**Introduction:**

**Automation vs. Plug-ins:**

* Automation
  + Solving specific problems:
    - Existing solutions require a lot of boring repetitive tasks.
    - Existing solutions are insufficient to address the question you need to address.
  + Often intended to be used just once or a few times
  + No need for fancy interfaces, intensive error trapping, etc.
* Plug-Ins:
  + Solving General Problems.
  + Intended to be distributed for other people to use.
  + Generally require a level of programming competence.
    - Object Oriented
    - Event Driven programming
    - Error trapping
    - Graphical User Interface (GUI).

A package is just a directory that contains one or more modules, a module is just a directory that contains one or more scripts, and a script is just a text file containing python code.

* A logical way to break down how python is organised.

**Core Modules:**

There are a number of core modules already in PYQGIS: (All the classes in PYQGIS are prefixed with Qgs before the class name, to avoid confusion about what class you are using if you import anything).

1. Qgis.core
   * This one you will use the most (Contains Classes like QgsVectorLayer, QgsFeature, QgsField, etc.)
   * The instance of the QgsVector Layer will be whatever active layer you have open in QGIS (Can have multiple instances in that case).
   * This class will have properties and methods that will allow you to work with active layers.
   * You can also get access to the component parts of the active layer (its features).
   * The active layer is made up of features and each feature is an instance of the QgsFeature class. It has its own properties and methods that allow you to do things like see what Field it has.
   * The Field is an instance of the QgsField class. We can use it to get and set individual attribute values.
2. Qgis.gui:
   * Provides access to the QGIS user interface.
3. Qgis.utils:
4. Qgis.analysis:
   * Allows you to perform complex raster analysis.
5. Qgis.networkanalysis:
   * Allows you to do things like routing through a network
6. Qgis.server:
   * Allows you to customise QGIS server capabilities.

QGIS was written in C++, so PYQGIS has to access C++ functions in order to execute a task.

* Any code that calls upon another piece of code to work (for example a function calling a function), are called wrappers.
* All PYQGIS classes are wrappers on the C++ classes.
* SIP : Is a built-in python tool for accessing C++ classes so users don’t have to worry about it.

QT:

* QT is a framework for creating graphical user interfaces.
* QGIS itself uses QT for its main application

Resources:

* PYQGIS Developers Cookbook.
* PYQGIS API Documentation (List of all classes, methods in PYQGIS)
* StackExchange
* Youtube