

---

# **chaste***codegenDocumentation*

***Release 0.10.3***

**chaste***codegenauthors*

**Oct 10, 2023**



# CONTENTS

<b>1</b>	<b>Updating SymPy or other python packages</b>	<b>3</b>
<b>2</b>	<b>Updating the ontology and including it in chaste_codegen</b>	<b>5</b>
<b>3</b>	<b>Doing a new chaste_codegen release</b>	<b>7</b>
<b>4</b>	<b>API documentation</b>	<b>9</b>
4.1	chaste_codegen Package . . . . .	9
<b>Python Module Index</b>		<b>23</b>
<b>Index</b>		<b>25</b>



chaste\_codegen is hosted on [GitHub](#), where you can find the code and installation instructions.



## UPDATING SYMPY OR OTHER PYTHON PACKAGES

Sympy or any other python package may need to be updated, especially as python versions evolve. To update the version:

- change the version listed in `setup.py`, e.g. for sympy it currently lists `'sympy>=1.9, <1.11'`, which means that the version is at least 1.9 and is less than 1.11.
- update `dev-requirements/dev.txt` if you want to also update your development pinned (fixed) versions
- create a new branch `git checkout -b <name_of_new_branch>`
- `git add`, `git commit` and `git push` the changes
- make a pull request. The tests may throw up some errors that may need fixing. The tests are in the `tests` folder and the reference data in `data/tests`. In `data/tests/chaste_reference_models` you'll see a few reference files ending in `.cpp_python36`, This is as due to sympy versions supported python 3.6 leads to an equivalent but subtly different generated model.
- Mention the changes made in the release notes `release.txt`
- To use the changes with chaste, do a new release of `chaste_codegen`.





## UPDATING THE ONTOLOGY AND INCLUDING IT IN CHASTE\_CODEGEN

- update the ontology according to the instructions in <https://github.com/ModellingWebLab/ontologies>
- the ontology is included in `chaste_codegen` via a submodule, update this with `git submodule update --remote chaste_codegen/ontologies`.
- *Please note* this same way can be used to update the `cellml` in `chaste_codegen/data/tests/cellml` submodule if required.
- create a new branch `git checkout -b <name_of_new_branch>`
- `git add`, `git commit` and `git push` the changes
- make a pull run the tests and fix any issues that arise
- update the release notes `release.txt` with information about the updated ontology.
- To use the changes with `chaste`, do a new release of `chaste_codegen`.



## DOING A NEW CHASTE\_CODEGEN RELEASE

- Update the release version number in `chaste_codegen/version.txt`.
- Update the release notes `release.txt` with the latest release number.
- For this version number: minor numbers will be picked up by chaste automatically, for major version numbers, `chaste_codegen.txt` will need updating in the chaste repository.
- Follow the following tutorial to publish the package: <https://packaging.python.org/en/latest/tutorials/packaging-projects/>
- You will need a login to pypi.org and the account you are using will need access to `chaste_codegen`.



## API DOCUMENTATION

### 4.1 chaste\_codegen Package

Main module for cardiac Chaste code generation

#### 4.1.1 Functions

---

*add\_conversions*(model[, use\_modifiers, ...])

---

*load\_model\_with\_conversions*(model\_file[, ...])

---

<i>load_template</i> (*name)	Loads a template from the local template directory.
------------------------------	---

---

<i>subs_math_func_placeholders</i> (expr)	Substitutes the placeholder math functions in expr for their corresponding Sympy functions :param expr: sympy expression
---	--

---

<i>version</i> ([formatted])	Returns the version number, as a 3-part integer (major, minor, revision).
------------------------------	---

---

#### **add\_conversions**

`chaste_codegen.add_conversions(model, use_modifiers=True, skip_chaste_stimulus_conversion=False)`

#### **load\_model\_with\_conversions**

`chaste_codegen.load_model_with_conversions(model_file, use_modifiers=False, quiet=False, skip_singularity_fixes=False, skip_conversions=False)`

## load\_template

`chaste_codegen.load_template(*name)`

Loads a template from the local template directory.

Templates can be specified as a single filename, e.g. `load_template('temp.txt')`, or loaded from subdirectories using e.g. `load_template('subdir_1', 'subdir_2', 'file.txt')`.

## subs\_math\_func\_placeholders

`chaste_codegen.subs_math_func_placeholders(expr)`

Substitutes the placeholder math functions in `expr` for their corresponding Sympy functions :param `expr`: sympy expression

Example: `>> str(expr) '2.0 * exp(V)' >> subs_math_func_placeholders(expr) '2.0 * exp(V)'`

### Returns

`expr` with all placeholder functions replaced by sympy functions.

## version

`chaste_codegen.version(formatted=False)`

Returns the version number, as a 3-part integer (major, minor, revision). If `formatted=True`, it returns a string formatted version (e.g. "chaste\_codegen 1.0.0").



## 4.1.2 Classes

<i>BackwardEulerModel</i> (model, file_name, **kwargs)	Holds template and information specific for the Backwards Euler model type
<i>BackwardEulerOptModel</i> (model, file_name, **kwargs)	Holds information specific for the Optimised Backward Euler model type.
<i>ChasteModel</i> (model, file_name, **kwargs)	Holds information about a cellml model for which chaste code is to be generated.
<i>ChastePrinter</i> ([symbol_function, ...])	Converts Sympy expressions to strings for use in Chaste code generation.
<i>CodegenError</i>	
<i>CvodeChasteModel</i> (model, file_name, **kwargs)	Holds template and information specific for the CVODE model type
<i>GeneralisedRushLarsenFirstOrderModel</i> (model, ...)	Holds template and information specific for the GeneralisedRushLarsen model type
<i>GeneralisedRushLarsenFirstOrderModelOpt</i> (...)	Holds template and information specific for the GeneralisedRushLarsenOpt model type
<i>GeneralisedRushLarsenSecondOrderModel</i> (model, ...)	Holds template and information specific for the GeneralisedRushLarsen model type
<i>GeneralisedRushLarsenSecondOrderModelOpt</i> (...)	Holds template and information specific for the GeneralisedRushLarsenOpt model type
<i>LabviewPrinter</i> ([symbol_function, ...])	Converts Sympy expressions to strings for use in Chaste code generation.
<i>NormalChasteModel</i> (model, file_name, **kwargs)	Holds template and information specific for the Normal model type
<i>OptChasteModel</i> (model, file_name, **kwargs)	Holds information specific for the Optimised model type.
<i>OptCvodeChasteModel</i> (model, file_name, **kwargs)	Holds information specific for the Cvode Optimised model type.
<i>RealFunction</i> (*args)	
<i>RushLarsenC</i> (model, file_name, **kwargs)	Holds template and information specific for the Rush-Larsen model type
<i>RushLarsenLabview</i> (model, file_name, **kwargs)	Holds template and information specific for the Rush-Larsen model type
<i>RushLarsenModel</i> (model, file_name, **kwargs)	Holds template and information specific for the Rush-Larsen model type
<i>RushLarsenOptModel</i> (model, file_name, **kwargs)	Holds template and information specific for the Rush-Larsen model type
<i>Transpiler</i> ([symbol_generator, number_generator]) <i>abs_</i> (*args)	Handles conversion of MathmL to Sympy exprerssions.
<i>acos_</i> (*args)	
<i>cos_</i> (*args)	
<i>exp_</i> (*args)	
<i>sin_</i> (*args)	
<i>sqrt_</i> (*args)	



## BackwardEulerModel

**class** chaste\_codegen.**BackwardEulerModel**(*model, file\_name, \*\*kwargs*)

Bases: [ChasteModel](#)

Holds template and information specific for the Backwards Euler model type

### Methods Summary

<i>format_derivative_equation</i> (eq, ...)	Format an individual derivative equation specified so that other model types can specify more detailed printing
<i>format_jacobian</i> ()	Format the jacobian to allow opt model to update what belongs were
<i>format_linear_deriv_eqs</i> (linear_deriv_eqs)	Format linear derivative equations belonging, to update what belongs were
<i>format_nonlinear_state_vars</i> ()	
<i>format_residual_equations</i> ()	Update the state vars, savings residual and jacobian info for outputing

### Methods Documentation

**format\_derivative\_equation**(*eq, modifiers\_with\_defining\_eqs*)

Format an individual derivative equation specified so that other model types can specify more detailed printing

**format\_jacobian**()

Format the jacobian to allow opt model to update what belongs were

**format\_linear\_deriv\_eqs**(*linear\_deriv\_eqs*)

Format linear derivative equations belonging, to update what belongs were

**format\_nonlinear\_state\_vars**()

**format\_residual\_equations**()

Update the state vars, savings residual and jacobian info for outputing

**format\_derivative\_equation**(*eq, modifiers\_with\_defining\_eqs*)

Format an individual derivative equation specified so that other model types can specify more detailed printing

**format\_linear\_deriv\_eqs**(*linear\_deriv\_eqs*)

Format linear derivative equations belonging, to update what belongs were

**format\_residual\_equations**()

Update the state vars, savings residual and jacobian info for outputing

**format\_jacobian**()

Format the jacobian to allow opt model to update what belongs were

## BackwardEulerOptModel

**class** chaste\_codegen.**BackwardEulerOptModel**(*model*, *file\_name*, *\*\*kwargs*)

Bases: *BackwardEulerModel*

Holds information specific for the Optimised Backward Euler model type.

### Methods Summary

---

<i>format_jacobian()</i>	Format the jacobian to update what belongs were
<i>format_linear_deriv_eqs</i> ( <i>linear_deriv_eqs</i> )	Format linear derivative equations belonging, to update what belongs were

---

### Methods Documentation

**format\_jacobian()**

Format the jacobian to update what belongs were

**format\_linear\_deriv\_eqs**(*linear\_deriv\_eqs*)

Format linear derivative equations belonging, to update what belongs were

**format\_linear\_deriv\_eqs**(*linear\_deriv\_eqs*)

Format linear derivative equations belonging, to update what belongs were

**format\_jacobian()**

Format the jacobian to update what belongs were

## ChasteModel

**class** chaste\_codegen.**ChasteModel**(*model*, *file\_name*, *\*\*kwargs*)

Bases: *object*

Holds information about a cellml model for which chaste code is to be generated.

It also holds relevant formatted equations and derivatives. Please Note: this calass cannot generate chaste code directly, instead use a subclass of the model type

### Attributes Summary

---

<i>DEFAULT_EXTENSIONS</i>
---------------------------

---

## Methods Summary

<code>format_derivative_equation(eq, ...)</code>	Format an individual derivative equation specified so that other model types can specify more detailed printing
<code>generate_chaste_code()</code>	Generates and stores chaste code

## Attributes Documentation

**DEFAULT\_EXTENSIONS** = ('.hpp', '.cpp')

## Methods Documentation

**format\_derivative\_equation**(*eq*, *modifiers\_with\_defining\_eqs*)

Format an individual derivative equation specified so that other model types can specify more detailed printing

**generate\_chaste\_code**()

Generates and stores chaste code

**format\_derivative\_equation**(*eq*, *modifiers\_with\_defining\_eqs*)

Format an individual derivative equation specified so that other model types can specify more detailed printing

**generate\_chaste\_code**()

Generates and stores chaste code

## ChastePrinter

**class** chaste\_codegen.**ChastePrinter**(*symbol\_function=None*, *derivative\_function=None*, *lookup\_table\_function=<function ChastePrinter.<lambda>>>*)

Bases: [Printer](#)

Converts Sympy expressions to strings for use in Chaste code generation.

To use, create a [ChastePrinter](#) instance, and call its method `doprint()` with a Sympy expression argument.

Arguments:

**symbol\_function**

A function that converts symbols to strings (variable names).

**derivative\_function**

A function that converts derivatives to strings.

**lookup\_table\_function**

A function that prints lookup table expressions or returns None if the expression is not in the lookup table.

## CodegenError

**exception** chaste\_codegen.CodegenError

Bases: [Exception](#)

## CvodeChasteModel

**class** chaste\_codegen.CvodeChasteModel(model, file\_name, \*\*kwargs)

Bases: [ChasteModel](#)

Holds template and information specific for the CVODE model type

## GeneralisedRushLarsenFirstOrderModel

**class** chaste\_codegen.GeneralisedRushLarsenFirstOrderModel(model, file\_name, \*\*kwargs)

Bases: [ChasteModel](#)

Holds template and information specific for the GeneralisedRushLarsen model type

## Methods Summary

---

<a href="#">eq_in_evaluate_partial_derivative</a> (eq, ...)	Indicate if the lhs of equation eq appears in used_jacobian_vars specified here so derived model types can specify in detail what happens here
<a href="#">eq_in_evaluate_y_derivative</a> (eq, used_equations)	Indicate if the lhs of equation eq appears in used_equations specified here so derived model types can specify in detail what happens here

---

## Methods Documentation

**eq\_in\_evaluate\_partial\_derivative**(eq, used\_jacobian\_vars)

Indicate if the lhs of equation eq appears in used\_jacobian\_vars specified here so derived model types can specify in detail what happens here

**eq\_in\_evaluate\_y\_derivative**(eq, used\_equations)

Indicate if the lhs of equation eq appears in used\_equations specified here so derived model types can specify in detail what happens here

**eq\_in\_evaluate\_y\_derivative**(eq, used\_equations)

Indicate if the lhs of equation eq appears in used\_equations specified here so derived model types can specify in detail what happens here

**eq\_in\_evaluate\_partial\_derivative**(eq, used\_jacobian\_vars)

Indicate if the lhs of equation eq appears in used\_jacobian\_vars specified here so derived model types can specify in detail what happens here

## GeneralisedRushLarsenFirstOrderModelOpt

**class** chaste\_codegen.GeneralisedRushLarsenFirstOrderModelOpt(*model, file\_name, \*\*kwargs*)

Bases: [GeneralisedRushLarsenFirstOrderModel](#)

Holds template and information specific for the GeneralisedRushLarsenOpt model type

### Methods Summary

<a href="#">eq_in_evaluate_partial_derivative</a> (eq, ...)	Indicate if the lhs of equation eq appears in used_jacobian_vars
<a href="#">eq_in_evaluate_y_derivative</a> (eq, used_equations)	Indicate if the lhs of equation eq appears in used_equations
<a href="#">format_derivative_equation</a> (eq, ...)	Format an individual derivative equation

### Methods Documentation

**eq\_in\_evaluate\_partial\_derivative**(*eq, used\_jacobian\_vars*)

Indicate if the lhs of equation eq appears in used\_jacobian\_vars

**eq\_in\_evaluate\_y\_derivative**(*eq, used\_equations*)

Indicate if the lhs of equation eq appears in used\_equations

**format\_derivative\_equation**(*eq, modifiers\_with\_defining\_eqs*)

Format an individual derivative equation

**format\_derivative\_equation**(*eq, modifiers\_with\_defining\_eqs*)

Format an individual derivative equation

**eq\_in\_evaluate\_y\_derivative**(*eq, used\_equations*)

Indicate if the lhs of equation eq appears in used\_equations

**eq\_in\_evaluate\_partial\_derivative**(*eq, used\_jacobian\_vars*)

Indicate if the lhs of equation eq appears in used\_jacobian\_vars

## GeneralisedRushLarsenSecondOrderModel

**class** chaste\_codegen.GeneralisedRushLarsenSecondOrderModel(*model, file\_name, \*\*kwargs*)

Bases: [GeneralisedRushLarsenFirstOrderModel](#)

Holds template and information specific for the GeneralisedRushLarsen model type

## GeneralisedRushLarsenSecondOrderModelOpt

**class** chaste\_codegen.GeneralisedRushLarsenSecondOrderModelOpt(*model, file\_name, \*\*kwargs*)

Bases: [GeneralisedRushLarsenFirstOrderModelOpt](#)

Holds template and information specific for the GeneralisedRushLarsenOpt model type

## LabviewPrinter

```
class chaste_codegen.LabviewPrinter(symbol_function=None, derivative_function=None,  
                                   lookup_table_function=<function ChastePrinter.<lambda>>)
```

Bases: [ChastePrinter](#)

Converts Sympy expressions to strings for use in Chaste code generation.

To use, create a [ChastePrinter](#) instance, and call its method `doprint()` with a Sympy expression argument.

Arguments:

**symbol\_function**

A function that converts symbols to strings (variable names).

**derivative\_function**

A function that converts derivatives to strings.

**lookup\_table\_function**

A function that prints lookup table expressions or returns None if the expression is not in the lookup table.

## NormalChasteModel

```
class chaste_codegen.NormalChasteModel(model, file_name, **kwargs)
```

Bases: [ChasteModel](#)

Holds template and information specific for the Normal model type

## OptChasteModel

```
class chaste_codegen.OptChasteModel(model, file_name, **kwargs)
```

Bases: [NormalChasteModel](#)

Holds information specific for the Optimised model type. Builds on Normal model type

## OptCvodeChasteModel

```
class chaste_codegen.OptCvodeChasteModel(model, file_name, **kwargs)
```

Bases: [CvodeChasteModel](#)

Holds information specific for the Cvode Optimised model type. Builds on Cvode model type

## RealFunction

```
class chaste_codegen.RealFunction(*args)
```

Bases: `Function`

## RushLarsenC

**class** chaste\_codegen.**RushLarsenC**(*model*, *file\_name*, **\*\*kwargs**)

Bases: [RushLarsenModel](#)

Holds template and information specific for the RushLarsen model type

### Attributes Summary

---

*DEFAULT\_EXTENSIONS*

---

### Methods Summary

---

<i>format_derivative_equation</i> ( <i>eq</i> , ...)	Format an individual derivative equation specified so that other model types can specify more detailed printing
--	---

---

### Attributes Documentation

**DEFAULT\_EXTENSIONS** = ('.h', '.c')

### Methods Documentation

**format\_derivative\_equation**(*eq*, *modifiers\_with\_defining\_eqs*)

Format an individual derivative equation specified so that other model types can specify more detailed printing

**format\_derivative\_equation**(*eq*, *modifiers\_with\_defining\_eqs*)

Format an individual derivative equation specified so that other model types can specify more detailed printing

## RushLarsenLabview

**class** chaste\_codegen.**RushLarsenLabview**(*model*, *file\_name*, **\*\*kwargs**)

Bases: [RushLarsenC](#)

Holds template and information specific for the RushLarsen model type

## Attributes Summary

---

*DEFAULT\_EXTENSIONS*

---

## Attributes Documentation

**DEFAULT\_EXTENSIONS** = (None, '.txt')

## RushLarsenModel

**class** chaste\_codegen.**RushLarsenModel**(*model, file\_name, \*\*kwargs*)

Bases: *ChasteModel*

Holds template and information specific for the RushLarsen model type

## Methods Summary

---

<i>format_deriv_eqs_EvaluateEquations(...)</i>	Format derivative equations belonging to EvaluateEquations, to allow opt model to update what belongs were
--	--

---

## Methods Documentation

**format\_deriv\_eqs\_EvaluateEquations**(*deriv\_eqs\_EvaluateEquations*)

Format derivative equations belonging to EvaluateEquations, to allow opt model to update what belongs were

**format\_deriv\_eqs\_EvaluateEquations**(*deriv\_eqs\_EvaluateEquations*)

Format derivative equations belonging to EvaluateEquations, to allow opt model to update what belongs were

## RushLarsenOptModel

**class** chaste\_codegen.**RushLarsenOptModel**(*model, file\_name, \*\*kwargs*)

Bases: *RushLarsenModel*

Holds template and information specific for the RushLarsen model type



## Methods Summary

<i>format_deriv_eqs_EvaluateEquations(...)</i>	Format derivative equations belonging to EvaluateEquations, to update what equation belongs were
--	--

## Methods Documentation

**format\_deriv\_eqs\_EvaluateEquations**(*deriv\_eqs\_EvaluateEquations*)

Format derivative equations belonging to EvaluateEquations, to update what equation belongs were

**format\_deriv\_eqs\_EvaluateEquations**(*deriv\_eqs\_EvaluateEquations*)

Format derivative equations belonging to EvaluateEquations, to update what equation belongs were

### abs

**class** chaste\_codegen.**abs**\_\*args)

Bases: *RealFunction*

**fdiff**(*argindex=1*)

Returns the first derivative of this function.

### acos

**class** chaste\_codegen.**acos**\_\*args)

Bases: *RealFunction*

**fdiff**(*argindex=1*)

Returns the first derivative of this function.

### cos

**class** chaste\_codegen.**cos**\_\*args)

Bases: *RealFunction*

**fdiff**(*argindex=1*)

Returns the first derivative of this function.

### exp

**class** chaste\_codegen.**exp**\_\*args)

Bases: *RealFunction*

**fdiff**(*argindex=1*)

Returns the first derivative of this function.

## sin

**class** chaste\_codegen.**sin\_**(\*args)

Bases: *RealFunction*

**fdiff**(argindex=1)

Returns the first derivative of this function.

## sqrt

**class** chaste\_codegen.**sqrt\_**(\*args)

Bases: *RealFunction*

**fdiff**(argindex=1)

Returns the first derivative of this function.

## PYTHON MODULE INDEX

### C

chaste\_codegen, [9](#)



## A

`abs_` (class in `chaste_codegen`), 21  
`acos_` (class in `chaste_codegen`), 21  
`add_conversions()` (in module `chaste_codegen`), 9

## B

`BackwardEulerModel` (class in `chaste_codegen`), 13  
`BackwardEulerOptModel` (class in `chaste_codegen`), 14

## C

`chaste_codegen`  
 module, 9  
`ChasteModel` (class in `chaste_codegen`), 14  
`ChastePrinter` (class in `chaste_codegen`), 15  
`CodeGenError`, 16  
`cos_` (class in `chaste_codegen`), 21  
`CvodeChasteModel` (class in `chaste_codegen`), 16

## D

`DEFAULT_EXTENSIONS` (`chaste_codegen.ChasteModel`  
 attribute), 15  
`DEFAULT_EXTENSIONS` (`chaste_codegen.RushLarsenC`  
 attribute), 19  
`DEFAULT_EXTENSIONS` (`chaste_codegen.RushLarsenLabview`  
 attribute), 20

## E

`eq_in_evaluate_partial_derivative()`  
 (`chaste_codegen.GeneralisedRushLarsenFirstOrderModel`  
 method), 16  
`eq_in_evaluate_partial_derivative()`  
 (`chaste_codegen.GeneralisedRushLarsenFirstOrderModelOpt`  
 method), 17  
`eq_in_evaluate_y_derivative()`  
 (`chaste_codegen.GeneralisedRushLarsenFirstOrderModel`  
 method), 16  
`eq_in_evaluate_y_derivative()`  
 (`chaste_codegen.GeneralisedRushLarsenFirstOrderModelOpt`  
 method), 17  
`exp_` (class in `chaste_codegen`), 21

## F

`fdiff()` (`chaste_codegen.abs_` method), 21  
`fdiff()` (`chaste_codegen.acos_` method), 21  
`fdiff()` (`chaste_codegen.cos_` method), 21  
`fdiff()` (`chaste_codegen.exp_` method), 21  
`fdiff()` (`chaste_codegen.sin_` method), 22  
`fdiff()` (`chaste_codegen.sqrt_` method), 22  
`format_deriv_eqs_EvaluateEquations()`  
 (`chaste_codegen.RushLarsenModel` method),  
 20  
`format_deriv_eqs_EvaluateEquations()`  
 (`chaste_codegen.RushLarsenOptModel`  
 method), 21  
`format_derivative_equation()`  
 (`chaste_codegen.BackwardEulerModel`  
 method), 13  
`format_derivative_equation()`  
 (`chaste_codegen.ChasteModel` method),  
 15  
`format_derivative_equation()`  
 (`chaste_codegen.GeneralisedRushLarsenFirstOrderModelOpt`  
 method), 17  
`format_derivative_equation()`  
 (`chaste_codegen.RushLarsenC` method),  
 19  
`format_jacobian()` (`chaste_codegen.BackwardEulerModel`  
 method), 13  
`format_jacobian()` (`chaste_codegen.BackwardEulerOptModel`  
 method), 14  
`format_linear_deriv_eqs()`  
 (`chaste_codegen.BackwardEulerModel`  
 method), 13  
`format_linear_deriv_eqs()`  
 (`chaste_codegen.BackwardEulerOptModel`  
 method), 14  
`format_nonlinear_state_vars()`  
 (`chaste_codegen.BackwardEulerModel`  
 method), 13  
`format_residual_equations()`  
 (`chaste_codegen.BackwardEulerModel`  
 method), 13

## G

GeneralisedRushLarsenFirstOrderModel (class in *chaste\_codegen*), 16

GeneralisedRushLarsenFirstOrderModelOpt (class in *chaste\_codegen*), 17

GeneralisedRushLarsenSecondOrderModel (class in *chaste\_codegen*), 17

GeneralisedRushLarsenSecondOrderModelOpt (class in *chaste\_codegen*), 17

generate\_chaste\_code()  
(*chaste\_codegen.ChasteModel* method),  
15

## L

LabviewPrinter (class in *chaste\_codegen*), 18

load\_model\_with\_conversions() (in module *chaste\_codegen*), 9

load\_template() (in module *chaste\_codegen*), 10

## M

module  
*chaste\_codegen*, 9

## N

NormalChasteModel (class in *chaste\_codegen*), 18

## O

OptChasteModel (class in *chaste\_codegen*), 18

OptCvodeChasteModel (class in *chaste\_codegen*), 18

## R

RealFunction (class in *chaste\_codegen*), 18

RushLarsenC (class in *chaste\_codegen*), 19

RushLarsenLabview (class in *chaste\_codegen*), 19

RushLarsenModel (class in *chaste\_codegen*), 20

RushLarsenOptModel (class in *chaste\_codegen*), 20

## S

sin\_ (class in *chaste\_codegen*), 22

sqrt\_ (class in *chaste\_codegen*), 22

subs\_math\_func\_placeholders() (in module *chaste\_codegen*), 10

## V

version() (in module *chaste\_codegen*), 10