.16.2.2 `def Ecodena.Question.models.Level.getLevelName ( self )`

Definition at line 14 of file models.py.

6.16.2.3 `def Ecodena.Question.models.Level.setLevelName ( self, lname )`

Definition at line 17 of file models.py.

### 6.16.3 Member Data Documentation

6.16.3.1 `tuple Ecodena.Question.models.Level.levelID_f =  
models.AutoField(primary_key=True) [static]`

Definition at line 11 of file models.py.

6.16.3.2 `Ecodena.Question.models.Level.levelName_f`

Definition at line 17 of file models.py.

6.16.3.3 `tuple Ecodena.Question.models.Level.levelName_f = models.CharField("Level  
Name", max_length = 20 , null = False, unique=True) [static]`

Definition at line 12 of file models.py.

### 6.16.4 Property Documentation

6.16.4.1 `Ecodena.Question.models.Level.levelName = property(getLevelName, setLevelName)  
[static]`

Definition at line 20 of file models.py.

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

- [/home/panx/ecodena/src/Ecodena/Question/models.py](#)

## 6.17 Ecodena.Question.admin.LevelInline Class Reference

### Static Public Attributes

- `model` = `Level`
- `int extra` = `2`

### 6.17.1 Detailed Description

Definition at line 10 of file admin.py.

### 6.17.2 Member Data Documentation

6.17.2.1 `int Ecodena.Question.admin.LevelInline.extra = 2` `[static]`

Definition at line 12 of file admin.py.

6.17.2.2 `Ecodena.Question.admin.LevelInline.model = Level` `[static]`

Definition at line 11 of file admin.py.

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

- [/home/panx/ecodena/src/Ecodena/Question/admin.py](#)

## 6.18 Ecodena.Question.models.Level.Meta Class Reference

### Static Public Attributes

- string `verbose_name` = 'level'
- string `verbose_name_plural` = 'levels'

### 6.18.1 Detailed Description

Definition at line 22 of file models.py.

### 6.18.2 Member Data Documentation

6.18.2.1 `string Ecodena.Question.models.Level.Meta.verbose_name = 'level'`  
`[static]`

Definition at line 23 of file models.py.

6.18.2.2 `string Ecodena.Question.models.Level.Meta.verbose_name_plural = 'levels'` `[static]`

Definition at line 24 of file models.py.

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

- [/home/panx/ecodena/src/Ecodena/Question/models.py](#)

## 6.19 Ecodena.Compiler.models.Language.Meta Class Reference

### Static Public Attributes

- string `verbose_name` = 'language'
- string `verbose_name_plural` = 'languages'

#### 6.19.1 Detailed Description

Definition at line 18 of file `models.py`.

#### 6.19.2 Member Data Documentation

6.19.2.1 string `Ecodena.Compiler.models.Language.Meta.verbose_name` = 'language' `[static]`

Definition at line 19 of file `models.py`.

6.19.2.2 string `Ecodena.Compiler.models.Language.Meta.verbose_name_plural` = 'languages' `[static]`

Definition at line 20 of file `models.py`.

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

- `/home/panx/ecodena/src/Ecodena/Compiler/models.py`

## 6.20 Ecodena.Question.models.Type.Meta Class Reference

### Static Public Attributes

- string `verbose_name` = 'type'
- string `verbose_name_plural` = 'types'

#### 6.20.1 Detailed Description

Definition at line 36 of file `models.py`.

#### 6.20.2 Member Data Documentation

6.20.2.1 string `Ecodena.Question.models.Type.Meta.verbose_name` = 'type' `[static]`

Definition at line 37 of file `models.py`.



6.20.2.2 string Ecodena.Question.models.Type.Meta.verbose\_name\_plural = 'types'  
[static]

Definition at line 38 of file models.py.

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

- /home/panx/ecodena/src/Ecodena/Question/[models.py](#)

## 6.21 Ecodena.Compiler.models.CompilerVersion.Meta Class Reference

### Static Public Attributes

- string [verbose\\_name](#) = 'compiler version'
- string [verbose\\_name\\_plural](#) = 'compiler versions'

#### 6.21.1 Detailed Description

Definition at line 52 of file models.py.

#### 6.21.2 Member Data Documentation

6.21.2.1 string Ecodena.Compiler.models.CompilerVersion.Meta.verbose\_name =  
'compiler version' [static]

Definition at line 53 of file models.py.

6.21.2.2 string Ecodena.Compiler.models.CompilerVersion.Meta.verbose\_name\_  
plural = 'compiler versions' [static]

Definition at line 54 of file models.py.

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

- /home/panx/ecodena/src/Ecodena/Compiler/[models.py](#)

## 6.22 Ecodena.Question.models.Question.Meta Class Reference

### Static Public Attributes

- string [verbose\\_name](#) = 'question'
- string [verbose\\_name\\_plural](#) = 'questions'

### 6.22.1 Detailed Description

Definition at line 95 of file models.py.

### 6.22.2 Member Data Documentation

6.22.2.1 `string Ecodena.Question.models.Question.Meta.verbose_name = 'question'`  
`[static]`

Definition at line 96 of file models.py.

6.22.2.2 `string Ecodena.Question.models.Question.Meta.verbose_name_plural = 'questions'` `[static]`

Definition at line 97 of file models.py.

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

- </home/panx/ecodena/src/Ecodena/Question/models.py>

## 6.23 Ecodena.Attempt.models.ErrorReport.Meta Class Reference

### Static Public Attributes

- string `verbose_name` = 'error report'
- string `verbose_name_plural` = 'error reports'

### 6.23.1 Detailed Description

Definition at line 58 of file models.py.

### 6.23.2 Member Data Documentation

6.23.2.1 `string Ecodena.Attempt.models.ErrorReport.Meta.verbose_name = 'error report'` `[static]`

Definition at line 59 of file models.py.

6.23.2.2 `string Ecodena.Attempt.models.ErrorReport.Meta.verbose_name_plural = 'error reports'` `[static]`

Definition at line 60 of file models.py.

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

- [/home/panx/ecodena/src/Ecodena/Attempt/models.py](#)

## 6.24 Ecodena.Question.models.TestCase.Meta Class Reference

### Static Public Attributes

- string `verbose_name` = 'test case'
- string `verbose_name_plural` = 'test cases'

#### 6.24.1 Detailed Description

Definition at line 142 of file models.py.

#### 6.24.2 Member Data Documentation

6.24.2.1 string `Ecodena.Question.models.TestCase.Meta.verbose_name` = 'test case' `[static]`

Definition at line 143 of file models.py.

6.24.2.2 string `Ecodena.Question.models.TestCase.Meta.verbose_name_plural` = 'test cases' `[static]`

Definition at line 144 of file models.py.

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

- [/home/panx/ecodena/src/Ecodena/Question/models.py](#)

## 6.25 Ecodena.Comment.models.Comment.Meta Class Reference

### Static Public Attributes

- string `verbose_name` = 'comment'
- string `verbose_name_plural` = 'comments'

#### 6.25.1 Detailed Description

Definition at line 71 of file models.py.

### 6.25.2 Member Data Documentation

6.25.2.1 `string Ecodena.Comment.models.Comment.Meta.verbose_name = 'comment' [static]`

Definition at line 72 of file models.py.

6.25.2.2 `string Ecodena.Comment.models.Comment.Meta.verbose_name_plural = 'comments' [static]`

Definition at line 73 of file models.py.

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

- [/home/panx/ecodena/src/Ecodena/Comment/models.py](#)

## 6.26 Ecodena.Attempt.models.Attempt.Meta Class Reference

### Static Public Attributes

- string `verbose_name` = 'attempt'
- string `verbose_name_plural` = 'attempts'

### 6.26.1 Detailed Description

Definition at line 133 of file models.py.

### 6.26.2 Member Data Documentation

6.26.2.1 `string Ecodena.Attempt.models.Attempt.Meta.verbose_name = 'attempt' [static]`

Definition at line 134 of file models.py.

6.26.2.2 `string Ecodena.Attempt.models.Attempt.Meta.verbose_name_plural = 'attempts' [static]`

Definition at line 135 of file models.py.

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

- [/home/panx/ecodena/src/Ecodena/Attempt/models.py](#)

## 6.27 Ecodena.Query.models.Query.Meta Class Reference

### Static Public Attributes

- string `verbose_name` = 'query'
- string `verbose_name_plural` = 'queries'

#### 6.27.1 Detailed Description

Definition at line 73 of file models.py.

#### 6.27.2 Member Data Documentation

6.27.2.1 string `Ecodena.Query.models.Query.Meta.verbose_name` = 'query'  
[static]

Definition at line 74 of file models.py.

6.27.2.2 string `Ecodena.Query.models.Query.Meta.verbose_name_plural` = 'queries'  
[static]

Definition at line 75 of file models.py.

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

- `/home/panx/ecodena/src/Ecodena/Query/models.py`

## 6.28 Ecodena.User.models.Profile Class Reference

### Public Member Functions

- def `getDOB`
- def `setDOB`
- def `getAddress`
- def `setAddress`
- def `getGender`
- def `setGender`
- def `getUserID`
- def `isProgrammer`
- def `isAdmin`
- def `getNumberOfAttempts`
- def `setNumberOfAttempts`

## Public Attributes

- [dob\\_f](#)
- [address\\_f](#)
- [gender\\_f](#)
- [numberOfAttempts\\_f](#)

## Static Public Attributes

- tuple [profileID\\_f](#) = models.AutoField(primary\_key=True)
- tuple [dob\\_f](#) = models.DateField("User Birth Date")
- tuple [address\\_f](#) = models.TextField("User Address")
- tuple [GENDER\\_TYPE](#) = ( (0, 'Male'), (1, 'Female'))
- tuple [gender\\_f](#) = models.SmallIntegerField(choices=[GENDER\\_TYPE](#), verbose\_name="Gender")
- tuple [userID\\_f](#) = models.ForeignKey(User, verbose\_name="User ID of user whose profile is this", null=False, unique=True)
- tuple [isProgrammer\\_f](#) = models.BooleanField("Is a [Programmer](#)", default=True)
- tuple [isAdmin\\_f](#) = models.BooleanField("Is Administrator", default=False)
- tuple [numberOfAttempts\\_f](#) = models.IntegerField("Number of Attempts")

## Properties

- [dob](#) = property(getDOB,setDOB)
- [address](#) = property(getAddress,setAddress)
- [gender](#) = property(getGender, setGender)
- [numberOfAttempts](#) = property(getNumberOfAttempts,setNumberOfAttempts)

### 6.28.1 Detailed Description

Here Profile is a subclass of models.Model and creates an entity named Profile.  
Basically it represents the profiles of users and admins of Ecodena.

Definition at line 129 of file models.py.

### 6.28.2 Member Function Documentation

#### 6.28.2.1 def Ecodena.User.models.Profile.getAddress ( )

Definition at line 150 of file models.py.

#### 6.28.2.2 def Ecodena.User.models.Profile.getDOB ( )

Definition at line 143 of file models.py.

**6.28.2.3** `def Ecodena.User.models.Profile.getGender ( )`

Definition at line 157 of file models.py.

**6.28.2.4** `def Ecodena.User.models.Profile.getNumberOfAttempts ( self )`

Definition at line 173 of file models.py.

**6.28.2.5** `def Ecodena.User.models.Profile.getUserID ( )`

Definition at line 164 of file models.py.

**6.28.2.6** `def Ecodena.User.models.Profile.isAdmin ( )`

Definition at line 170 of file models.py.

**6.28.2.7** `def Ecodena.User.models.Profile.isProgrammer ( )`

Definition at line 167 of file models.py.

**6.28.2.8** `def Ecodena.User.models.Profile.setAddress ( address )`

Definition at line 152 of file models.py.

**6.28.2.9** `def Ecodena.User.models.Profile.setDOB ( date )`

Definition at line 145 of file models.py.

**6.28.2.10** `def Ecodena.User.models.Profile.setGender ( )`

Definition at line 159 of file models.py.

**6.28.2.11** `def Ecodena.User.models.Profile.setNumberOfAttempts ( self, attempt )`

Definition at line 175 of file models.py.

**6.28.3 Member Data Documentation****6.28.3.1** `tuple Ecodena.User.models.Profile.address_f = models.TextField("User Address") [static]`

Definition at line 135 of file models.py.

### 6.28.3.2 **Ecodena.User.models.Profile.address\_f**

Definition at line 152 of file models.py.

### 6.28.3.3 **Ecodena.User.models.Profile.dob\_f**

Definition at line 145 of file models.py.

**6.28.3.4** `tuple Ecodena.User.models.Profile.dob_f = models.DateField("User Birth Date")`  
`[static]`

Definition at line 134 of file models.py.

**6.28.3.5** `tuple Ecodena.User.models.Profile.gender_f =`  
`models.SmallIntegerField(choices=GENDER_TYPE, verbose_name="Gender")`  
`[static]`

Definition at line 137 of file models.py.

### 6.28.3.6 **Ecodena.User.models.Profile.gender\_f**

Definition at line 159 of file models.py.

**6.28.3.7** `tuple Ecodena.User.models.Profile.GENDER_TYPE = ( (0, 'Male'), (1, 'Female'))`  
`[static]`

Definition at line 136 of file models.py.

**6.28.3.8** `tuple Ecodena.User.models.Profile.isAdmin_f = models.BooleanField("Is`  
`Administrator", default=False) [static]`

Definition at line 140 of file models.py.

**6.28.3.9** `tuple Ecodena.User.models.Profile.isProgrammer_f = models.BooleanField("Is`  
`a Programmer", default=True) [static]`

Definition at line 139 of file models.py.

**6.28.3.10** `tuple Ecodena.User.models.Profile.numberOfAttempts_f =`  
`models.IntegerField("Number of Attempts") [static]`

Definition at line 141 of file models.py.



#### 6.28.3.11 Ecodena.User.models.Profile.numberOfAttempts\_f

Definition at line 175 of file models.py.

#### 6.28.3.12 tuple Ecodena.User.models.Profile.profileID\_f = models.AutoField(primary\_key=True) [static]

Definition at line 132 of file models.py.

#### 6.28.3.13 tuple Ecodena.User.models.Profile.userID\_f = models.ForeignKey(User, verbose\_name="User ID of user whose profile is this", null=False, unique=True) [static]

Definition at line 138 of file models.py.

### 6.28.4 Property Documentation

#### 6.28.4.1 Ecodena.User.models.Profile.address = property(getAddress, setAddress) [static]

Definition at line 155 of file models.py.

#### 6.28.4.2 Ecodena.User.models.Profile.dob = property(getDOB, setDOB) [static]

Definition at line 148 of file models.py.

#### 6.28.4.3 Ecodena.User.models.Profile.gender = property(getGender, setGender) [static]

Definition at line 162 of file models.py.

#### 6.28.4.4 Ecodena.User.models.Profile.numberOfAttempts = property(getNumberOfAttempts, setNumberOfAttempts) [static]

Definition at line 178 of file models.py.

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

- /home/panx/ecodena/src/Ecodena/User/[models.py](#)

## 6.29 Ecodena.User.admin.ProfileAdmin Class Reference

### Static Public Attributes

- tuple `list_display` = ('profileID\_f', 'dob\_f', 'address\_f', 'gender\_f', 'userID\_f', 'isProgrammer\_f', 'isAdmin\_f', 'numberOfAttempts\_f')

#### 6.29.1 Detailed Description

Definition at line 9 of file admin.py.

#### 6.29.2 Member Data Documentation

6.29.2.1 tuple `Ecodena.User.admin.ProfileAdmin.list_display` = ('profileID\_f', 'dob\_f', 'address\_f', 'gender\_f', 'userID\_f', 'isProgrammer\_f', 'isAdmin\_f', 'numberOfAttempts\_f')  
[static]

Definition at line 10 of file admin.py.

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

- /home/panx/ecodena/src/Ecodena/User/admin.py

### 6.30 Ecodena.User.views.ProfileForm Class Reference

#### Static Public Attributes

- tuple `username` = forms.CharField()
- tuple `dob` = forms.DateField()
- tuple `GENDER_CHOICES`
- tuple `gender` = forms.ChoiceField(choices=`GENDER_CHOICES`)
- tuple `email` = forms.EmailField()
- tuple `address` = forms.CharField(widget=forms.Textarea)

#### 6.30.1 Detailed Description

Definition at line 12 of file views.py.

#### 6.30.2 Member Data Documentation

6.30.2.1 tuple `Ecodena.User.views.ProfileForm.address` =  
`forms.CharField(widget=forms.Textarea)` [static]

Definition at line 21 of file views.py.

6.30.2.2 tuple Ecodena.User.views.ProfileForm.dob = forms.DateField() [static]

Definition at line 14 of file views.py.

6.30.2.3 tuple Ecodena.User.views.ProfileForm.email = forms.EmailField()  
[static]

Definition at line 20 of file views.py.

6.30.2.4 tuple Ecodena.User.views.ProfileForm.gender =  
forms.ChoiceField(choices=GENDER\_CHOICES) [static]

Definition at line 19 of file views.py.

6.30.2.5 tuple Ecodena.User.views.ProfileForm.GENDER\_CHOICES [static]

**Initial value:**

```
(  
    ('M', 'Male'),  
    ('F', 'Female'),  
)
```

Definition at line 15 of file views.py.

6.30.2.6 tuple Ecodena.User.views.ProfileForm.username = forms.CharField()  
[static]

Definition at line 13 of file views.py.

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

- /home/panx/ecodena/src/Ecodena/User/[views.py](#)

## 6.31 Ecodena.User.models.Programmer Class Reference

Inherits User.

### Public Member Functions

- def [getLevel](#)
- def [setLevel](#)

## Public Attributes

- [level\\_f](#)

## Static Public Attributes

- tuple [level\\_f](#) = models.CharField("Level of User", max\_length=20, null=False)

## Properties

- [level](#) = property(getLevel,setLevel)

### 6.31.1 Detailed Description

Here Programmer is a subclass of object and creates an entity named Programmer.

Basically it represents the users of ECODENA. They can submit their solutions of various parameters: User --> its a class that helps a user to log-in to Ecodena.

Definition at line 116 of file models.py.

### 6.31.2 Member Function Documentation

#### 6.31.2.1 def Ecodena.User.models.Programmer.getLevel ( )

Definition at line 122 of file models.py.

#### 6.31.2.2 def Ecodena.User.models.Programmer.setLevel ( *level* )

Definition at line 124 of file models.py.

### 6.31.3 Member Data Documentation

#### 6.31.3.1 tuple Ecodena.User.models.Programmer.level\_f = models.CharField("Level of User", max\_length=20, null=False) `[static]`

Definition at line 120 of file models.py.

#### 6.31.3.2 Ecodena.User.models.Programmer.level\_f

Definition at line 124 of file models.py.

### 6.31.4 Property Documentation

6.31.4.1 Ecodena.User.models.Programmer.level = property(getLevel,setLevel) [static]

Definition at line 126 of file models.py.

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

- /home/panx/ecodena/src/Ecodena/User/models.py

## 6.32 Ecodena.Query.models.Query Class Reference

### Classes

- class [Meta](#)

### Public Member Functions

- def [getQueryText](#)
- def [setQueryText](#)
- def [getQueryID](#)
- def [setQueryID](#)
- def [getSolution](#)
- def [setSolution](#)
- def [getQueryTime](#)
- def [setQueryTime](#)
- def [getReplyTime](#)
- def [setReplyTime](#)
- def [getUserID](#)
- def [setUserID](#)
- def [getAdminID](#)
- def [setAdminID](#)
- def [\\_\\_unicode\\_\\_](#)

### Public Attributes

- [queryText\\_f](#)
- [queryID\\_f](#)
- [solution\\_f](#)
- [queryTime\\_f](#)
- [replyTime\\_f](#)
- [userID\\_f](#)
- [adminID\\_f](#)

### Static Public Attributes

- tuple `queryID_f` = `models.AutoField(primary_key=True)`
- tuple `queryText_f` = `models.TextField("The Text explaining the Query", null=False)`
- tuple `solution_f` = `models.TextField("The Solution of the Query", blank=True)`
- tuple `userID_f` = `models.ForeignKey(User, verbose_name="User ID of user posting the query", null=False, related_name='+')`
- tuple `adminID_f` = `models.ForeignKey(User, verbose_name="User ID of admin resolving the query", related_name='+', blank=True, null=True)`
- tuple `queryTime_f` = `models.DateTimeField("Time of posting Query", null=False)`
- tuple `replyTime_f` = `models.DateTimeField("Time of resolving Query", blank=True, null=True)`

### Properties

- `queryText` = `property(getQueryText, setQueryText)`
- `queryID` = `property(getQueryID, setQueryID)`
- `solution` = `property(getSolution, setSolution)`
- `queryTime` = `property(getQueryTime, setQueryTime)`
- `replyTime` = `property(getReplyTime, setReplyTime)`
- `userID` = `property(getUserID, setUserID)`
- `adminID` = `property(getAdminID, setAdminID)`

### 6.32.1 Detailed Description

Here `Query` is subclass of `models.Model` and Creates an entity named `Query`.

Definition at line 8 of file `models.py`.

### 6.32.2 Member Function Documentation

#### 6.32.2.1 `def Ecodena.Query.models.Query.__unicode__( self )`

Definition at line 77 of file `models.py`.

#### 6.32.2.2 `def Ecodena.Query.models.Query.getAdminID( self )`

Definition at line 66 of file `models.py`.

#### 6.32.2.3 `def Ecodena.Query.models.Query.getQueryID( self )`

Definition at line 26 of file `models.py`.

**6.32.2.4 def Ecodena.Query.models.Query.getQueryText ( self )**

Definition at line 19 of file models.py.

**6.32.2.5 def Ecodena.Query.models.Query.getQueryTime ( self )**

Definition at line 40 of file models.py.

**6.32.2.6 def Ecodena.Query.models.Query.getReplyTime ( self )**

Definition at line 51 of file models.py.

**6.32.2.7 def Ecodena.Query.models.Query.getSolution ( self )**

Definition at line 33 of file models.py.

**6.32.2.8 def Ecodena.Query.models.Query.getUserID ( self )**

Definition at line 59 of file models.py.

**6.32.2.9 def Ecodena.Query.models.Query.setAdminID ( self, aID )**

Definition at line 68 of file models.py.

**6.32.2.10 def Ecodena.Query.models.Query.setQueryID ( self, qID )**

Definition at line 28 of file models.py.

**6.32.2.11 def Ecodena.Query.models.Query.setQueryText ( self, text )**

Definition at line 21 of file models.py.

**6.32.2.12 def Ecodena.Query.models.Query.setQueryTime ( self, Timestamp )**

Definition at line 42 of file models.py.

**6.32.2.13 def Ecodena.Query.models.Query.setReplyTime ( self, Timestamp )**

Definition at line 53 of file models.py.

**6.32.2.14** `def Ecodena.Query.models.Query.setSolution ( self, code )`

Definition at line 35 of file models.py.

**6.32.2.15** `def Ecodena.Query.models.Query.setUserID ( self, uid )`

Definition at line 61 of file models.py.

### 6.32.3 Member Data Documentation

**6.32.3.1** `tuple Ecodena.Query.models.Query.adminID_f = models.ForeignKey(User, verbose_name="User ID of admin resolving the query", related_name='+', blank=True, null=True) [static]`

Definition at line 14 of file models.py.

**6.32.3.2** `Ecodena.Query.models.Query.adminID_f`

Definition at line 68 of file models.py.

**6.32.3.3** `tuple Ecodena.Query.models.Query.queryID_f = models.AutoField(primary_key=True) [static]`

Definition at line 10 of file models.py.

**6.32.3.4** `Ecodena.Query.models.Query.queryID_f`

Definition at line 28 of file models.py.

**6.32.3.5** `Ecodena.Query.models.Query.queryText_f`

Definition at line 21 of file models.py.

**6.32.3.6** `tuple Ecodena.Query.models.Query.queryText_f = models.TextField("The Text explaining the Query", null=False) [static]`

Definition at line 11 of file models.py.

**6.32.3.7** `Ecodena.Query.models.Query.queryTime_f`

Definition at line 42 of file models.py.



**6.32.3.8** tuple `Ecodena.Query.models.Query.queryTime_f = models.DateTimeField("Time of posting Query", null=False)` `[static]`

Definition at line 15 of file models.py.

**6.32.3.9** tuple `Ecodena.Query.models.Query.replyTime_f = models.DateTimeField("Time of resolving Query", blank=True, null=True)` `[static]`

Definition at line 16 of file models.py.

**6.32.3.10** `Ecodena.Query.models.Query.replyTime_f`

Definition at line 53 of file models.py.

**6.32.3.11** `Ecodena.Query.models.Query.solution_f`

Definition at line 35 of file models.py.

**6.32.3.12** tuple `Ecodena.Query.models.Query.solution_f = models.TextField("The Solution of the Query", blank=True)` `[static]`

Definition at line 12 of file models.py.

**6.32.3.13** tuple `Ecodena.Query.models.Query.userID_f = models.ForeignKey(User, verbose_name="User ID of user posting the query", null=False, related_name='+')` `[static]`

Definition at line 13 of file models.py.

**6.32.3.14** `Ecodena.Query.models.Query.userID_f`

Definition at line 61 of file models.py.

## 6.32.4 Property Documentation

**6.32.4.1** `Ecodena.Query.models.Query.adminID = property(getAdminID, setAdminID)` `[static]`

Definition at line 70 of file models.py.

**6.32.4.2** `Ecodena.Query.models.Query.queryID = property(getQueryID, setQueryID)`  
`[static]`

Definition at line 30 of file models.py.

**6.32.4.3** `Ecodena.Query.models.Query.queryText = property(getQueryText, setQueryText)`  
`[static]`

Definition at line 23 of file models.py.

**6.32.4.4** `Ecodena.Query.models.Query.queryTime = property(getQueryTime, setQueryTime)`  
`[static]`

Definition at line 47 of file models.py.

**6.32.4.5** `Ecodena.Query.models.Query.replyTime = property(getReplyTime, setReplyTime)`  
`[static]`

Definition at line 56 of file models.py.

**6.32.4.6** `Ecodena.Query.models.Query.solution = property(getSolution, setSolution)`  
`[static]`

Definition at line 37 of file models.py.

**6.32.4.7** `Ecodena.Query.models.Query.userID = property(getUserID, setUserID)` `[static]`

Definition at line 63 of file models.py.

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

- `/home/panx/ecodena/src/Ecodena/Query/models.py`

## 6.33 Ecodena.Query.admin.QueryAdmin Class Reference

### Static Public Attributes

- tuple `list_display` = ( 'queryID', 'queryText', 'userID', 'queryTime', 'solution', 'adminID', 'replyTime')

### 6.33.1 Detailed Description

Definition at line 4 of file admin.py.

### 6.33.2 Member Data Documentation

6.33.2.1 tuple `Ecodena.Query.admin.QueryAdmin.list_display` = ( 'queryID', 'queryText', 'userID', 'queryTime', 'solution', 'adminID', 'replyTime') `[static]`

Definition at line 5 of file `admin.py`.

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

- `/home/panx/ecodena/src/Ecodena/Query/admin.py`

## 6.34 Ecodena.Query.views.QueryForm Class Reference

### Static Public Attributes

- tuple `text` = `forms.CharField(widget=forms.Textarea)`

### 6.34.1 Detailed Description

Its a form which user have to fill in order to post a query

Definition at line 10 of file `views.py`.

### 6.34.2 Member Data Documentation

6.34.2.1 tuple `Ecodena.Query.views.QueryForm.text` = `forms.CharField(widget=forms.Textarea)` `[static]`

Definition at line 12 of file `views.py`.

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

- `/home/panx/ecodena/src/Ecodena/Query/views.py`

## 6.35 Ecodena.Question.models.Question Class Reference

### Classes

- class `Meta`

### Public Member Functions

- def `getQuestionID`
- def `setQuestionID`

- def [getQuestionText](#)
- def [setQuestionText](#)
- def [getQuestionTitle](#)
- def [setQuestionTitle](#)
- def [getComments](#)
- def [addComments](#)
- def [setCommentList](#)
- def [getTestCasesList](#)
- def [setTestCasesList](#)
- def [removeTestCase](#)
- def [\\_\\_unicode\\_\\_](#)

### Public Attributes

- [questionID\\_f](#)
- [questionText\\_f](#)
- [questionTitle\\_f](#)
- [commentList\\_f](#)
- [testCasesList\\_f](#)

### Static Public Attributes

- tuple [questionID\\_f](#) = models.AutoField(primary\_key=True)
- tuple [questionText\\_f](#) = models.TextField("Problem Statement", null=False)
- tuple [questionTitle\\_f](#) = models.CharField("Problem Title", max\_length = 50, null = False)
- tuple [level\\_f](#) = models.ForeignKey([Level](#), verbose\_name="Question [Level](#)", null=False)
- tuple [type\\_f](#) = models.ForeignKey([Type](#), verbose\_name="Type of [Question](#)", null=False)
- tuple [timeLimit\\_f](#) = models.FloatField("Time Limit for [Question](#)",null=False)
- list [commentList\\_f](#) = []
- list [testCasesList\\_f](#) = []

### Properties

- [questionID](#) = property(getQuestionID,setQuestionID)
- [questionText](#) = property(getQuestionText,setQuestionText)
- [questionTitle](#) = property(getQuestionTitle, setQuestionTitle)
- [commentList](#) = property(getComments,setCommentList)
- [testCasesList](#) = property(getTestCasesList,setTestCasesList)

## 6.35.1 Detailed Description

Creates the entity question  
which is a subclass of models.Model

Definition at line 44 of file models.py.

### 6.35.2 Member Function Documentation

6.35.2.1 `def Ecodena.Question.models.Question.__unicode__( self )`

Definition at line 99 of file models.py.

6.35.2.2 `def Ecodena.Question.models.Question.addComments ( self, comment )`

Definition at line 78 of file models.py.

6.35.2.3 `def Ecodena.Question.models.Question.getComments ( self )`

Definition at line 76 of file models.py.

6.35.2.4 `def Ecodena.Question.models.Question.getQuestionID ( self )`

Definition at line 56 of file models.py.

6.35.2.5 `def Ecodena.Question.models.Question.getQuestionText ( self )`

Definition at line 63 of file models.py.

6.35.2.6 `def Ecodena.Question.models.Question.getQuestionTitle ( self )`

Definition at line 70 of file models.py.

6.35.2.7 `def Ecodena.Question.models.Question.getTestCasesList ( self )`

Definition at line 86 of file models.py.

6.35.2.8 `def Ecodena.Question.models.Question.removeTestCase ( self, caseID )`

Definition at line 90 of file models.py.

6.35.2.9 `def Ecodena.Question.models.Question.setCommentList ( self, commentList )`

Definition at line 81 of file models.py.

6.35.2.10 `def Ecodena.Question.models.Question.setQuestionID ( self, qID )`

Definition at line 58 of file models.py.

6.35.2.11 `def Ecodena.Question.models.Question.setQuestionText ( self, qtext )`

Definition at line 65 of file models.py.

6.35.2.12 `def Ecodena.Question.models.Question.setQuestionTitle ( self, title )`

Definition at line 72 of file models.py.

6.35.2.13 `def Ecodena.Question.models.Question.setTestCasesList ( self, testCasesList )`

Definition at line 88 of file models.py.

### 6.35.3 Member Data Documentation

6.35.3.1 `list Ecodena.Question.models.Question.commentList_f = [] [static]`

Definition at line 53 of file models.py.

6.35.3.2 `Ecodena.Question.models.Question.commentList_f`

Definition at line 81 of file models.py.

6.35.3.3 `tuple Ecodena.Question.models.Question.level_f = models.ForeignKey(Level, verbose_name="Question Level", null=False) [static]`

Definition at line 50 of file models.py.

6.35.3.4 `tuple Ecodena.Question.models.Question.questionID_f = models.AutoField(primary_key=True) [static]`

Definition at line 47 of file models.py.

6.35.3.5 `Ecodena.Question.models.Question.questionID_f`

Definition at line 58 of file models.py.

6.35.3.6 `tuple Ecodena.Question.models.Question.questionText_f = models.TextField("Problem Statement", null=False) [static]`

Definition at line 48 of file models.py.

**6.35.3.7 Ecodena.Question.models.Question.questionText\_f**

Definition at line 65 of file models.py.

**6.35.3.8 Ecodena.Question.models.Question.questionTitle\_f**

Definition at line 72 of file models.py.

```
6.35.3.9 tuple Ecodena.Question.models.Question.questionTitle_f =  
models.CharField("Problem Title", max_length = 50, null = False) [static]
```

Definition at line 49 of file models.py.

```
6.35.3.10 list Ecodena.Question.models.Question.testCasesList_f = []  
[static]
```

Definition at line 54 of file models.py.

**6.35.3.11 Ecodena.Question.models.Question.testCasesList\_f**

Definition at line 88 of file models.py.

```
6.35.3.12 tuple Ecodena.Question.models.Question.timeLimit_f =  
models.FloatField("Time Limit for Question",null=False) [static]
```

Definition at line 52 of file models.py.

```
6.35.3.13 tuple Ecodena.Question.models.Question.type_f = models.ForeignKey(Type,  
verbose_name="Type of Question", null=False) [static]
```

Definition at line 51 of file models.py.

**6.35.4 Property Documentation**

```
6.35.4.1 Ecodena.Question.models.Question.commentList =  
property(getComments,setCommentList) [static]
```

Definition at line 83 of file models.py.

```
6.35.4.2 Ecodena.Question.models.Question.questionID =  
property(getQuestionID,setQuestionID) [static]
```

Definition at line 60 of file models.py.

**6.35.4.3** `Ecodena.Question.models.Question.questionText = property(getQuestionText,setQuestionText) [static]`

Definition at line 67 of file models.py.

**6.35.4.4** `Ecodena.Question.models.Question.questionTitle = property(getQuestionTitle, setQuestionTitle) [static]`

Definition at line 74 of file models.py.

**6.35.4.5** `Ecodena.Question.models.Question.testCasesList = property(getTestCasesList,setTestCasesList) [static]`

Definition at line 93 of file models.py.

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

- [/home/panx/ecodena/src/Ecodena/Question/models.py](#)

## 6.36 Ecodena.Question.admin.QuestionAdmin Class Reference

### Static Public Attributes

- list `inlines1` = [[TypeInline](#)]
- list `inlines2` = [[LevelInline](#)]
- tuple `list_display` = ('questionID\_f', 'questionTitle', 'level\_f', 'type\_f', 'timeLimit\_f')

### 6.36.1 Detailed Description

Definition at line 14 of file admin.py.

### 6.36.2 Member Data Documentation

**6.36.2.1** list `Ecodena.Question.admin.QuestionAdmin.inlines1` = [[TypeInline](#)]  
[static]

Definition at line 15 of file admin.py.

**6.36.2.2** list `Ecodena.Question.admin.QuestionAdmin.inlines2` = [[LevelInline](#)]  
[static]

Definition at line 16 of file admin.py.



6.36.2.3 tuple `Ecodena.Question.admin.QuestionAdmin.list_display` = ('questionID\_f', 'questionTitle', 'level\_f', 'type\_f', 'timeLimit\_f') `[static]`

Definition at line 17 of file `admin.py`.

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

- `/home/panx/ecodena/src/Ecodena/Question/admin.py`

## 6.37 Ecodena.Query.tests.SimpleTest Class Reference

Inherits `TestCase`.

### Public Member Functions

- def `test_basic_addition`

#### 6.37.1 Detailed Description

Definition at line 11 of file `tests.py`.

#### 6.37.2 Member Function Documentation

6.37.2.1 def `Ecodena.Query.tests.SimpleTest.test_basic_addition ( self )`

Tests that `1 + 1` always equals `2`.

Definition at line 12 of file `tests.py`.

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

- `/home/panx/ecodena/src/Ecodena/Query/tests.py`

## 6.38 Ecodena.Compiler.tests.SimpleTest Class Reference

Inherits `TestCase`.

### Public Member Functions

- def `test_basic_addition`

#### 6.38.1 Detailed Description

Definition at line 11 of file `tests.py`.

## 6.38.2 Member Function Documentation

### 6.38.2.1 `def Ecodena.Compiler.tests.SimpleTest.test_basic_addition ( self )`

Tests that 1 + 1 always equals 2.

Definition at line 12 of file tests.py.

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

- `/home/panx/ecodena/src/Ecodena/Compiler/tests.py`

## 6.39 Ecodena.Main.tests.SimpleTest Class Reference

Inherits TestCase.

### Public Member Functions

- `def test\_basic\_addition`

### 6.39.1 Detailed Description

Definition at line 11 of file tests.py.

## 6.39.2 Member Function Documentation

### 6.39.2.1 `def Ecodena.Main.tests.SimpleTest.test_basic_addition ( self )`

Tests that 1 + 1 always equals 2.

Definition at line 12 of file tests.py.

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

- `/home/panx/ecodena/src/Ecodena/Main/tests.py`

## 6.40 Ecodena.Question.tests.SimpleTest Class Reference

Inherits TestCase.

### Public Member Functions

- `def test\_basic\_addition`

### 6.40.1 Detailed Description

Definition at line 11 of file tests.py.

### 6.40.2 Member Function Documentation

#### 6.40.2.1 def Ecodena.Question.tests.SimpleTest.test\_basic\_addition ( self )

Tests that 1 + 1 always equals 2.

Definition at line 12 of file tests.py.

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

- /home/panx/ecodena/src/Ecodena/Question/[tests.py](#)

## 6.41 Ecodena.Comment.tests.SimpleTest Class Reference

Inherits TestCase.

### Public Member Functions

- def [test\\_basic\\_addition](#)

### 6.41.1 Detailed Description

Definition at line 11 of file tests.py.

### 6.41.2 Member Function Documentation

#### 6.41.2.1 def Ecodena.Comment.tests.SimpleTest.test\_basic\_addition ( self )

Tests that 1 + 1 always equals 2.

Definition at line 12 of file tests.py.

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

- /home/panx/ecodena/src/Ecodena/Comment/[tests.py](#)

## 6.42 Ecodena.Attempt.tests.SimpleTest Class Reference

Inherits TestCase.

## Public Member Functions

- def [test\\_basic\\_addition](#)

### 6.42.1 Detailed Description

Definition at line 11 of file tests.py.

### 6.42.2 Member Function Documentation

#### 6.42.2.1 def Ecodena.Attempt.tests.SimpleTest.test\_basic\_addition ( self )

Tests that 1 + 1 always equals 2.

Definition at line 12 of file tests.py.

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

- /home/panx/ecodena/src/Ecodena/Attempt/[tests.py](#)

## 6.43 Ecodena.User.tests.SimpleTest Class Reference

Inherits TestCase.

## Public Member Functions

- def [test\\_basic\\_addition](#)

### 6.43.1 Detailed Description

Definition at line 11 of file tests.py.

### 6.43.2 Member Function Documentation

#### 6.43.2.1 def Ecodena.User.tests.SimpleTest.test\_basic\_addition ( self )

Tests that 1 + 1 always equals 2.

Definition at line 12 of file tests.py.

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

- /home/panx/ecodena/src/Ecodena/User/[tests.py](#)

## 6.44 Ecodena.Attempt.views.SolutionForm Class Reference

### Static Public Attributes

- tuple [text](#) = forms.CharField(widget=forms.Textarea)
- tuple [version](#) = forms.ChoiceField()

#### 6.44.1 Detailed Description

creates a form for pasting the solution of a question

Definition at line 18 of file views.py.

#### 6.44.2 Member Data Documentation

6.44.2.1 tuple **Ecodena.Attempt.views.SolutionForm.text** =  
forms.CharField(widget=forms.Textarea) [static]

Definition at line 20 of file views.py.

6.44.2.2 tuple **Ecodena.Attempt.views.SolutionForm.version** = forms.ChoiceField()  
[static]

Definition at line 21 of file views.py.

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

- /home/panx/ecodena/src/Ecodena/Attempt/[views.py](#)

## 6.45 Ecodena.Question.models.TestCase Class Reference

### Classes

- class [Meta](#)

### Public Member Functions

- def [getTestID](#)
- def [setTestID](#)
- def [getCaseType](#)
- def [setCaseType](#)
- def [getOutput](#)
- def [setOutput](#)
- def [getQuestion](#)

- def [setQuestion](#)
- def [\\_\\_unicode\\_\\_](#)

### Public Attributes

- [caseID\\_f](#)
- [caseType\\_f](#)
- [output\\_f](#)
- [questionID\\_f](#)

### Static Public Attributes

- tuple [caseID\\_f](#) = models.AutoField(primary\_key=True)
- tuple [questionID\\_f](#) = models.ForeignKey([Question](#), verbose\_name="Question ID", null=False)
- tuple [CASE\\_TYPE](#) = ((0, 'Low'), (1, 'Medium'), (2, 'High'))
- tuple [caseType\\_f](#) = models.CharField(max\_length=40, choices=[CASE\\_TYPE](#), verbose\_name="Type of Case")
- tuple [input\\_f](#) = models.TextField("Input")
- tuple [output\\_f](#) = models.TextField("Output")

### Properties

- [caseID](#) = property(getTestID,setTestID)
- [caseType](#) = property(getCaseType,setCaseType)
- [output](#) = property(getOutput,setOutput)
- [questionID](#) = property(getQuestion,setQuestion)

### 6.45.1 Detailed Description

Creates the entity TestCase which is a subclass of models.Model

Definition at line 106 of file models.py.

### 6.45.2 Member Function Documentation

#### 6.45.2.1 def Ecodena.Question.models.TestCase.\_\_unicode\_\_( self )

Definition at line 146 of file models.py.

#### 6.45.2.2 def Ecodena.Question.models.TestCase.getCaseType ( self )

Definition at line 123 of file models.py.

**6.45.2.3** `def Ecodena.Question.models.TestCase.getOutput ( self )`

Definition at line 130 of file models.py.

**6.45.2.4** `def Ecodena.Question.models.TestCase.getQuestion ( self )`

Definition at line 136 of file models.py.

**6.45.2.5** `def Ecodena.Question.models.TestCase.getTestID ( self )`

Definition at line 116 of file models.py.

**6.45.2.6** `def Ecodena.Question.models.TestCase.setCaseType ( self, ctype )`

Definition at line 125 of file models.py.

**6.45.2.7** `def Ecodena.Question.models.TestCase.setOutput ( self, text )`

Definition at line 132 of file models.py.

**6.45.2.8** `def Ecodena.Question.models.TestCase.setQuestion ( self, qID )`

Definition at line 138 of file models.py.

**6.45.2.9** `def Ecodena.Question.models.TestCase.setTestID ( self, tID )`

Definition at line 118 of file models.py.

**6.45.3 Member Data Documentation****6.45.3.1** `tuple Ecodena.Question.models.TestCase.CASE_TYPE = ((0, 'Low'), (1, 'Medium'), (2, 'High'))` `[static]`

Definition at line 110 of file models.py.

**6.45.3.2** `tuple Ecodena.Question.models.TestCase.caseID_f = models.AutoField(primary_key=True)` `[static]`

Definition at line 108 of file models.py.

**6.45.3.3** `Ecodena.Question.models.TestCase.caseID_f`

Definition at line 118 of file models.py.

```
6.45.3.4 tuple Ecodena.Question.models.TestCase.caseType_f =  
        models.CharField(max_length=40, choices=CASE_TYPE, verbose_name="Type of  
        Case") [static]
```

Definition at line 111 of file models.py.

```
6.45.3.5 Ecodena.Question.models.TestCase.caseType_f
```

Definition at line 125 of file models.py.

```
6.45.3.6 tuple Ecodena.Question.models.TestCase.input_f = models.TextField("Input")  
        [static]
```

Definition at line 112 of file models.py.

```
6.45.3.7 tuple Ecodena.Question.models.TestCase.output_f =  
        models.TextField("Output") [static]
```

Definition at line 113 of file models.py.

```
6.45.3.8 Ecodena.Question.models.TestCase.output_f
```

Definition at line 132 of file models.py.

```
6.45.3.9 tuple Ecodena.Question.models.TestCase.questionID_f =  
        models.ForeignKey(Question , verbose_name="Question ID", null=False)  
        [static]
```

Definition at line 109 of file models.py.

```
6.45.3.10 Ecodena.Question.models.TestCase.questionID_f
```

Definition at line 138 of file models.py.

## 6.45.4 Property Documentation

```
6.45.4.1 Ecodena.Question.models.TestCase.caseID = property(getTestID,setTestID)  
        [static]
```

Definition at line 120 of file models.py.



6.45.4.2 `Ecodena.Question.models.TestCase.caseType = property(getCaseType,setCaseType)`  
`[static]`

Definition at line 127 of file models.py.

6.45.4.3 `Ecodena.Question.models.TestCase.output = property(getOutput,setOutput)`  
`[static]`

Definition at line 134 of file models.py.

6.45.4.4 `Ecodena.Question.models.TestCase.questionID = property(getQuestion,setQuestion)`  
`[static]`

Definition at line 140 of file models.py.

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

- [/home/panx/ecodena/src/Ecodena/Question/models.py](#)

## 6.46 Ecodena.Question.models.Type Class Reference

### Classes

- class [Meta](#)

### Public Member Functions

- def [\\_\\_unicode\\_\\_](#)

### Static Public Attributes

- tuple [typeID\\_f](#) = `models.AutoField(primary_key=True)`
- tuple [typeName\\_f](#) = `models.CharField("Type Name" , max_length = 30 , null = False)`

### 6.46.1 Detailed Description

Creates the entity Type which is a subclass of models.Model class

Definition at line 31 of file models.py.

### 6.46.2 Member Function Documentation

6.46.2.1 `def Ecodena.Question.models.Type.__unicode__( self )`

Definition at line 40 of file models.py.

### 6.46.3 Member Data Documentation

6.46.3.1 `tuple Ecodena.Question.models.Type.typeID_f = models.AutoField(primary_key=True) [static]`

Definition at line 33 of file models.py.

6.46.3.2 `tuple Ecodena.Question.models.Type.typeName_f = models.CharField("Type Name", max_length = 30, null = False) [static]`

Definition at line 34 of file models.py.

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

- [/home/panx/ecodena/src/Ecodena/Question/models.py](#)

## 6.47 Ecodena.Question.admin.TypeInline Class Reference

### Static Public Attributes

- `model = Type`

### 6.47.1 Detailed Description

Definition at line 7 of file admin.py.

### 6.47.2 Member Data Documentation

6.47.2.1 `Ecodena.Question.admin.TypeInline.model = Type [static]`

Definition at line 8 of file admin.py.

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

- [/home/panx/ecodena/src/Ecodena/Question/admin.py](#)

## Chapter 7

# File Documentation

### 7.1 /home/panx/ecodena/src/Ecodena/\_\_init\_\_.py File Reference

#### Packages

- package [Ecodena](#)

### 7.2 /home/panx/ecodena/src/Ecodena/Attempt/\_\_init\_\_.py File Reference

#### Packages

- package [Ecodena.Attempt](#)

### 7.3 /home/panx/ecodena/src/Ecodena/Comment/\_\_init\_\_.py File Reference

#### Packages

- package [Ecodena.Comment](#)

### 7.4 /home/panx/ecodena/src/Ecodena/Compiler/\_\_init\_\_.py File Reference

#### Packages

- package [Ecodena.Compiler](#)

## 7.5 /home/panx/ecodena/src/Ecodena/Main/\_\_init\_\_.py File Reference

### Packages

- package [Ecodena.Main](#)

## 7.6 /home/panx/ecodena/src/Ecodena/Query/\_\_init\_\_.py File Reference

### Packages

- package [Ecodena.Query](#)

## 7.7 /home/panx/ecodena/src/Ecodena/Question/\_\_init\_\_.py File Reference

### Packages

- package [Ecodena.Question](#)

## 7.8 /home/panx/ecodena/src/Ecodena/User/\_\_init\_\_.py File Reference

### Packages

- package [Ecodena.User](#)

## 7.9 /home/panx/ecodena/src/Ecodena/Attempt/admin.py File Reference

### Classes

- class [Ecodena.Attempt.admin.AttemptAdmin](#)
- class [Ecodena.Attempt.admin.ErrorReportAdmin](#)

### Packages

- package [Ecodena.Attempt.admin](#)

## 7.10 /home/panx/ecodena/src/Ecodena/Comment/admin.py File Reference

### Classes

- class [Ecodena.Comment.admin.CommentAdmin](#)

### Packages

- package [Ecodena.Comment.admin](#)

## 7.11 /home/panx/ecodena/src/Ecodena/Compiler/admin.py File Reference

### Classes

- class [Ecodena.Compiler.admin.LanguageAdmin](#)
- class [Ecodena.Compiler.admin.CompilerVersionAdmin](#)

### Packages

- package [Ecodena.Compiler.admin](#)

## 7.12 /home/panx/ecodena/src/Ecodena/Query/admin.py File Reference

### Classes

- class [Ecodena.Query.admin.QueryAdmin](#)

### Packages

- package [Ecodena.Query.admin](#)

## 7.13 /home/panx/ecodena/src/Ecodena/Question/admin.py File Reference

### Classes

- class [Ecodena.Question.admin.TypeInline](#)

- class [Ecodena.Question.admin.LevelInline](#)
- class [Ecodena.Question.admin.QuestionAdmin](#)

### Packages

- package [Ecodena.Question.admin](#)

## 7.14 /home/panx/ecodena/src/Ecodena/User/admin.py File Reference

### Classes

- class [Ecodena.User.admin.ProfileAdmin](#)

### Packages

- package [Ecodena.User.admin](#)

## 7.15 /home/panx/ecodena/src/Ecodena/Attempt/models.py File Reference

### Classes

- class [Ecodena.Attempt.models.ErrorReport](#)
- class [Ecodena.Attempt.models.ErrorReport.Meta](#)
- class [Ecodena.Attempt.models.Attempt](#)
- class [Ecodena.Attempt.models.Attempt.Meta](#)

### Packages

- package [Ecodena.Attempt.models](#)

## 7.16 /home/panx/ecodena/src/Ecodena/Comment/models.py File Reference

### Classes

- class [Ecodena.Comment.models.Comment](#)
- class [Ecodena.Comment.models.Comment.Meta](#)

## Packages

- package [Ecodena.Comment.models](#)

## 7.17 /home/panx/ecodena/src/Ecodena/Compiler/models.py File Reference

### Classes

- class [Ecodena.Compiler.models.Language](#)
- class [Ecodena.Compiler.models.Language.Meta](#)
- class [Ecodena.Compiler.models.CompilerVersion](#)
- class [Ecodena.Compiler.models.CompilerVersion.Meta](#)
- class [Ecodena.Compiler.models.Compiler](#)
- class [Ecodena.Compiler.models.CCompiler111](#)
- class [Ecodena.Compiler.models.JavaCompiler111](#)

## Packages

- package [Ecodena.Compiler.models](#)

## 7.18 /home/panx/ecodena/src/Ecodena/Main/models.py File Reference

### Packages

- package [Ecodena.Main.models](#)

## 7.19 /home/panx/ecodena/src/Ecodena/Query/models.py File Reference

### Classes

- class [Ecodena.Query.models.Query](#)
- class [Ecodena.Query.models.Query.Meta](#)

## Packages

- package [Ecodena.Query.models](#)

## 7.20 `/home/panx/ecodena/src/Ecodena/Question/models.py` File Reference

### Classes

- class [Ecodena.Question.models.Level](#)
- class [Ecodena.Question.models.Level.Meta](#)
- class [Ecodena.Question.models.Type](#)
- class [Ecodena.Question.models.Type.Meta](#)
- class [Ecodena.Question.models.Question](#)
- class [Ecodena.Question.models.Question.Meta](#)
- class [Ecodena.Question.models.BattleQuestion](#)
- class [Ecodena.Question.models.TestCase](#)
- class [Ecodena.Question.models.TestCase.Meta](#)

### Packages

- package [Ecodena.Question.models](#)

## 7.21 `/home/panx/ecodena/src/Ecodena/User/models.py` File Reference

### Classes

- class [Ecodena.User.models.Programmer](#)
- class [Ecodena.User.models.Profile](#)

### Packages

- package [Ecodena.User.models](#)

## 7.22 `/home/panx/ecodena/src/Ecodena/Attempt/tests.py` File Reference

### Classes

- class [Ecodena.Attempt.tests.SimpleTest](#)

### Packages

- package [Ecodena.Attempt.tests](#)



## 7.23 /home/panx/ecodena/src/Ecodena/Comment/tests.py File Reference

### Classes

- class [Ecodena.Comment.tests.SimpleTest](#)

### Packages

- package [Ecodena.Comment.tests](#)

## 7.24 /home/panx/ecodena/src/Ecodena/Compiler/tests.py File Reference

### Classes

- class [Ecodena.Compiler.tests.SimpleTest](#)

### Packages

- package [Ecodena.Compiler.tests](#)

## 7.25 /home/panx/ecodena/src/Ecodena/Main/tests.py File Reference

### Classes

- class [Ecodena.Main.tests.SimpleTest](#)

### Packages

- package [Ecodena.Main.tests](#)

## 7.26 /home/panx/ecodena/src/Ecodena/Query/tests.py File Reference

### Classes

- class [Ecodena.Query.tests.SimpleTest](#)

### Packages

- package [Ecodena.Query.tests](#)

## 7.27 [/home/panx/ecodena/src/Ecodena/Question/tests.py](#) File Reference

### Classes

- class [Ecodena.Question.tests.SimpleTest](#)

### Packages

- package [Ecodena.Question.tests](#)

## 7.28 [/home/panx/ecodena/src/Ecodena/User/tests.py](#) File Reference

### Classes

- class [Ecodena.User.tests.SimpleTest](#)

### Packages

- package [Ecodena.User.tests](#)

## 7.29 [/home/panx/ecodena/src/Ecodena/Attempt/views.py](#) File Reference

### Classes

- class [Ecodena.Attempt.views.SolutionForm](#)

### Packages

- package [Ecodena.Attempt.views](#)

### Functions

- def [Ecodena::Attempt::views.listSolutions](#)
- def [Ecodena::Attempt::views.viewSolution](#)

- def [Ecodena::Attempt::views.submitSolution](#)
- def [Ecodena::Attempt::views.compileSolution](#)

## **7.30 /home/panx/ecodena/src/Ecodena/Comment/views.py File Reference**

### **Packages**

- package [Ecodena.Comment.views](#)

## **7.31 /home/panx/ecodena/src/Ecodena/Compiler/views.py File Reference**

### **Packages**

- package [Ecodena.Compiler.views](#)

### **Functions**

- def [Ecodena::Compiler::views.CCompile110](#)
- def [Ecodena::Compiler::views.JavaCompile110](#)

## **7.32 /home/panx/ecodena/src/Ecodena/Main/views.py File Reference**

### **Packages**

- package [Ecodena.Main.views](#)

### **Functions**

- def [Ecodena::Main::views.home](#)

## **7.33 /home/panx/ecodena/src/Ecodena/Query/views.py File Reference**

### **Classes**

- class [Ecodena.Query.views.QueryForm](#)

### Packages

- package [Ecodena.Query.views](#)

### Functions

- def [Ecodena::Query::views.postQuery](#)

## 7.34 [/home/panx/ecodena/src/Ecodena/Question/views.py](#) File Reference

### Classes

- class [Ecodena.Question.views.CommentForm](#)

### Packages

- package [Ecodena.Question.views](#)

### Functions

- def [Ecodena::Question::views.test](#)
- def [Ecodena::Question::views.listQuestions](#)
- def [Ecodena::Question::views.viewQuestionById](#)

## 7.35 [/home/panx/ecodena/src/Ecodena/User/views.py](#) File Reference

### Classes

- class [Ecodena.User.views.ProfileForm](#)

### Packages

- package [Ecodena.User.views](#)

### Functions

- def [Ecodena::User::views.viewProfile](#)
- def [Ecodena::User::views.editProfile](#)

### 7.36 /home/panx/ecodena/src/Ecodena/Compiler/dictionary.py File Reference 103

## 7.36 /home/panx/ecodena/src/Ecodena/Compiler/dictionary.py File Reference

### Packages

- package [Ecodena.Compiler.dictionary](#)

### Variables

- dictionary [Ecodena::Compiler::dictionary.compilers](#)

## 7.37 /home/panx/ecodena/src/Ecodena/manage.py File Reference

### Packages

- package [Ecodena.manage](#)

## 7.38 /home/panx/ecodena/src/Ecodena/settings.py File Reference

### Packages

- package [Ecodena.settings](#)

### Variables

- [Ecodena::settings.DEBUG](#) = True
- [Ecodena::settings.TEMPLATE\\_DEBUG](#) = DEBUG
- tuple [Ecodena::settings.ADMINS](#)
- [Ecodena::settings.MANAGERS](#) = ADMINS
- dictionary [Ecodena::settings.DATABASES](#)
- string [Ecodena::settings.TIME\\_ZONE](#) = 'America/Chicago'
- string [Ecodena::settings.LANGUAGE\\_CODE](#) = 'en-us'
- int [Ecodena::settings.SITE\\_ID](#) = 1
- [Ecodena::settings.USE\\_I18N](#) = True
- [Ecodena::settings.USE\\_L10N](#) = True
- string [Ecodena::settings.MEDIA\\_ROOT](#) = "
- string [Ecodena::settings.MEDIA\\_URL](#) = "
- string [Ecodena::settings.STATIC\\_ROOT](#) = "
- string [Ecodena::settings.STATIC\\_URL](#) = '/static/'
- string [Ecodena::settings.ADMIN\\_MEDIA\\_PREFIX](#) = '/static/admin/'
- tuple [Ecodena::settings.STATICFILES\\_DIRS](#)
- tuple [Ecodena::settings.STATICFILES\\_FINDERS](#)

- string [Ecodena::settings.SECRET\\_KEY](#) = 'k!\*n&!9\$t9sp61=vq(%t^(k@yz7o^=\*y=i2kmxtpi!4+88gyhl'
- tuple [Ecodena::settings.TEMPLATE\\_LOADERS](#)
- tuple [Ecodena::settings.MIDDLEWARE\\_CLASSES](#)
- string [Ecodena::settings.ROOT\\_URLCONF](#) = 'Ecodena.urls'
- tuple [Ecodena::settings.TEMPLATE\\_DIRS](#)
- tuple [Ecodena::settings.INSTALLED\\_APPS](#)
- string [Ecodena::settings.AUTH\\_PROFILE\\_MODULE](#) = 'Ecodena.User.Profile'
- dictionary [Ecodena::settings.LOGGING](#)

### 7.39 /home/panx/ecodena/src/Ecodena/urls.py File Reference

#### Packages

- package [Ecodena.urls](#)

#### Variables

- tuple [Ecodena::urls.urlpatterns](#)

# Index

/home/panx/ecodena/src/ECodena/Attempt/\_ - \_init\_\_.py, 94  
\_init\_\_.py, 93 /home/panx/ecodena/src/ECodena/Query/admin.py,  
/home/panx/ecodena/src/ECodena/Attempt/admin.py, 95  
94 /home/panx/ecodena/src/ECodena/Query/models.py,  
/home/panx/ecodena/src/ECodena/Attempt/models.py, 97  
96 /home/panx/ecodena/src/ECodena/Query/tests.py,  
/home/panx/ecodena/src/ECodena/Attempt/tests.py, 99  
98 /home/panx/ecodena/src/ECodena/Query/views.py,  
/home/panx/ecodena/src/ECodena/Attempt/views.py, 101  
100 /home/panx/ecodena/src/ECodena/Question/\_ -  
/home/panx/ecodena/src/ECodena/Comment/\_ - \_init\_\_.py, 94  
\_init\_\_.py, 93 /home/panx/ecodena/src/ECodena/Question/admin.py,  
/home/panx/ecodena/src/ECodena/Comment/admin.py, 95  
95 /home/panx/ecodena/src/ECodena/Question/models.py,  
/home/panx/ecodena/src/ECodena/Comment/models.py, 98  
96 /home/panx/ecodena/src/ECodena/Question/tests.py,  
/home/panx/ecodena/src/ECodena/Comment/tests.py, 100  
99 /home/panx/ecodena/src/ECodena/Question/views.py,  
/home/panx/ecodena/src/ECodena/Comment/views.py, 102  
101 /home/panx/ecodena/src/ECodena/User/\_ -  
/home/panx/ecodena/src/ECodena/Compiler/\_ - \_init\_\_.py, 94  
\_init\_\_.py, 93 /home/panx/ecodena/src/ECodena/User/admin.py,  
/home/panx/ecodena/src/ECodena/Compiler/admin.py, 96  
95 /home/panx/ecodena/src/ECodena/User/models.py,  
/home/panx/ecodena/src/ECodena/Compiler/dictionary, 99  
103 /home/panx/ecodena/src/ECodena/User/tests.py,  
/home/panx/ecodena/src/ECodena/Compiler/models.py, 100  
97 /home/panx/ecodena/src/ECodena/User/views.py,  
/home/panx/ecodena/src/ECodena/Compiler/tests.py, 102  
99 /home/panx/ecodena/src/ECodena/\_ \_init\_ -  
/home/panx/ecodena/src/ECodena/Compiler/views.py, \_ .py, 93  
101 /home/panx/ecodena/src/ECodena/manage.py,  
/home/panx/ecodena/src/ECodena/Main/\_ - 103  
\_init\_\_.py, 94 /home/panx/ecodena/src/ECodena/settings.py,  
/home/panx/ecodena/src/ECodena/Main/models.py, 103  
97 /home/panx/ecodena/src/ECodena/urls.py,  
/home/panx/ecodena/src/ECodena/Main/tests.py, 104  
99 \_ \_unicode\_ \_  
/home/panx/ecodena/src/ECodena/Main/views.py, ECodena::Attempt::models::Attempt, 28  
101 ECodena::Attempt::models::ErrorReport,  
/home/panx/ecodena/src/ECodena/Query/\_ - 48

Ecodena::Comment::models::Comment, 36  
 Ecodena::Compiler::models::CompilerVersionType\_f, 45  
 Ecodena::Compiler::models::Language, 53  
 Ecodena::Query::models::Query, 72  
 Ecodena::Question::models::Level, 55  
 Ecodena::Question::models::Question, 79  
 Ecodena::Question::models::TestCase, 88  
 Ecodena::Question::models::Type, 92  
 Ecodena::Question::models::TestCase, 90  
 Ecodena::Question::models::TestCase, 89, 90  
 CCompile110  
 Ecodena::Compiler::views, 14  
 commentID\_f  
 Ecodena::Comment::models::Comment, 37  
 commentList  
 Ecodena::Question::models::Question, 81  
 commentList\_f  
 Ecodena::Question::models::Question, 80  
 commentText  
 Ecodena::Comment::models::Comment, 39  
 commentText\_f  
 Ecodena::Comment::models::Comment, 37, 38  
 compile  
 Ecodena::Compiler::models::CCompiler111, 34  
 Ecodena::Compiler::models::Compiler, 42  
 Ecodena::Compiler::models::JavaCompiler111, 52  
 compilers  
 Ecodena::Compiler::dictionary, 13  
 CompilerVersion  
 Ecodena::Compiler::models::Compiler, 43  
 compilerVersion  
 Ecodena::Attempt::models::Attempt, 32  
 Ecodena::Attempt::models::Attempt, 30  
 Ecodena::Compiler::models::Compiler, 43  
 compilerVersion\_f  
 Ecodena::Attempt::models::Attempt, 30  
 compileSolution  
 Ecodena::Attempt::views, 11  
 CASE\_TYPE  
 Ecodena::Question::models::TestCase, 89  
 caseID  
 Ecodena::Question::models::TestCase, 90  
 caseID\_f  
 Ecodena::Question::models::TestCase, 89  
 caseType  
 Ecodena::User::models::Profile, 67  
 Ecodena::User::views::ProfileForm, 68  
 ADMIN\_MEDIA\_PREFIX  
 Ecodena::settings, 19  
 adminID  
 Ecodena::Query::models::Query, 75  
 adminID\_f  
 Ecodena::Query::models::Query, 74  
 ADMINS  
 Ecodena::settings, 19  
 attempt  
 Ecodena::Compiler::models::Compiler, 43  
 attemptID  
 Ecodena::Attempt::models::Attempt, 32  
 attemptID\_f  
 Ecodena::Attempt::models::Attempt, 30  
 AUTH\_PROFILE\_MODULE  
 Ecodena::settings, 19  
 DATABASES  
 Ecodena::settings, 19  
 DEBUG  
 Ecodena::settings, 20  
 Ecodena::User::models::Profile, 67  
 Ecodena::User::views::ProfileForm, 68



- dob\_f
  - Ecodena::User::models::Profile, 66
- Ecodena, 9
- Ecodena.Attempt, 9
- Ecodena.Attempt.admin, 9
- Ecodena.Attempt.models, 10
- Ecodena.Attempt.tests, 10
- Ecodena.Attempt.views, 10
- Ecodena.Comment, 11
- Ecodena.Comment.admin, 11
- Ecodena.Comment.models, 12
- Ecodena.Comment.tests, 12
- Ecodena.Comment.views, 12
- Ecodena.Compiler, 12
- Ecodena.Compiler.admin, 12
- Ecodena.Compiler.dictionary, 13
- Ecodena.Compiler.models, 13
- Ecodena.Compiler.tests, 13
- Ecodena.Compiler.views, 14
- Ecodena.Main, 14
- Ecodena.Main.models, 14
- Ecodena.Main.tests, 14
- Ecodena.Main.views, 15
- Ecodena.manage, 15
- Ecodena.Query, 15
- Ecodena.Query.admin, 15
- Ecodena.Query.models, 15
- Ecodena.Query.tests, 16
- Ecodena.Query.views, 16
- Ecodena.Question, 16
- Ecodena.Question.admin, 17
- Ecodena.Question.models, 17
- Ecodena.Question.tests, 17
- Ecodena.Question.views, 17
- Ecodena.settings, 18
- Ecodena.urls, 23
- Ecodena.User, 24
- Ecodena.User.admin, 24
- Ecodena.User.models, 24
- Ecodena.User.tests, 25
- Ecodena.User.views, 25
- Ecodena::Attempt::admin::AttemptAdmin, 33
  - list\_display, 33
- Ecodena::Attempt::admin::ErrorReportAdmin, 51
  - list\_display, 52
- Ecodena::Attempt::models::Attempt, 27
  - \_\_unicode\_\_, 28
  - attemptID, 32
  - attemptID\_f, 30
  - compilerVersion, 32
  - compilerVersion\_f, 30
  - errorReportID, 32
  - errorReportID\_f, 31
  - getAttemptID, 28
  - getCompilerVersionOfAttempt, 29
  - getErrorReportID, 29
  - getQuestionID, 29
  - getSolution, 29
  - getTimeOfSubmission, 29
  - getUserID, 29
  - isCorrect, 29
  - questionID, 32
  - questionID\_f, 31
  - setAttemptID, 29
  - setCompilerVersionOfAttempt, 29
  - setErrorReportID, 29
  - setQuestionID, 30
  - setSolution, 30
  - setStatus, 30
  - setTimeOfSubmission, 30
  - setUserID, 30
  - solution\_f, 31
  - solutionText, 32
  - status, 33
  - status\_f, 31
  - timeOfSubmission, 33
  - timeOfSubmission\_f, 31, 32
  - userID, 33
  - userID\_f, 32
- Ecodena::Attempt::models::Attempt::Meta, 62
  - verbose\_name, 62
  - verbose\_name\_plural, 62
- Ecodena::Attempt::models::ErrorReport, 47
  - \_\_unicode\_\_, 48
  - ERROR\_TYPE, 49
  - errorMessage, 51
  - errorMessage\_f, 49
  - errorReportID\_f, 49
  - errorType, 51
  - errorType\_f, 50
  - getErrorMessage, 48
  - getErrorType, 48
  - getMemory, 48
  - getTestCaseLevel, 48
  - getTimeRequirement, 48
  - memory, 51
  - memory\_f, 50

- setErrorMessage, [49](#)
- setErrorType, [49](#)
- setMemory, [49](#)
- setTestCaseLevel, [49](#)
- setTimeRequirement, [49](#)
- testCaseLevel, [51](#)
- testCaseLevel\_f, [50](#)
- timeRequirement, [51](#)
- timeRequirement\_f, [50](#)
- Ecodena::Attempt::models::ErrorReport::Meta, [60](#)
  - verbose\_name, [60](#)
  - verbose\_name\_plural, [60](#)
- Ecodena::Attempt::tests::SimpleTest, [85](#)
  - test\_basic\_addition, [86](#)
- Ecodena::Attempt::views
  - compileSolution, [11](#)
  - listSolutions, [11](#)
  - submitSolution, [11](#)
  - viewSolution, [11](#)
- Ecodena::Attempt::views::SolutionForm, [87](#)
  - text, [87](#)
  - version, [87](#)
- Ecodena::Comment::admin::CommentAdmin, [40](#)
  - list\_display, [40](#)
- Ecodena::Comment::models::Comment, [34](#)
  - \_\_unicode\_\_, [36](#)
  - commentID\_f, [37](#)
  - commentText, [39](#)
  - commentText\_f, [37](#), [38](#)
  - getCommentText, [36](#)
  - getQuestionID, [36](#)
  - getReportingUserID, [36](#)
  - getTimeStamp, [36](#)
  - getUserID, [36](#)
  - isReported, [36](#), [39](#)
  - isReported\_f, [38](#)
  - questionID, [39](#)
  - questionID\_f, [38](#)
  - report, [37](#)
  - reportingUserID, [39](#)
  - reportingUserID\_f, [38](#)
  - setCommentText, [37](#)
  - setQuestionID, [37](#)
  - setReportingUserID, [37](#)
  - setTimeStamp, [37](#)
  - setUserID, [37](#)
  - timeStamp, [39](#)
  - timeStamp\_f, [38](#)
  - userID, [39](#)
  - userID\_f, [39](#)
- Ecodena::Comment::models::Comment::Meta, [61](#)
  - verbose\_name, [62](#)
  - verbose\_name\_plural, [62](#)
- Ecodena::Comment::tests::SimpleTest, [85](#)
  - test\_basic\_addition, [85](#)
- Ecodena::Compiler::admin::CompilerVersionAdmin, [46](#)
  - list\_display, [47](#)
- Ecodena::Compiler::admin::LanguageAdmin, [54](#)
  - list\_display, [54](#)
- Ecodena::Compiler::dictionary
  - compilers, [13](#)
- Ecodena::Compiler::models::CCompiler111, [34](#)
  - compile, [34](#)
- Ecodena::Compiler::models::Compiler, [41](#)
  - attempt, [43](#)
  - compile, [42](#)
  - CompilerVersion, [43](#)
  - compilerVersion, [43](#)
  - errorReport, [43](#)
  - generateErrorReport, [42](#)
  - getAttempt, [42](#)
  - getCompilerVersion, [42](#)
  - getLanguage, [42](#)
  - getQuestion, [42](#)
  - language, [43](#)
  - question, [43](#)
  - setAttempt, [42](#)
  - setCompilerVersion, [42](#)
  - setLanguage, [43](#)
  - setQuestion, [43](#)
- Ecodena::Compiler::models::CompilerVersion, [44](#)
  - \_\_unicode\_\_, [45](#)
  - getLanguage, [45](#)
  - getVersionName, [45](#)
  - language, [46](#)
  - language\_f, [45](#)
  - languageName, [45](#)
  - setLanguage, [45](#)
  - setVersionName, [45](#)
  - versionID\_f, [45](#)
  - versionName, [46](#)
  - versionName\_f, [46](#)

Ecodena::Compiler::models::CompilerVersion::Meta, 59  
     verbose\_name, 59  
     verbose\_name\_plural, 59  
 Ecodena::Compiler::models::JavaCompiler111, 52  
     compile, 52  
 Ecodena::Compiler::models::Language, 52  
     \_\_unicode\_\_, 53  
     getLanguageName, 53  
     languageID\_f, 54  
     languageName, 54  
     languageName\_f, 54  
     setLanguageName, 53  
 Ecodena::Compiler::models::Language::Meta, 58  
     verbose\_name, 58  
     verbose\_name\_plural, 58  
 Ecodena::Compiler::tests::SimpleTest, 83  
     test\_basic\_addition, 84  
 Ecodena::Compiler::views  
     CCompile110, 14  
     JavaCompile110, 14  
 Ecodena::Main::tests::SimpleTest, 84  
     test\_basic\_addition, 84  
 Ecodena::Main::views  
     home, 15  
 Ecodena::Query::admin::QueryAdmin, 76  
     list\_display, 77  
 Ecodena::Query::models::Query, 71  
     \_\_unicode\_\_, 72  
     adminID, 75  
     adminID\_f, 74  
     getAdminID, 72  
     getQueryID, 72  
     getQueryText, 72  
     getQueryTime, 73  
     getReplyTime, 73  
     getSolution, 73  
     getUserID, 73  
     queryID, 75  
     queryID\_f, 74  
     queryText, 76  
     queryText\_f, 74  
     queryTime, 76  
     queryTime\_f, 74  
     replyTime, 76  
     replyTime\_f, 75  
     setAdminID, 73  
     setQueryID, 73  
     setQueryText, 73  
     setQueryTime, 73  
     setReplyTime, 73  
     setSolution, 73  
     setUserID, 74  
     solution, 76  
     solution\_f, 75  
     userID, 76  
     userID\_f, 75  
 Ecodena::Query::models::Query::Meta, 63  
     verbose\_name, 63  
     verbose\_name\_plural, 63  
 Ecodena::Query::tests::SimpleTest, 83  
     test\_basic\_addition, 83  
 Ecodena::Query::views  
     postQuery, 16  
 Ecodena::Query::views::QueryForm, 77  
     text, 77  
 Ecodena::Question::admin::LevelInline, 56  
     extra, 57  
     model, 57  
 Ecodena::Question::admin::QuestionAdmin, 82  
     inlines1, 82  
     inlines2, 82  
     list\_display, 82  
 Ecodena::Question::admin::TypeInline, 92  
     model, 92  
 Ecodena::Question::models::BattleQuestion, 34  
 Ecodena::Question::models::Level, 55  
     \_\_unicode\_\_, 55  
     getLevelName, 55  
     levelID\_f, 56  
     levelName, 56  
     levelName\_f, 56  
     setLevelName, 56  
 Ecodena::Question::models::Level::Meta, 57  
     verbose\_name, 57  
     verbose\_name\_plural, 57  
 Ecodena::Question::models::Question, 77  
     \_\_unicode\_\_, 79  
     addComments, 79  
     commentList, 81  
     commentList\_f, 80  
     getComments, 79  
     getQuestionID, 79  
     getQuestionText, 79  
     getQuestionTitle, 79  
     getTestCasesList, 79

- level\_f, 80
- questionID, 81
- questionID\_f, 80
- questionText, 81
- questionText\_f, 80
- questionTitle, 82
- questionTitle\_f, 81
- removeTestCase, 79
- setCommentList, 79
- setQuestionID, 79
- setQuestionText, 79
- setQuestionTitle, 80
- setTestCasesList, 80
- testCasesList, 82
- testCasesList\_f, 81
- timeLimit\_f, 81
- type\_f, 81
- Ecodena::Question::models::Question::Meta, 59
  - verbose\_name, 60
  - verbose\_name\_plural, 60
- Ecodena::Question::models::TestCase, 87
  - \_\_unicode\_\_, 88
  - CASE\_TYPE, 89
  - caseID, 90
  - caseID\_f, 89
  - caseType, 90
  - caseType\_f, 89, 90
  - getCaseType, 88
  - getOutput, 88
  - getQuestion, 89
  - getTestID, 89
  - input\_f, 90
  - output, 91
  - output\_f, 90
  - questionID, 91
  - questionID\_f, 90
  - setCaseType, 89
  - setOutput, 89
  - setQuestion, 89
  - setTestID, 89
- Ecodena::Question::models::TestCase::Meta, 61
  - verbose\_name, 61
  - verbose\_name\_plural, 61
- Ecodena::Question::models::Type, 91
  - \_\_unicode\_\_, 92
  - typeID\_f, 92
  - typeName\_f, 92
- Ecodena::Question::models::Type::Meta, 58
  - verbose\_name, 58
  - verbose\_name\_plural, 58
- Ecodena::Question::tests::SimpleTest, 84
  - test\_basic\_addition, 85
- Ecodena::Question::views
  - listQuestions, 18
  - test, 18
  - viewQuestionByID, 18
- Ecodena::Question::views::CommentForm, 40
  - text, 41
- Ecodena::settings
  - ADMIN\_MEDIA\_PREFIX, 19
  - ADMINS, 19
  - AUTH\_PROFILE\_MODULE, 19
  - DATABASES, 19
  - DEBUG, 20
  - INSTALLED\_APPS, 20
  - LANGUAGE\_CODE, 20
  - LOGGING, 20
  - MANAGERS, 21
  - MEDIA\_ROOT, 21
  - MEDIA\_URL, 21
  - MIDDLEWARE\_CLASSES, 21
  - ROOT\_URLCONF, 21
  - SECRET\_KEY, 21
  - SITE\_ID, 21
  - STATIC\_ROOT, 22
  - STATIC\_URL, 22
  - STATICFILES\_DIRS, 22
  - STATICFILES\_FINDERS, 22
  - TEMPLATE\_DEBUG, 22
  - TEMPLATE\_DIRS, 22
  - TEMPLATE\_LOADERS, 23
  - TIME\_ZONE, 23
  - USE\_I18N, 23
  - USE\_L10N, 23
- Ecodena::urls
  - urlpatterns, 23
- Ecodena::User::admin::ProfileAdmin, 67
  - list\_display, 68
- Ecodena::User::models::Profile, 63
  - address, 67
  - address\_f, 65
  - dob, 67
  - dob\_f, 66
  - gender, 67
  - gender\_f, 66
  - GENDER\_TYPE, 66
  - getAddress, 64

- getDOB, [64](#)
- getGender, [64](#)
- getNumberOfAttempts, [65](#)
- getUserID, [65](#)
- isAdmin, [65](#)
- isAdmin\_f, [66](#)
- isProgrammer, [65](#)
- isProgrammer\_f, [66](#)
- numberOfAttempts, [67](#)
- numberOfAttempts\_f, [66](#)
- profileID\_f, [67](#)
- setAddress, [65](#)
- setDOB, [65](#)
- setGender, [65](#)
- setNumberOfAttempts, [65](#)
- userID\_f, [67](#)
- Ecodena::User::models::Programmer, [69](#)
  - getLevel, [70](#)
  - level, [71](#)
  - level\_f, [70](#)
  - setLevel, [70](#)
- Ecodena::User::tests::SimpleTest, [86](#)
  - test\_basic\_addition, [86](#)
- Ecodena::User::views
  - editProfile, [25](#)
  - viewProfile, [25](#)
- Ecodena::User::views::ProfileForm, [68](#)
  - address, [68](#)
  - dob, [68](#)
  - email, [69](#)
  - gender, [69](#)
  - GENDER\_CHOICES, [69](#)
  - username, [69](#)
- editProfile
  - Ecodena::User::views, [25](#)
- email
  - Ecodena::User::views::ProfileForm, [69](#)
- ERROR\_TYPE
  - Ecodena::Attempt::models::ErrorReport, [49](#)
- errorMessage
  - Ecodena::Attempt::models::ErrorReport, [51](#)
- errorMessage\_f
  - Ecodena::Attempt::models::ErrorReport, [49](#)
- errorReport
  - Ecodena::Compiler::models::Compiler, [43](#)
- errorReportID
  - Ecodena::Attempt::models::Attempt, [32](#)
- errorReportID\_f
  - Ecodena::Attempt::models::Attempt, [31](#)
  - Ecodena::Attempt::models::ErrorReport, [49](#)
- errorType
  - Ecodena::Attempt::models::ErrorReport, [51](#)
- errorType\_f
  - Ecodena::Attempt::models::ErrorReport, [50](#)
- extra
  - Ecodena::Question::admin::LevelInline, [57](#)
- gender
  - Ecodena::User::models::Profile, [67](#)
  - Ecodena::User::views::ProfileForm, [69](#)
- GENDER\_CHOICES
  - Ecodena::User::views::ProfileForm, [69](#)
- gender\_f
  - Ecodena::User::models::Profile, [66](#)
- GENDER\_TYPE
  - Ecodena::User::models::Profile, [66](#)
- generateErrorReport
  - Ecodena::Compiler::models::Compiler, [42](#)
- getAddress
  - Ecodena::User::models::Profile, [64](#)
- getAdminID
  - Ecodena::Query::models::Query, [72](#)
- getAttempt
  - Ecodena::Compiler::models::Compiler, [42](#)
- getAttemptID
  - Ecodena::Attempt::models::Attempt, [28](#)
- getCaseType
  - Ecodena::Question::models::TestCase, [88](#)
- getComments
  - Ecodena::Question::models::Question, [79](#)
- getCommentText
  - Ecodena::Comment::models::Comment, [36](#)
- getCompilerVersion
  - Ecodena::Compiler::models::Compiler, [42](#)
- getCompilerVersionOfAttempt
  - Ecodena::Attempt::models::Attempt, [29](#)

getDOB	Ecodena::User::models::Profile, 64	getQuestionText	Ecodena::Question::models::Question, 79
getErrorMessage	Ecodena::Attempt::models::ErrorReport, 48	getQuestionTitle	Ecodena::Question::models::Question, 79
getErrorReportID	Ecodena::Attempt::models::Attempt, 29	getReplyTime	Ecodena::Query::models::Query, 73
getErrorType	Ecodena::Attempt::models::ErrorReport, 48	getReportingUserID	Ecodena::Comment::models::Comment, 36
getGender	Ecodena::User::models::Profile, 64	getSolution	Ecodena::Attempt::models::Attempt, 29
getLanguage	Ecodena::Compiler::models::Compiler, 42	Ecodena::Query::models::Query, 73	
	Ecodena::Compiler::models::CompilerVersion, 45	getTestCaseLevel	Ecodena::Attempt::models::ErrorReport, 48
getLanguageName	Ecodena::Compiler::models::Language, 53	getTestCasesList	Ecodena::Question::models::Question, 79
getLevel	Ecodena::User::models::Programmer, 70	getTestID	Ecodena::Question::models::TestCase, 89
getLevelName	Ecodena::Question::models::Level, 55	getTimeOfSubmission	Ecodena::Attempt::models::Attempt, 29
getMemory	Ecodena::Attempt::models::ErrorReport, 48	getTimeRequirement	Ecodena::Attempt::models::ErrorReport, 48
getNumberOfAttempts	Ecodena::User::models::Profile, 65	getTimeStamp	Ecodena::Comment::models::Comment, 36
getOutput	Ecodena::Question::models::TestCase, 88	getUserID	Ecodena::Attempt::models::Attempt, 29
getQueryID	Ecodena::Query::models::Query, 72	Ecodena::Comment::models::Comment, 36	
getQueryText	Ecodena::Query::models::Query, 72	Ecodena::Query::models::Query, 73	
getQueryTime	Ecodena::Query::models::Query, 73	Ecodena::User::models::Profile, 65	
getQuestion	Ecodena::Compiler::models::Compiler, 42	getVersionName	Ecodena::Compiler::models::CompilerVersion, 45
	Ecodena::Question::models::TestCase, 89	home	Ecodena::Main::views, 15
getQuestionID	Ecodena::Attempt::models::Attempt, 29	inlines1	Ecodena::Question::admin::QuestionAdmin, 82
	Ecodena::Comment::models::Comment, 36	inlines2	Ecodena::Question::admin::QuestionAdmin, 82
	Ecodena::Question::models::Question, 79		

input\_f  
     Ecodena::Question::models::TestCase, 90  
 INSTALLED\_APPS  
     Ecodena::settings, 20  
 isAdmin  
     Ecodena::User::models::Profile, 65  
 isAdmin\_f  
     Ecodena::User::models::Profile, 66  
 isCorrect  
     Ecodena::Attempt::models::Attempt, 29  
 isProgrammer  
     Ecodena::User::models::Profile, 65  
 isProgrammer\_f  
     Ecodena::User::models::Profile, 66  
 isReported  
     Ecodena::Comment::models::Comment, 36, 39  
 isReported\_f  
     Ecodena::Comment::models::Comment, 38  
 JavaCompile110  
     Ecodena::Compiler::views, 14  
 language  
     Ecodena::Compiler::models::Compiler, 43  
     Ecodena::Compiler::models::CompilerVersion, 46  
 LANGUAGE\_CODE  
     Ecodena::settings, 20  
 language\_f  
     Ecodena::Compiler::models::CompilerVersion, 45  
 languageID\_f  
     Ecodena::Compiler::models::Language, 54  
 languageName  
     Ecodena::Compiler::models::CompilerVersion, 45  
     Ecodena::Compiler::models::Language, 54  
 languageName\_f  
     Ecodena::Compiler::models::Language, 54  
 level  
     Ecodena::User::models::Programmer, 71  
 level\_f  
     Ecodena::Question::models::Question, 80  
     Ecodena::User::models::Programmer, 70  
 levelID\_f  
     Ecodena::Question::models::Level, 56  
 levelName  
     Ecodena::Question::models::Level, 56  
 levelName\_f  
     Ecodena::Question::models::Level, 56  
 list\_display  
     Ecodena::Attempt::admin::AttemptAdmin, 33  
     Ecodena::Attempt::admin::ErrorReportAdmin, 52  
     Ecodena::Comment::admin::CommentAdmin, 40  
     Ecodena::Compiler::admin::CompilerVersionAdmin, 47  
     Ecodena::Compiler::admin::LanguageAdmin, 54  
     Ecodena::Query::admin::QueryAdmin, 77  
     Ecodena::Question::admin::QuestionAdmin, 82  
     Ecodena::User::admin::ProfileAdmin, 68  
 listQuestions  
     Ecodena::Question::views, 18  
 listSolutions  
     Ecodena::Attempt::views, 11  
 LOGGING  
     Ecodena::settings, 20  
 MANAGERS  
     Ecodena::settings, 21  
 MEDIA\_ROOT  
     Ecodena::settings, 21  
 MEDIA\_URL  
     Ecodena::settings, 21  
 memory  
     Ecodena::Attempt::models::ErrorReport, 51  
     Ecodena::Attempt::models::ErrorReport, 50  
 MIDDLEWARE\_CLASSES  
     Ecodena::settings, 21  
 model

Ecodena::Question::admin::LevelInline, 57  
 Ecodena::Question::admin::TypeInline, 92  
 numberOfAttempts  
     Ecodena::User::models::Profile, 67  
 numberOfAttempts\_f  
     Ecodena::User::models::Profile, 66  
 output  
     Ecodena::Question::models::TestCase, 91  
 output\_f  
     Ecodena::Question::models::TestCase, 90  
 postQuery  
     Ecodena::Query::views, 16  
 profileID\_f  
     Ecodena::User::models::Profile, 67  
 queryID  
     Ecodena::Query::models::Query, 75  
 queryID\_f  
     Ecodena::Query::models::Query, 74  
 queryText  
     Ecodena::Query::models::Query, 76  
 queryText\_f  
     Ecodena::Query::models::Query, 74  
 queryTime  
     Ecodena::Query::models::Query, 76  
 queryTime\_f  
     Ecodena::Query::models::Query, 74  
 question  
     Ecodena::Compiler::models::Compiler, 43  
 questionID  
     Ecodena::Attempt::models::Attempt, 32  
     Ecodena::Comment::models::Comment, 39  
     Ecodena::Question::models::Question, 81  
     Ecodena::Question::models::TestCase, 91  
 questionID\_f  
     Ecodena::Attempt::models::Attempt, 31  
     Ecodena::Comment::models::Comment, 38  
     Ecodena::Question::models::Question, 80  
     Ecodena::Question::models::TestCase, 90  
     questionText\_f, 80  
     questionTitle, 82  
     questionTitle\_f, 81  
 removeTestCase  
     Ecodena::Question::models::Question, 79  
 replyTime  
     Ecodena::Query::models::Query, 76  
 replyTime\_f  
     Ecodena::Query::models::Query, 75  
 report  
     Ecodena::Comment::models::Comment, 37  
 reportingUserID  
     Ecodena::Comment::models::Comment, 39  
 reportingUserID\_f  
     Ecodena::Comment::models::Comment, 38  
 ROOT\_URLCONF  
     Ecodena::settings, 21  
 SECRET\_KEY  
     Ecodena::settings, 21  
 setAddress  
     Ecodena::User::models::Profile, 65  
 setAdminID  
     Ecodena::Query::models::Query, 73  
 setAttempt  
     Ecodena::Compiler::models::Compiler, 42  
 setAttemptID  
     Ecodena::Attempt::models::Attempt, 29  
 setCaseType  
     Ecodena::Question::models::TestCase, 89  
 setCommentList



Ecodena::Question::models::Question, 79	Ecodena::Compiler::models::Compiler, 43
setCommentText Ecodena::Comment::models::Comment, 37	Ecodena::Question::models::TestCase, 89
setCompilerVersion Ecodena::Compiler::models::Compiler, 42	setQuestionID Ecodena::Attempt::models::Attempt, 30
setCompilerVersionOfAttempt Ecodena::Attempt::models::Attempt, 29	Ecodena::Comment::models::Comment, 37
setDOB Ecodena::User::models::Profile, 65	Ecodena::Question::models::Question, 79
setErrorMessage Ecodena::Attempt::models::ErrorReport, 49	setQuestionText Ecodena::Question::models::Question, 79
setErrorReportID Ecodena::Attempt::models::Attempt, 29	setQuestionTitle Ecodena::Question::models::Question, 80
setErrorType Ecodena::Attempt::models::ErrorReport, 49	setReplyTime Ecodena::Query::models::Query, 73
setGender Ecodena::User::models::Profile, 65	setReportingUserID Ecodena::Comment::models::Comment, 37
setLanguage Ecodena::Compiler::models::Compiler, 43	setSolution Ecodena::Attempt::models::Attempt, 30
Ecodena::Compiler::models::CompilerVersion, 45	Ecodena::Query::models::Query, 73
setLanguageName Ecodena::Compiler::models::Language, 53	setStatus Ecodena::Attempt::models::Attempt, 30
setLevel Ecodena::User::models::Programmer, 70	setTestCaseLevel Ecodena::Attempt::models::ErrorReport, 49
setLevelName Ecodena::Question::models::Level, 56	setTestCasesList Ecodena::Question::models::Question, 80
setMemory Ecodena::Attempt::models::ErrorReport, 49	setTestID Ecodena::Question::models::TestCase, 89
setNumberOfAttempts Ecodena::User::models::Profile, 65	setTimeOfSubmission Ecodena::Attempt::models::Attempt, 30
setOutput Ecodena::Question::models::TestCase, 89	setTimeRequirement Ecodena::Attempt::models::ErrorReport, 49
setQueryID Ecodena::Query::models::Query, 73	setTimeStamp Ecodena::Comment::models::Comment, 37
setQueryText Ecodena::Query::models::Query, 73	setUserID Ecodena::Attempt::models::Attempt, 30
setQueryTime Ecodena::Query::models::Query, 73	Ecodena::Comment::models::Comment, 37
setQuestion	Ecodena::Query::models::Query, 74
	setVersionName

Ecodena::Compiler::models::CompilerVersion, 45  
 Ecodena::Attempt::models::ErrorReport, 50  
 SITE\_ID  
     Ecodena::settings, 21  
 solution  
     Ecodena::Query::models::Query, 76  
 solution\_f  
     Ecodena::Attempt::models::Attempt, 31  
     Ecodena::Query::models::Query, 75  
 solutionText  
     Ecodena::Attempt::models::Attempt, 32  
 STATIC\_ROOT  
     Ecodena::settings, 22  
 STATIC\_URL  
     Ecodena::settings, 22  
 STATICFILES\_DIRS  
     Ecodena::settings, 22  
 STATICFILES\_FINDERS  
     Ecodena::settings, 22  
 status  
     Ecodena::Attempt::models::Attempt, 33  
 status\_f  
     Ecodena::Attempt::models::Attempt, 31  
 submitSolution  
     Ecodena::Attempt::views, 11  
 TEMPLATE\_DEBUG  
     Ecodena::settings, 22  
 TEMPLATE\_DIRS  
     Ecodena::settings, 22  
 TEMPLATE\_LOADERS  
     Ecodena::settings, 23  
 test  
     Ecodena::Question::views, 18  
 test\_basic\_addition  
     Ecodena::Attempt::tests::SimpleTest, 86  
     Ecodena::Comment::tests::SimpleTest, 85  
     Ecodena::Compiler::tests::SimpleTest, 84  
     Ecodena::Main::tests::SimpleTest, 84  
     Ecodena::Query::tests::SimpleTest, 83  
     Ecodena::Question::tests::SimpleTest, 85  
     Ecodena::User::tests::SimpleTest, 86  
 testCaseLevel  
     Ecodena::Attempt::models::ErrorReport, 51  
 testCaseLevel\_f  
     Ecodena::Attempt::models::ErrorReport, 50  
     testCasesList  
         Ecodena::Question::models::Question, 82  
     testCasesList\_f  
         Ecodena::Question::models::Question, 81  
     text  
         Ecodena::Attempt::views::SolutionForm, 87  
         Ecodena::Query::views::QueryForm, 77  
         Ecodena::Question::views::CommentForm, 41  
 TIME\_ZONE  
     Ecodena::settings, 23  
 timeLimit\_f  
     Ecodena::Question::models::Question, 81  
 timeOfSubmission  
     Ecodena::Attempt::models::Attempt, 33  
 timeOfSubmission\_f  
     Ecodena::Attempt::models::Attempt, 31, 32  
 timeRequirement  
     Ecodena::Attempt::models::ErrorReport, 51  
 timeRequirement\_f  
     Ecodena::Attempt::models::ErrorReport, 50  
 timeStamp  
     Ecodena::Comment::models::Comment, 39  
 timeStamp\_f  
     Ecodena::Comment::models::Comment, 38  
 type\_f  
     Ecodena::Question::models::Question, 81  
 typeID\_f  
     Ecodena::Question::models::Type, 92  
 typeName\_f  
     Ecodena::Question::models::Type, 92  
 urlpatterns  
     Ecodena::urls, 23  
 USE\_I18N  
     Ecodena::settings, 23  
 USE\_L10N  
     Ecodena::settings, 23

userID  
     Ecodena::Attempt::models::Attempt, [33](#)  
     Ecodena::Comment::models::Comment, [39](#)  
     Ecodena::Query::models::Query, [76](#)  
 userID\_f  
     Ecodena::Attempt::models::Attempt, [32](#)  
     Ecodena::Comment::models::Comment, [39](#)  
     Ecodena::Query::models::Query, [75](#)  
     Ecodena::User::models::Profile, [67](#)  
 username  
     Ecodena::User::views::ProfileForm, [69](#)  
 verbose\_name  
     Ecodena::Attempt::models::Attempt::Meta, [62](#)  
     Ecodena::Attempt::models::ErrorReport::Meta, [60](#)  
     Ecodena::Comment::models::Comment::Meta, [62](#)  
     Ecodena::Compiler::models::CompilerVersion::Meta, [59](#)  
     Ecodena::Compiler::models::Language::Meta, [58](#)  
     Ecodena::Query::models::Query::Meta, [63](#)  
     Ecodena::Question::models::Level::Meta, [57](#)  
     Ecodena::Question::models::Question::Meta, [60](#)  
     Ecodena::Question::models::TestCase::Meta, [61](#)  
     Ecodena::Question::models::Type::Meta, [58](#)  
 verbose\_name\_plural  
     Ecodena::Attempt::models::Attempt::Meta, [62](#)  
     Ecodena::Attempt::models::ErrorReport::Meta, [60](#)  
     Ecodena::Comment::models::Comment::Meta, [62](#)  
     Ecodena::Compiler::models::CompilerVersion::Meta, [59](#)  
     Ecodena::Compiler::models::Language::Meta, [58](#)  
     Ecodena::Query::models::Query::Meta, [63](#)  
     Ecodena::Question::models::Level::Meta, [57](#)  
     Ecodena::Question::models::Question::Meta, [60](#)  
     Ecodena::Question::models::TestCase::Meta, [61](#)  
     Ecodena::Question::models::Type::Meta, [58](#)  
     Ecodena::Attempt::views::SolutionForm, [87](#)  
     Ecodena::Compiler::models::CompilerVersion, [45](#)  
     Ecodena::Compiler::models::CompilerVersion, [46](#)  
     Ecodena::Compiler::models::CompilerVersion, [46](#)  
     Ecodena::User::views, [25](#)  
     Ecodena::Question::views, [18](#)  
     Ecodena::Attempt::views, [11](#)