

Unidad 07

QGIS y Python

Curso de geoprocесamiento
de datos con Python
2016



Cayetano Benavent Viñuales

Analista GIS en Geographica
cayetano.benavent@geographica.gs

Unidad 07 - Sumario de contenidos

1. Fundamentos de PyQGIS.
2. Geoprocesamiento en QGIS desde el intérprete de Python.
3. Creación de herramientas de geoprocesamiento en QGIS.
4. Bibliografía



El hecho de que una parte muy importante de QGIS esté escrito en **Python***, hace que cualquier desarrollo (o librería externa) en este lenguaje tiene un encaje muy directo en dicha aplicación.

Ello es clave, pues casi todas las aplicaciones del mundo geoespacial tienen API o binding a Python, ya sean software libre o no (recordemos ArcPy de ArcGIS, su potente entorno de geoprocesamiento).

** La casi totalidad de los plugins y herramientas de geoprocesamiento están escritos en Python.*



Una de las ventajas de la enorme integración de **Python** con QGIS es que podemos trabajar con este lenguaje en varios entornos diferentes dentro de la aplicación.

Cada uno de estos entornos ofrece tanto funcionalidades como niveles de dificultad y complejidad muy distintos.

A continuación veremos las distintas opciones que QGIS nos ofrece para trabajar con Python.



- Escribiendo código desde la consola interactiva de Python.

A screenshot of the Python Console window in QGIS. The window has a title bar 'Python Console' and a toolbar with icons for running, saving, and other actions. The main area contains a list of instructions: '1 Python Console', '2 Use iface to access QGIS API interface or Type help(iface) for more info', and '3'. Below this, there is a prompt '>>>' indicating where to enter code.

```
Python Console
1 Python Console
2 Use iface to access QGIS API interface or Type help(iface) for more info
3
>>>
```

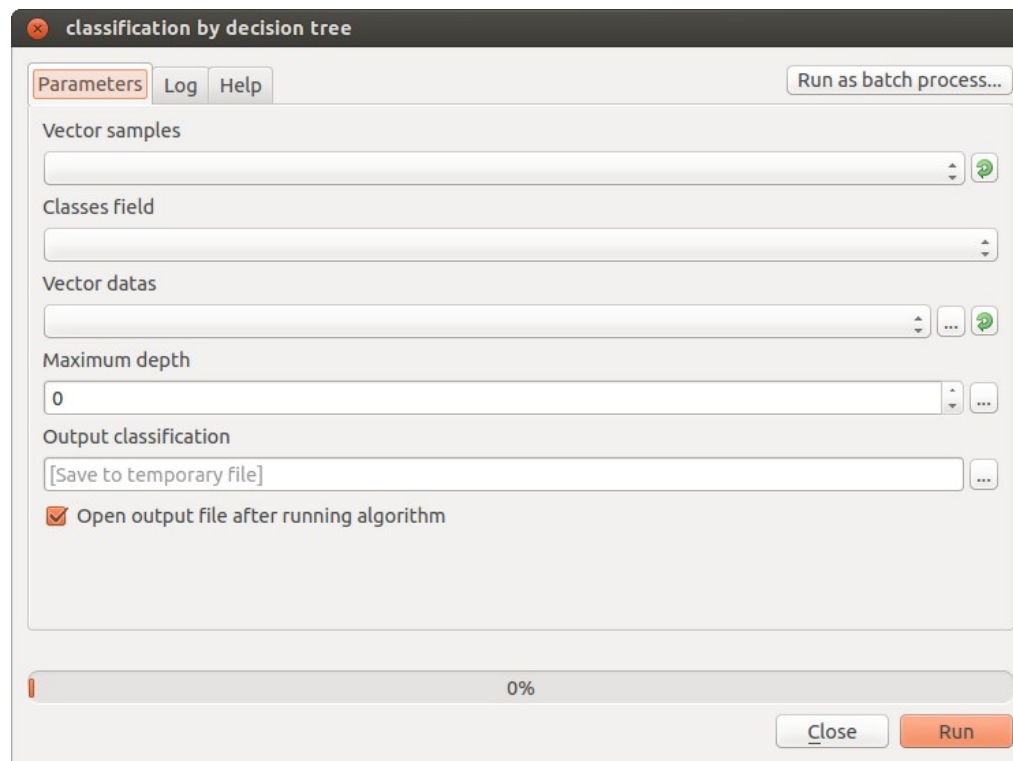


- Escribiendo código (o abriendo scripts externos) y ejecutándolo desde el editor integrado con la consola interactiva.

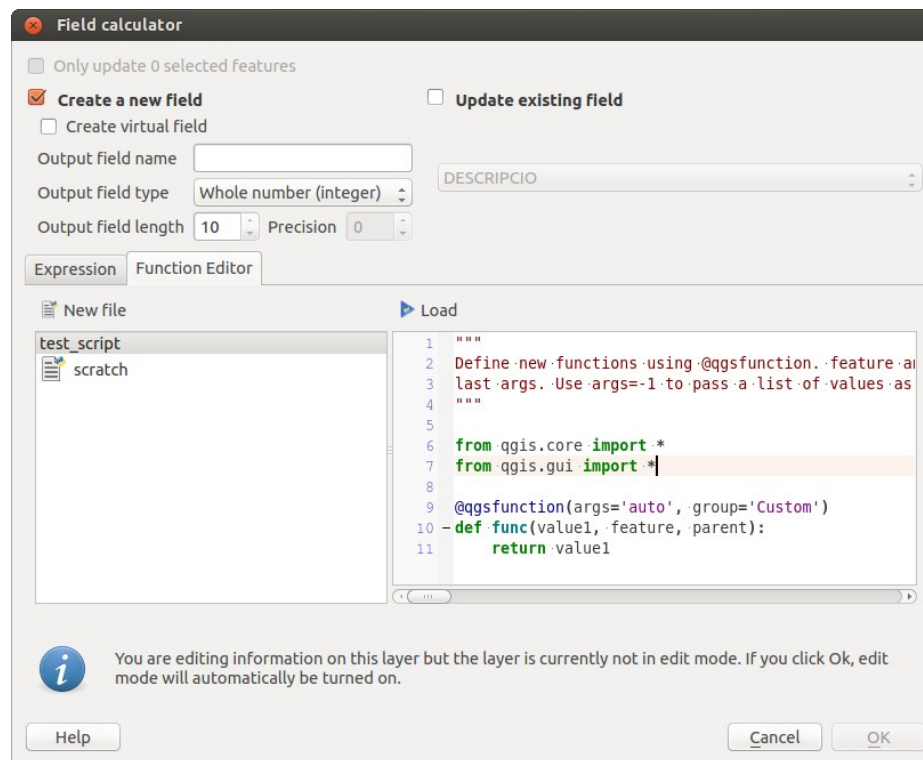
```
Python Console
1 Python Console
2 Use iface to access QGIS API interface or Type help(if
3 ace) for more info
4 >>> import processing
5 >>> from PyQt4.QtCore import *
6 >>> from PyQt4.QtGui import *
>>>

temp_bbox_creation.py create_poly.py
7 - coords = [
8     (-174.934,-23.6423), (-174.948,-23.6485),
9     (-174.96,-23.619), (-174.909,-23.5065),
10    (-174.728,-23.1291), (-174.72,-23.1119),
11    (-174.648,-23.0879), (-174.6,-22.988),
12    (-174.552,-22.968), (-174.547,-22.9825),
13    (-174.648,-23.184), (-174.716,-23.2504),
14    (-174.716,-23.3303), (-174.716,-23.3417),
15    (-174.918,-23.6251), (-174.934,-23.6423)
16 ]
17
18 - def createPoly(coords):
19     pl_ly = QgsVectorLayer("Polygon", "temporary_poi
20
21     prov = pl_ly.dataProvider()
22
23     feat = QgsFeature()
```

- Escribiendo geoprosesos en Python en el Processing Toolbox.



- Escribiendo funciones Python en la calculadora de campos.



- Desarrollando plugins. Podemos subirlos (o no) al hub de QGIS (<http://plugins.qgis.org>) e instalarlos para su uso dentro de la interfaz de QGIS.



A screenshot of the QGIS Python Plugins Repository website. The page shows a list of installed plugins under the heading 'My Plugins'. The table lists three plugins: 'MBTiles images extract', 'Search & format EPSG CRS Plugin', and 'Worldwide night'. Each plugin entry includes a checkmark, a star icon, a download icon, the author's name, the creation date, the number of stars (votes), and version information for stable and experimental releases. A yellow banner at the bottom of the table states 'Deprecated plugins are printed in red.' The left sidebar contains navigation links for 'Share a plugin', 'Plugins', and 'Plugin tags'.

- Desarrollando aplicaciones independientes con PyQGIS. De esta manera podemos acceder a todo el conjunto de herramientas de análisis y procesamiento de QGIS sin hacer uso de su GUI.
- Escribir funciones PL/Python y ejecutarlas desde el administrador de bases de datos para PostgreSQL/PostGIS.
- Ejecutar código Python en el arranque de la aplicación.



A continuación vamos a desarrollar una serie de ejercicios prácticos para ver algunas de las posibilidades de uso de Python con QGIS mencionadas.



Una de las ventajas de la enorme integración de **Python** con QGIS es que podemos trabajar con este lenguaje en varios entornos diferentes dentro de la aplicación.

Cada uno de estos entornos ofrece tanto funcionalidades como niveles de dificultad y complejidad muy distintos.

A continuación veremos las distintas opciones que QGIS nos ofrece para trabajar con Python.

Geoprocesamiento en QGIS desde el intérprete de Python

Para entender mejor cómo llevar a cabo tareas de geoprocesamiento en QGIS desde el intérprete de Python, realizaremos un ejercicio práctico siguiendo el Notebook unit07_01.



Jupyter Notebook

Geoprocesamiento en QGIS desde el intérprete de Python

Para entender mejor cómo llevar a cabo tareas de geoprocesamiento en QGIS desde el intérprete de Python, haciendo uso de scripts almacenados externamente (editor integrado), realizaremos un ejercicio práctico siguiendo el Notebook unit07_02.



Jupyter Notebook

Creación de herramientas de geoprocésamiento en QGIS

Para entender mejor cómo construir herramientas de geoprocésamiento en QGIS integradas en el processing toolbox, realizaremos un ejercicio práctico siguiendo el Notebook unit07_03.



Jupyter Notebook

Selección bibliográfica:

- QGIS Developers (2016): “PyQGIS developer cookbook”. QGIS Project.
- Lawhead, J. (2015): “QGIS Python Programming Cookbook”. Packtpub.
- Sherman, G. (2014): “The PyQGIS Programmer's Guide”. Locate Press.