

# Solar Farm Calculator

v0.1

Generated by Doxygen 1.8.3.1

Tue Oct 8 2013 22:10:33



# Contents

<b>1</b>	<b>Solar Farm Calculator</b>	<b>1</b>
<b>2</b>	<b>Hierarchical 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>Class Documentation</b>	<b>7</b>
4.1	SolarCalculator.Assets.AC1Cable Class Reference . . . . .	7
4.1.1	Detailed Description . . . . .	7
4.1.2	Constructor & Destructor Documentation . . . . .	8
4.1.2.1	__init__ . . . . .	8
4.1.3	Member Function Documentation . . . . .	8
4.1.3.1	getDiameter . . . . .	8
4.1.3.2	getLength . . . . .	8
4.1.3.3	getMaterial . . . . .	8
4.1.3.4	getStrandNum . . . . .	8
4.2	SolarCalculator.Assets.AC2Cable Class Reference . . . . .	8
4.2.1	Detailed Description . . . . .	9
4.2.2	Constructor & Destructor Documentation . . . . .	9
4.2.2.1	__init__ . . . . .	9
4.2.3	Member Function Documentation . . . . .	9
4.2.3.1	getDiameter . . . . .	9
4.2.3.2	getLength . . . . .	9
4.2.3.3	getMaterial . . . . .	9
4.2.3.4	getStrandNum . . . . .	9
4.3	SolarCalculator.Utils.PyExchangeRates.AccessDataFailure Class Reference . . . . .	9
4.3.1	Detailed Description . . . . .	10
4.4	SolarCalculator.GUI.ApplicationFrame Class Reference . . . . .	10
4.4.1	Detailed Description . . . . .	13
4.5	SolarCalculator.Assets.Asset Class Reference . . . . .	14

4.5.1	Detailed Description	14
4.5.2	Constructor & Destructor Documentation	15
4.5.2.1	__init__	15
4.5.3	Member Function Documentation	15
4.5.3.1	getCost	15
4.5.3.2	getCurrency	15
4.5.3.3	getDepRate	15
4.5.3.4	getDepreciatedValue	15
4.6	SolarCalculator.Utils.PyExchangeRates.BadAppID Class Reference	15
4.6.1	Detailed Description	15
4.7	SolarCalculator.Assets.CircuitBreaker Class Reference	16
4.7.1	Detailed Description	16
4.7.2	Constructor & Destructor Documentation	16
4.7.2.1	__init__	16
4.8	SolarCalculator.Utils.ReverseGeocode.CountryNotFound Class Reference	16
4.8.1	Detailed Description	16
4.9	SolarCalculator.Utils.PyExchangeRates.Currency Class Reference	17
4.9.1	Detailed Description	17
4.9.2	Constructor & Destructor Documentation	17
4.9.2.1	__init__	17
4.10	SolarCalculator.GUI.DateError Class Reference	17
4.10.1	Detailed Description	18
4.11	SolarCalculator.Assets.DCCable Class Reference	18
4.11.1	Detailed Description	18
4.11.2	Constructor & Destructor Documentation	19
4.11.2.1	__init__	19
4.11.3	Member Function Documentation	19
4.11.3.1	getDiameter	19
4.11.3.2	getLength	19
4.11.3.3	getMaterial	19
4.12	main.DialogBox_DateError Class Reference	19
4.12.1	Constructor & Destructor Documentation	19
4.12.1.1	__init__	19
4.12.2	Member Function Documentation	20
4.12.2.1	evt_dialogOK_clicked	20
4.13	main.DialogBox_FatalError Class Reference	20
4.13.1	Constructor & Destructor Documentation	20
4.13.1.1	__init__	20
4.13.2	Member Function Documentation	20
4.13.2.1	evt_dialogCloseProgram_clicked	20

4.14	main.DialogBox_GeoCodeError Class Reference	20
4.14.1	Constructor & Destructor Documentation	21
4.14.1.1	__init__	21
4.14.2	Member Function Documentation	21
4.14.2.1	evt_dialogOK_clicked	21
4.15	main.DialogBox_IncompleteForm Class Reference	21
4.15.1	Constructor & Destructor Documentation	22
4.15.1.1	__init__	22
4.15.2	Member Function Documentation	22
4.15.2.1	evt_dialogOK_clicked	22
4.16	main.DialogBox_NoInternet Class Reference	22
4.16.1	Constructor & Destructor Documentation	22
4.16.1.1	__init__	22
4.16.2	Member Function Documentation	22
4.16.2.1	evt_dialogOK_clicked	22
4.17	main.DialogBox_ProgressDialog Class Reference	23
4.17.1	Constructor & Destructor Documentation	23
4.17.1.1	__init__	23
4.17.2	Member Function Documentation	23
4.17.2.1	closeDialog	23
4.17.2.2	update	23
4.18	main.DialogBox_SimulationResults Class Reference	23
4.18.1	Constructor & Destructor Documentation	24
4.18.1.1	__init__	24
4.18.2	Member Function Documentation	24
4.18.2.1	evt_dialogOK_clicked	24
4.19	SolarCalculator.Utils.PyExchangeRates.Exchange Class Reference	24
4.19.1	Detailed Description	25
4.19.2	Member Function Documentation	25
4.19.2.1	withdraw	25
4.20	SolarCalculator.GUI.FatalError Class Reference	25
4.20.1	Detailed Description	26
4.21	SolarCalculator.Assets.Financial Class Reference	26
4.21.1	Detailed Description	26
4.21.2	Constructor & Destructor Documentation	26
4.21.2.1	__init__	26
4.21.3	Member Function Documentation	27
4.21.3.1	accumulateDailyInterest	27
4.21.3.2	addToLoan	27
4.21.3.3	amountInBaseCurrency	27

4.21.3.4	<a href="#">getBaseCurrency</a>	27
4.21.3.5	<a href="#">getCurrentLoanValue</a>	27
4.21.3.6	<a href="#">getDailyMaintenance</a>	27
4.21.3.7	<a href="#">getPowerPrice</a>	27
4.21.3.8	<a href="#">makeLoanPayment</a>	27
4.22	<a href="#">SolarCalculator.GUI.GeoCodeError Class Reference</a>	27
4.22.1	<a href="#">Detailed Description</a>	28
4.23	<a href="#">SolarCalculator.GUI.IncompleteForm Class Reference</a>	28
4.23.1	<a href="#">Detailed Description</a>	29
4.24	<a href="#">main.InputField Class Reference</a>	29
4.24.1	<a href="#">Detailed Description</a>	29
4.24.2	<a href="#">Member Function Documentation</a>	30
4.24.2.1	<a href="#">setFieldValue</a>	30
4.24.2.2	<a href="#">setLabelColour</a>	30
4.24.2.3	<a href="#">validateField</a>	30
4.25	<a href="#">SolarCalculator.Utils.PyExchangeRates.InvalidCurrencyKey Class Reference</a>	30
4.25.1	<a href="#">Detailed Description</a>	30
4.26	<a href="#">SolarCalculator.Assets.Inverter Class Reference</a>	30
4.26.1	<a href="#">Detailed Description</a>	31
4.26.2	<a href="#">Constructor &amp; Destructor Documentation</a>	31
4.26.2.1	<a href="#">__init__</a>	31
4.26.3	<a href="#">Member Function Documentation</a>	31
4.26.3.1	<a href="#">getEfficiency</a>	31
4.26.3.2	<a href="#">getPowerFactor</a>	31
4.26.3.3	<a href="#">getVoltage</a>	31
4.27	<a href="#">SolarCalculator.Assets.Material Class Reference</a>	31
4.27.1	<a href="#">Detailed Description</a>	32
4.27.2	<a href="#">Constructor &amp; Destructor Documentation</a>	32
4.27.2.1	<a href="#">__init__</a>	32
4.27.3	<a href="#">Member Function Documentation</a>	32
4.27.3.1	<a href="#">getResistivity</a>	32
4.27.3.2	<a href="#">getTempCoefficient</a>	32
4.28	<a href="#">SolarCalculator.Utils.PyExchangeRates.Money Class Reference</a>	32
4.28.1	<a href="#">Detailed Description</a>	33
4.28.2	<a href="#">Member Function Documentation</a>	33
4.28.2.1	<a href="#">__add__</a>	33
4.28.2.2	<a href="#">__div__</a>	33
4.28.2.3	<a href="#">__mul__</a>	33
4.28.2.4	<a href="#">__str__</a>	33
4.28.2.5	<a href="#">__sub__</a>	33

4.28.2.6	convert	33
4.28.2.7	getAmount	34
4.28.2.8	getCurrencyKey	34
4.28.2.9	getExchange	34
4.28.2.10	setAmount	34
4.29	SolarCalculator.GUI.NoInternet Class Reference	34
4.29.1	Detailed Description	35
4.30	main.OptionalInputField Class Reference	35
4.30.1	Constructor & Destructor Documentation	35
4.30.1.1	__init__	35
4.30.2	Member Function Documentation	35
4.30.2.1	getCheckboxState	35
4.30.2.2	validateField	35
4.31	SolarCalculator.Assets.PVArray Class Reference	36
4.31.1	Detailed Description	36
4.31.2	Constructor & Destructor Documentation	36
4.31.2.1	__init__	36
4.31.3	Member Function Documentation	36
4.31.3.1	getAngle	36
4.31.3.2	getArea	37
4.31.3.3	getModuleNum	37
4.31.3.4	getModuleType	37
4.31.3.5	getVoltage	37
4.32	SolarCalculator.Assets.PVModule Class Reference	37
4.32.1	Detailed Description	38
4.32.2	Constructor & Destructor Documentation	38
4.32.2.1	__init__	38
4.32.3	Member Function Documentation	38
4.32.3.1	getArea	38
4.32.3.2	getPanelNum	38
4.32.3.3	getPanelType	38
4.32.3.4	getVoltage	38
4.33	SolarCalculator.Assets.PVPanel Class Reference	38
4.33.1	Detailed Description	39
4.33.2	Constructor & Destructor Documentation	39
4.33.2.1	__init__	39
4.33.3	Member Function Documentation	39
4.33.3.1	getArea	39
4.33.3.2	getDegradationRate	39
4.33.3.3	getEfficiency	39

4.33.3.4	<a href="#">getRating</a>	39
4.33.3.5	<a href="#">getVoltage</a>	40
4.34	<a href="#">SolarCalculator.Utils.PyExchangeRates.SimpleObjectEncoder Class Reference</a>	40
4.34.1	<a href="#">Detailed Description</a>	40
4.35	<a href="#">SolarCalculator.Simulation.Simulation Class Reference</a>	40
4.35.1	<a href="#">Detailed Description</a>	41
4.35.2	<a href="#">Constructor &amp; Destructor Documentation</a>	41
4.35.2.1	<a href="#">__init__</a>	41
4.35.3	<a href="#">Member Function Documentation</a>	41
4.35.3.1	<a href="#">getFinancialResults</a>	41
4.35.3.2	<a href="#">getPowerProgress</a>	41
4.35.3.3	<a href="#">getPowerResults</a>	41
4.35.3.4	<a href="#">runFinancial</a>	41
4.35.3.5	<a href="#">runPower</a>	41
4.36	<a href="#">Simulation.Simulation Class Reference</a>	42
4.36.1	<a href="#">Detailed Description</a>	42
4.36.2	<a href="#">Constructor &amp; Destructor Documentation</a>	42
4.36.2.1	<a href="#">__init__</a>	42
4.36.3	<a href="#">Member Function Documentation</a>	43
4.36.3.1	<a href="#">getFinancialResults</a>	43
4.36.3.2	<a href="#">getPowerProgress</a>	43
4.36.3.3	<a href="#">getPowerResults</a>	43
4.36.3.4	<a href="#">runFinancial</a>	43
4.36.3.5	<a href="#">runPower</a>	43
4.37	<a href="#">Simulation.SimulationDay Class Reference</a>	43
4.37.1	<a href="#">Detailed Description</a>	44
4.37.2	<a href="#">Member Function Documentation</a>	44
4.37.2.1	<a href="#">getElectricalEfficiency</a>	44
4.37.2.2	<a href="#">getElectricalEnergy</a>	44
4.37.2.3	<a href="#">getTotalEfficiency</a>	44
4.37.2.4	<a href="#">setElectricalEfficiency</a>	44
4.37.2.5	<a href="#">setElectricalEnergy</a>	44
4.37.2.6	<a href="#">setTotalEfficiency</a>	44
4.38	<a href="#">SolarCalculator.Simulation.SimulationDay Class Reference</a>	45
4.38.1	<a href="#">Detailed Description</a>	45
4.38.2	<a href="#">Member Function Documentation</a>	45
4.38.2.1	<a href="#">getElectricalEfficiency</a>	45
4.38.2.2	<a href="#">getElectricalEnergy</a>	45
4.38.2.3	<a href="#">getTotalEfficiency</a>	46
4.38.2.4	<a href="#">setElectricalEfficiency</a>	46



4.38.2.5	setElectricalEnergy	46
4.38.2.6	setTotalEfficiency	46
4.39	SolarCalculator.GUI.SimulationResults Class Reference	46
4.39.1	Detailed Description	47
4.40	SolarCalculator.Assets.Site Class Reference	47
4.40.1	Detailed Description	47
4.40.2	Constructor & Destructor Documentation	48
4.40.2.1	__init__	48
4.40.3	Member Function Documentation	48
4.40.3.1	getArrayNum	48
4.40.3.2	getCircuitBreakerNum	48
4.40.3.3	getInverterNum	48
4.40.3.4	getLatitude	48
4.40.3.5	getLongitude	48
4.40.3.6	getTemperature	48
4.40.3.7	getTransformerNum	48
4.41	main.SolarFarmCalculator Class Reference	48
4.41.1	Constructor & Destructor Documentation	49
4.41.1.1	__init__	49
4.41.2	Member Function Documentation	49
4.41.2.1	evt_calculateTXCableLength_checked	49
4.41.2.2	evt_closeApp_clicked	49
4.41.2.3	evt_runSimulation_clicked	49
4.42	Simulation.thread_SimulateDay Class Reference	50
4.42.1	Constructor & Destructor Documentation	50
4.42.1.1	__init__	50
4.43	SolarCalculator.Simulation.thread_SimulateDay Class Reference	50
4.43.1	Constructor & Destructor Documentation	51
4.43.1.1	__init__	51
4.44	SolarCalculator.Assets.Transformer Class Reference	51
4.44.1	Detailed Description	51
4.44.2	Constructor & Destructor Documentation	51
4.44.2.1	__init__	51
4.44.3	Member Function Documentation	52
4.44.3.1	getEfficiency	52
4.44.3.2	getScrapValue	52
4.44.3.3	getVARating	52
4.44.3.4	getVoltage	52

**Index****52**

# Chapter 1

## Solar Farm Calculator

A software tool to model a generic solar photovoltaic farm.

### Author

Ashok Fernandez, Jarrad Raumati, Darren O'Neill

### Description:

Models the technical and financial details of a photovoltaic farm. The user specifies the parameters of the site which then calculates the expected output energy at the grid entry point (GEP).

A site and grid entry point can be selected using GPS coordinates which assist in calculating the insolation from the sun using `PySolar`. System components such as panels, DC and AC cables, inverter, transformer and transmission lines can be specified using the graphical user interface, which will then run the model simulation.

The outputs from the simulation can be listed and plotted for the user to determine if the parameters produce a viable solar farm.

### How to run

In it's current state, the calculator should run on any operating system that supports python and the dependant packages.

### Install Dependencies

The packages that this project depends on are

- `NumPy`
- `Matplotlib`
- `PySolar`
- `wxPython 2.9`

Special thanks goes out to the above projects for providing such great tools!

### OpenExchangeRates.org API Key

An API key is required from `OpenExchangeRates.org`. The free API key will suffice for this project as there is a limit to how often the software will hit the API for new data. When you have gotten an API key place it at the top

of Assets.py to ensure the exchange rates are retrieved.

## Chapter 2

# Hierarchical Index

### 2.1 Class Hierarchy

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

Dialog	
SolarCalculator.GUI.DateError . . . . .	17
main.DialogBox_DateError . . . . .	19
SolarCalculator.GUI.FatalError . . . . .	25
main.DialogBox_FatalError . . . . .	20
SolarCalculator.GUI.GeoCodeError . . . . .	27
main.DialogBox_GeoCodeError . . . . .	20
SolarCalculator.GUI.IncompleteForm . . . . .	28
main.DialogBox_IncompleteForm . . . . .	21
SolarCalculator.GUI.NoInternet . . . . .	34
main.DialogBox_NoInternet . . . . .	22
SolarCalculator.GUI.SimulationResults . . . . .	46
main.DialogBox_SimulationResults . . . . .	23
Exception	
SolarCalculator.Utils.PyExchangeRates.AccessDataFailure . . . . .	9
SolarCalculator.Utils.PyExchangeRates.BadAppID . . . . .	15
SolarCalculator.Utils.PyExchangeRates.InvalidCurrencyKey . . . . .	30
SolarCalculator.Utils.ReverseGeocode.CountryNotFound . . . . .	16
Frame	
SolarCalculator.GUI.ApplicationFrame . . . . .	10
main.SolarFarmCalculator . . . . .	48
JSONEncoder	
SolarCalculator.Utils.PyExchangeRates.SimpleObjectEncoder . . . . .	40
object	
main.DialogBox_ProgressDialog . . . . .	23
main.InputField . . . . .	29
main.OptionalInputField . . . . .	35
Simulation.Simulation . . . . .	42
Simulation.SimulationDay . . . . .	43
SolarCalculator.Assets.Asset . . . . .	14
SolarCalculator.Assets.AC1Cable . . . . .	7
SolarCalculator.Assets.AC2Cable . . . . .	8
SolarCalculator.Assets.CircuitBreaker . . . . .	16
SolarCalculator.Assets.DCCable . . . . .	18
SolarCalculator.Assets.Inverter . . . . .	30
SolarCalculator.Assets.PVArray . . . . .	36
SolarCalculator.Assets.PVModule . . . . .	37

SolarCalculator.Assets.PVPanel . . . . .	38
SolarCalculator.Assets.Site . . . . .	47
SolarCalculator.Assets.Transformer . . . . .	51
SolarCalculator.Assets.Financial . . . . .	26
SolarCalculator.Assets.Material . . . . .	31
SolarCalculator.Simulation.Simulation . . . . .	40
SolarCalculator.Simulation.SimulationDay . . . . .	45
SolarCalculator.Utils.PyExchangeRates.Currency . . . . .	17
SolarCalculator.Utils.PyExchangeRates.Exchange . . . . .	24
SolarCalculator.Utils.PyExchangeRates.Money . . . . .	32
Thread	
Simulation.thread_SimulateDay . . . . .	50
SolarCalculator.Simulation.thread_SimulateDay . . . . .	50

## Chapter 3

# Class Index

### 3.1 Class List

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

SolarCalculator.Assets.AC1Cable	7
SolarCalculator.Assets.AC2Cable	8
SolarCalculator.Utils.PyExchangeRates.AccessDataFailure	9
SolarCalculator.GUI.ApplicationFrame	
Class ApplicationFrame	10
SolarCalculator.Assets.Asset	14
SolarCalculator.Utils.PyExchangeRates.BadAppID	15
SolarCalculator.Assets.CircuitBreaker	16
SolarCalculator.Utils.ReverseGeocode.CountryNotFound	16
SolarCalculator.Utils.PyExchangeRates.Currency	17
SolarCalculator.GUI.DateError	
Class DateError	17
SolarCalculator.Assets.DCCable	18
main.DialogBox_DateError	19
main.DialogBox_FatalError	20
main.DialogBox_GeoCodeError	20
main.DialogBox_IncompleteForm	21
main.DialogBox_NoInternet	22
main.DialogBox_ProgressDialog	23
main.DialogBox_SimulationResults	23
SolarCalculator.Utils.PyExchangeRates.Exchange	24
SolarCalculator.GUI.FatalError	
Class FatalError	25
SolarCalculator.Assets.Financial	26
SolarCalculator.GUI.GeoCodeError	
Class GeoCodeError	27
SolarCalculator.GUI.IncompleteForm	
Class IncompleteForm	28
main.InputField	29
SolarCalculator.Utils.PyExchangeRates.InvalidCurrencyKey	30
SolarCalculator.Assets.Inverter	30
SolarCalculator.Assets.Material	31
SolarCalculator.Utils.PyExchangeRates.Money	32
SolarCalculator.GUI.NoInternet	
Class NoInternet	34
main.OptionalInputField	35
SolarCalculator.Assets.PVArray	36
SolarCalculator.Assets.PVModule	37

SolarCalculator.Assets.PVPanel . . . . .	38
SolarCalculator.Utils.PyExchangeRates.SimpleObjectEncoder . . . . .	40
SolarCalculator.Simulation.Simulation . . . . .	40
Simulation.Simulation . . . . .	42
Simulation.SimulationDay . . . . .	43
SolarCalculator.Simulation.SimulationDay . . . . .	45
SolarCalculator.GUI.SimulationResults . . . . .	
Class SimulationResults . . . . .	46
SolarCalculator.Assets.Site . . . . .	47
main.SolarFarmCalculator . . . . .	48
Simulation.thread_SimulateDay . . . . .	50
SolarCalculator.Simulation.thread_SimulateDay . . . . .	50
SolarCalculator.Assets.Transformer . . . . .	51

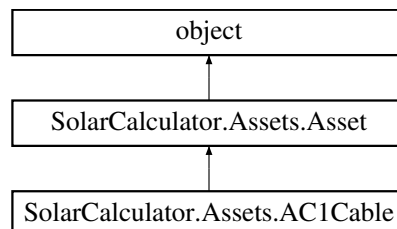


## Chapter 4

# Class Documentation

### 4.1 SolarCalculator.Assets.AC1Cable Class Reference

Inheritance diagram for SolarCalculator.Assets.AC1Cable:



#### Public Member Functions

- def [\\_\\_init\\_\\_](#)
- def [getStrandNum](#)
- def [getDiameter](#)
- def [getMaterial](#)
- def [getLength](#)

#### Public Attributes

- **strandNum**
- **diameter**
- **material**
- **length**

#### Additional Inherited Members

##### 4.1.1 Detailed Description

Class that stores the information relating to the AC cable between the inverter and the transformer.

### 4.1.2 Constructor & Destructor Documentation

**4.1.2.1** `def SolarCalculator.Assets.AC1Cable.__init__( self, strandNum, diameter, material, length, costPerMeter, currency = 'USD', depRate = 0 )`

Initialise the AC cable object.

### 4.1.3 Member Function Documentation

**4.1.3.1** `def SolarCalculator.Assets.AC1Cable.getDiameter ( self )`

Return the cable diameter.

**4.1.3.2** `def SolarCalculator.Assets.AC1Cable.getLength ( self )`

Return the length of the cable.

**4.1.3.3** `def SolarCalculator.Assets.AC1Cable.getMaterial ( self )`

Return the cable material.

**4.1.3.4** `def SolarCalculator.Assets.AC1Cable.getStrandNum ( self )`

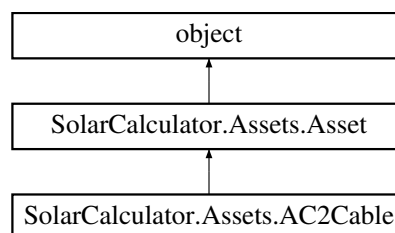
Return the number of strands for the AC1 cable.

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

- SolarCalculator/Assets.py

## 4.2 SolarCalculator.Assets.AC2Cable Class Reference

Inheritance diagram for SolarCalculator.Assets.AC2Cable:



### Public Member Functions

- `def \_\_init\_\_`
- `def getStrandNum`
- `def getDiameter`
- `def getMaterial`
- `def getLength`

## Public Attributes

- **strandNum**
- **diameter**
- **material**
- **length**

## Additional Inherited Members

### 4.2.1 Detailed Description

Class that stores the information relating the transmission line between the solar farm and the grid entry point

### 4.2.2 Constructor & Destructor Documentation

**4.2.2.1** `def SolarCalculator.Assets.AC2Cable.__init__( self, strandNum, diameter, material, length, costPerMeter, currency = 'USD', depRate = 0 )`

Initialise the GEP object

### 4.2.3 Member Function Documentation

**4.2.3.1** `def SolarCalculator.Assets.AC2Cable.getDiameter ( self )`

Return the strand diameter

**4.2.3.2** `def SolarCalculator.Assets.AC2Cable.getLength ( self )`

Return the length of the strand

**4.2.3.3** `def SolarCalculator.Assets.AC2Cable.getMaterial ( self )`

Return the strand material

**4.2.3.4** `def SolarCalculator.Assets.AC2Cable.getStrandNum ( self )`

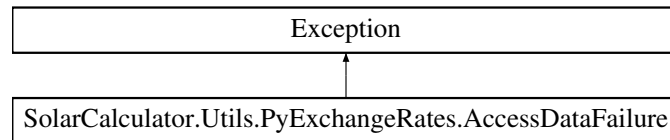
Return the number of strands in ACC or ACSR cable

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

- SolarCalculator/Assets.py

## 4.3 SolarCalculator.Utils.PyExchangeRates.AccessDataFailure Class Reference

Inheritance diagram for SolarCalculator.Utils.PyExchangeRates.AccessDataFailure:



#### 4.3.1 Detailed Description

Exception for when there is no data on the currencies - either from file or from the internet

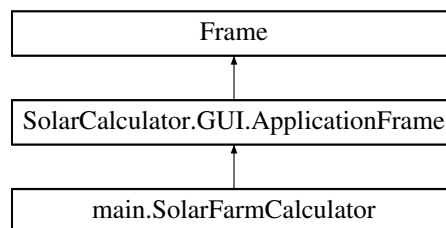
The documentation for this class was generated from the following file:

- SolarCalculator/Utils/PyExchangeRates.py

### 4.4 SolarCalculator.GUI.ApplicationFrame Class Reference

Class [ApplicationFrame](#).

Inheritance diagram for SolarCalculator.GUI.ApplicationFrame:



#### Public Member Functions

- `def __init__`
- `def __del__`
- `def evt_closeApp_clicked`
- `def evt_textEnter_validateNumber`
- `def evt_runSimulation_clicked`
- `def evt_calculateTXCableLength_checked`

#### Public Attributes

- **ApplicationTabs**
- **Calculator**
- **siteCost\_label**
- **siteCost\_input**
- **siteCost\_currency**
- **siteAppreciation\_label**
- **siteAppreciation\_input**
- **siteAppreciation\_units**
- **siteLatitude\_label**
- **siteLatitude\_input**
- **siteLatitude\_units**
- **siteLongitude\_label**

- siteLongitude\_input
- siteLongitude\_units
- siteGridLatitude\_label
- siteGridLatitude\_input
- siteGridLatitude\_units
- siteGridLongitude\_label
- siteGridLongitude\_input
- siteGridLongitude\_units
- siteNumPanels\_label
- siteNumPanels\_input
- siteNumModules\_label
- siteNumModules\_input
- siteNumArrays\_label
- siteNumArrays\_input
- siteNumTransformers\_label
- siteNumTransformers\_input
- siteNumInverters\_label
- siteNumInverters\_input
- siteNumCircuitBreakers\_label
- siteNumCircuitBreakers\_input
- financialCurrency\_label
- financialCurrency\_currency
- financialInterestRate\_label
- financialInterestRate\_input
- financialInterestRate\_units
- financialMiscExpenses\_label
- financialMiscExpenses\_input
- financialMaintenance\_label
- financialMaintenance\_input
- financialMaintenance\_units
- financialPowerPrice\_label
- financialPowerPrice\_input
- financialPowerPrice\_units
- simulationStart\_label
- simulationStart\_input
- simulationEnd\_label
- simulationEnd\_input
- runSimulation\_button
- panelVoltage\_label
- panelVoltage\_input
- panelVoltage\_units
- panelAngle\_label
- panelAngle\_input
- panelAngle\_units
- panelRating\_label
- panelRating\_input
- panelRating\_units
- panelDegradation\_label
- panelDegradation\_input
- panelDegradation\_units
- panelArea\_label
- panelArea\_input
- panelArea\_units
- panelCost\_label
- panelCost\_input

- panelCost\_currency
- panelCost\_units
- panelDepreciation\_label
- panelDepreciation\_input
- panelDepreciation\_units
- circuitBreakerCost\_label
- circuitBreakerCost\_input
- circuitBreakerCost\_currency
- circuitBreakerCost\_units
- circuitBreakerDepreciation\_label
- circuitBreakerDepreciation\_input
- circuitBreakerDepreciation\_units
- DCCableDiameter\_label
- DCCableDiameter\_input
- DCCableDiameter\_units
- DCCableMaterial\_label
- DCCableMaterial\_input
- DCCableLength\_label
- DCCableLength\_input
- DCCableLength\_units
- DCCableCost\_label
- DCCableCost\_input
- DCCableCost\_currency
- DCCableCost\_units
- DCCableDepreciation\_label
- DCCableDepreciation\_input
- DCCableDepreciation\_units
- inverterPowerFactor\_label
- inverterPowerFactor\_input
- inverterPowerFactor\_units
- inverterEfficiency\_label
- inverterEfficiency\_input
- inverterEfficiency\_units
- inverterOutputVoltage\_label
- inverterOutputVoltage\_input
- inverterOutputVoltage\_units
- inverterCost\_label
- inverterCost\_input
- inverterCost\_currency
- inverterCost\_units
- inverterDepreciation\_label
- inverterDepreciation\_input
- inverterDepreciation\_units
- ACCableDiameter\_label
- ACCableDiameter\_input
- ACCableDiameter\_units
- ACCableNumStrands\_label
- ACCableNumStrands\_input
- ACCableMaterial\_label
- ACCableMaterial\_input
- ACCableLength\_label
- ACCableLength\_input
- ACCableLength\_units
- ACCableCost\_label
- ACCableCost\_input

- ACCableCost\_currency
- ACCableCost\_units
- ACCableDepreciation\_label
- ACCableDepreciation\_input
- ACCableDepreciation\_units
- transformerOutputVoltage\_label
- transformerOutputVoltage\_input
- transformerOutputVoltage\_units
- transformerEfficiency\_label
- transformerEfficiency\_input
- transformerEfficiency\_units
- transformerRating\_label
- transformerRating\_input
- transformerRating\_units
- transformerCost\_label
- transformerCost\_input
- transformerCost\_currency
- transformerCost\_units
- transformerDepreciation\_label
- transformerDepreciation\_input
- transformerDepreciation\_units
- TXCableDiameter\_label
- TXCableDiameter\_input
- TXCableDiameter\_units
- TXCableNumStrands\_label
- TXCableNumStrands\_input
- TXCableMaterial\_label
- TXCableMaterial\_input
- TXCableLength\_label
- TXCableLength\_input
- TXCableLength\_units
- TXCableCalculateLength\_label
- TXCableCalculateLength\_checkBox
- TXCableCost\_label
- TXCableCost\_input
- TXCableCost\_currency
- TXCableCost\_units
- TXCableDepreciation\_label
- TXCableDepreciation\_input
- TXCableDepreciation\_units
- SolarFarmModel
- solarFarmModelBitmap
- Settings
- settingsPlaceholderLabel
- Help
- settingsPlaceholderLabel1

#### 4.4.1 Detailed Description

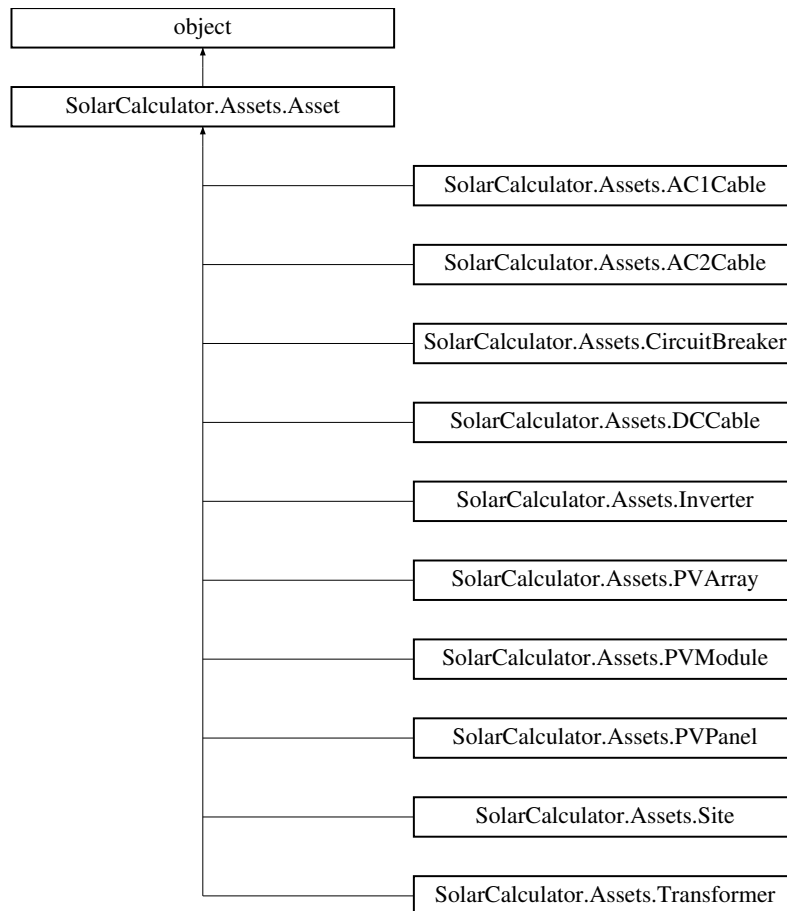
Class [ApplicationFrame](#).

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

- SolarCalculator/GUI.py

## 4.5 SolarCalculator.Assets.Asset Class Reference

Inheritance diagram for SolarCalculator.Assets.Asset:



### Public Member Functions

- def `__init__`
- def `getCost`
- def `getDepreciatedValue`
- def `getDepRate`
- def `getCurrency`

### Public Attributes

- **cost**
- **depRate**

### Static Public Attributes

- **exchange** = CURRENCY\_EXCHANGE

#### 4.5.1 Detailed Description

Asset superclass for PV farm components. Contains the financial data relating to the asset.



## 4.5.2 Constructor & Destructor Documentation

### 4.5.2.1 `def SolarCalculator.Assets.Asset.__init__( self, cost, currency, depRate = 0 )`

Initialise the asset superclass object.

## 4.5.3 Member Function Documentation

### 4.5.3.1 `def SolarCalculator.Assets.Asset.getCost ( self )`

Return the cost of the asset.

### 4.5.3.2 `def SolarCalculator.Assets.Asset.getCurrency ( self )`

Return the currency of the asset's cost.

### 4.5.3.3 `def SolarCalculator.Assets.Asset.getDepRate ( self )`

Return the asset depreciation rate of the asset.

### 4.5.3.4 `def SolarCalculator.Assets.Asset.getDepreciatedValue ( self, timedelta )`

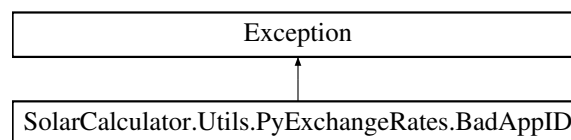
Returns the assets value factoring in depreciation over the given time.

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

- SolarCalculator/Assets.py

## 4.6 SolarCalculator.Utills.PyExchangeRates.BadAppID Class Reference

Inheritance diagram for SolarCalculator.Utills.PyExchangeRates.BadAppID:



### 4.6.1 Detailed Description

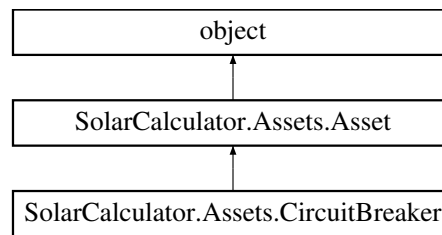
Thrown when the app id for the open currency exchange is invalid

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

- SolarCalculator/Utills/PyExchangeRates.py

## 4.7 SolarCalculator.Assets.CircuitBreaker Class Reference

Inheritance diagram for SolarCalculator.Assets.CircuitBreaker:



### Public Member Functions

- `def \_\_init\_\_`

### Additional Inherited Members

#### 4.7.1 Detailed Description

Class for storing information relating to a circuit breaker

#### 4.7.2 Constructor & Destructor Documentation

**4.7.2.1** `def SolarCalculator.Assets.CircuitBreaker.__init__( self, cost, currency = ' USD ' , depRate = 0 )`

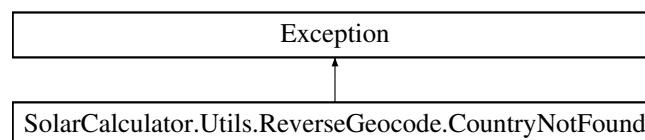
Initialise the circuit breaker class object

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

- SolarCalculator/Assets.py

## 4.8 SolarCalculator.Utils.ReverseGeocode.CountryNotFound Class Reference

Inheritance diagram for SolarCalculator.Utils.ReverseGeocode.CountryNotFound:



#### 4.8.1 Detailed Description

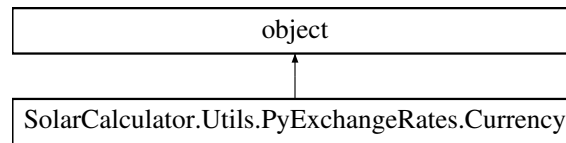
Raised when the geocode could not find a country

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

- SolarCalculator/Utils/ReverseGeocode.py

## 4.9 SolarCalculator.Utils.PyExchangeRates.Currency Class Reference

Inheritance diagram for SolarCalculator.Utils.PyExchangeRates.Currency:



### Public Member Functions

- `def \_\_init\_\_`
- `def \_\_str\_\_`

### Public Attributes

- **key**
- **baseKey**
- **rate**
- **name**

#### 4.9.1 Detailed Description

Class to store information about a currency

#### 4.9.2 Constructor & Destructor Documentation

4.9.2.1 `def SolarCalculator.Utils.PyExchangeRates.Currency.__init__( self, key, baseKey, rate, name )`

Initialise a currency object

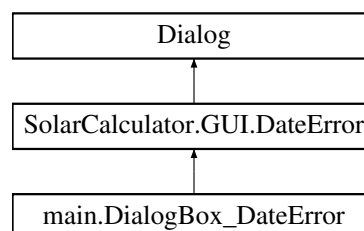
The documentation for this class was generated from the following file:

- SolarCalculator/Utils/PyExchangeRates.py

## 4.10 SolarCalculator.GUI.DateError Class Reference

Class [DateError](#).

Inheritance diagram for SolarCalculator.GUI.DateError:



## Public Member Functions

- `def __init__`
- `def __del__`
- `def evt_dialogOK_clicked`

## Public Attributes

- `dateErrorTitleLabel`
- `dateErrorFormLabel`
- `dateError_button`

### 4.10.1 Detailed Description

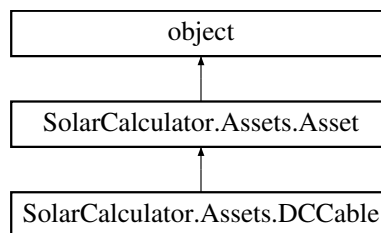
Class [DateError](#).

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

- `SolarCalculator/GUI.py`

## 4.11 SolarCalculator.Assets.DCCable Class Reference

Inheritance diagram for `SolarCalculator.Assets.DCCable`:



## Public Member Functions

- `def __init__`
- `def getDiameter`
- `def getMaterial`
- `def getLength`

## Public Attributes

- `diameter`
- `material`
- `length`

## Additional Inherited Members

### 4.11.1 Detailed Description

Class to store the information relating to the DC cable between the PV array and the inverter.

### 4.11.2 Constructor & Destructor Documentation

4.11.2.1 `def SolarCalculator.Assets.DCCable.__init__( self, diameter, material, length, costPerMeter, currency = ' USD ', depRate = 0 )`

Initialise a DC cable object.

### 4.11.3 Member Function Documentation

4.11.3.1 `def SolarCalculator.Assets.DCCable.getDiameter ( self )`

Return the cable diameter.

4.11.3.2 `def SolarCalculator.Assets.DCCable.getLength ( self )`

Return the length of the cable.

4.11.3.3 `def SolarCalculator.Assets.DCCable.getMaterial ( self )`

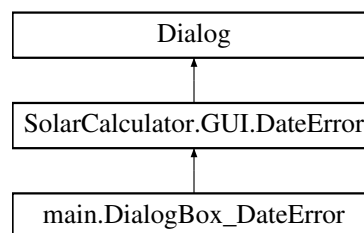
Return the cable material.

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

- SolarCalculator/Assets.py

## 4.12 main.DialogBox\_DateError Class Reference

Inheritance diagram for main.DialogBox\_DateError:



### Public Member Functions

- `def __init__`
- `def evt_dialogOK_clicked`

### Additional Inherited Members

### 4.12.1 Constructor & Destructor Documentation

4.12.1.1 `def main.DialogBox_DateError.__init__( self )`

Creates the "Date Error" dialog box and uses the given string as the error message.  
the program will quit when the dialog is dismissed

### 4.12.2 Member Function Documentation

#### 4.12.2.1 `def main.DialogBox_DateError.evt_dialogOK_clicked ( self, event )`

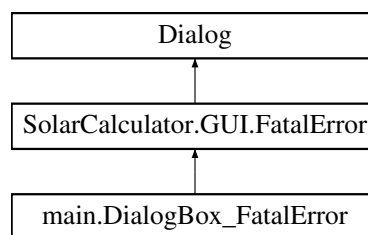
Closes the window when the OK button is pressed

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

- main.py

## 4.13 `main.DialogBox_FatalError` Class Reference

Inheritance diagram for `main.DialogBox_FatalError`:



### Public Member Functions

- `def \_\_init\_\_`
- `def evt\_dialogCloseProgram\_clicked`

### Additional Inherited Members

#### 4.13.1 Constructor & Destructor Documentation

##### 4.13.1.1 `def main.DialogBox_FatalError.__init__ ( self, errorMessage )`

Creates the "Fatal Error" dialog box and uses the given string as the error message.  
the program will quit when the dialog is dismissed

### 4.13.2 Member Function Documentation

#### 4.13.2.1 `def main.DialogBox_FatalError.evt_dialogCloseProgram_clicked ( self, event )`

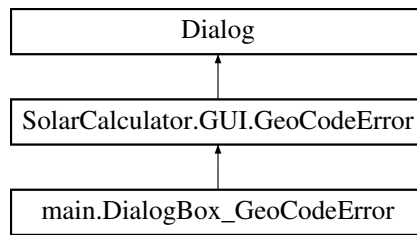
Terminates the program after the user has been notified of a fatal error

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

- main.py

## 4.14 `main.DialogBox_GeoCodeError` Class Reference

Inheritance diagram for `main.DialogBox_GeoCodeError`:



### Public Member Functions

- def `__init__`
- def `evt_dialogOK_clicked`

### Additional Inherited Members

#### 4.14.1 Constructor & Destructor Documentation

##### 4.14.1.1 def main.DialogBox\_GeoCodeError.\_\_init\_\_( self )

Creates the "GeoCode Error" dialog box and uses the given string as the error message.  
the program will quit when the dialog is dismissed

#### 4.14.2 Member Function Documentation

##### 4.14.2.1 def main.DialogBox\_GeoCodeError.evt\_dialogOK\_clicked( self, event )

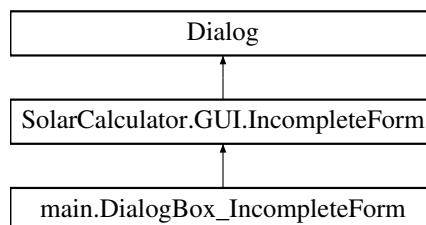
Closes the window when the OK button is pressed

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

- main.py

## 4.15 main.DialogBox\_IncompleteForm Class Reference

Inheritance diagram for main.DialogBox\_IncompleteForm:



### Public Member Functions

- def `__init__`
- def `evt_dialogOK_clicked`

## Additional Inherited Members

### 4.15.1 Constructor & Destructor Documentation

#### 4.15.1.1 `def main.DialogBox_IncompleteForm.__init__( self )`

Creates the "Incomplete Form" dialog box and shows it as a modal dialog which blocks the program until it is dismissed

### 4.15.2 Member Function Documentation

#### 4.15.2.1 `def main.DialogBox_IncompleteForm.evt_dialogOK_clicked ( self, event )`

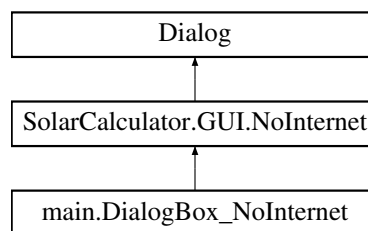
Closes the window when the OK button is pressed

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

- `main.py`

## 4.16 `main.DialogBox_NoInternet` Class Reference

Inheritance diagram for `main.DialogBox_NoInternet`:



## Public Member Functions

- `def \_\_init\_\_`
- `def evt\_dialogOK\_clicked`

## Additional Inherited Members

### 4.16.1 Constructor & Destructor Documentation

#### 4.16.1.1 `def main.DialogBox_NoInternet.__init__( self )`

Creates the "No Internet" dialog box and shows it as a modal dialog which blocks the program until it is dismissed

### 4.16.2 Member Function Documentation

#### 4.16.2.1 `def main.DialogBox_NoInternet.evt_dialogOK_clicked ( self, event )`

Closes the window when the OK button is pressed

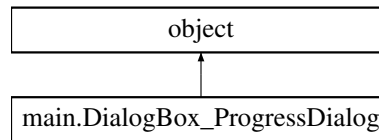
The documentation for this class was generated from the following file:



- main.py

## 4.17 main.DialogBox\_ProgressDialog Class Reference

Inheritance diagram for main.DialogBox\_ProgressDialog:



### Public Member Functions

- def [\\_\\_init\\_\\_](#)
- def [update](#)
- def [closeDialog](#)

### Public Attributes

- **progressBox**

#### 4.17.1 Constructor & Destructor Documentation

4.17.1.1 `def main.DialogBox_ProgressDialog.__init__( self, parent, maxItems = 100 )`

Dialog box to show while the simulation is in progress

#### 4.17.2 Member Function Documentation

4.17.2.1 `def main.DialogBox_ProgressDialog.closeDialog( self )`

Closes the dialog box

4.17.2.2 `def main.DialogBox_ProgressDialog.update( self, itemsLeft, newMessage = None )`

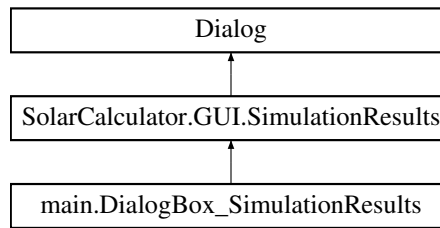
Passes the dialog the updated amount of items left in the simulation queue

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

- main.py

## 4.18 main.DialogBox\_SimulationResults Class Reference

Inheritance diagram for main.DialogBox\_SimulationResults:



## Public Member Functions

- def `__init__`
- def `evt_dialogOK_clicked`

## Additional Inherited Members

### 4.18.1 Constructor & Destructor Documentation

#### 4.18.1.1 def `main.DialogBox_SimulationResults.__init__( self, simulationResults )`

Creates the "Fatal Error" dialog box and uses the given string as the error message. the program will quit when the dialog is dismissed

### 4.18.2 Member Function Documentation

#### 4.18.2.1 def `main.DialogBox_SimulationResults.evt_dialogOK_clicked ( self, event )`

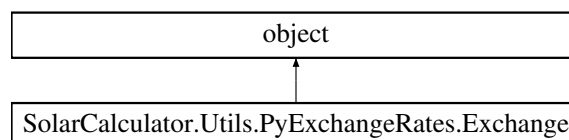
Closes the window when the OK button is pressed

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

- `main.py`

## 4.19 SolarCalculator.Utils.PyExchangeRates.Exchange Class Reference

Inheritance diagram for `SolarCalculator.Utils.PyExchangeRates.Exchange`:



## Public Member Functions

- def `__init__`
- def `withdraw`

## Public Attributes

- **applID**
- **filename**
- **currencies**
- **lastUpdated**

### 4.19.1 Detailed Description

Object to store currencies and update them with the OpenExchangeRates API

### 4.19.2 Member Function Documentation

#### 4.19.2.1 `def SolarCalculator.Utils.PyExchangeRates.Exchange.withdraw ( self, amount, currencyKey )`

Creates a money object that points to this exchange with the given amount of the given currency

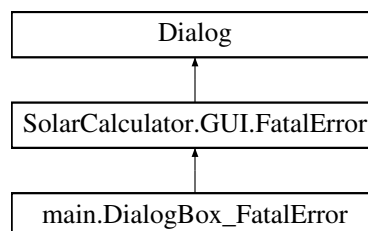
The documentation for this class was generated from the following file:

- SolarCalculator/Utils/PyExchangeRates.py

## 4.20 SolarCalculator.GUI.FatalError Class Reference

Class [FatalError](#).

Inheritance diagram for SolarCalculator.GUI.FatalError:



## Public Member Functions

- `def __init__`
- `def __del__`
- `def evt_dialogCloseProgram_clicked`

## Public Attributes

- **fatalErrorTitleLabel**
- **fatalErrorLabel**
- **fatalError\_button**

### 4.20.1 Detailed Description

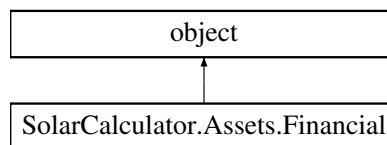
Class [FatalError](#).

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

- SolarCalculator/GUI.py

## 4.21 SolarCalculator.Assets.Financial Class Reference

Inheritance diagram for SolarCalculator.Assets.Financial:



### Public Member Functions

- `def \_\_init\_\_`
- `def getDailyMaintenance`
- `def addToLoan`
- `def makeLoanPayment`
- `def accumulateDailyInterest`
- `def getCurrentLoanValue`
- `def amountInBaseCurrency`
- `def getBaseCurrency`
- `def getPowerPrice`

### Public Attributes

- **`baseCurrency`**
- **`interestRate`**
- **`maintenance`**
- **`loan`**
- **`powerPrice`**

### Static Public Attributes

- **`exchange = CURRENCY_EXCHANGE`**

### 4.21.1 Detailed Description

Class that stores the information relating to the financial data that is independent of the solar farm

### 4.21.2 Constructor & Destructor Documentation

- 4.21.2.1 `def SolarCalculator.Assets.Financial.__init__( self, maintenance, miscExpenses, interestRate, powerPrice, baseCurrency = 'USD' )`

Initialise the Financial object

### 4.21.3 Member Function Documentation

#### 4.21.3.1 `def SolarCalculator.Assets.Financial.accumulateDailyInterest ( self )`

Adds interest to the initial expenses loan

#### 4.21.3.2 `def SolarCalculator.Assets.Financial.addToLoan ( self, cost )`

Adds money to the initial cost

#### 4.21.3.3 `def SolarCalculator.Assets.Financial.amountInBaseCurrency ( self, money )`

Returns the value of a money object in the base currency of the loan

#### 4.21.3.4 `def SolarCalculator.Assets.Financial.getBaseCurrency ( self )`

Returns the three letter code of the base currency

#### 4.21.3.5 `def SolarCalculator.Assets.Financial.getCurrentLoanValue ( self )`

Returns the current value of the loan

#### 4.21.3.6 `def SolarCalculator.Assets.Financial.getDailyMaintenance ( self )`

Return the maintenance budget per year

#### 4.21.3.7 `def SolarCalculator.Assets.Financial.getPowerPrice ( self )`

Return the selling rate of power

#### 4.21.3.8 `def SolarCalculator.Assets.Financial.makeLoanPayment ( self, payment )`

Pays back money to the loan for the initial costs

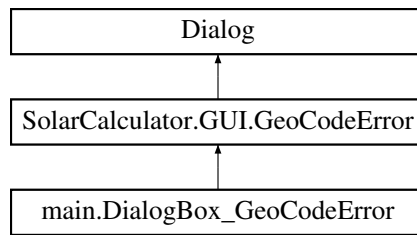
The documentation for this class was generated from the following file:

- SolarCalculator/Assets.py

## 4.22 SolarCalculator.GUI.GeoCodeError Class Reference

Class [GeoCodeError](#).

Inheritance diagram for SolarCalculator.GUI.GeoCodeError:



### Public Member Functions

- def **\_\_init\_\_**
- def **\_\_del\_\_**
- def **evt\_dialogOK\_clicked**

### Public Attributes

- **geoCodeErrorTitleLabel**
- **geoCodeErrorFormLabel**
- **geoCodeError\_button**

#### 4.22.1 Detailed Description

Class [GeoCodeError](#).

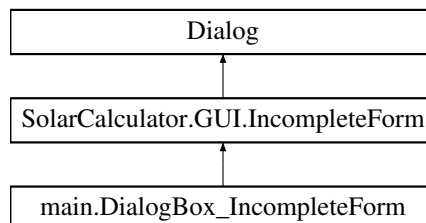
The documentation for this class was generated from the following file:

- SolarCalculator/GUI.py

## 4.23 SolarCalculator.GUI.IncompleteForm Class Reference

Class [IncompleteForm](#).

Inheritance diagram for SolarCalculator.GUI.IncompleteForm:



### Public Member Functions

- def **\_\_init\_\_**
- def **\_\_del\_\_**
- def **evt\_dialogOK\_clicked**

### Public Attributes

- `incompleteFormTitleLabel`
- `incompleteFormLabel`
- `incompleteForm_button`

#### 4.23.1 Detailed Description

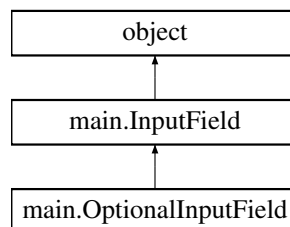
Class `IncompleteForm`.

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

- `SolarCalculator/GUI.py`

## 4.24 main.InputField Class Reference

Inheritance diagram for `main.InputField`:



### Public Member Functions

- `def __init__`
- `def setLabelColour`
- `def setFieldValue`
- `def validateField`

### Public Attributes

- `field`
- `label`
- `condition`
- `upperLimit`
- `lowerLimit`

### Static Public Attributes

- tuple `RED` = (255,0,0, 200)
- tuple `BLACK` = (0,0,0)
- tuple `WHITE` = (255,255,255,255)

#### 4.24.1 Detailed Description

Stores an input field and corresponding label and encapsulates validation of the input field

## 4.24.2 Member Function Documentation

### 4.24.2.1 `def main.InputField.setFieldValue ( self, value )`

Allows an external program to set the value in the field for testing purposes

### 4.24.2.2 `def main.InputField.setLabelColour ( self, colour )`

Sets the colour of the wxStaticText label, colour is a 3 or 4 length tuple of RGB or RGBA values between 0-255

### 4.24.2.3 `def main.InputField.validateField ( self )`

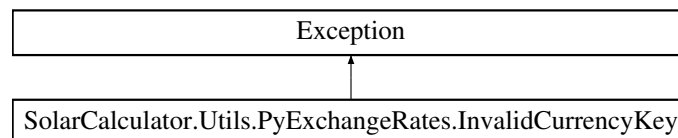
Validates the input in the wxTextCtrl inputField to be purely numeric. Condition can be either a string containing "p" or "n" to constrain the inputField to positive or negative numbers respectively, and can also contain "i" to specify the number must be an integer. These can be combined, for instance "pi" specifies a positive integer. If the field is valid the wxStaticText label fieldLabel's colour text is set to black, otherwise it is set to red. The function returns the fields value if the input was valid, otherwise it returns false.

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

- main.py

## 4.25 SolarCalculator.Utills.PyExchangeRates.InvalidCurrencyKey Class Reference

Inheritance diagram for SolarCalculator.Utills.PyExchangeRates.InvalidCurrencyKey:



### 4.25.1 Detailed Description

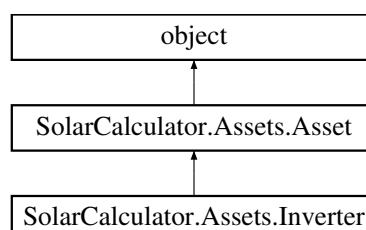
Thrown when a currency key is given that is not in the Exchange

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

- SolarCalculator/Utills/PyExchangeRates.py

## 4.26 SolarCalculator.Assets.Inverter Class Reference

Inheritance diagram for SolarCalculator.Assets.Inverter:





## Public Member Functions

- def `__init__`
- def `getPowerFactor`
- def `getEfficiency`
- def `getVoltage`

## Public Attributes

- `powerFactor`
- `efficiency`
- `voltage`

## Additional Inherited Members

### 4.26.1 Detailed Description

Class to store the information relating to the Inverter.

### 4.26.2 Constructor & Destructor Documentation

**4.26.2.1** `def SolarCalculator.Assets.Inverter.__init__( self, powerFactor, efficiency, voltage, cost, currency = 'USD', depRate = 0 )`

Initialise an inverter object.

### 4.26.3 Member Function Documentation

**4.26.3.1** `def SolarCalculator.Assets.Inverter.getEfficiency( self )`

Return the efficiency of the inverter between 0 and 1.

**4.26.3.2** `def SolarCalculator.Assets.Inverter.getPowerFactor( self )`

Return the power factor.

**4.26.3.3** `def SolarCalculator.Assets.Inverter.getVoltage( self )`

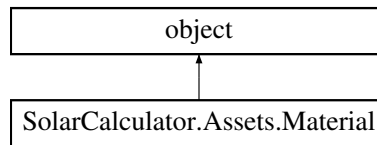
Return the output voltage of the inverter.

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

- SolarCalculator/Assets.py

## 4.27 SolarCalculator.Assets.Material Class Reference

Inheritance diagram for SolarCalculator.Assets.Material:



### Public Member Functions

- `def __init__`
- `def getResistivity`
- `def getTempCoefficient`

### Public Attributes

- **name**
- **resistivity**
- **tempCoefficient**

#### 4.27.1 Detailed Description

Class object for a material.

#### 4.27.2 Constructor & Destructor Documentation

##### 4.27.2.1 `def SolarCalculator.Assets.Material.__init__( self, name, resistivity, tempCoefficient )`

Initialise a material object.

#### 4.27.3 Member Function Documentation

##### 4.27.3.1 `def SolarCalculator.Assets.Material.getResistivity( self )`

Return the resistivity of the material.

##### 4.27.3.2 `def SolarCalculator.Assets.Material.getTempCoefficient( self )`

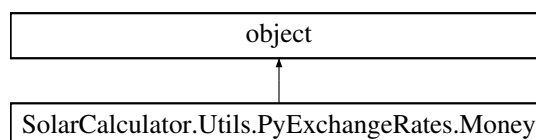
Return the temperature coefficient of the material.

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

- SolarCalculator/Assets.py

## 4.28 SolarCalculator.Utils.PyExchangeRates.Money Class Reference

Inheritance diagram for SolarCalculator.Utils.PyExchangeRates.Money:



## Public Member Functions

- def `__init__`
- def `getAmount`
- def `setAmount`
- def `getCurrencyKey`
- def `getExchange`
- def `convert`
- def `__str__`
- def `__add__`
- def `__sub__`
- def `__mul__`
- def `__div__`

## Public Attributes

- `amount`
- `currencyKey`
- `exchange`

### 4.28.1 Detailed Description

Object to represent a certain amount of a currency withdrawn from an exchange

### 4.28.2 Member Function Documentation

#### 4.28.2.1 `def SolarCalculator.Utils.PyExchangeRates.Money.__add__( self, other )`

Adds two currencies together. The resulting currency is United States Dollars

#### 4.28.2.2 `def SolarCalculator.Utils.PyExchangeRates.Money.__div__( self, other )`

Divides two currencies together. The resulting currency is United States Dollars by default

#### 4.28.2.3 `def SolarCalculator.Utils.PyExchangeRates.Money.__mul__( self, other )`

Multiplies currencies together. The resulting currency is United States Dollars by default

#### 4.28.2.4 `def SolarCalculator.Utils.PyExchangeRates.Money.__str__( self )`

Allows the object to be printed using "print"

#### 4.28.2.5 `def SolarCalculator.Utils.PyExchangeRates.Money.__sub__( self, other )`

Subtracts two currencies together. The resulting currency is United States Dollars

#### 4.28.2.6 `def SolarCalculator.Utils.PyExchangeRates.Money.convert( self, currencyKey )`

Takes a given currency key and returns a new Money object of the given currency

#### 4.28.2.7 `def SolarCalculator.Utils.PyExchangeRates.Money.getAmount ( self )`

Returns the amount

#### 4.28.2.8 `def SolarCalculator.Utils.PyExchangeRates.Money.getCurrencyKey ( self )`

Returns the currency key

#### 4.28.2.9 `def SolarCalculator.Utils.PyExchangeRates.Money.getExchange ( self )`

Returns the pointer to the exchange that the money was withdrawn from

#### 4.28.2.10 `def SolarCalculator.Utils.PyExchangeRates.Money.setAmount ( self, amount )`

Sets the amount

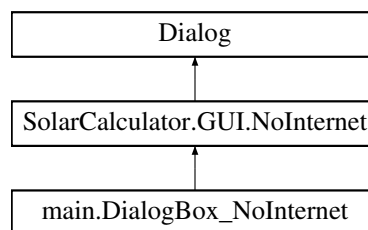
The documentation for this class was generated from the following file:

- SolarCalculator/Utils/PyExchangeRates.py

## 4.29 SolarCalculator.GUI.NoInternet Class Reference

Class [NoInternet](#).

Inheritance diagram for SolarCalculator.GUI.NoInternet:



### Public Member Functions

- `def __init__`
- `def __del__`
- `def evt_dialogOK_clicked`

### Public Attributes

- `noInternettitleLabel`
- `noInternetLabel`
- `noInternet_button`

### 4.29.1 Detailed Description

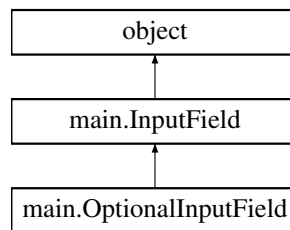
Class [NoInternet](#).

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

- SolarCalculator/GUI.py

## 4.30 main.OptionalInputField Class Reference

Inheritance diagram for main.OptionalInputField:



### Public Member Functions

- `def \_\_init\_\_`
- `def getCheckboxState`
- `def validateField`

### Public Attributes

- `checkbox`

### Additional Inherited Members

#### 4.30.1 Constructor & Destructor Documentation

**4.30.1.1** `def main.OptionalInputField.__init__( self, field, label, checkbox, condition = ' ', upperLimit=False, lowerLimit=False )`

Initialises the input field superclass and saves the checkbox

#### 4.30.2 Member Function Documentation

**4.30.2.1** `def main.OptionalInputField.getCheckboxState( self )`

Returns the state of the checkbox so the simulation can be told whether or not the value is valid or needs to be calculated. True if the checkbox is checked, otherwise False.

**4.30.2.2** `def main.OptionalInputField.validateField( self )`

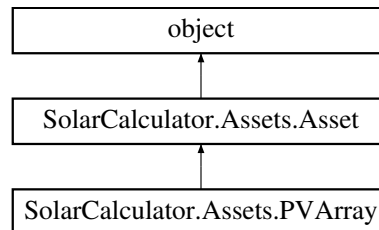
If the field is to be calculated from other field values return a placeholder value, otherwise validate like normal. The placeholder value is required as the fields that depend on this one need to be validated first

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

- main.py

## 4.31 SolarCalculator.Assets.PVArray Class Reference

Inheritance diagram for SolarCalculator.Assets.PVArray:



### Public Member Functions

- def `__init__`
- def `getModuleType`
- def `getModuleNum`
- def `getVoltage`
- def `getAngle`
- def `getArea`

### Public Attributes

- `moduleType`
- `moduleNum`
- `angle`
- `voltage`

### Additional Inherited Members

#### 4.31.1 Detailed Description

Class to store the information relating to a PV array. An array contains PV modules.

#### 4.31.2 Constructor & Destructor Documentation

4.31.2.1 `def SolarCalculator.Assets.PVArray.__init__( self, moduleType, moduleNum, arrayAngle )`

Initialise a PV array object.

#### 4.31.3 Member Function Documentation

4.31.3.1 `def SolarCalculator.Assets.PVArray.getAngle( self )`

Return the angle of the PV panels.

4.31.3.2 `def SolarCalculator.Assets.PVArray.getArea ( self )`

Calculates the total area of the panels in m<sup>2</sup>.

4.31.3.3 `def SolarCalculator.Assets.PVArray.getModuleNum ( self )`

return the number of modules within the array.

4.31.3.4 `def SolarCalculator.Assets.PVArray.getModuleType ( self )`

Return the module type within the array.

4.31.3.5 `def SolarCalculator.Assets.PVArray.getVoltage ( self )`

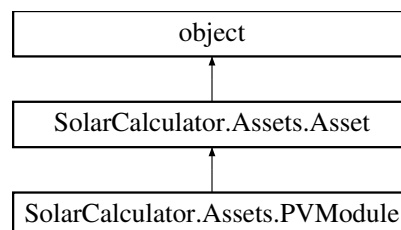
Return the voltage of the array.

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

- SolarCalculator/Assets.py

## 4.32 SolarCalculator.Assets.PVModule Class Reference

Inheritance diagram for SolarCalculator.Assets.PVModule:



### Public Member Functions

- `def \_\_init\_\_`
- `def getPanelType`
- `def getVoltage`
- `def getArea`
- `def getPanelNum`

### Public Attributes

- **panelType**
- **panelNum**
- **voltage**

## Additional Inherited Members

### 4.32.1 Detailed Description

Class to store information relating to a solar PV module. A module contains PV panels.

### 4.32.2 Constructor & Destructor Documentation

#### 4.32.2.1 `def SolarCalculator.Assets.PVModule.__init__( self, panelType, panelNum )`

Initialise a PV module object.

### 4.32.3 Member Function Documentation

#### 4.32.3.1 `def SolarCalculator.Assets.PVModule.getArea ( self )`

Calculates the total area of the panels in m<sup>2</sup>.

#### 4.32.3.2 `def SolarCalculator.Assets.PVModule.getPanelNum ( self )`

Return the number of panels within a module

#### 4.32.3.3 `def SolarCalculator.Assets.PVModule.getPanelType ( self )`

Return the panel object within the module.

#### 4.32.3.4 `def SolarCalculator.Assets.PVModule.getVoltage ( self )`

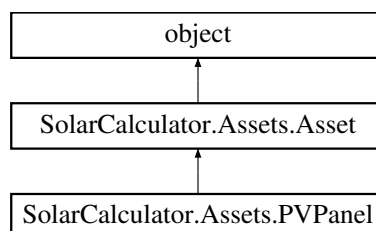
Return the module voltage.

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

- SolarCalculator/Assets.py

## 4.33 SolarCalculator.Assets.PVPanel Class Reference

Inheritance diagram for SolarCalculator.Assets.PVPanel:





## Public Member Functions

- def `__init__`
- def `getVoltage`
- def `getEfficiency`
- def `getDegradationRate`
- def `getArea`
- def `getRating`

## Public Attributes

- `voltage`
- `degradationRate`
- `area`
- `rating`

## Additional Inherited Members

### 4.33.1 Detailed Description

Class to store information relating to a solar PV panel.

### 4.33.2 Constructor & Destructor Documentation

**4.33.2.1** `def SolarCalculator.Assets.PVPanel.__init__( self, voltage, rating, degradationRate, area, cost, currency = 'USD', depRate = 0 )`

Initialise a PV panel object.

### 4.33.3 Member Function Documentation

**4.33.3.1** `def SolarCalculator.Assets.PVPanel.getArea ( self )`

Return the panel surface area.

**4.33.3.2** `def SolarCalculator.Assets.PVPanel.getDegradationRate ( self )`

Return the panel asset degradation rate.

**4.33.3.3** `def SolarCalculator.Assets.PVPanel.getEfficiency ( self )`

Return the panel efficiency between 0 and 1 (converts from a percentage).

**4.33.3.4** `def SolarCalculator.Assets.PVPanel.getRating ( self )`

Return the rating of the panel in watts.

#### 4.33.3.5 `def SolarCalculator.Assets.PVPanel.getVoltage ( self )`

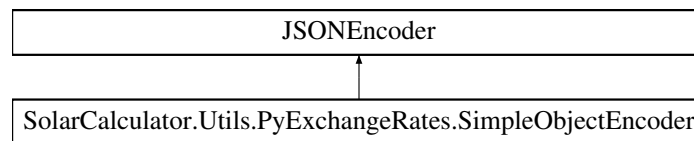
Return the panel voltage.

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

- SolarCalculator/Assets.py

### 4.34 SolarCalculator.Utils.PyExchangeRates.SimpleObjectEncoder Class Reference

Inheritance diagram for SolarCalculator.Utils.PyExchangeRates.SimpleObjectEncoder:



#### Public Member Functions

- `def default`

#### 4.34.1 Detailed Description

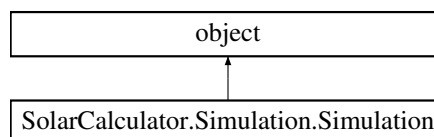
Allows simple objects which contain no methods to be encoded by returning them as a dictionary

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

- SolarCalculator/Utils/PyExchangeRates.py

### 4.35 SolarCalculator.Simulation.Simulation Class Reference

Inheritance diagram for SolarCalculator.Simulation.Simulation:



#### Public Member Functions

- `def \_\_init\_\_`
- `def getStartDate`
- `def setStartDate`
- `def getFinishDate`
- `def setFinishDate`
- `def runPower`
- `def getPowerProgress`
- `def getPowerResults`
- `def runFinancial`
- `def getFinancialResults`

## Public Attributes

- **start**
- **finish**
- **numDays**
- **days**
- **numThreads**
- **simulationTimestepMins**
- **parameters**
- **powerResults**
- **financialResults**
- **inputQueue**
- **outputQueue**

### 4.35.1 Detailed Description

Object to contain the simulation parameters

### 4.35.2 Constructor & Destructor Documentation

**4.35.2.1** `def SolarCalculator.Simulation.Simulation.__init__( self, start, finish, PVPanel, PVModule, PVArray, DCCable, Inverter, AC1Cable, Transformer, AC2Cable, CircuitBreaker, Site, Financial, numThreads = 5, simulationTimestepMins = 30 )`

Initilise the simulation

### 4.35.3 Member Function Documentation

**4.35.3.1** `def SolarCalculator.Simulation.Simulation.getFinancialResults ( self )`

Returns the financial results

**4.35.3.2** `def SolarCalculator.Simulation.Simulation.getPowerProgress ( self )`

Returns percentage of days simulated in the power simulation as a number between 0 and 100

**4.35.3.3** `def SolarCalculator.Simulation.Simulation.getPowerResults ( self )`

Processes and gets the power results. Blocks until the power simulation is finished.

**4.35.3.4** `def SolarCalculator.Simulation.Simulation.runFinancial ( self )`

Runs the finicial simulation, requires the results from power flow simulation. Blocks until complete

**4.35.3.5** `def SolarCalculator.Simulation.Simulation.runPower ( self )`

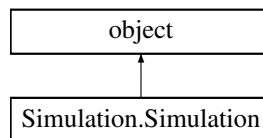
Runs the power flow simulation

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

- SolarCalculator/Simulation.py

## 4.36 Simulation.Simulation Class Reference

Inheritance diagram for Simulation.Simulation:



### Public Member Functions

- def [\\_\\_init\\_\\_](#)
- def **getStartDate**
- def **setStartDate**
- def **getFinishDate**
- def **setFinishDate**
- def [runPower](#)
- def [getPowerProgress](#)
- def [getPowerResults](#)
- def [runFinancial](#)
- def [getFinancialResults](#)

### Public Attributes

- **start**
- **finish**
- **numDays**
- **days**
- **numThreads**
- **simulationTimestepMins**
- **parameters**
- **powerResults**
- **financialResults**
- **inputQueue**
- **outputQueue**

### 4.36.1 Detailed Description

Object to contain the simulation parameters

### 4.36.2 Constructor & Destructor Documentation

4.36.2.1 **def Simulation.Simulation.\_\_init\_\_( self, start, finish, PVPanel, PVModule, PVArray, DCCable, Inverter, AC1Cable, Transformer, AC2Cable, CircuitBreaker, Site, Financial, numThreads = 5, simulationTimestepMins = 30 )**

Initilise the simulation

### 4.36.3 Member Function Documentation

#### 4.36.3.1 def Simulation.Simulation.getFinancialResults ( self )

Returns the financial results

#### 4.36.3.2 def Simulation.Simulation.getPowerProgress ( self )

Returns percentage of days simulated in the power simulation as a number between 0 and 100

#### 4.36.3.3 def Simulation.Simulation.getPowerResults ( self )

Processes and gets the power results. Blocks until the power simulation is finished.

#### 4.36.3.4 def Simulation.Simulation.runFinancial ( self )

Runs the finicial simulation, requires the results from power flow simulation. Blocks until complete

#### 4.36.3.5 def Simulation.Simulation.runPower ( self )

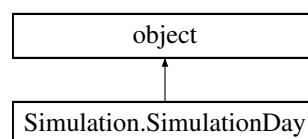
Runs the power flow simulation. Spawns

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

- Simulation.py

## 4.37 Simulation.SimulationDay Class Reference

Inheritance diagram for Simulation.SimulationDay:



### Public Member Functions

- def `__init__`
- def `setElectricalEnergy`
- def `getElectricalEnergy`
- def `setElectricalEfficiency`
- def `getElectricalEfficiency`
- def `setTotalEfficiency`
- def `getTotalEfficiency`

## Public Attributes

- **date**
- **parameters**
- **peakCurrent\_DC**
- **peakCurrent\_AC1**
- **peakCurrent\_AC2**
- **averagePower**
- **electricalEnergy**
- **electricalEfficiency**
- **totalEfficiency**
- **sunnyTime**

### 4.37.1 Detailed Description

Contains a date object with a day to simulate, as well as the output data from the simulation

### 4.37.2 Member Function Documentation

#### 4.37.2.1 `def Simulation.SimulationDay.getElectricalEfficiency ( self )`

Gets the average electrical efficiency from the day simulated, that is the efficiency between the output of the solar panels and the grid connection point

#### 4.37.2.2 `def Simulation.SimulationDay.getElectricalEnergy ( self )`

Gets the electrical energy from the simulation

#### 4.37.2.3 `def Simulation.SimulationDay.getTotalEfficiency ( self )`

Gets the average total efficiency from the day simulated, that is the efficiency between the solar energy in and the energy out at the grid connection point

#### 4.37.2.4 `def Simulation.SimulationDay.setElectricalEfficiency ( self, electricalEfficiency )`

Sets the average electrical efficiency from the day simulated, that is the efficiency between the output of the solar panels and the grid connection point

#### 4.37.2.5 `def Simulation.SimulationDay.setElectricalEnergy ( self, electricalEnergy )`

Sets the electrical energy from the simulation

#### 4.37.2.6 `def Simulation.SimulationDay.setTotalEfficiency ( self, totalEfficiency )`

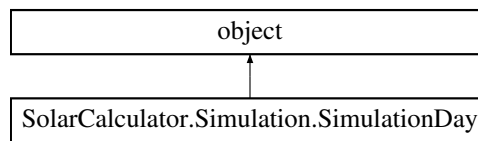
Sets the average total efficiency from the day simulated, that is the efficiency between the solar energy in and the energy out at the grid connection point

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

- Simulation.py

## 4.38 SolarCalculator.Simulation.SimulationDay Class Reference

Inheritance diagram for SolarCalculator.Simulation.SimulationDay:



### Public Member Functions

- def `__init__`
- def `setElectricalEnergy`
- def `getElectricalEnergy`
- def `setElectricalEfficiency`
- def `getElectricalEfficiency`
- def `setTotalEfficiency`
- def `getTotalEfficiency`

### Public Attributes

- `date`
- `parameters`
- `peakCurrent_DC`
- `peakCurrent_AC1`
- `peakCurrent_AC2`
- `averagePower`
- `electricalEnergy`
- `electricalEfficiency`
- `totalEfficiency`
- `sunnyTime`

### 4.38.1 Detailed Description

Contains a date object with a day to simulate, as well as the output data from the simulation

### 4.38.2 Member Function Documentation

#### 4.38.2.1 def SolarCalculator.Simulation.SimulationDay.getElectricalEfficiency ( self )

Gets the average electrical efficiency from the day simulated, that is the efficiency between the output of the solar panels and the grid connection point

#### 4.38.2.2 def SolarCalculator.Simulation.SimulationDay.getElectricalEnergy ( self )

Gets the electrical energy from the simulation

#### 4.38.2.3 `def SolarCalculator.Simulation.SimulationDay.getTotalEfficiency ( self )`

Gets the average total efficiency from the day simulated, that is the efficiency between the solar energy in and the energy out at the grid connection point

#### 4.38.2.4 `def SolarCalculator.Simulation.SimulationDay.setElectricalEfficiency ( self, electricalEfficiency )`

Sets the average electrical efficiency from the day simulated, that is the efficiency between the output of the solar panels and the grid connection point

#### 4.38.2.5 `def SolarCalculator.Simulation.SimulationDay.setElectricalEnergy ( self, electricalEnergy )`

Sets the electrical energy from the simulation

#### 4.38.2.6 `def SolarCalculator.Simulation.SimulationDay.setTotalEfficiency ( self, totalEfficiency )`

Sets the average total efficiency from the day simulated, that is the efficiency between the solar energy in and the energy out at the grid connection point

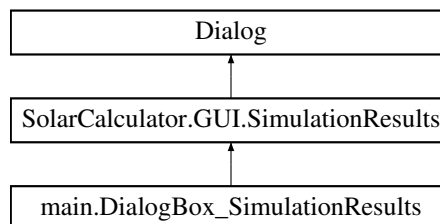
The documentation for this class was generated from the following file:

- SolarCalculator/Simulation.py

## 4.39 SolarCalculator.GUI.SimulationResults Class Reference

Class [SimulationResults](#).

Inheritance diagram for SolarCalculator.GUI.SimulationResults:



### Public Member Functions

- `def __init__`
- `def __del__`
- `def evt_dialogOK_clicked`

### Public Attributes

- `simulationResultsTitleLabel`
- `simulationResultsLabel`
- `simulationResults_button`



### 4.39.1 Detailed Description

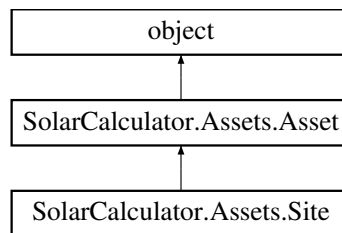
Class [SimulationResults](#).

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

- SolarCalculator/GUI.py

## 4.40 SolarCalculator.Assets.Site Class Reference

Inheritance diagram for SolarCalculator.Assets.Site:



### Public Member Functions

- def [\\_\\_init\\_\\_](#)
- def [getTransformerNum](#)
- def [getArrayNum](#)
- def [getCircuitBreakerNum](#)
- def [getInverterNum](#)
- def [getLatitude](#)
- def [getLongitude](#)
- def [getTemperature](#)

### Public Attributes

- **transformerNum**
- **arrayNum**
- **circuitBreakerNum**
- **inverterNum**
- **temperature**
- **landPrice**
- **landAppRate**
- **latitude**
- **longitude**

### Additional Inherited Members

#### 4.40.1 Detailed Description

Class that stores the information relating to the solar farm site

#### 4.40.2 Constructor & Destructor Documentation

**4.40.2.1** `def SolarCalculator.Assets.Site.__init__( self, transformerNum, arrayNum, circuitBreakerNum, inverterNum, latitude, longitude, temperature, landPrice, landAppRate = 0, currency = ' USD' )`

Initialise the solar farm site object

#### 4.40.3 Member Function Documentation

**4.40.3.1** `def SolarCalculator.Assets.Site.getArrayNum ( self )`

Return the number of arrays within the site

**4.40.3.2** `def SolarCalculator.Assets.Site.getCircuitBreakerNum ( self )`

Return the number of circuit breakers within the site

**4.40.3.3** `def SolarCalculator.Assets.Site.getInverterNum ( self )`

Return the number of inverters

**4.40.3.4** `def SolarCalculator.Assets.Site.getLatitude ( self )`

Return the site latitude

**4.40.3.5** `def SolarCalculator.Assets.Site.getLongitude ( self )`

Return the site longitude

**4.40.3.6** `def SolarCalculator.Assets.Site.getTemperature ( self, month )`

Get the temperature of the site during the given month, months are specified as 1 for January and 12 for December

**4.40.3.7** `def SolarCalculator.Assets.Site.getTransformerNum ( self )`

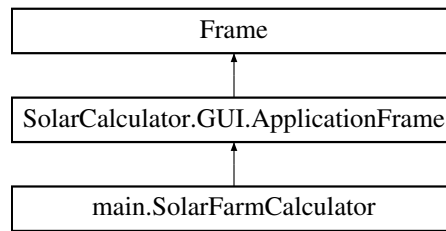
Return the number of transformers within the site

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

- SolarCalculator/Assets.py

#### 4.41 main.SolarFarmCalculator Class Reference

Inheritance diagram for main.SolarFarmCalculator:



## Public Member Functions

- `def \_\_init\_\_`
- `def evt\_closeApp\_clicked`
- `def evt\_runSimulation\_clicked`
- `def evt\_calculateTXCableLength\_checked`

## Public Attributes

- **inputFields**
- **optionalInputFields**
- **selectors**

### 4.41.1 Constructor & Destructor Documentation

#### 4.41.1.1 `def main.SolarFarmCalculator.__init__( self, parent )`

Initialises the main parent window of the program

### 4.41.2 Member Function Documentation

#### 4.41.2.1 `def main.SolarFarmCalculator.evt_calculateTXCableLength_checked( self, event )`

Enables and disables the tx cable length text ctrl when the "Calculate Cable Length" checkbox is toggled

#### 4.41.2.2 `def main.SolarFarmCalculator.evt_closeApp_clicked( self, event )`

Terminates the program when the red cross is clicked on the main window

#### 4.41.2.3 `def main.SolarFarmCalculator.evt_runSimulation_clicked( self, event )`

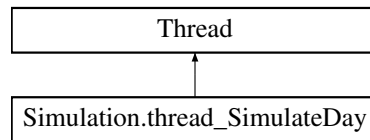
Event that is run when the "Run Simulation" button is clicked. This will validate all the inputs, check for an internet connection and run the simulation if all the inputs are correct. Otherwise an error dialog is shown telling the user what they did wrong

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

- `main.py`

## 4.42 Simulation.thread\_SimulateDay Class Reference

Inheritance diagram for Simulation.thread\_SimulateDay:



### Public Member Functions

- `def \_\_init\_\_`
- `def run`

### Public Attributes

- `timestep_mins`
- `inputQueue`
- `outputQueue`

### 4.42.1 Constructor & Destructor Documentation

4.42.1.1 `def Simulation.thread_SimulateDay.__init__( self, inputQueue, outputQueue, timestep_mins )`

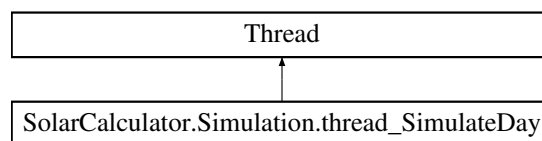
Takes an input of `SimulationDay` objects, runs the simulation for that day and stores the result inside the `SimulationDay` object before pushing it to the output queue

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

- `Simulation.py`

## 4.43 SolarCalculator.Simulation.thread\_SimulateDay Class Reference

Inheritance diagram for `SolarCalculator.Simulation.thread_SimulateDay`:



### Public Member Functions

- `def \_\_init\_\_`
- `def run`

### Public Attributes

- `timestep_mins`
- `inputQueue`
- `outputQueue`

### 4.43.1 Constructor & Destructor Documentation

#### 4.43.1.1 `def SolarCalculator.Simulation.thread.SimulateDay.__init__( self, inputQueue, outputQueue, timestep.mins )`

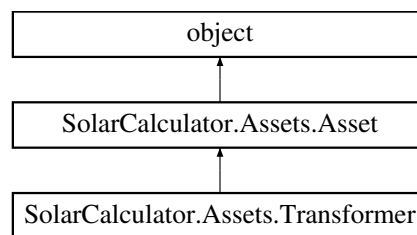
Takes an input of `SimulationDay` objects, runs the simulation for that day and stores the result inside the `SimulationDay` object before pushing it to the output queue

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

- `SolarCalculator/Simulation.py`

## 4.44 SolarCalculator.Assets.Transformer Class Reference

Inheritance diagram for `SolarCalculator.Assets.Transformer`:



### Public Member Functions

- `def __init__`
- `def getVoltage`
- `def getEfficiency`
- `def getVARating`
- `def getScrapValue`

### Public Attributes

- **voltage**
- **efficiency**
- **VARating**

### Additional Inherited Members

#### 4.44.1 Detailed Description

Class that stores the information relating to a transformer.

#### 4.44.2 Constructor & Destructor Documentation

##### 4.44.2.1 `def SolarCalculator.Assets.Transformer.__init__( self, voltage, efficiency, VARating, cost, currency = 'USD', depRate = 0 )`

Initialise the transformer object

#### 4.44.3 Member Function Documentation

##### 4.44.3.1 `def SolarCalculator.Assets.Transformer.getEfficiency ( self )`

Return the efficiency of the transformer between 0 and 1

##### 4.44.3.2 `def SolarCalculator.Assets.Transformer.getScrapValue ( self )`

Return the scrap value of the cable

##### 4.44.3.3 `def SolarCalculator.Assets.Transformer.getVARating ( self )`

Return the rating of the transformer (MVA)

##### 4.44.3.4 `def SolarCalculator.Assets.Transformer.getVoltage ( self )`

Return the high voltage side of the transformer

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

- SolarCalculator/Assets.py

# Index

- `__add__`
  - SolarCalculator::Utils::PyExchangeRates::Money, [33](#)
- `__div__`
  - SolarCalculator::Utils::PyExchangeRates::Money, [33](#)
- `__init__`
  - main::DialogBox\_DateError, [19](#)
  - main::DialogBox\_FatalError, [20](#)
  - main::DialogBox\_GeoCodeError, [21](#)
  - main::DialogBox\_IncompleteForm, [22](#)
  - main::DialogBox\_NoInternet, [22](#)
  - main::DialogBox\_ProgressDialog, [23](#)
  - main::DialogBox\_SimulationResults, [24](#)
  - main::OptionalInputField, [35](#)
  - main::SolarFarmCalculator, [49](#)
  - Simulation::Simulation, [42](#)
  - Simulation::thread\_SimulateDay, [50](#)
  - SolarCalculator::Assets::AC1Cable, [8](#)
  - SolarCalculator::Assets::AC2Cable, [9](#)
  - SolarCalculator::Assets::Asset, [15](#)
  - SolarCalculator::Assets::CircuitBreaker, [16](#)
  - SolarCalculator::Assets::DCCable, [19](#)
  - SolarCalculator::Assets::Financial, [26](#)
  - SolarCalculator::Assets::Inverter, [31](#)
  - SolarCalculator::Assets::Material, [32](#)
  - SolarCalculator::Assets::PVArray, [36](#)
  - SolarCalculator::Assets::PVModule, [38](#)
  - SolarCalculator::Assets::PVPanel, [39](#)
  - SolarCalculator::Assets::Site, [48](#)
  - SolarCalculator::Assets::Transformer, [51](#)
  - SolarCalculator::Simulation::Simulation, [41](#)
  - SolarCalculator::Simulation::thread\_SimulateDay, [51](#)
  - SolarCalculator::Utils::PyExchangeRates::Currency, [17](#)
- `__mul__`
  - SolarCalculator::Utils::PyExchangeRates::Money, [33](#)
- `__str__`
  - SolarCalculator::Utils::PyExchangeRates::Money, [33](#)
- `__sub__`
  - SolarCalculator::Utils::PyExchangeRates::Money, [33](#)
- accumulateDailyInterest
  - SolarCalculator::Assets::Financial, [27](#)
- addToLoan
  - SolarCalculator::Assets::Financial, [27](#)
- amountInBaseCurrency
  - SolarCalculator::Assets::Financial, [27](#)
- closeDialog
  - main::DialogBox\_ProgressDialog, [23](#)
- convert
  - SolarCalculator::Utils::PyExchangeRates::Money, [33](#)
- evt\_calculateTXCableLength\_checked
  - main::SolarFarmCalculator, [49](#)
- evt\_closeApp\_clicked
  - main::SolarFarmCalculator, [49](#)
- evt\_dialogCloseProgram\_clicked
  - main::DialogBox\_FatalError, [20](#)
- evt\_dialogOK\_clicked
  - main::DialogBox\_DateError, [20](#)
  - main::DialogBox\_GeoCodeError, [21](#)
  - main::DialogBox\_IncompleteForm, [22](#)
  - main::DialogBox\_NoInternet, [22](#)
  - main::DialogBox\_SimulationResults, [24](#)
- evt\_runSimulation\_clicked
  - main::SolarFarmCalculator, [49](#)
- getAmount
  - SolarCalculator::Utils::PyExchangeRates::Money, [33](#)
- getAngle
  - SolarCalculator::Assets::PVArray, [36](#)
- getArea
  - SolarCalculator::Assets::PVArray, [36](#)
  - SolarCalculator::Assets::PVModule, [38](#)
  - SolarCalculator::Assets::PVPanel, [39](#)
- getArrayNum
  - SolarCalculator::Assets::Site, [48](#)
- getBaseCurrency
  - SolarCalculator::Assets::Financial, [27](#)
- getCheckboxState
  - main::OptionalInputField, [35](#)
- getCircuitBreakerNum
  - SolarCalculator::Assets::Site, [48](#)
- getCost
  - SolarCalculator::Assets::Asset, [15](#)
- getCurrency
  - SolarCalculator::Assets::Asset, [15](#)
- getCurrencyKey
  - SolarCalculator::Utils::PyExchangeRates::Money, [34](#)
- getCurrentLoanValue
  - SolarCalculator::Assets::Financial, [27](#)

- getDailyMaintenance
  - SolarCalculator::Assets::Financial, 27
- getDegradationRate
  - SolarCalculator::Assets::PVPanel, 39
- getDepRate
  - SolarCalculator::Assets::Asset, 15
- getDepreciatedValue
  - SolarCalculator::Assets::Asset, 15
- getDiameter
  - SolarCalculator::Assets::AC1Cable, 8
  - SolarCalculator::Assets::AC2Cable, 9
  - SolarCalculator::Assets::DCCable, 19
- getEfficiency
  - SolarCalculator::Assets::Inverter, 31
  - SolarCalculator::Assets::PVPanel, 39
  - SolarCalculator::Assets::Transformer, 52
- getElectricalEfficiency
  - Simulation::SimulationDay, 44
  - SolarCalculator::Simulation::SimulationDay, 45
- getElectricalEnergy
  - Simulation::SimulationDay, 44
  - SolarCalculator::Simulation::SimulationDay, 45
- getExchange
  - SolarCalculator::Utils::PyExchangeRates::Money, 34
- getFinancialResults
  - Simulation::Simulation, 43
  - SolarCalculator::Simulation::Simulation, 41
- getInverterNum
  - SolarCalculator::Assets::Site, 48
- getLatitude
  - SolarCalculator::Assets::Site, 48
- getLength
  - SolarCalculator::Assets::AC1Cable, 8
  - SolarCalculator::Assets::AC2Cable, 9
  - SolarCalculator::Assets::DCCable, 19
- getLongitude
  - SolarCalculator::Assets::Site, 48
- getMaterial
  - SolarCalculator::Assets::AC1Cable, 8
  - SolarCalculator::Assets::AC2Cable, 9
  - SolarCalculator::Assets::DCCable, 19
- getModuleNum
  - SolarCalculator::Assets::PVArray, 37
- getModuleType
  - SolarCalculator::Assets::PVArray, 37
- getPanelNum
  - SolarCalculator::Assets::PVMModule, 38
- getPanelType
  - SolarCalculator::Assets::PVMModule, 38
- getPowerFactor
  - SolarCalculator::Assets::Inverter, 31
- getPowerPrice
  - SolarCalculator::Assets::Financial, 27
- getPowerProgress
  - Simulation::Simulation, 43
  - SolarCalculator::Simulation::Simulation, 41
- getPowerResults
  - Simulation::Simulation, 43
  - SolarCalculator::Simulation::Simulation, 41
- getRating
  - SolarCalculator::Assets::PVPanel, 39
- getResistivity
  - SolarCalculator::Assets::Material, 32
- getScrapValue
  - SolarCalculator::Assets::Transformer, 52
- getStrandNum
  - SolarCalculator::Assets::AC1Cable, 8
  - SolarCalculator::Assets::AC2Cable, 9
- getTempCoefficient
  - SolarCalculator::Assets::Material, 32
- getTemperature
  - SolarCalculator::Assets::Site, 48
- getTotalEfficiency
  - Simulation::SimulationDay, 44
  - SolarCalculator::Simulation::SimulationDay, 45
- getTransformerNum
  - SolarCalculator::Assets::Site, 48
- getVARating
  - SolarCalculator::Assets::Transformer, 52
- getVoltage
  - SolarCalculator::Assets::Inverter, 31
  - SolarCalculator::Assets::PVArray, 37
  - SolarCalculator::Assets::PVMModule, 38
  - SolarCalculator::Assets::PVPanel, 39
  - SolarCalculator::Assets::Transformer, 52
- main.DialogBox\_DateError, 19
- main.DialogBox\_FatalError, 20
- main.DialogBox\_GeoCodeError, 20
- main.DialogBox\_IncompleteForm, 21
- main.DialogBox\_NoInternet, 22
- main.DialogBox\_ProgressDialog, 23
- main.DialogBox\_SimulationResults, 23
- main.InputField, 29
- main.OptionalInputField, 35
- main.SolarFarmCalculator, 48
- main::DialogBox\_DateError
  - \_\_init\_\_, 19
  - evt\_dialogOK\_clicked, 20
- main::DialogBox\_FatalError
  - \_\_init\_\_, 20
  - evt\_dialogCloseProgram\_clicked, 20
- main::DialogBox\_GeoCodeError
  - \_\_init\_\_, 21
  - evt\_dialogOK\_clicked, 21
- main::DialogBox\_IncompleteForm
  - \_\_init\_\_, 22
  - evt\_dialogOK\_clicked, 22
- main::DialogBox\_NoInternet
  - \_\_init\_\_, 22
  - evt\_dialogOK\_clicked, 22
- main::DialogBox\_ProgressDialog
  - \_\_init\_\_, 23
  - closeDialog, 23
  - update, 23
- main::DialogBox\_SimulationResults



- \_\_init\_\_, 24
  - evt\_dialogOK\_clicked, 24
- main::InputField
  - setFieldValue, 30
  - setLabelColour, 30
  - validateField, 30
- main::OptionalInputField
  - \_\_init\_\_, 35
  - getCheckboxState, 35
  - validateField, 35
- main::SolarFarmCalculator
  - \_\_init\_\_, 49
  - evt\_calculateTXCableLength\_checked, 49
  - evt\_closeApp\_clicked, 49
  - evt\_runSimulation\_clicked, 49
- makeLoanPayment
  - SolarCalculator::Assets::Financial, 27
- runFinancial
  - Simulation::Simulation, 43
  - SolarCalculator::Simulation::Simulation, 41
- runPower
  - Simulation::Simulation, 43
  - SolarCalculator::Simulation::Simulation, 41
- setAmount
  - SolarCalculator::Utils::PyExchangeRates::Money, 34
- setElectricalEfficiency
  - Simulation::SimulationDay, 44
  - SolarCalculator::Simulation::SimulationDay, 46
- setElectricalEnergy
  - Simulation::SimulationDay, 44
  - SolarCalculator::Simulation::SimulationDay, 46
- setFieldValue
  - main::InputField, 30
- setLabelColour
  - main::InputField, 30
- setTotalEfficiency
  - Simulation::SimulationDay, 44
  - SolarCalculator::Simulation::SimulationDay, 46
- Simulation.Simulation, 42
- Simulation.SimulationDay, 43
- Simulation.thread\_SimulateDay, 50
- Simulation::Simulation
  - \_\_init\_\_, 42
  - getFinancialResults, 43
  - getPowerProgress, 43
  - getPowerResults, 43
  - runFinancial, 43
  - runPower, 43
- Simulation::SimulationDay
  - getElectricalEfficiency, 44
  - getElectricalEnergy, 44
  - getTotalEfficiency, 44
  - setElectricalEfficiency, 44
  - setElectricalEnergy, 44
  - setTotalEfficiency, 44
- Simulation::thread\_SimulateDay
  - \_\_init\_\_, 50
- SolarCalculator.Assets.AC1Cable, 7
- SolarCalculator.Assets.AC2Cable, 8
- SolarCalculator.Assets.Asset, 14
- SolarCalculator.Assets.CircuitBreaker, 16
- SolarCalculator.Assets.DCCable, 18
- SolarCalculator.Assets.Financial, 26
- SolarCalculator.Assets.Inverter, 30
- SolarCalculator.Assets.Material, 31
- SolarCalculator.Assets.PVArray, 36
- SolarCalculator.Assets.PVModule, 37
- SolarCalculator.Assets.PVPanel, 38
- SolarCalculator.Assets.Site, 47
- SolarCalculator.Assets.Transformer, 51
- SolarCalculator.GUI.ApplicationFrame, 10
- SolarCalculator.GUI.DateError, 17
- SolarCalculator.GUI.FatalError, 25
- SolarCalculator.GUI.GeoCodeError, 27
- SolarCalculator.GUI.IncompleteForm, 28
- SolarCalculator.GUI.NoInternet, 34
- SolarCalculator.GUI.SimulationResults, 46
- SolarCalculator.Simulation.Simulation, 40
- SolarCalculator.Simulation.SimulationDay, 45
- SolarCalculator.Simulation.thread\_SimulateDay, 50
- SolarCalculator.Utils.PyExchangeRates.AccessData-Failure, 9
- SolarCalculator.Utils.PyExchangeRates.BadAppID, 15
- SolarCalculator.Utils.PyExchangeRates.Currency, 17
- SolarCalculator.Utils.PyExchangeRates.Exchange, 24
- SolarCalculator.Utils.PyExchangeRates.InvalidCurrency-Key, 30
- SolarCalculator.Utils.PyExchangeRates.Money, 32
- SolarCalculator.Utils.PyExchangeRates.SimpleObject-Encoder, 40
- SolarCalculator.Utils.ReverseGeocode.CountryNot-Found, 16
- SolarCalculator::Assets::AC1Cable
  - \_\_init\_\_, 8
  - getDiameter, 8
  - getLength, 8
  - getMaterial, 8
  - getStrandNum, 8
- SolarCalculator::Assets::AC2Cable
  - \_\_init\_\_, 9
  - getDiameter, 9
  - getLength, 9
  - getMaterial, 9
  - getStrandNum, 9
- SolarCalculator::Assets::Asset
  - \_\_init\_\_, 15
  - getCost, 15
  - getCurrency, 15
  - getDepRate, 15
  - getDepreciatedValue, 15
- SolarCalculator::Assets::CircuitBreaker
  - \_\_init\_\_, 16
- SolarCalculator::Assets::DCCable
  - \_\_init\_\_, 19

- getDiameter, [19](#)
- getLength, [19](#)
- getMaterial, [19](#)
- SolarCalculator::Assets::Financial
  - \_\_init\_\_, [26](#)
  - accumulateDailyInterest, [27](#)
  - addToLoan, [27](#)
  - amountInBaseCurrency, [27](#)
  - getBaseCurrency, [27](#)
  - getCurrentLoanValue, [27](#)
  - getDailyMaintenance, [27](#)
  - getPowerPrice, [27](#)
  - makeLoanPayment, [27](#)
- SolarCalculator::Assets::Inverter
  - \_\_init\_\_, [31](#)
  - getEfficiency, [31](#)
  - getPowerFactor, [31](#)
  - getVoltage, [31](#)
- SolarCalculator::Assets::Material
  - \_\_init\_\_, [32](#)
  - getResistivity, [32](#)
  - getTempCoefficient, [32](#)
- SolarCalculator::Assets::PVArray
  - \_\_init\_\_, [36](#)
  - getAngle, [36](#)
  - getArea, [36](#)
  - getModuleNum, [37](#)
  - getModuleType, [37](#)
  - getVoltage, [37](#)
- SolarCalculator::Assets::PVModule
  - \_\_init\_\_, [38](#)
  - getArea, [38](#)
  - getPanelNum, [38](#)
  - getPanelType, [38](#)
  - getVoltage, [38](#)
- SolarCalculator::Assets::PVPanel
  - \_\_init\_\_, [39](#)
  - getArea, [39](#)
  - getDegradationRate, [39](#)
  - getEfficiency, [39](#)
  - getRating, [39](#)
  - getVoltage, [39](#)
- SolarCalculator::Assets::Site
  - \_\_init\_\_, [48](#)
  - getArrayNum, [48](#)
  - getCircuitBreakerNum, [48](#)
  - getInverterNum, [48](#)
  - getLatitude, [48](#)
  - getLongitude, [48](#)
  - getTemperature, [48](#)
  - getTransformerNum, [48](#)
- SolarCalculator::Assets::Transformer
  - \_\_init\_\_, [51](#)
  - getEfficiency, [52](#)
  - getScrapValue, [52](#)
  - getVARating, [52](#)
  - getVoltage, [52](#)
- SolarCalculator::Simulation::Simulation
  - \_\_init\_\_, [41](#)
  - getFinancialResults, [41](#)
  - getPowerProgress, [41](#)
  - getPowerResults, [41](#)
  - runFinancial, [41](#)
  - runPower, [41](#)
- SolarCalculator::Simulation::SimulationDay
  - getElectricalEfficiency, [45](#)
  - getElectricalEnergy, [45](#)
  - getTotalEfficiency, [45](#)
  - setElectricalEfficiency, [46](#)
  - setElectricalEnergy, [46](#)
  - setTotalEfficiency, [46](#)
- SolarCalculator::Simulation::thread\_SimulateDay
  - \_\_init\_\_, [51](#)
- SolarCalculator::Utils::PyExchangeRates::Currency
  - \_\_init\_\_, [17](#)
- SolarCalculator::Utils::PyExchangeRates::Exchange
  - withdraw, [25](#)
- SolarCalculator::Utils::PyExchangeRates::Money
  - \_\_add\_\_, [33](#)
  - \_\_div\_\_, [33](#)
  - \_\_mul\_\_, [33](#)
  - \_\_str\_\_, [33](#)
  - \_\_sub\_\_, [33](#)
  - convert, [33](#)
  - getAmount, [33](#)
  - getCurrencyKey, [34](#)
  - getExchange, [34](#)
  - setAmount, [34](#)
- update
  - main::DialogBox\_ProgressDialog, [23](#)
- validateField
  - main::InputField, [30](#)
  - main::OptionalInputField, [35](#)
- withdraw
  - SolarCalculator::Utils::PyExchangeRates::Exchange, [25](#)