# Week 9 (review):

* *Graphical User Interface* (GUI) with PyQT5 (QTWidgets, QTGui, QTCore, QTDesigner, pyuic5)
* Used (& modified) our classes for rankine and steam with a GUI to allow user to control inputs.
* ‘*signal’* and ‘*slot’* mechanism used to connect signals issued by a widget to an action (slot).

# Week 10:

1. A bit more on slots and signals:

## Signals and Slots: Custom Signals

## Events and Event Filters

1. Reading and processing information from a file containing data: ‘*A Word about Parsing Text*’
   1. Get the path to your file from a dialog box.
   2. Opening and reading the file.
   3. Parsing the file into useful data with the *keys and values*.
   4. Parsing a file with know structure.

c. The string functions: split(), strip(), replace()

d. Basic error handling: try/except structure

1. Displaying information:
   1. Graphical feedback with matplotlib.pyplot
      1. from matplotlib.backends.backend\_qt5agg import FigureCanvasQTAgg
      2. from matplotlib.figure import Figure
      3. canvas: the widget to add to your GUI.
      4. figure: an object of type Figure that holds subplots (e.g., plots laid out on a grid).
      5. axes: the axes where plotting and formatting are done.
2. Modifying data: the Model-View-Controller (MVC) design pattern

MVC is a way to organize our programs so that each part has a specific role.

* 1. Model:

The model contains the data for our object(s). The model should not contain display elements.

* 1. View:

The view is used to display the data of our model to the user and provide a way for the user to change the data. Think CLI or GUI.

* 1. Controller:

The controller coordinates the mapping of data to the view and handles changes to the data from the view. The controller should hold a reference to the model and to the view so that it can communicate effectively in both directions. Most of our coding goes into the controller.

* 1. Pipe network example
     1. QTreeWidget
        1. Editable, sortable
        2. Customizing interaction with install.eventFilter()
        3. Updating the PipeNetwork model
     2. Multiple levels in the TreeWidget
  2. The Rankine class