

Setting up a PyQGIS development machine

1. Install QGIS
2. Install ‘Plugin Builder’ and ‘Plugin Reloader’ QGIS plugins
3. Install ‘pip’
4. Install the ‘Plugin Builder Tool’ (pb_tool)
5. Mac/Linux: Install Qt Creator
6. Mac/Linux: Setup the environment paths
7. Install PyCharm Pro
8. Configure PyCharm Pro



Check if you have the OSGeo4W package installed, and correctly:

- Find OSGeo4W shell and run shell as administrator
- Type ‘pyrcc4’ and press enter
- If it gives an error move to the next step
- Find the Qt Designer/Creator shortcut
- Find the QGIS shortcut



Download and run the OSGeo4W Network installer

<https://trac.osgeo.org/osgeo4w/>

- Choose *Advanced Install*
- Select to install: Desktop -> qgis; Libs -> qt4-devel; Libs -> setuptools

OSGeo4W
Your Open Source Compass for Windows

Search

Login | Help/Guide | About Trac | Preferences

Wiki Timeline Roadmap Browse Source View Tickets Search

Start Page | Index | History

wiki: [WikiStart](#)

OSGeo4W

[Japanese](#) [French](#) [Polish](#) [Deutsch](#)

This is the web site, wiki and issue tracking database for the **OSGeo4W project**. OSGeo4W is a binary distribution of a broad set of open source geospatial software for Win32 environments (Windows XP, Vista, etc). OSGeo4W includes [GDAL/OGR](#), [GRASS](#), [MapServer](#), [OpenEV](#), [uDig](#), [QGIS](#) as well as [many other packages](#) (over 150).

This project is under the umbrella of the **Open Source Geospatial Foundation**, <http://www.osgeo.org/>.

Only authenticated (logged in) users can submit and edit tickets, and modify the wiki. Use your [OSGeo userid/password](#) to login.

[About licenses](#)

Quick Start for OSGeo4W Users

1. Download the [32bit](#) or [64bit](#) OSGeo4W network installer
2. Run the installer.
3. Select *Express Install*, and *Next*.
4. Pick one or more packages to install, and *Next*.
5. The selected packages and their required subpackages will be downloaded and installed automatically.

Table of Contents

Quick Start for OSGeo4W Users
OSGeo4W User Information
OSGeo4W Packager (Developer) Information
Communication / Coordination

<http://www.kyngchaos.com/software/qgis>

- Install GDAL Complete framework
- Install Matplotlib
- Install QGIS



"The beast is actively interested only in now, and, as it is always now and always shall be, there is an eternity of time for the accomplishment of objects."
- the wisdom of Tarzan

KYNGCHAOS
WIKI

You are here: KyngChaos Wiki » Software » QGIS

Login

Main Menu

Anime & Manga
Mac OS X Porting
Software

Software Menu

SumomOS
UNIX Porting Downloads

- Frameworks
- GRASS GIS
- QGIS
- PostgreSQL
- PHP
- MapServer
- Python Modules
- Download Archive

FAQ
Installation Guide
Developer Notes

QGIS

Mac OS X installers for QGIS. For OS X Lion and newer. Install the Current version to stay up to date on features. Install the Long Term Support version for feature stability for a year.

Current

Requirements:

- GDAL Complete 1.11 framework package
- Matplotlib Python module

Optional:

- other Python Modules for plugins

Download:

- QGIS 2.12.0-1 [181.6 MiB].

See the QGIS website for more information about, and help with, QGIS.

QGIS includes its own internal copies of GRASS, Orfeo Toolbox, SAGA and TauDEM.

Long Term Support

All required items are included on the disk image.

Optional:

- other Python Modules for plugins

Table of Contents

- QGIS
 - Current
 - Long Term Support
 - Development Builds

Install ‘Plugin Builder’

Plugins | Installed (260)

All
Installed
Not installed
Upgradeable
New
Invalid
Settings

Search plugin

MetaSearch Catalogue Client
mmqgis
MultiEdit
multiPrint
norGIS ALKIS-Einbindung
NTv2 Datum Transformations
Open Aerial Map (OAM)
OpenLayers Plugin
OSMDownloader
Oursins
PDOK BAG Geocoder
PDOK services plugin
Plugin Builder
Plugin Reloader
PostGIS geoprocessing tools
pyUPVBib
QGIS Cloud Plugin
QuickMultiAttributeEdit
QuickWKT
Raster Terrain Analysis plugin
Remote Debug
Road graph plugin
Search & format EPSG CRS Plugin
SLD4raster
Spatial Query Plugin

Plugin Builder

Creates a QGIS plugin template for use as a starting point in plugin development

Create a template for a QGIS plugin

★★★★★ 48 rating vote(s), 45411 downloads

Tags: development
More info: [homepage](#) [tracker](#) [code repository](#)

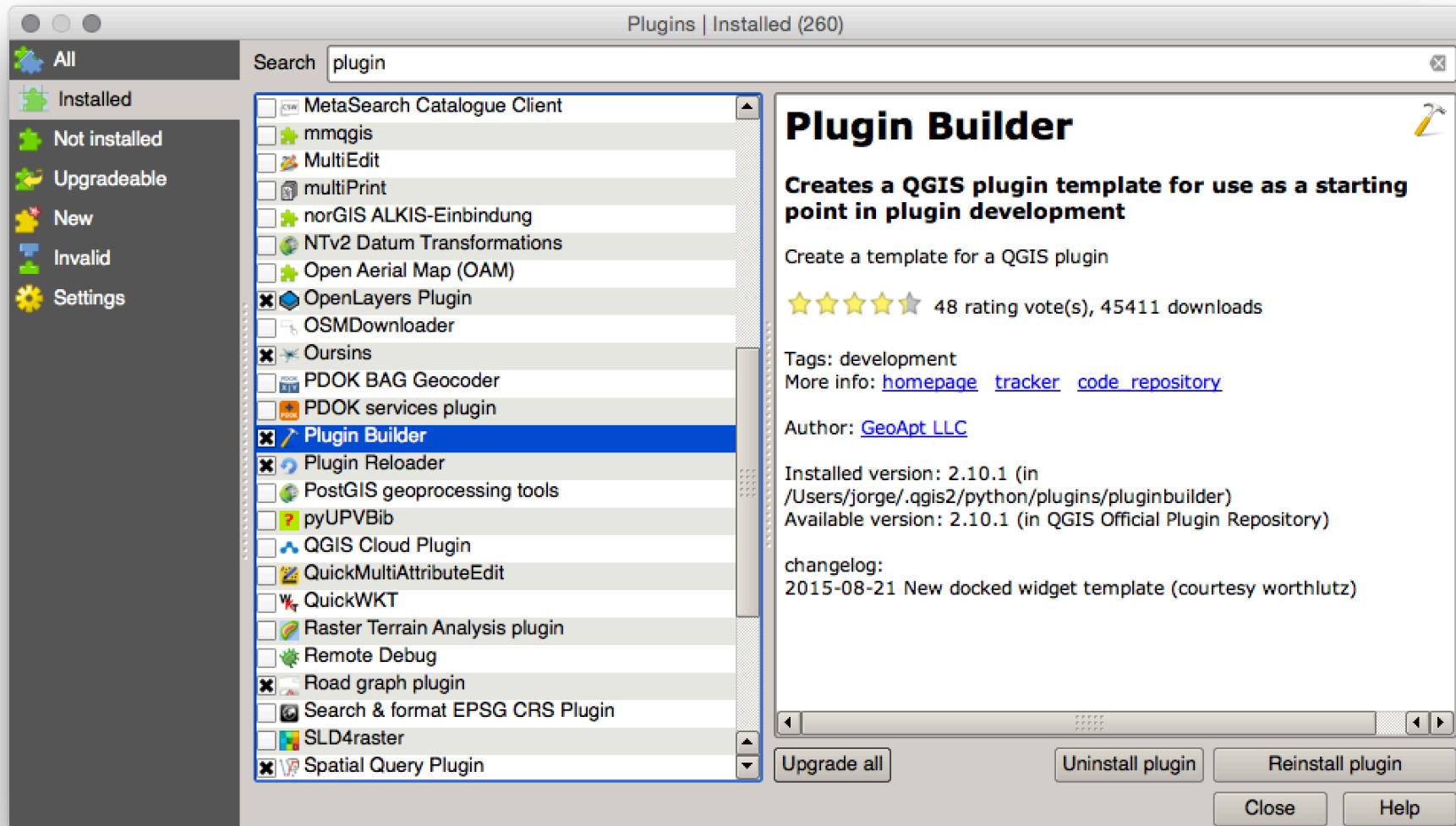
Author: [GeoApt LLC](#)

Installed version: 2.10.1 (in /Users/jorge/.qgis2/python/plugins/pluginbuilder)
Available version: 2.10.1 (in QGIS Official Plugin Repository)

changelog:
2015-08-21 New docked widget template (courtesy worthlutz)

Upgrade all Uninstall plugin Reinstall plugin

Close Help



Plugins | Installed (260)

All Search plugin

Installed

Not installed

Upgradeable

New

Invalid

Settings

MetaSearch Catalogue Client

mmqgis

MultiEdit

multiPrint

norGIS ALKIS-Einbindung

NTv2 Datum Transformations

Open Aerial Map (OAM)

OpenLayers Plugin

OSMDownloader

Oursins

PDOCK BAG Geocoder

PDOCK services plugin

Plugin Builder

Plugin Reloader

PostGIS geoprocessing tools

pyUPVBib

QGIS Cloud Plugin

QuickMultiAttributeEdit

QuickWKT

Raster Terrain Analysis plugin

Remote Debug

Road graph plugin

Search & format EPSG CRS Plugin

SLD4raster

Spatial Query Plugin

This plugin is experimental

Plugin Reloader

Reloads a chosen plugin in one click (only useful for Python Plugin Developers)

★★★★★ 33 rating vote(s), 12172 downloads

Tags: reloader, reload, python, development, developer
More info: [homepage](#) [tracker](#) [code repository](#)

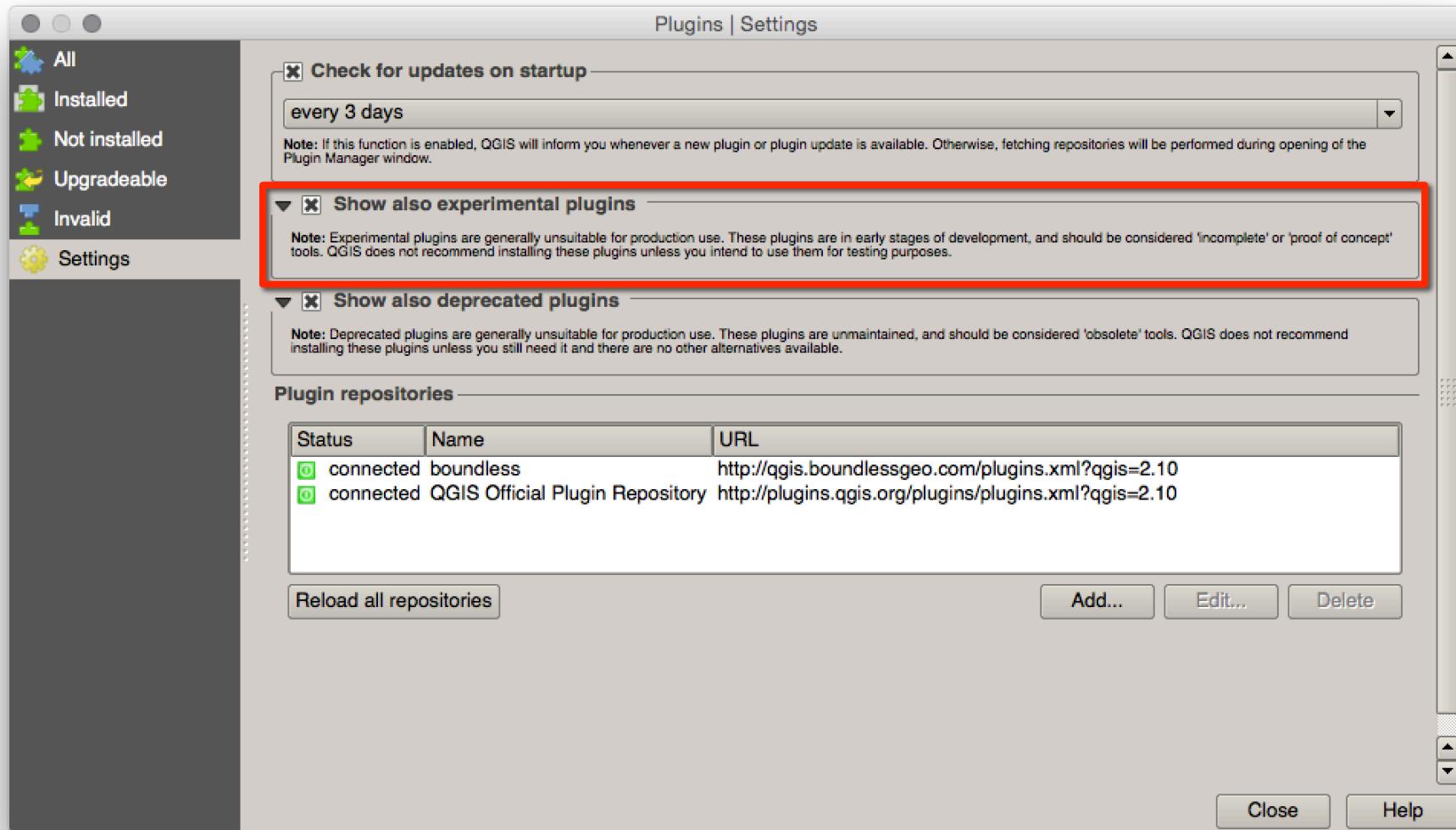
Author: [Borys Jurgiel](#)

Installed version: 0.6.3 (in /Users/jorge/.qgis2/python/plugins/plugin_reloader)
Available version: 0.6.3 (in QGIS Official Plugin Repository)

Upgrade all Uninstall plugin Reinstall plugin

Close Help

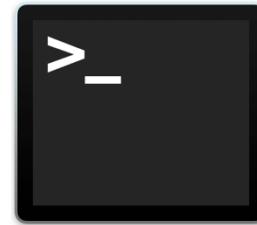
The screenshot shows the QGIS Plugin Manager interface. On the left, there's a sidebar with categories like All, Installed, Not installed, etc. The main area shows a list of plugins. The 'Plugin Reloader' plugin is selected and highlighted with a red box. Above the plugin's details, a yellow callout box contains the text 'This plugin is experimental'. The right side of the window displays the plugin's title 'Plugin Reloader', a brief description ('Reloads a chosen plugin in one click (only useful for Python Plugin Developers)'), its rating (5 stars from 33 votes), download count (12172), tags (reloader, reload, python, development, developer), and more info links (homepage, tracker, code repository). It also shows the author as 'Borys Jurgiel'. Below this, it lists the installed version (0.6.3) and available version (0.6.3) along with their respective paths. At the bottom, there are buttons for upgrading all, uninstalling, reinstalling, closing, and getting help.



Install ‘pip’

Pip is a command line Python package installer.

- Check if you have pip:
 - Windows: start the OSGeo4W Shell command line window as administrator;
 - Mac OSX: start the terminal;
 - Type ‘pip’ on the command prompt;
 - Press Enter.



```
jorge-macbook:Downloads jorge$ pip
Usage:
  pip <command> [options]

Commands:
  install           Install packages.
  uninstall         Uninstall packages.
  freeze            Output installed packages in requirements format.
  list              List installed packages.
  show              Show information about installed packages.
  search             Search PyPI for packages.
  wheel              Build wheels from your requirements.
  help               Show help for commands.

General Options:
  -h, --help          Show help.
  --isolated          Run pip in an isolated mode, ignoring
                      environment variables and user configuration.
  -v, --verbose        Give more output. Option is additive, and can be
                      used up to 3 times.
  -V, --version        Show version and exit.
  -q, --quiet          Give less output.
  --log <path>         Path to a verbose appending log.
  --proxy <proxy>      Specify a proxy in the form
                      [user:password@]proxy.server:port.
  --retries <retries> Maximum number of retries each connection should
                      attempt (default 5 times).
  --timeout <sec>       Set the socket timeout (default 15 seconds).
  --exists-action <action> Default action when a path already exists:
                      (s)witch, (i)gnore, (w)ipe, (b)ackup.
  --trusted-host <hostname> Mark this host as trusted, even though it does
                      not have valid or any HTTPS.
  --cert <path>          Path to alternate CA bundle.
  --client-cert <path> Path to SSL client certificate, a single file
                      containing the private key and the certificate
                      in PEM format.
  --cache-dir <dir>     Store the cache data in <dir>.
  --no-cache-dir        Disable the cache.
  --disable-pip-version-check
                        Don't periodically check PyPI to determine
                        whether a new version of pip is available for
                        download. Implied with --no-index.

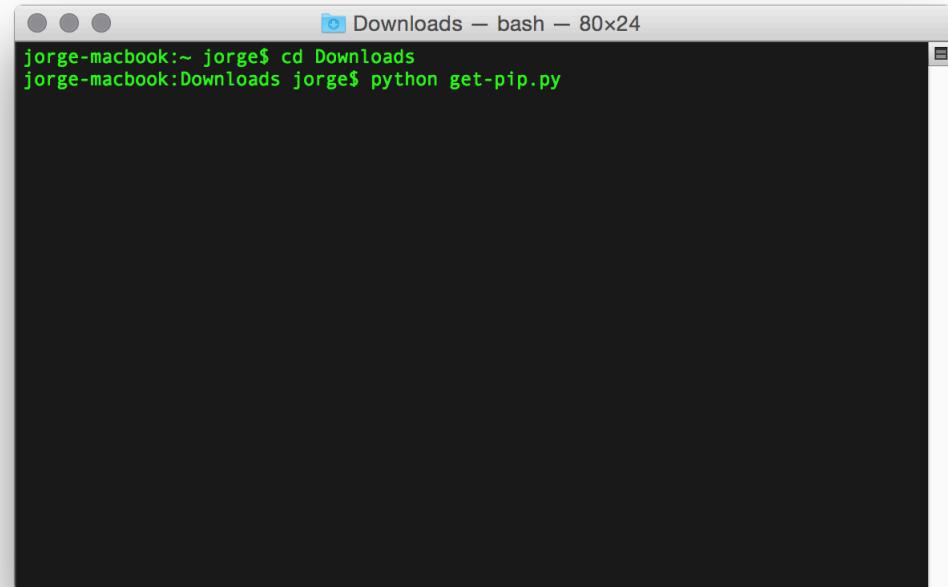
jorge-macbook:Downloads jorge$
```

Install ‘pip’

If needed download: <https://bootstrap.pypa.io/get-pip.py>

- Install ‘pip’:

- Using the Shell or Terminal, change directory to the download location (cd [path])
- Type and execute ‘python get-pip.py’
- MacOSX: ‘sudo python get-pip.py’
- Wait and it should work...
- Test by typing pip again

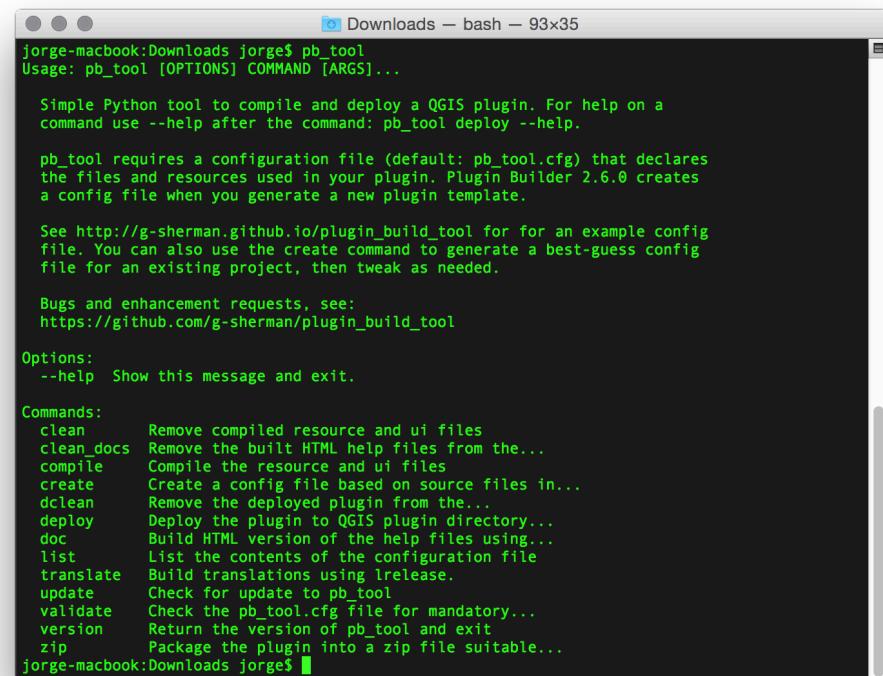


```
jorge-macbook:~ jorge$ cd Downloads
jorge-macbook:Downloads jorge$ python get-pip.py
```

Install the ‘Plugin Builder Tool’

To install use the Shell or the Terminal:

- Type and execute ‘pip install pb_tool’
- On MacOSX: ‘sudo pip install pb_tool’
- Close and reopen the terminal window
- Type and execute ‘pb_tool’ to test



```
jorge-macbook:Downloads jorge$ pb_tool
Usage: pb_tool [OPTIONS] COMMAND [ARGS]...

Simple Python tool to compile and deploy a QGIS plugin. For help on a
command use --help after the command: pb_tool deploy --help.

pb_tool requires a configuration file (default: pb_tool.cfg) that declares
the files and resources used in your plugin. Plugin Builder 2.6.0 creates
a config file when you generate a new plugin template.

See http://g-sherman.github.io/plugin_build_tool for for an example config
file. You can also use the create command to generate a best-guess config
file for an existing project, then tweak as needed.

Bugs and enhancement requests, see:
https://github.com/g-sherman/plugin_build_tool

Options:
--help Show this message and exit.

Commands:
clean      Remove compiled resource and ui files
clean_docs Remove the built HTML help files from the...
compile    Compile the resource and ui files
create     Create a config file based on source files in...
dclean    Remove the deployed plugin from the...
deploy     Deploy the plugin to QGIS plugin directory...
doc        Build HTML version of the help files using...
list       List the contents of the configuration file
translate  Build translations using lrelease.
update    Check for update to pb_tool
validate   Check the pb_tool.cfg file for mandatory...
version   Return the version of pb_tool and exit
zip       Package the plugin into a zip file suitable...

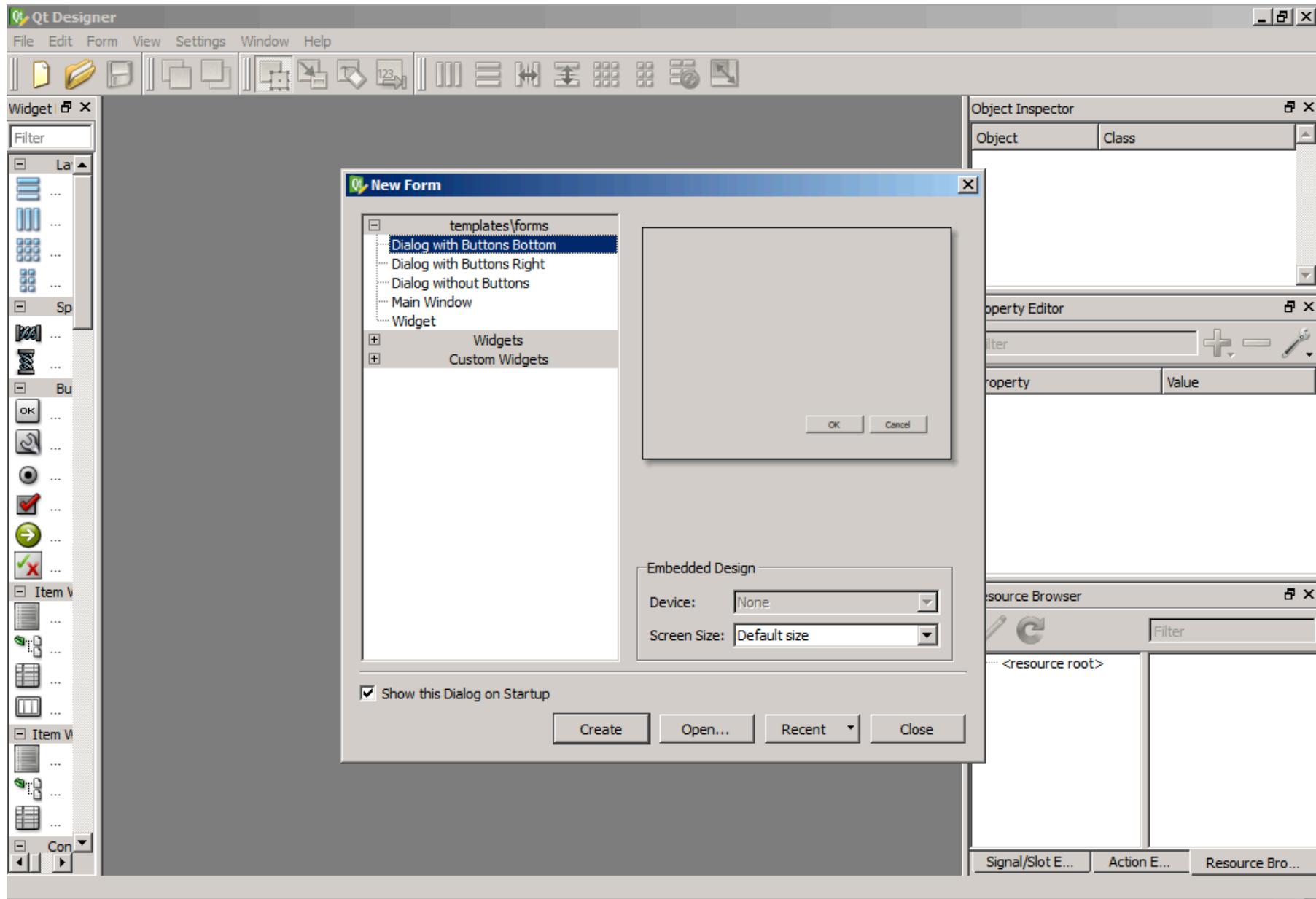
jorge-macbook:Downloads jorge$
```

For details see the website:

https://g-sherman.github.io/plugin_build_tool/

Start Qt Designer from the start menu

12



- Install Mac OSX developer tools from the ‘App Store’
- Start the Xcode App to agree with the license terms
- Download and install Qt Creator:

http://download.qt.io/official_releases/qtcreator/3.6/3.6.1/qt-creator-opensource-mac-x86_64-3.6.1.dmg

Edit the `.bash_profile` in your `$Home` folder to use the QGIS app folder, because this already includes sip and PyQt.

Start the Terminal, and type:

```
$ touch ~/.bash_profile (to create the file if it doesn't exist)  
$ open ~/.bash_profile (to open the file for editing)
```

Write the following commands in the text file:

```
QGISBASE="/Applications/QGIS.app/Contents"  
export PATH=/usr/bin:/bin:/usr/sbin:/sbin:/usr/local/bin:/System/  
Library/Frameworks/Python.framework/Versions/2.7/bin/:${QGISBASE}/  
MacOS/bin  
export DYLD_LIBRARY_PATH=${QGISBASE}/MacOS/lib  
export PYTHONPATH=:${QGISBASE}/Resources/python:$PYTHONPATH
```

(If you already have a profile, append the QGISBASE paths in **bold**)

Save and closeTextEdit.

PyCharm Community is a free Python IDE popular with QGIS plugin developers.

PyCharm Pro adds remote debugging capability, very useful for QGIS plugin development.

Download and install the version for your system:

<https://www.jetbrains.com/pycharm/download/>

It's a 30 days trial, you will get a commercial license later.

SpatialDecision - [C:/Users/jorge/development/GEO1005/SpatialDecision] - spatial_decision_dockwidget.py - PyCharm Edu 2.0.2

File Edit View Navigate Help

Project SpatialDecision (C:/Users/jorge/development/GEO1005/SpatialDecision)

14 *****
15 *
16 * This program is free software; you can redistribute it and/or modify
17 * it under the terms of the GNU General Public License as published by
18 * the Free Software Foundation; either version 2 of the License, or
19 * (at your option) any later version.
20 *
21 *****
22 """
23
24 +import ...
25
26 FORM_CLASS, _ = uic.loadUiType(os.path.join(
27 os.path.dirname(__file__), 'spatial_decision_dockwidget_base.ui'))
28
29 class SpatialDecisionDockWidget(QtGui.QDockWidget, FORM_CLASS):
30
31 closingPlugin = pyqtSignal()
32
33
34 def __init__(self, parent=None):
35 """Constructor."""
36 super(SpatialDecisionDockWidget, self).__init__(parent)
37 # Set up the user interface from Designer.
38 # After setupUI you can access any designer object by doing
39 # self.<objectname>, and you can use autoconnect slots - see
40 # http://qt-project.org/doc/qt-4.8/designer-using-a-ui-file.html
41 # #widgets-and-dialogs-with-auto-connect
42 self.setupUi(self)
43
44 #self.comboBox

Python Console Version Control

LF UTF-8 Git: master

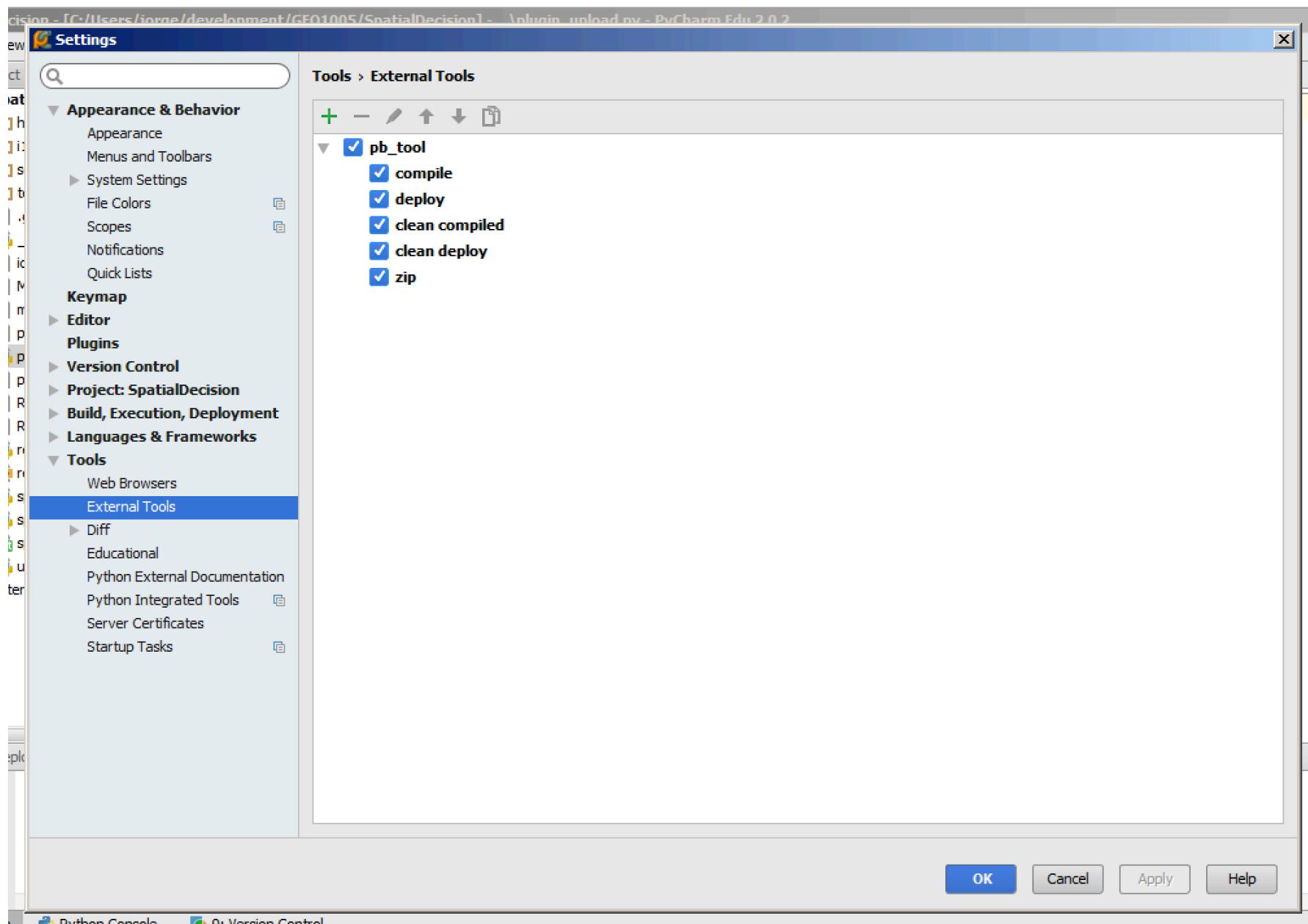
To configure PyCharm after installation:

- Click “Configure > Preferences...” in the Welcome window
- Go to “File > Settings...” (Windows)
- Go to “PyCharm > Preferences...” (Mac OSX)

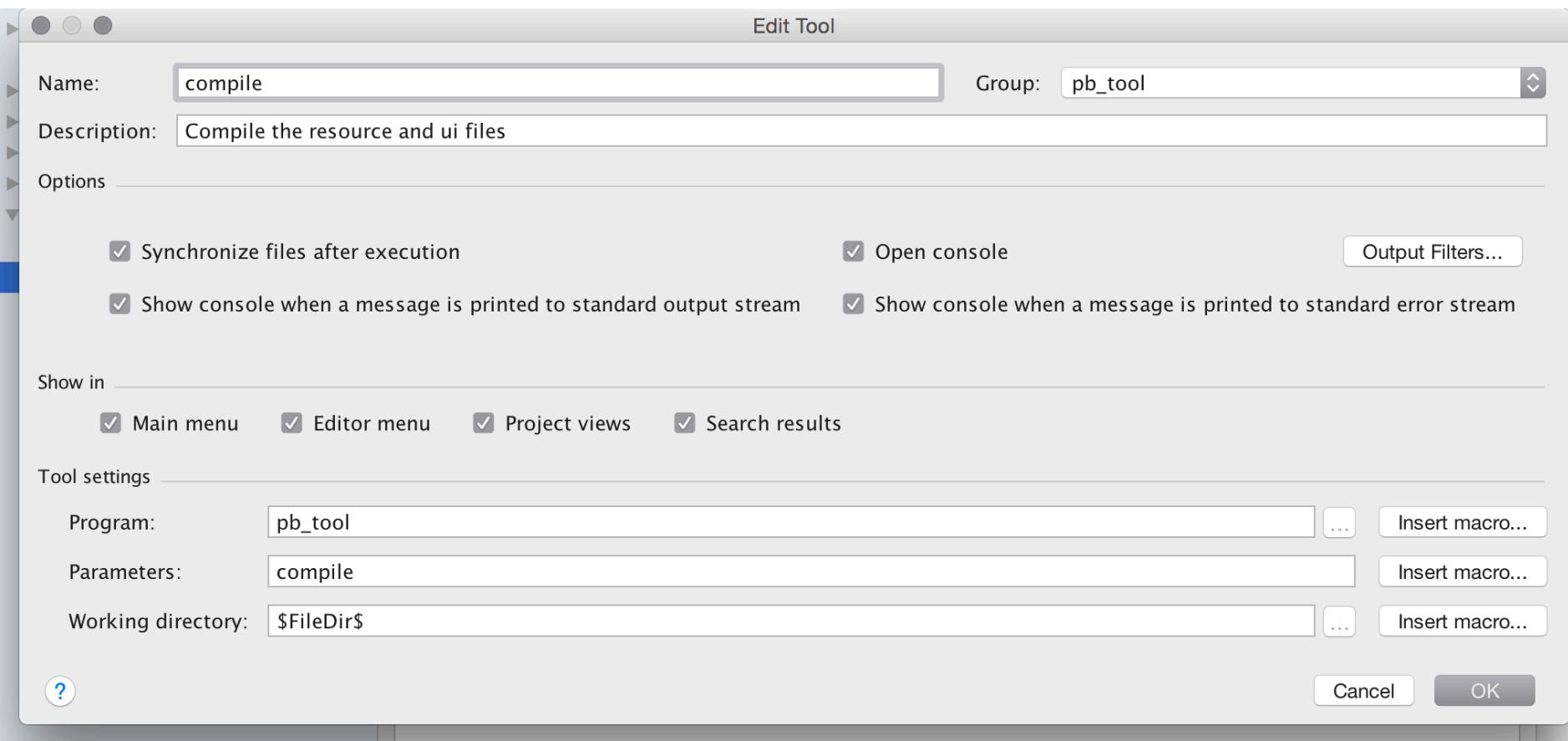
Configuration tasks:

- Create ‘External Tools’ for compiling and deployment
- Add ‘External Tools’ tools to menus
- Select a python interpreter that recognises QGIS
- Create a QGIS remote debug configuration
- Create a QGIS aware PyCharm command

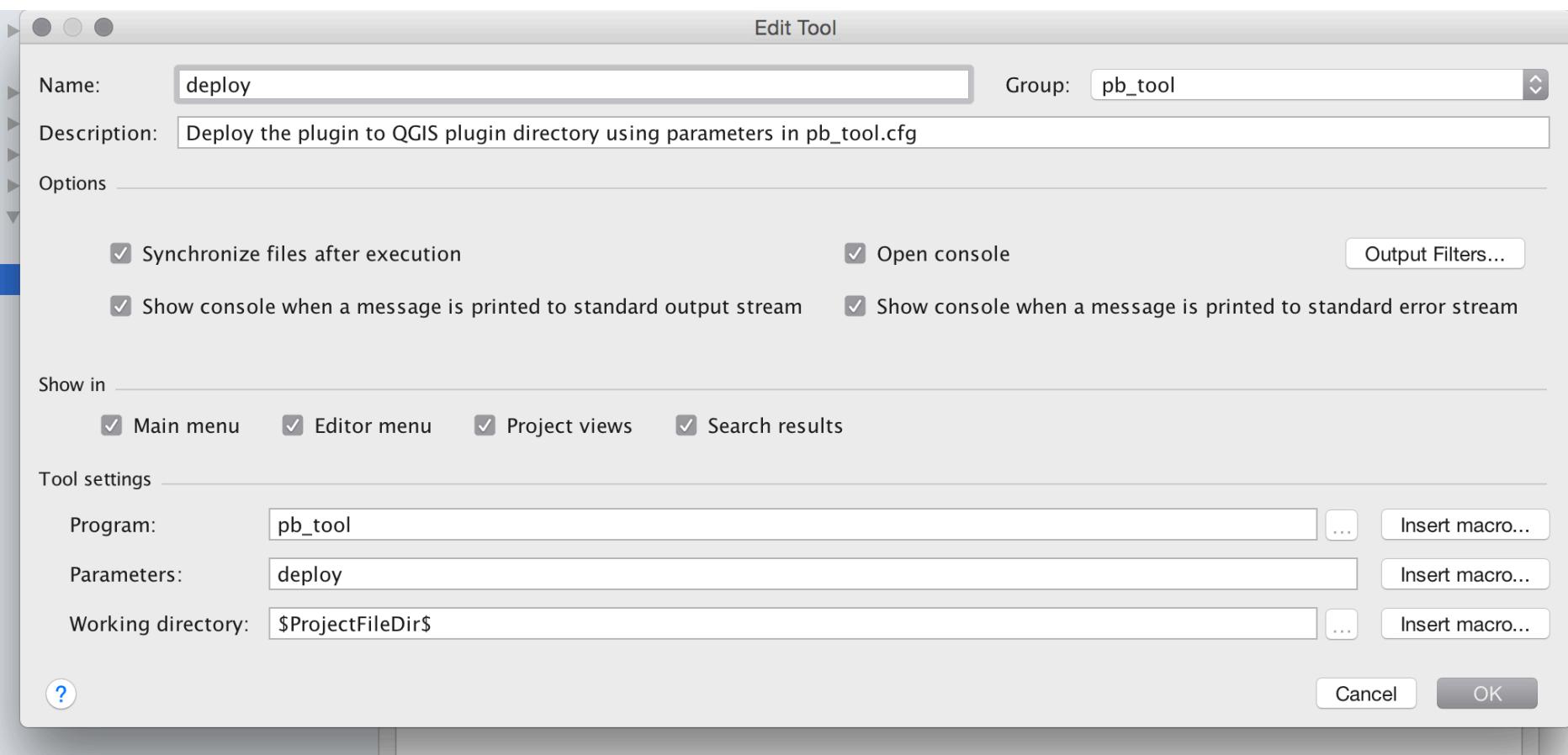
Create ‘External Tools’ for pb_tool



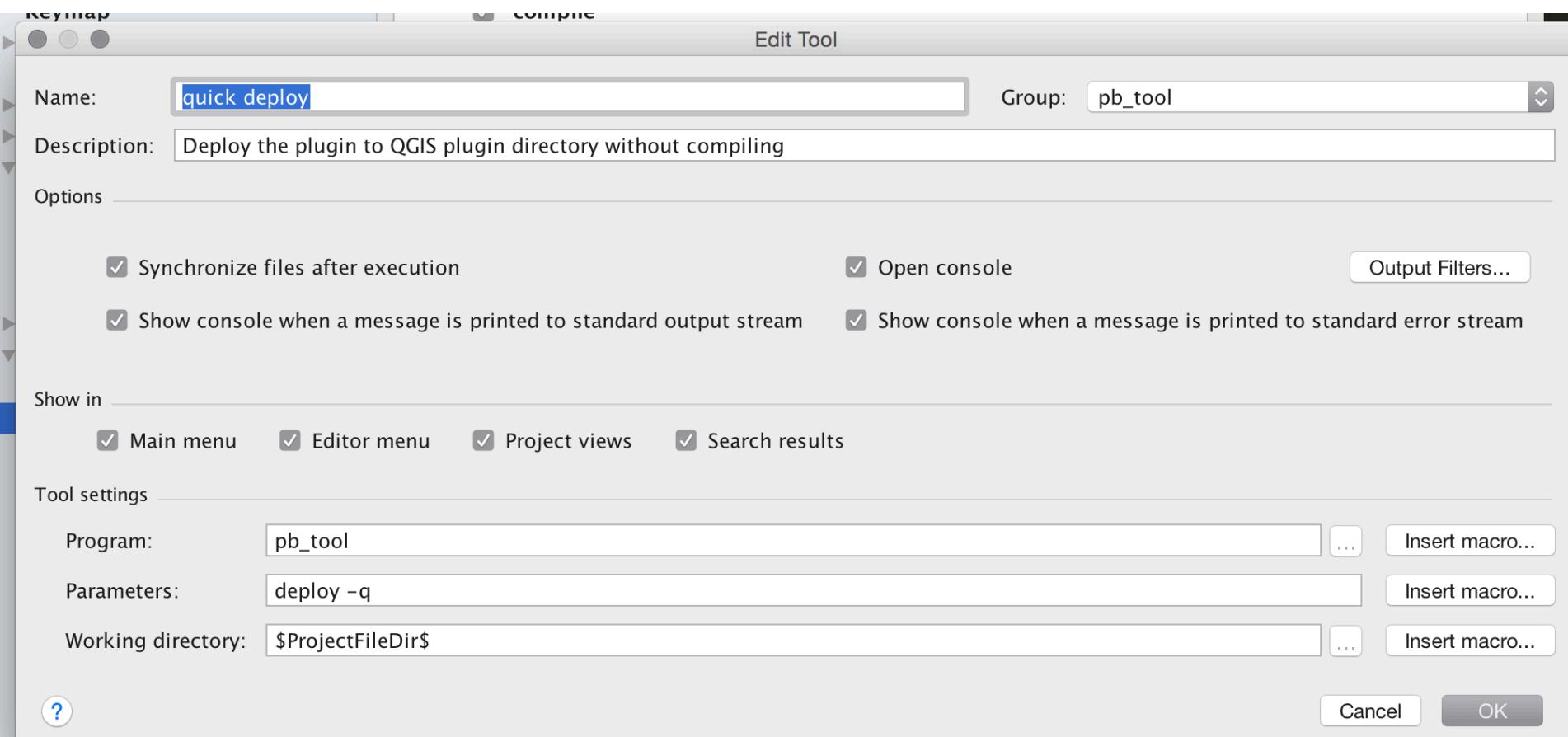
- Compiles the user interface and resources files



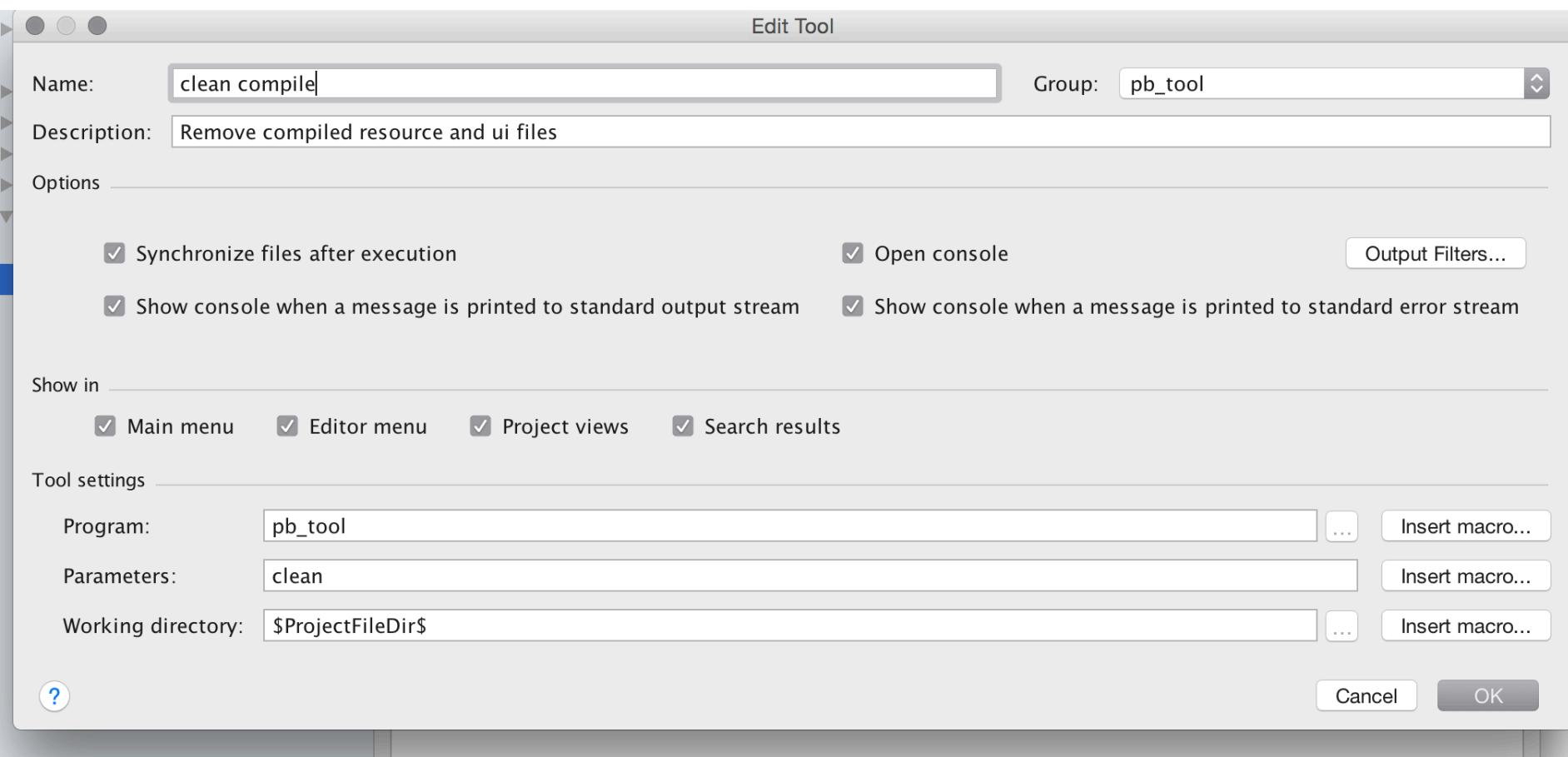
- Prepares the project and copies it into the plugins folder



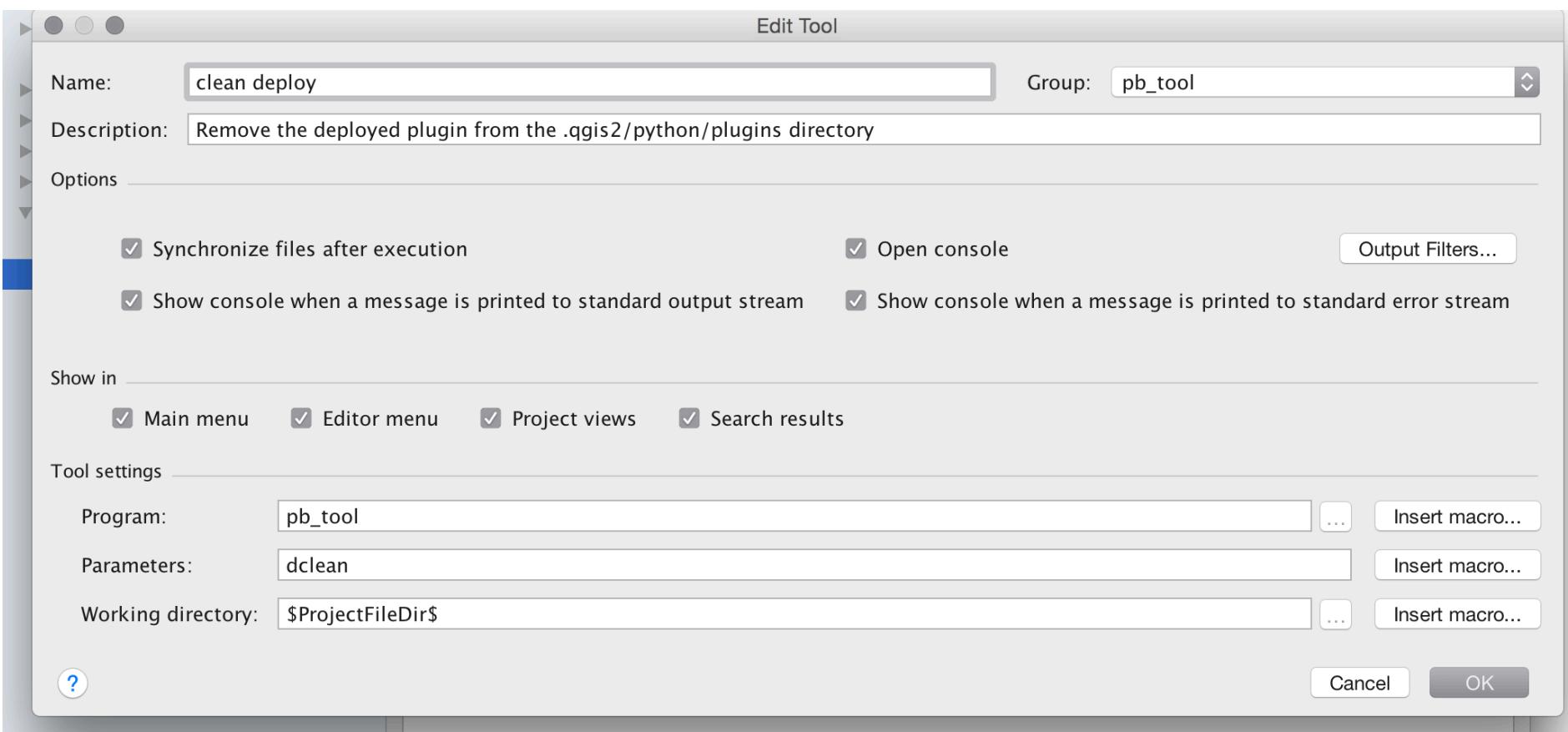
- Updates the project in the plugins folder



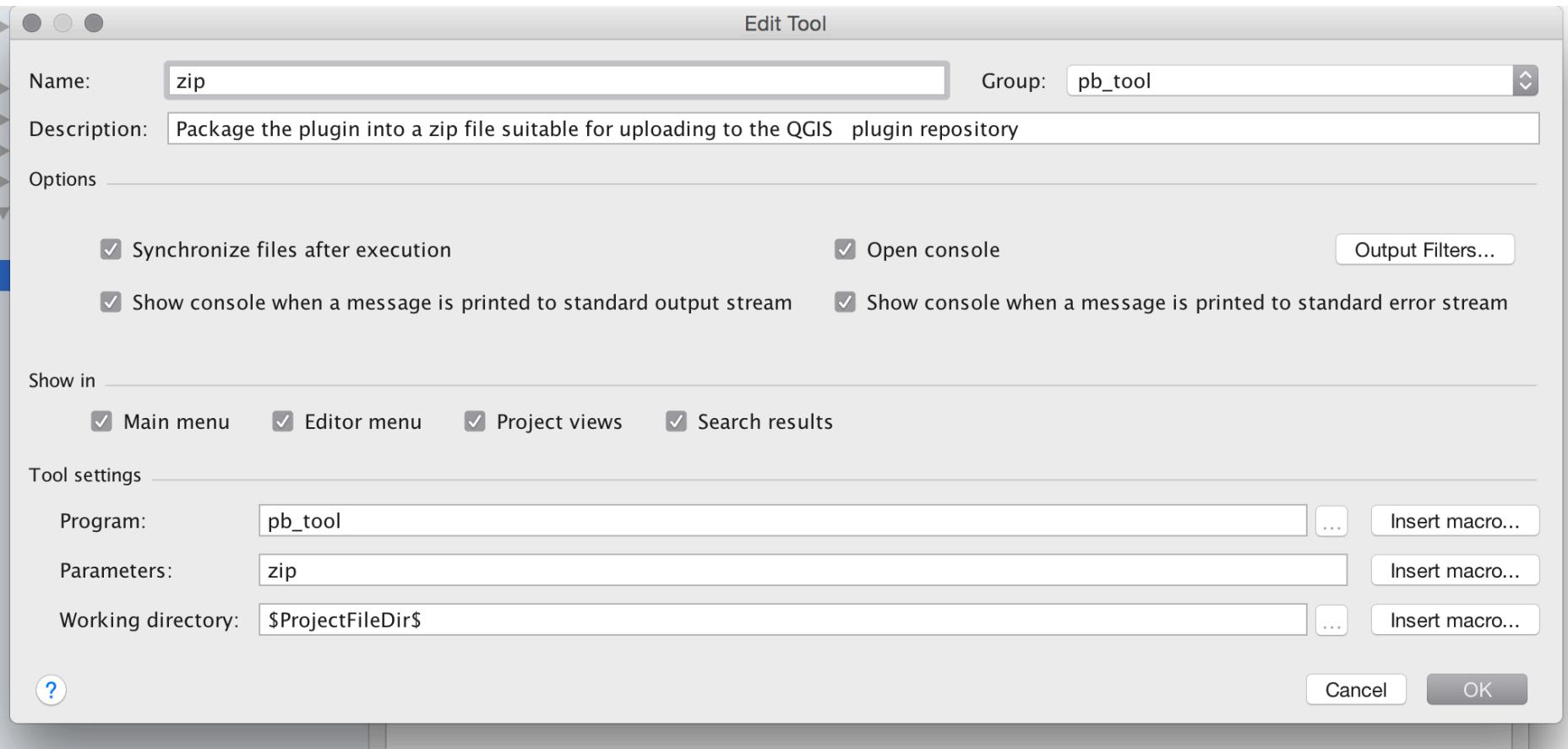
- Removes the previously compiled resource and ui files



- Removes the previously deployed plugin from the folder

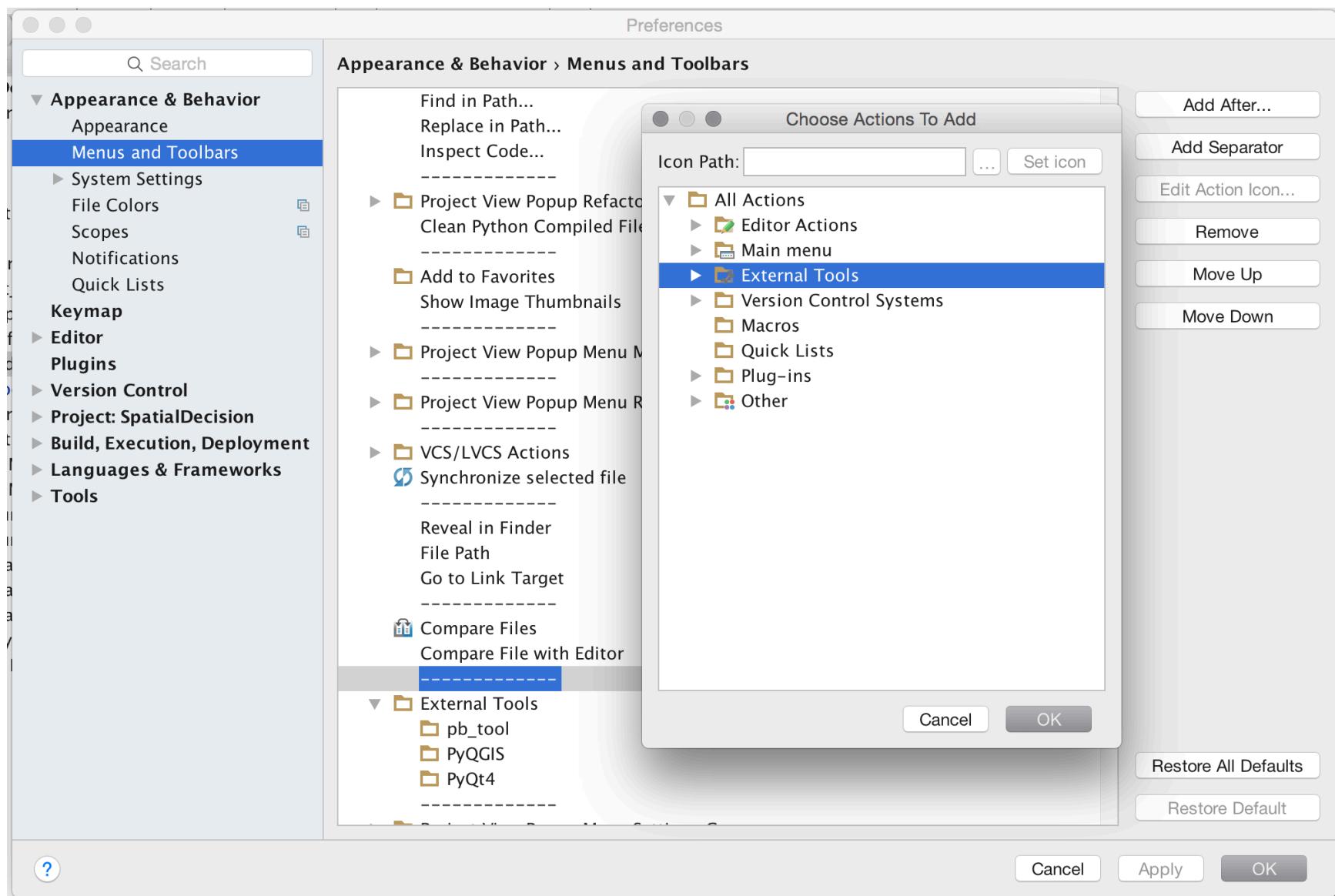


- Creates a clean zip file of the plugin for distribution



- Click 'Apply' after creating the External Tools

Add ‘External Tools’ to menus



- Click ‘OK’, Click ‘Apply’

Add 'Quick Compile' to the main toolbar

The screenshot shows the QGIS application's preferences dialog for "Appearance & Behavior > Menus and Toolbars". The "Main Toolbar" section is selected. A context menu is open over the "quick deploy" action, showing options like "Add After...", "Add Separator", "Edit Action Icon...", "Remove", "Move Up", and "Move Down".

Preferences

Appearance & Behavior > Menus and Toolbars

Main Toolbar

- Open...
-
- Undo
- Redo
-
- Cut
- Copy
- Paste
- Toolbar Find Actions
-
- Back
- Forward
- Toolbar Run Actions
-
- quick deploy
-
- MainToolBarSettings
-
- Help Topics
- Search Everywhere
- Editor Popup Menu
- Editor Gutter Popup Menu
- Editor Tab Popup Menu
- Project View Popup Menu
- Scope View Popup Menu
- Favorites View Popup Menu
- Navigation Bar

Choose Actions To Add

Icon Path: Set icon

All Actions

- Editor Actions
- Main menu
- External Tools
- PyQGIS
- PyQt4
- pb_tool

- clean compile
- clean deploy
- compile
- deploy
- quick deploy
- zip

Version Control Systems

Macros

Quick Lists

Plug-ins

Other

Action Options

- Add After...
- Add Separator
- Edit Action Icon...
- Remove
- Move Up
- Move Down

Cancel OK

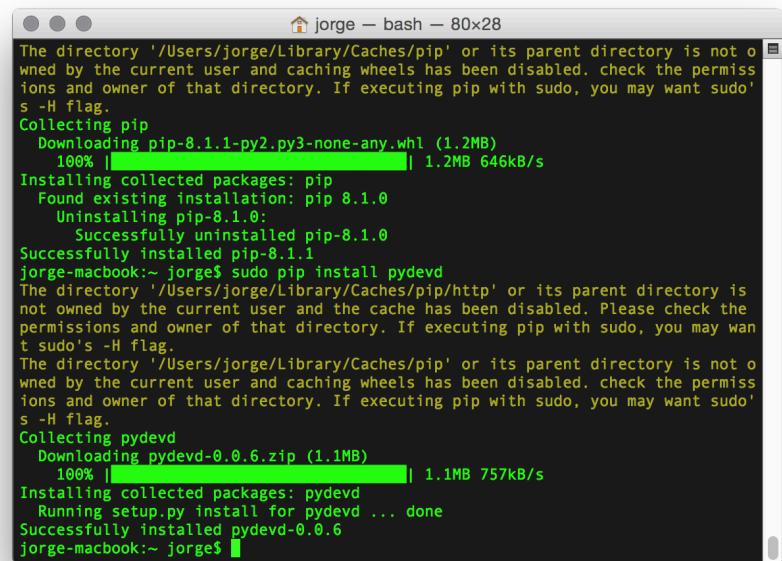
Restore All Defaults

Restore Default

Cancel Apply OK

Create ‘Remote Debug’ configuration

- Install the pydevd Python package using the pip tool:
 - Windows: \$ pip install pydevd
 - Mac OSX: \$ sudo pip install pydevd
- In PyCharm go to menu ‘Run’ > ‘Edit Configurations...’
- Click the ‘+’ sign to add a configuration
- Select ‘Python Remote Debug’
- Change the port to ‘53100’



```
jorge ~ bash - 80x28
The directory '/Users/jorge/Library/Caches/pip' or its parent directory is not owned by the current user and caching wheels has been disabled. check the permissions and owner of that directory. If executing pip with sudo, you may want sudo's -H flag.
Collecting pip
  Downloading pip-8.1.1-py2.py3-none-any.whl (1.2MB)
    100% |██████████| 1.2MB 646kB/s
Installing collected packages: pip
  Found existing installation: pip 8.1.0
    Uninstalling pip-8.1.0:
      Successfully uninstalled pip-8.1.0
Successfully installed pip-8.1.1
jorge-macbook:~ jorge$ sudo pip install pydevd
The directory '/Users/jorge/Library/Caches/pip/http' or its parent directory is not owned by the current user and the cache has been disabled. Please check the permissions and owner of that directory. If executing pip with sudo, you may want sudo's -H flag.
The directory '/Users/jorge/Library/Caches/pip' or its parent directory is not owned by the current user and caching wheels has been disabled. check the permissions and owner of that directory. If executing pip with sudo, you may want sudo's -H flag.
Collecting pydevd
  Downloading pydevd-0.0.6.zip (1.1MB)
    100% |██████████| 1.1MB 757kB/s
Installing collected packages: pydevd
  Running setup.py install for pydevd ... done
Successfully installed pydevd-0.0.6
jorge-macbook:~ jorge$
```

Create 'Remote Debug' configuration

Run/Debug Configurations

Name: Share Single instance only

Python Remote Debug **debug qgis plugin**

Defaults

- App Engine server
- Django server
- Django tests
- JavaScript Debug
- Node-webkit
- Pyramid server
- Python
- Python Remote Debug
- Python docs
- Python tests
- Query Language Console

Launch this debug configuration to start the debug server.

Update your script:

1. Add pycharm-debug.egg from the PyCharm installation to the Python path.
2. Add the following import statement:
`import pydevd`

3. Add the following command to connect to the debug server:
`pydevd.settrace('localhost', port=53100, stdoutToServer=True, stderrToServer=True)`

Local host name:

Port:

Path mappings: ...

Redirect output to console

Suspend after connect

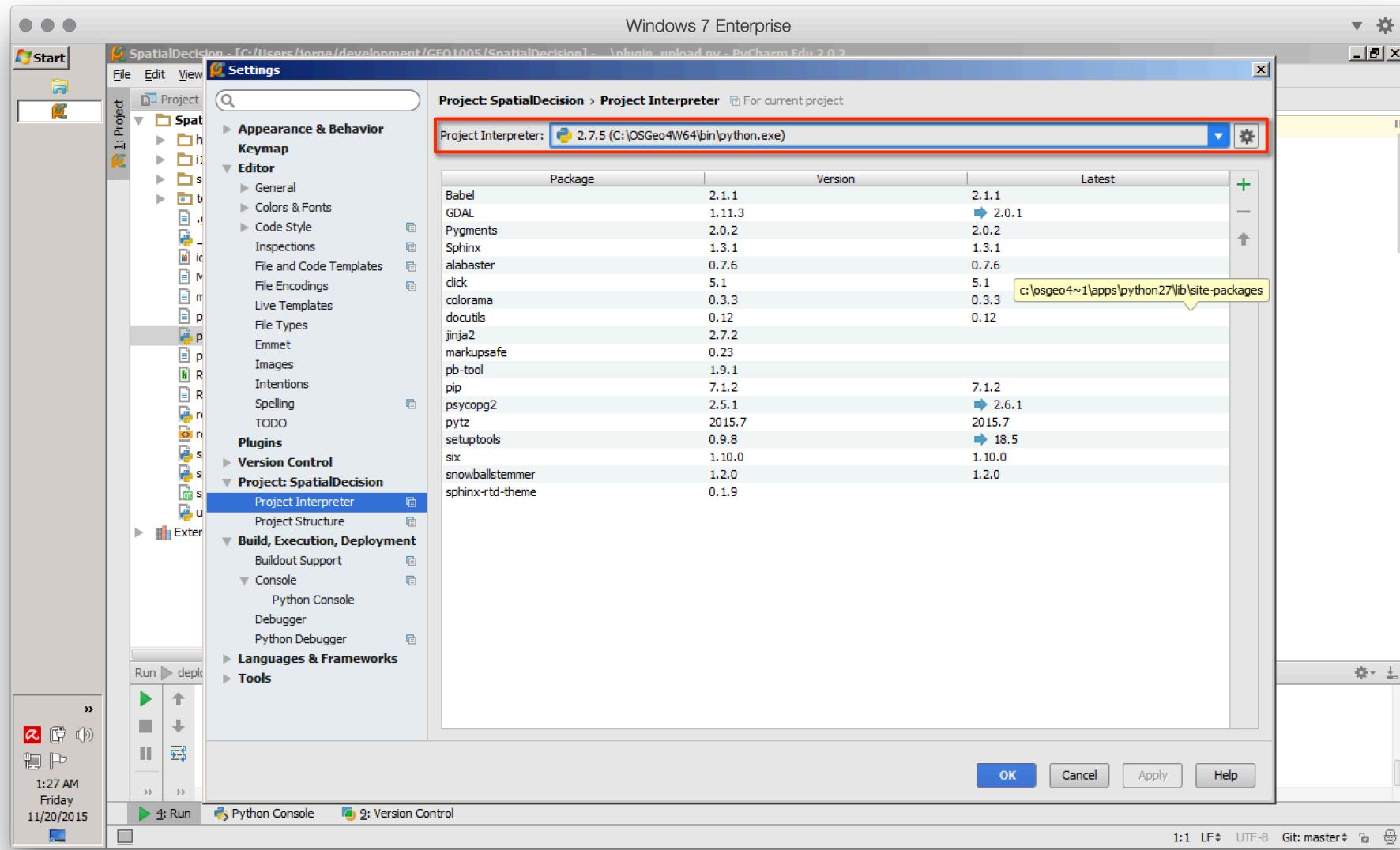
Before launch

There are no tasks to run before launch

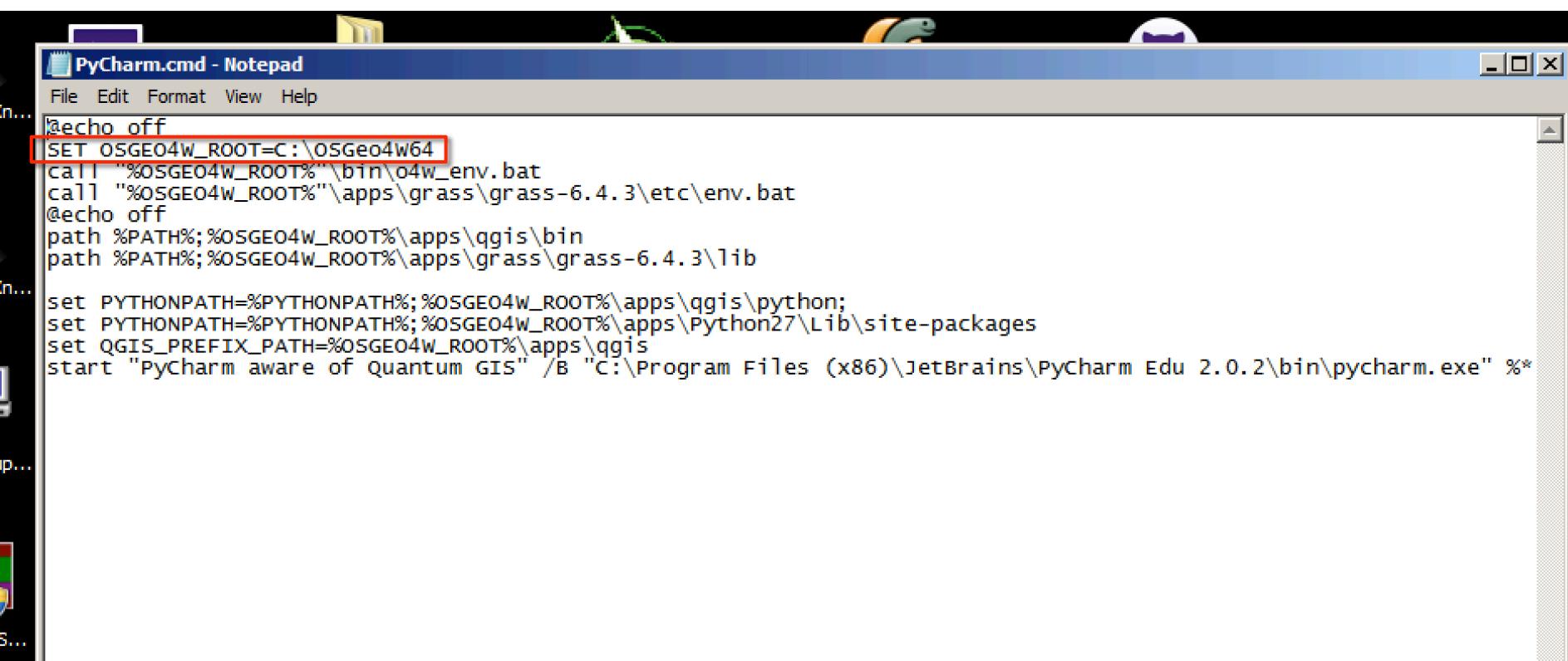
Cancel Apply OK

Select Python interpreter (Windows only)

- Set the interpreter to have code completion of QGIS API



- Copy this file to your desktop and use it to start Pycharm
- Change the OSGEO4W_ROOT variable path if required, based on your OSGeo4W installation path



```
PyCharm.cmd - Notepad
File Edit Format View Help
@echo off
SET OSGEO4W_ROOT=C:\OSGeo4W64
call "%OSGEO4W_ROOT%\bin\o4w_env.bat"
call "%OSGEO4W_ROOT%\apps\grass\grass-6.4.3\etc\env.bat"
@echo off
path %PATH%;%OSGEO4W_ROOT%\apps\qgis\bin
path %PATH%;%OSGEO4W_ROOT%\apps\grass\grass-6.4.3\lib

set PYTHONPATH=%PYTHONPATH%;%OSGEO4W_ROOT%\apps\qgis\python;
set PYTHONPATH=%PYTHONPATH%;%OSGEO4W_ROOT%\apps\Python27\Lib\site-packages
set QGIS_PREFIX_PATH=%OSGEO4W_ROOT%\apps\qgis
start "Pycharm aware of Quantum GIS" /B "C:\Program Files (x86)\JetBrains\PyCharm Edu 2.0.2\bin\pycharm.exe" %*
```