



Section : 13. Programming

Module : 13.2. QGIS Python Plugin Creation



Creating a QGIS plugin in Python

"In this module you will create a simple QGIS plugin from scratch."

If you understand a little python and the basic structure of a QGIS plugin, creating your own plugin is relatively straightforward. The QGIS documentation guide provides a detailed overview of this process, which is the basis for this lesson. Here are the three essential resources for getting up to speed with writing a plugin in QGIS:

1. [Plugin Writing Guide](#)
2. [QGIS Python API Documentation](#)
3. [The QGIS Python Cookbook](#)

TABLE OF CONTENTS

A GENTLE INTRODUCTION TO GIS

TRAINING MANUAL

USER GUIDE/MANUAL

USER GUIDE/MANUAL PDF'S

PYQGIS COOKBOOK

Introduction

Loading Projects

Loading Layers

Using Raster Layers

Using Vector Layers

Geometry Handling

Projections Support

Using the Map Canvas

Map Rendering and Printing

Expressions, Filtering and

Calculating Values

Reading And Storing Settings

Communicating with the user

Using Raster Layers



Warning

Despite our constant efforts, information beyond this line may not be updated for QGIS 3. Refer to <https://qgis.org/pyqgis/master> for the python API documentation or, give a hand to update the chapters you know about. Thanks.

- [Layer Details](#)
- [Renderer](#)
 - [Single Band Rasters](#)
 - [Multi Band Rasters](#)
- [Query Values](#)

The code snippets on this page needs the following imports if you're outside the pyqgis console:

```
from qgis.core import (  
    QgsRasterLayer,  
    QgsColorRampShader,  
    QgsSingleBandPseudoColorRenderer  
)
```

You try:

Goal: Create a simple 'hello world' plugin in QGIS