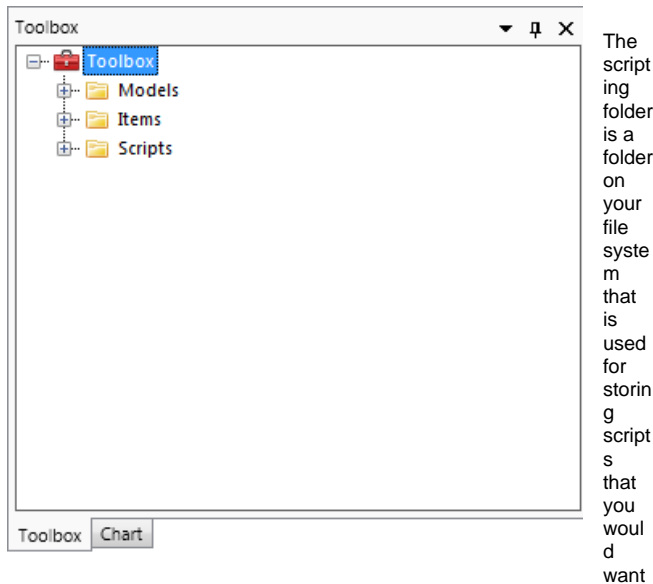# Toolbox overview

The toolbox in Delta-Shell consists of three parts. The first two parts "Models" and "Items" contain lists of models and items that can be added to your project. The third part "Scripts" shows all the folders and (python) script files that are in the scripting folder.

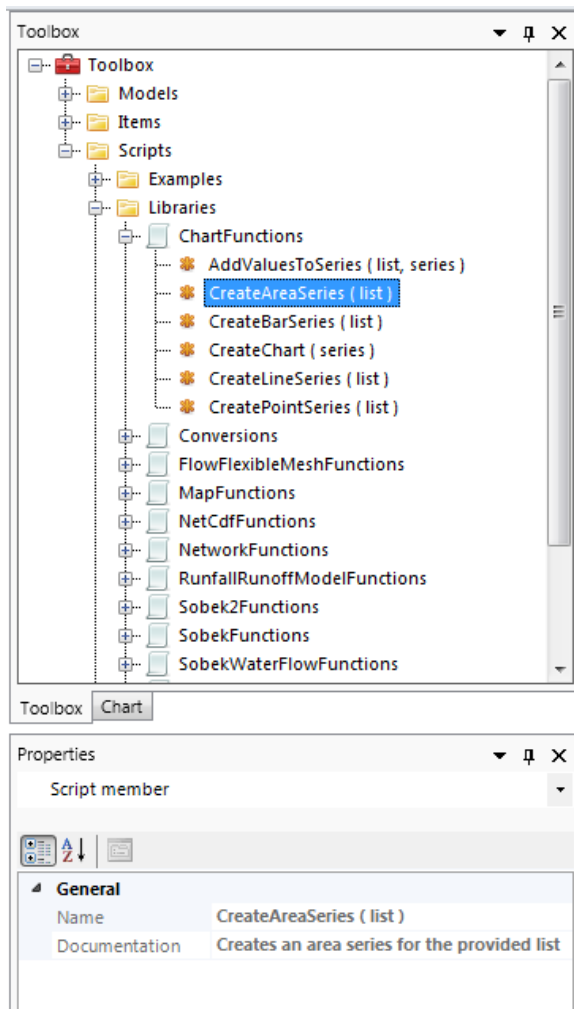 The scripting folder is a folder on your file system that is used for storing scripts that you would want to share between projects. This folder is synchronized with the file system so that you can manage the scripts in the toolbox and with your file browser (windows explorer). To change the location of the scripting folder right click on the toolbox node and select "Change scripting folder"

Exploring the scripting folder

The toolbox gives you an easy way to work and manage the scripts in the scripting folder.

By using the + sign next to the file you can see all the declared functions and classes that are in that script. By selecting the script or one of the functions you can also see the documentation of the selection in the property grid.

By double clicking one of the functions you will open the scripting editor at the start of the declaration of the function.

Working with scripts

After selecting a script in the toolbox and right clicking it, you get the following options.

Function	Description
Edit script	Opens the scripting editor for this script (see Scripting editor overview)
Run file	Runs the whole script without opening the editor
Insert code	Inserts the code of the selected script into the current script (the script open in the scripting editor)
Add as import	Adds the selected script as import to the current script (the script open in the scripting editor)
Open location with windows explorer	Opens an instance of windows explorer for the location of the selected script
Rename	Rename the script (also on file system)
Delete	Deletes the selected script (also on file system)

Working with folders under scripting folder

After selecting a folder under the scripting folder script in the toolbox and right clicking it, you get the following options.

Function	Description
Open location with windows explorer	Opens an instance of windows explorer for the location of the selected folder
Add new folder	Adds a new folder under the current selected folder and adds a __init__.py file to that folder (to make it work as import).
Add new script	Adds a new (python) script to the folder
Import script from existing	Adds a new (python) script to the folder with the content of an existing script (like a file copy)
Add as import	Adds the folder as import to the current script
Rename	Rename the folder (also on file system)
Delete	Deletes the selected folder (also on file system)