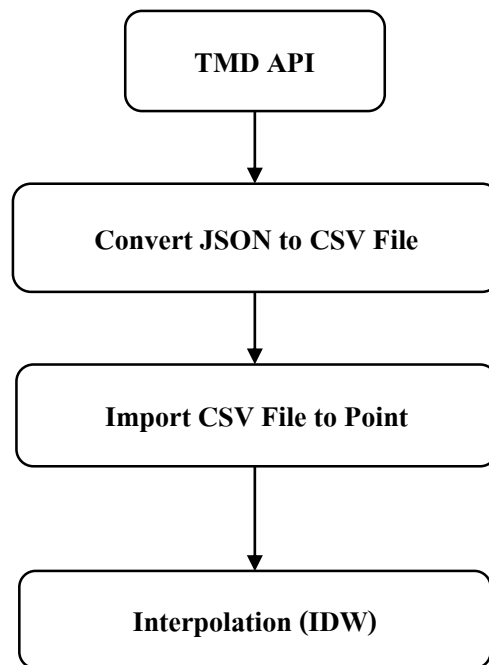


## การสร้าง QGIS Plugin

เอกสารนี้มีเนื้อหาเกี่ยวกับการสร้าง QGIS Plugin ด้วย QGIS Plugin builder การออกแบบ ปลั๊กอิน ด้วยโปรแกรม QT designer พร้อมกับตัวอย่างปลั๊กอิน โดยใช้ภาษา python

ตัวอย่างปลั๊กอินชื่อ TMD IDW คือปลั๊กอินสำหรับนำข้อมูล Weather To day จาก TMD API ของกรมอุตุนิยมวิทยา นำเอาข้อมูลเรื่องปริมาณน้ำฝน มาเขียนเขียนเป็นไฟล์ CSV และนำเข้ามาเป็น ข้อมูลจุด ในโปรแกรม QGIS แล้วทำการประมาณค่า (Interpolation) ด้วยวิธีการ IDW

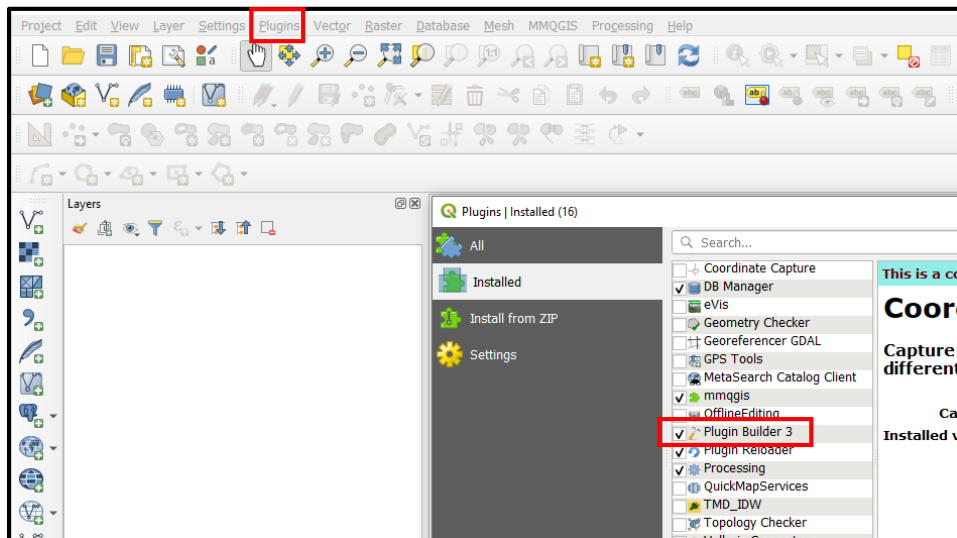


\*\*\* ควรดาวน์โหลดข้อมูลประกอบการดูเอกสารนี้จาก <https://github.com/Anon-B/QGIS-plugin>

สามารถติดต่อสอบถามเพิ่มเติมได้ที่ Email : [Anon.b@i-bitz.co.th](mailto:Anon.b@i-bitz.co.th)

## 1. การสร้าง Plugin ด้วย QGIS Plugin builder

### 1.1 ทำการติดตั้ง QGIS Plugin builder แล้วเรียกใช้เพื่อสร้างปลั๊กอิน



### 1.2 Plugin name and required information กรอกข้อมูลปลั๊กอินที่เราจะสร้างให้ครบ

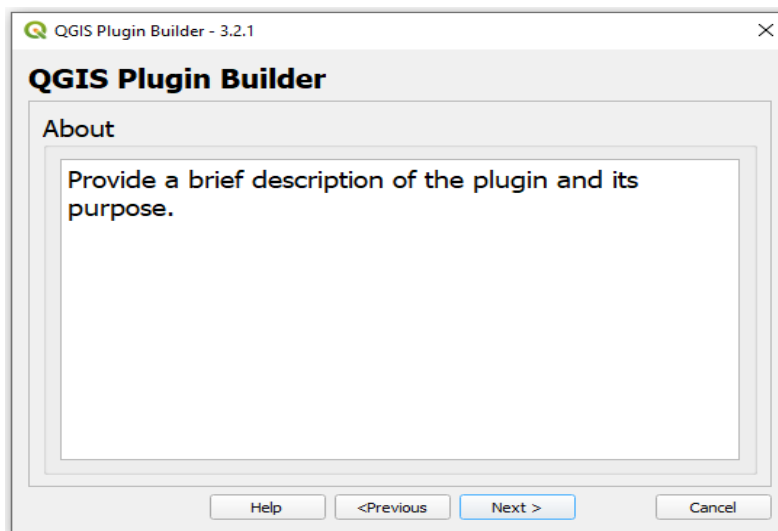
A screenshot of the 'QGIS Plugin Builder - 3.2.1' dialog box. The title bar says 'QGIS Plugin Builder - 3.2.1'. The main title is 'QGIS Plugin Builder'. The form contains the following fields:

- Class name: e.g. PhotoLinker
- Plugin name: e.g. Photo Linker
- Description: e.g. This plugin links points to photos
- Module name: e.g. photo\_linker
- Version number: 0.1
- Minimum QGIS version: 3.0
- Author/Company: e.g. Acme widgets Inc.
- Email address: e.g. jack@qgis.org

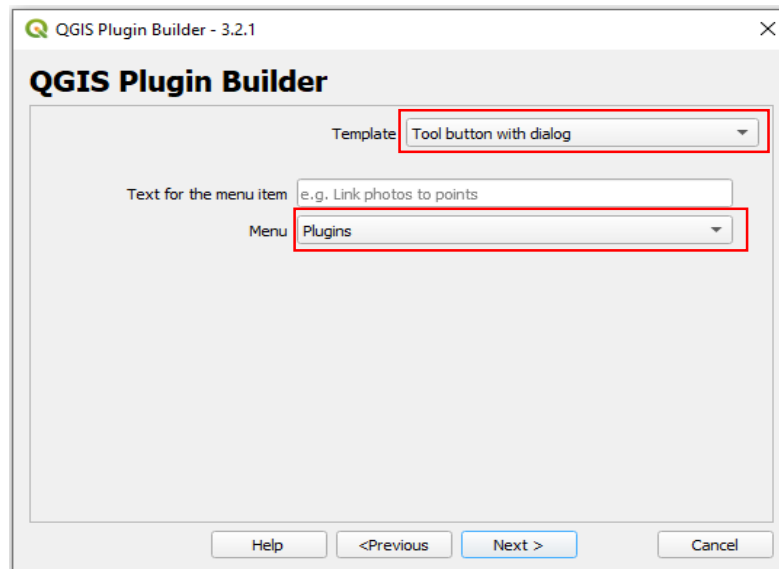
At the bottom, there are buttons for 'Help', '<Previous', 'Next >', and 'Cancel'.

- **Class name** ชื่อของ Python Class ในปลั๊กอิน (ห้ามเว้นวรรค)
- **Plugin name** ชื่อปลั๊กอิน
- **Description** คือ คำอธิบายสั้นๆเกี่ยวกับปลั๊กอิน
- **Module name** ชื่อโมดูลจะเป็นชื่อของไฟล์ไพธอนหลักสำหรับปลั๊กอิน(ห้ามเว้นวรรค)
- **Version number** เวอร์ชันของปลั๊กอิน
- **Minimum QGIS version** กำหนดเวอร์ชันขั้นต่ำของ QGIS ที่สามารถใช้ปลั๊กอิน
- **Author/Company** ระบุชื่อ/องค์กร/บริษัท ผู้พัฒนา
- **Email address** ระบุอีเมลเพื่อที่จะให้ผู้ใช้งาน Plugin สามารถติดต่อกลับถึงผู้สร้างได้

1.3 Detailed description เพิ่มคำอธิบายรายละเอียดของปลั๊กอินที่เราจะทำ เช่น การอธิบายวัตถุประสงค์ฟังก์ชันการทำงาน เป็นต้น (ภาษาอังกฤษ)

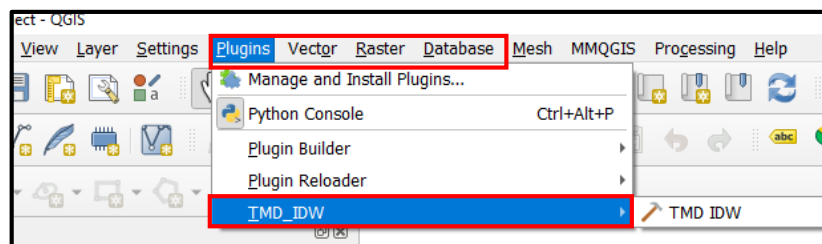


#### 1.4 Template specific parameters กำหนดรูปแบบของ ปลั๊กอิน



##### 1.4.1 Template เลือกรูปแบบของ Plugin ได้ 3 แบบคือ

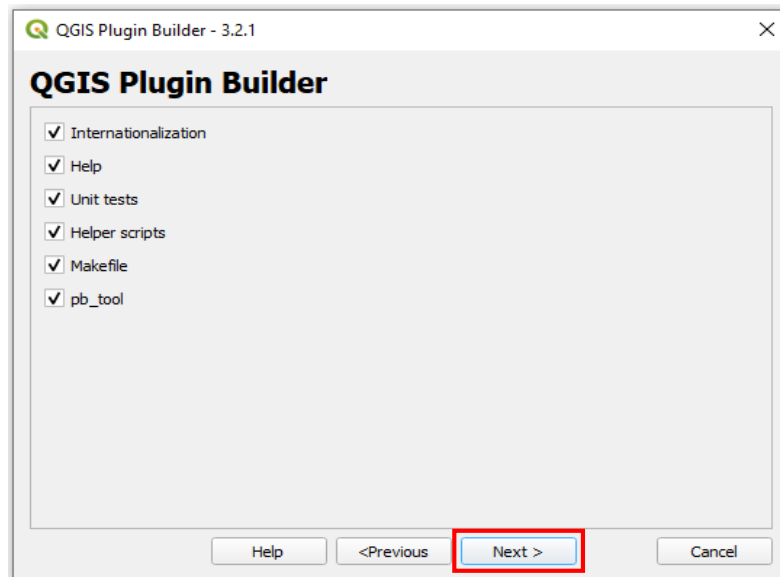
- Tool button with dialog แสดงเป็นหน้าต่างใหม่คล้าย Message box (เลือก)
- Tool button with dock widget แสดงในพื้นที่ด้านซ้าย ขวา หรือล่างในหน้าต่างของ QGIS
- Processing provider กำหนดให้แสดงในลักษณะ processing



##### 1.4.2 Text for the menu item ชื่อของเมนูไอเท็ม (ตั้ง TMD\_IDW)

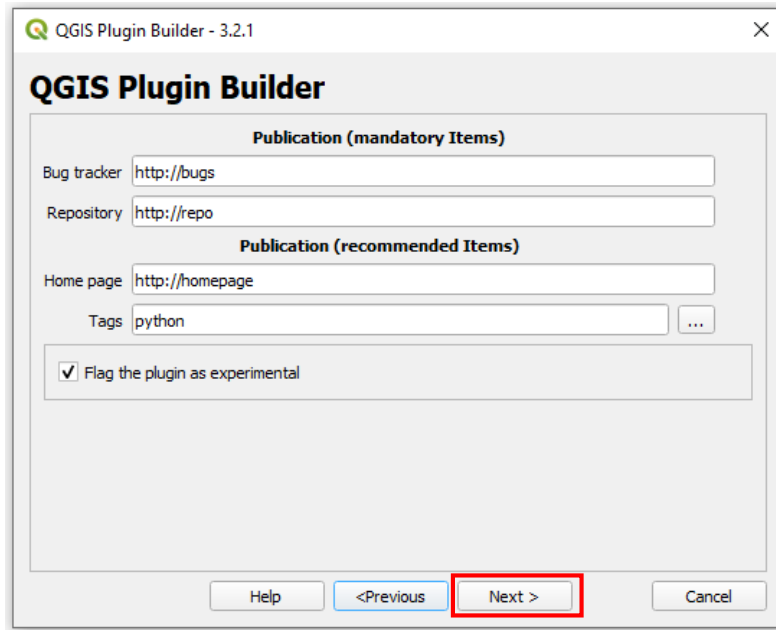
##### 1.4.3 Menu คือการกำหนดตำแหน่งให้ปลั๊กอินว่าจะอยู่ในเมนูไหน ได้แก่ Plugin, Database, Raster, Vector, Web (เลือก Plugin)

1.5 Additional components กำหนดค่าส่วนเพิ่มเติมที่แนะนำ เป็นตัวช่วยต่างๆที่ปลั๊กอินจะเพิ่มเข้ามาให้ตอนสร้างปลั๊กอิน จะเลือกหรือไม่เลือกก็ได้



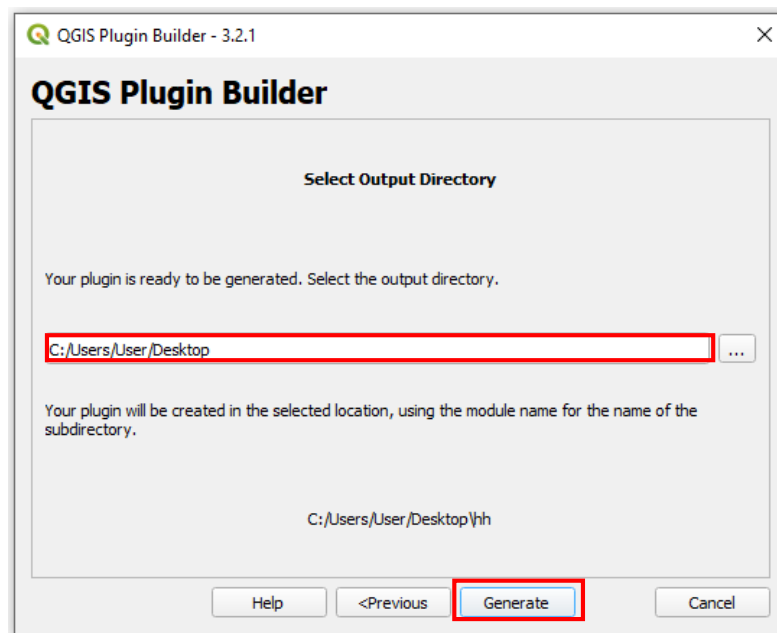
- Internationalization A stub for adding translated texts to your plugin.
- Help Creates a Sphinx template for generating a HTML help. See Documenting your Plugin for more information.
- Unit tests Creates a basic set of unit tests for your plugin.
- Helper scripts Adds a helper script for publishing your plugin on <http://plugins.qgis.org/> and additional scripts for internationalization and testing.
- Makefile Adds a Makefile for building your plugin with GNU make. See Using the Makefile for more information.
- pb\_tool Generates a configuration for pb\_tool, a Python command line tool for compiling and deploying QGIS plugins on Linux, Mac OS X, and Windows.

1.6 Publication information คือข้อมูลสำคัญที่ควรระบุ เพื่อเพิ่มความน่าเชื่อถือ ให้ได้รับการยอมรับจากผู้ใช้งาน  
เมื่อเราจะเผยแพร่ plugin ให้ผู้ใช้งานอื่นๆแบบสาธารณะ

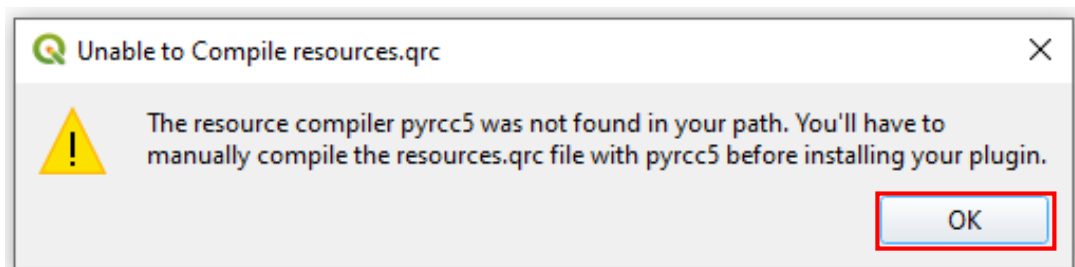


- Bug tracker URL สำหรับแจ้ง เมื่อเจอ Bug ต่างๆ
- Home page URL Home page ของปลั๊กอิน
- Repository URL ของที่เก็บซอร์สโค้ดสำหรับปลั๊กอินของคุณ
- Experimental การเลือกที่ “Flag the plugin as experimental” เพื่อจะระบุว่าอยู่ระหว่างการทดลองและพัฒนา
- Tags เพิ่มแท็กที่เกี่ยวข้องกับปลั๊กอินนี้โดยเพิ่มคำขึ้นด้วยเครื่องหมาย comma (,)

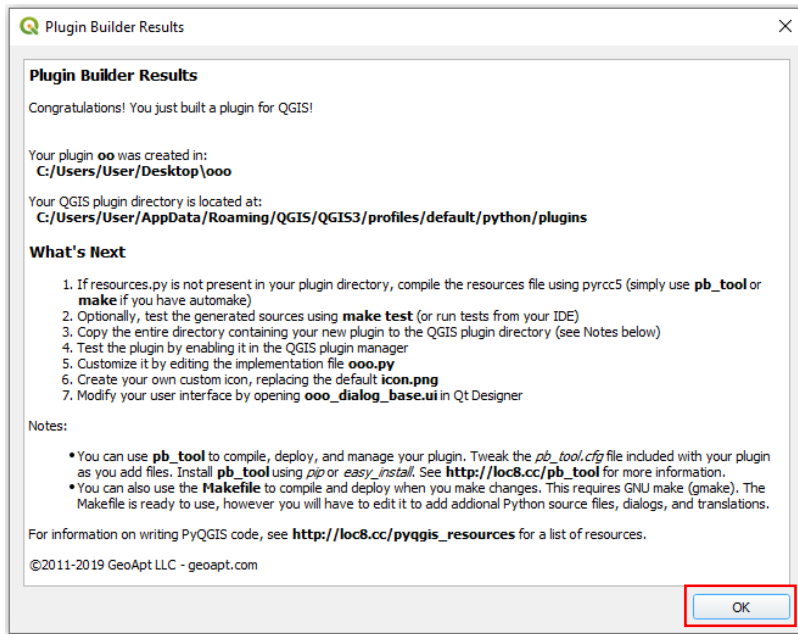
## 1.7 Deploying ให้เลือกโฟลเดอร์ที่จัดเก็บ Plugin



\*\*\*แนะนำให้สร้างไว้หน้า Desktop ก่อน



\*\*\* compile resource file ไม่ผ่าน ให้เรา compile resource file ก่อนแล้วค่อยติดตั้งปลั๊กอิน



1.8 การ compile resource file สร้าง Windows batch file (.bat) เพื่อ compile resource file ก่อนค่อยติดตั้ง Plugin (สำหรับ windows)

## Windows OS

@echo off

call "C:\Program Files\QGIS 3.6\bin\o4w\_env.bat"

call "C:\Program Files\QGIS 3.6\bin\qt5\_env.bat"

call "C:\Program Files\QGIS 3.6\bin\py3\_env.bat"

@echo on

pyrcc5 -o resources.py resources.qrc



- นำ Windows batch file (compile.bat) มาวางในโฟลเดอร์ปลั๊กอินของเรา แล้วกด run ก็จะได้ไฟล์ resources.py

help	10/28/2019 4:49 PM	File folder	
i18n	10/22/2019 4:39 PM	File folder	
scripts	10/22/2019 4:39 PM	File folder	
test	10/28/2019 4:49 PM	File folder	
init	10/28/2019 4:49 PM	Python File	2 KB
compile	10/28/2019 5:07 PM	Windows Batch File	1 KB
icon	10/22/2019 4:39 PM	PNG File	2 KB
Makefile	10/28/2019 4:49 PM	File	8 KB
metadata	10/28/2019 4:49 PM	Text Document	2 KB
pb_tool.cfg	10/28/2019 4:49 PM	CFG File	3 KB
plugin_upload	10/22/2019 4:39 PM	Python File	4 KB
pylintrc	10/22/2019 4:39 PM	File	9 KB
README	10/28/2019 4:49 PM	Chrome HTML Do...	2 KB
README	10/28/2019 4:49 PM	Text Document	1 KB
resources	10/28/2019 5:07 PM	Python File	6 KB
resources.qrc	10/28/2019 4:49 PM	QRC File	1 KB
TMD_IDW	10/28/2019 4:49 PM	Python File	7 KB
TMD_IDW_dialog	10/28/2019 4:49 PM	Python File	2 KB
TMD_IDW_dialog_base.ui	10/28/2019 4:49 PM	UI File	2 KB

### (Mac OS และ Linux OS)

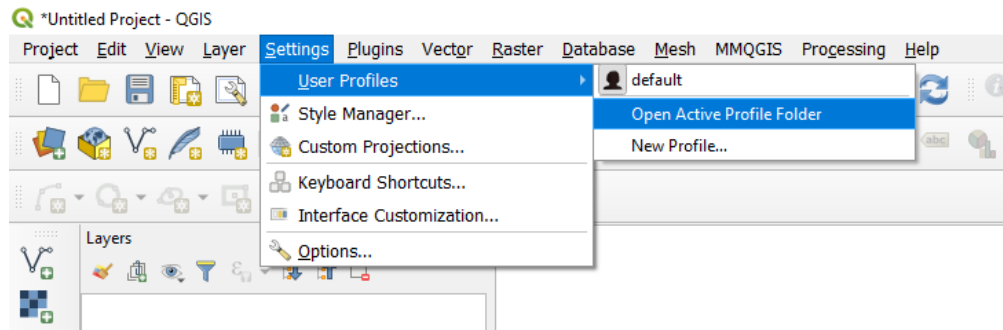
เปิด Terminal และ ไปที่ plugin directory แล้วใช้คำสั่ง

```
sudo pip3 install pb_tool
```

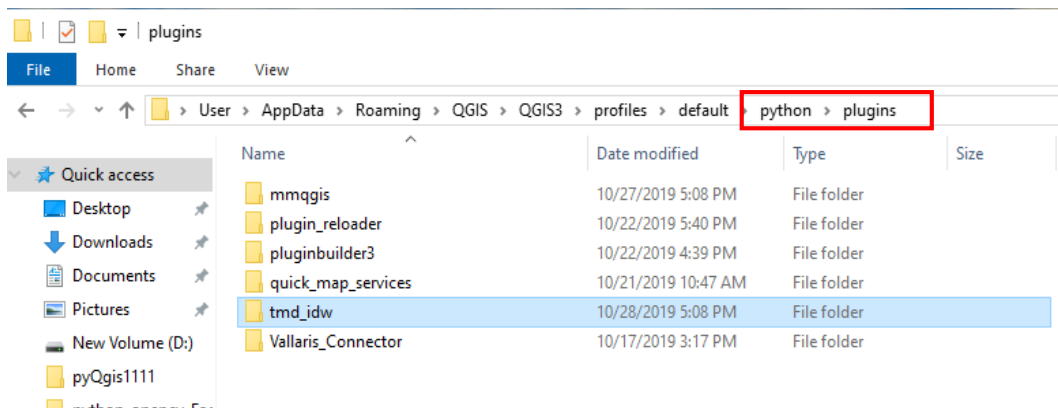
```
pb_tool compile
```

## 1.9 ติดตั้งปลั๊กอินใน QGIS

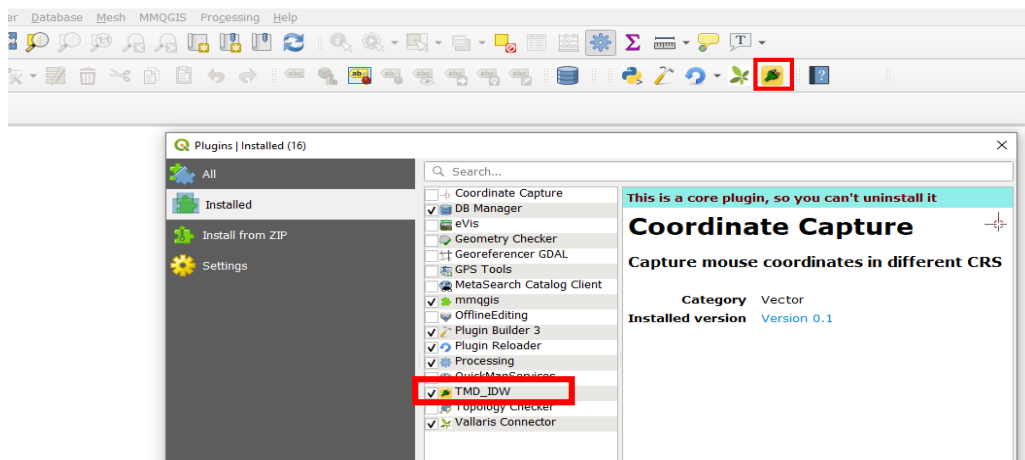
- Settings → User Profiles → Open Active Profile Folder



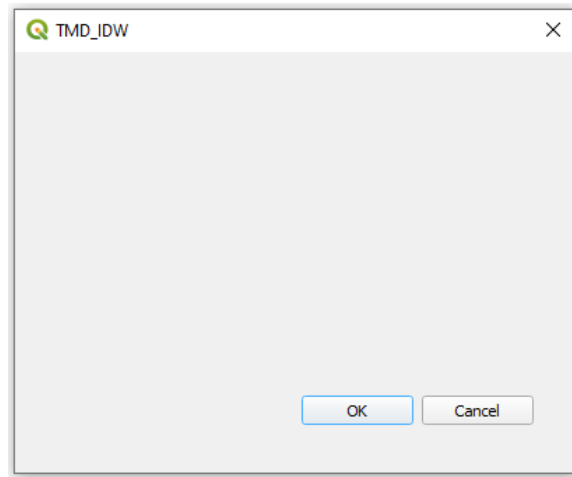
- นำไฟล์เดสก์ท็อปปลั๊กอินที่เราสร้างไว้หน้า Desktop ไปวางในไดเรกทอรี แล้วปิดโปรแกรม QGIS แล้วเปิดใหม่



- Plugins → manage and install Plugins → Installed เช็คว่าถูกเพื่อเปิดใช้งานปลั๊กอิน

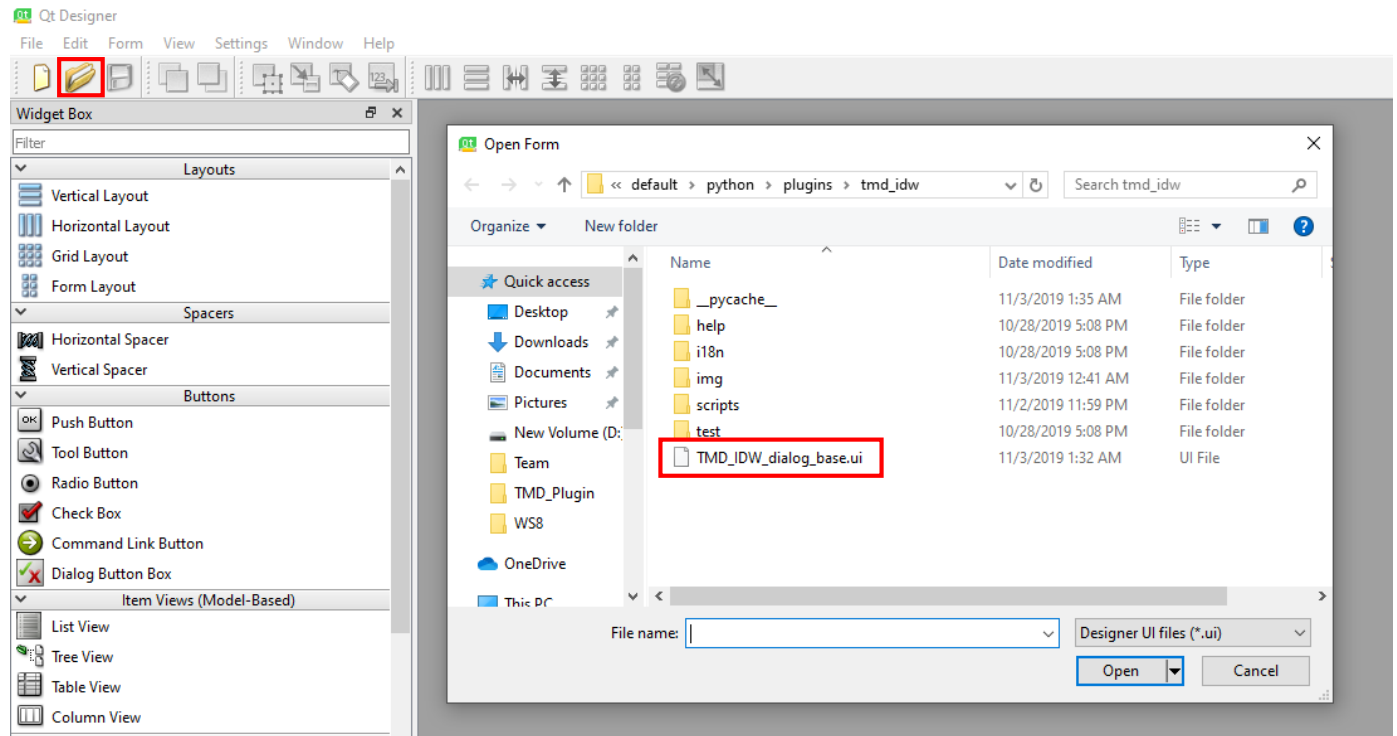


- เมื่อกดเปิดปลั๊กอินจะได้หน้าต่างดังนี้

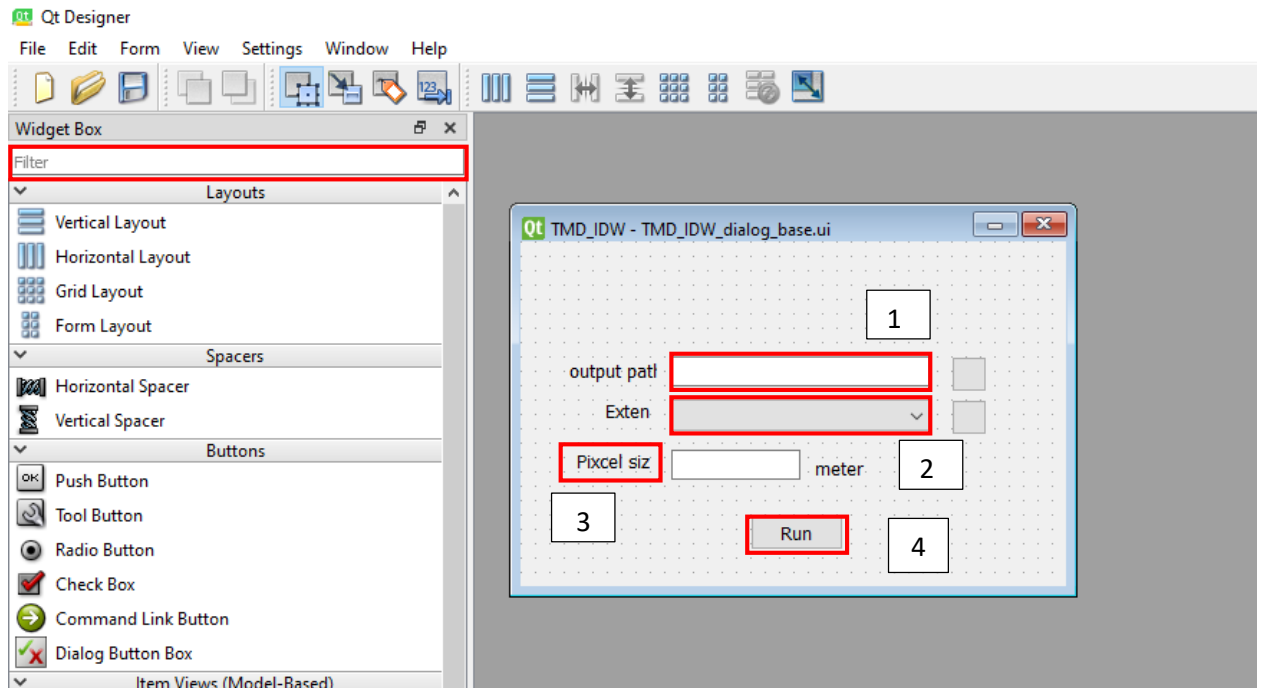


## 2.ออกแบบ UI โดยใช้โปรแกรม QT designer

2.1 เปิดโปรแกรม QT designer แล้วเลือกเปิดไฟล์ (.ui) เพื่อทำการออกแบบ Dialog ของปลั๊กอิน



## 2.2 ออกแบบ Dialog ให้เข้ากับการทำงานของฟังก์ชัน

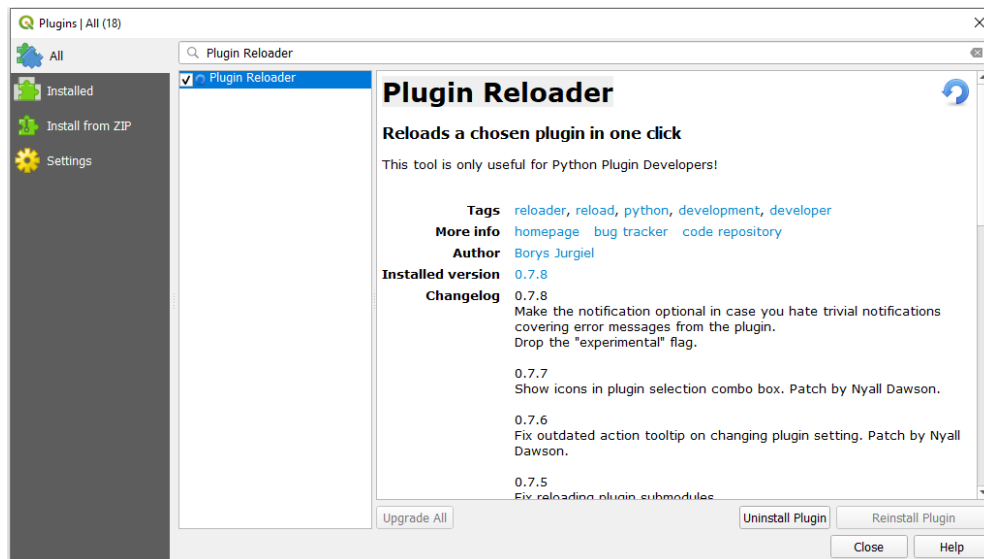


สามารถค้นหา filter ที่จะใช้ช่อง filter

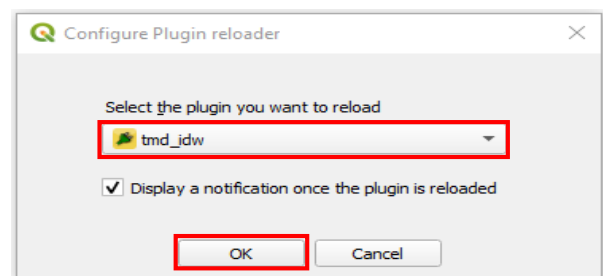
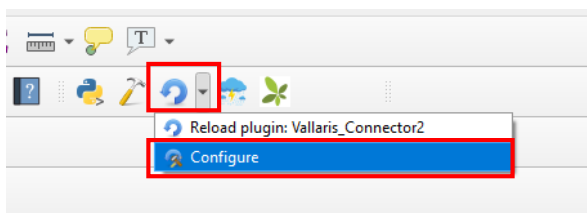
- Line Edit (1)
- Combo Box (2)
- Label (3)
- Push Button (4)
- objectName ควรตั้งชื่อให้สื่อ เพื่อง่ายต่อการเรียกใช้งาน (5)

Property	Value
QObject	
objectName	comboBox
QWidget	
enabled	<input checked="" type="checkbox"/>
geometry	[(105, 110), 180 x 22]
X	105
Y	110
Width	180
Height	22
sizePolicy	[Preferred, Fixed, 0, 0]
minimumSize	0 x 0
maximumSize	16777215 x 16777215
sizeIncrement	0 x 0
baseSize	0 x 0
palette	Inherited
font	A [MS Shell Dlg 2, 10]
cursor	Arrow
mouseTracking	<input type="checkbox"/>
tabletTracking	<input type="checkbox"/>
focusPolicy	WheelFocus

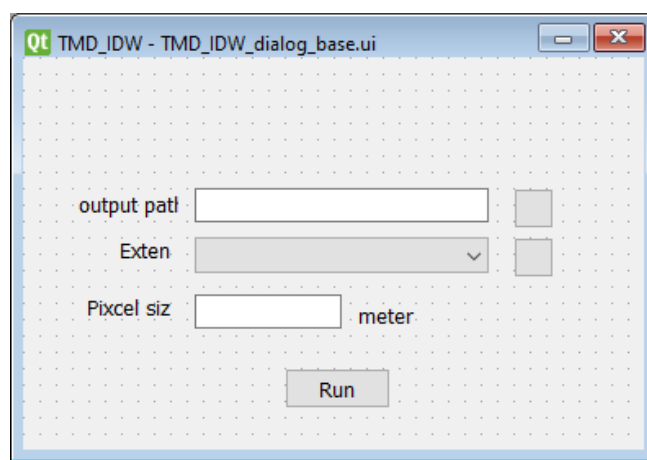
- ทำการ reload ปลั๊กอินโดยคลิกปลั๊กอิน Plugin Reloader เพื่ออัปเดตปลั๊กอิน



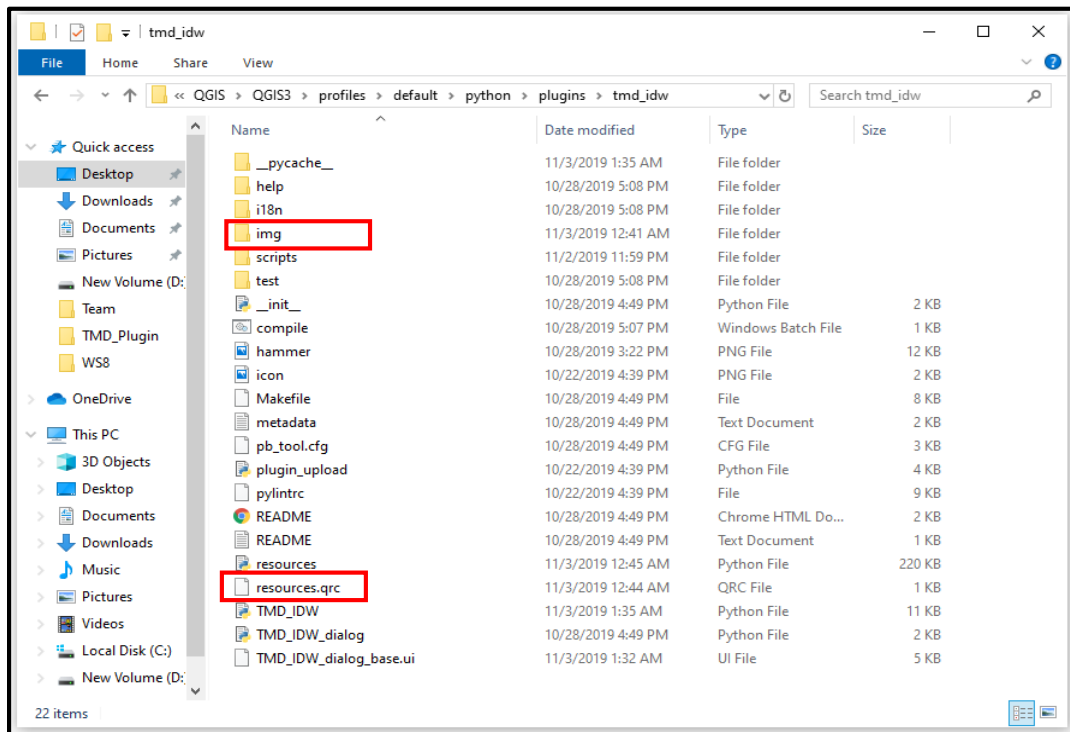
- ตั้งค่าการ configure ให้เป็นปลั๊กอินที่เราสร้าง แล้วกด reload



- แล้วเราจะได้ผลลัพธ์ตามที่เรากำหนด



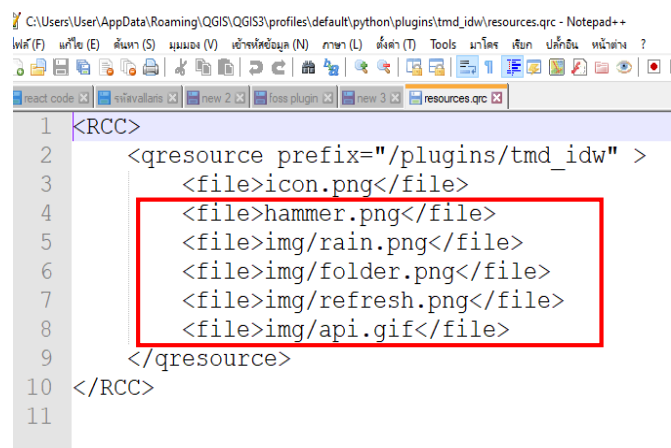
## 2.3 การเพิ่มเติม แก้ไข เปลี่ยนแปลงรูปภาพ และ Icon ในหน้าต่าง GUI ของปลั๊กอิน

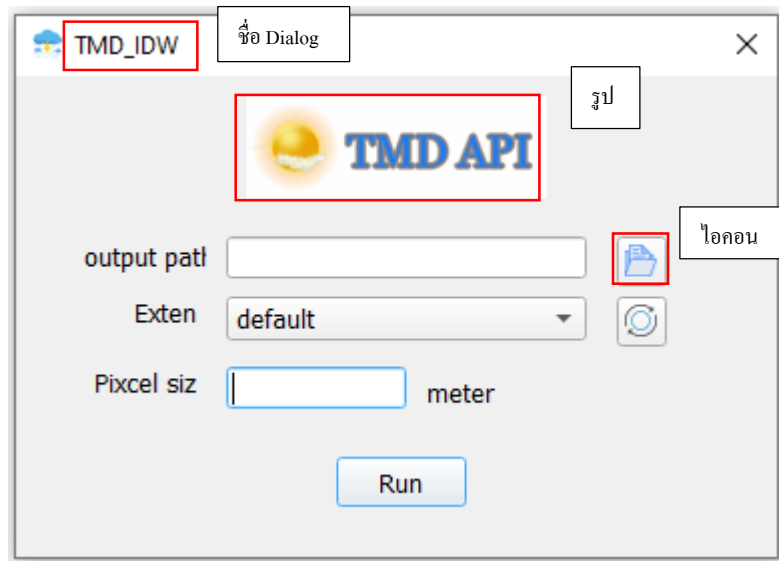


- สร้างโฟลเดอร์ img พร้อมรูปภาพที่ต้องใช้
- เลือกไฟล์ resources.qrc แล้วทำการแก้ไข

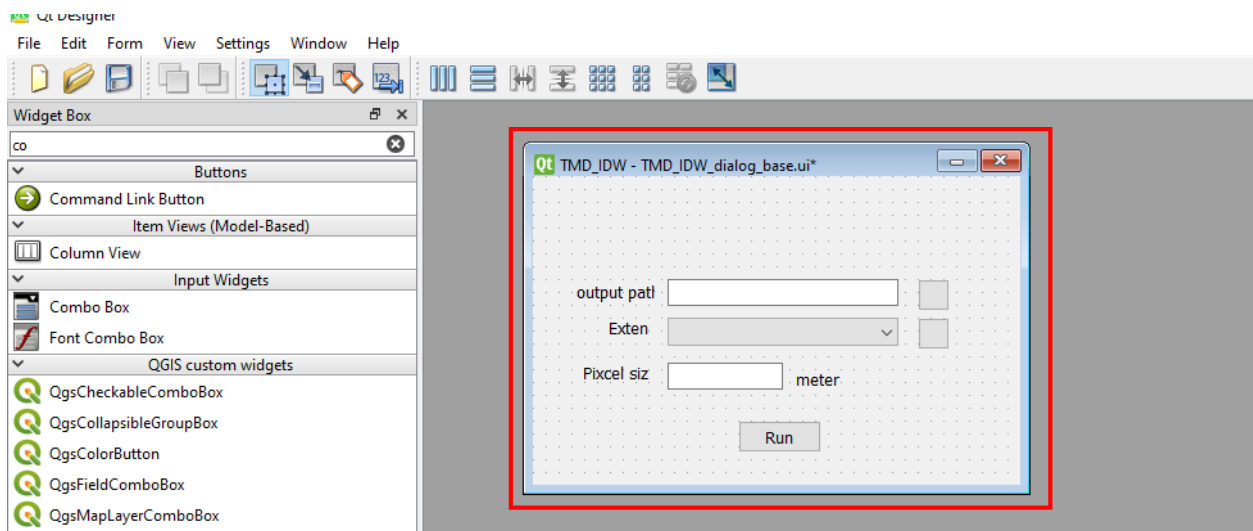
โดยเพิ่ม directory ของรูปที่เราต้องการใช้

- กด run compile.bat



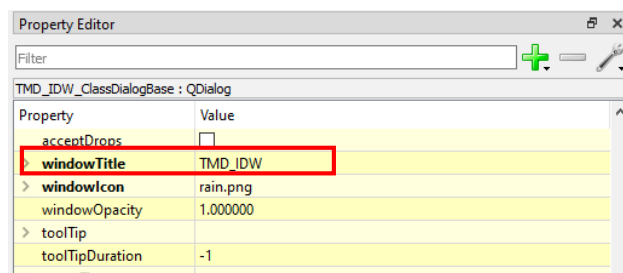


### 2.3.1 การเปลี่ยนชื่อ Dialog



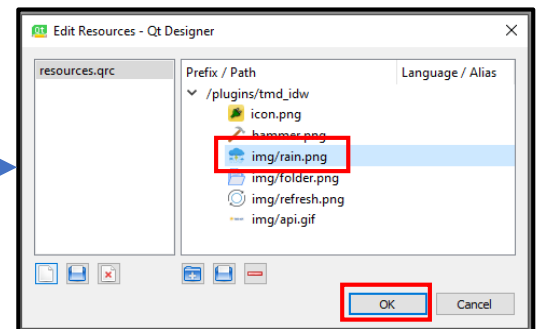
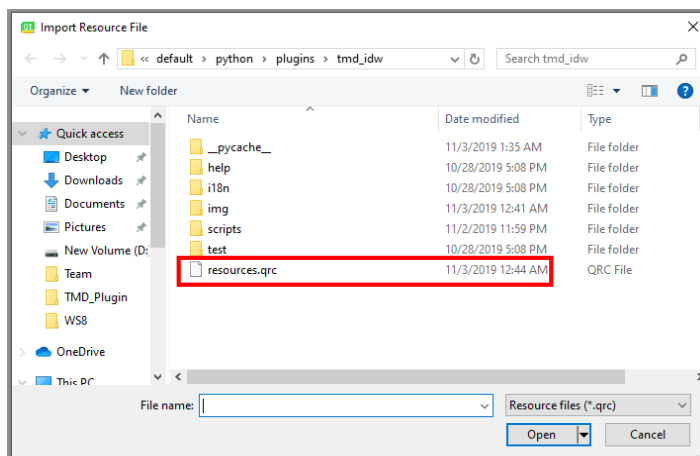
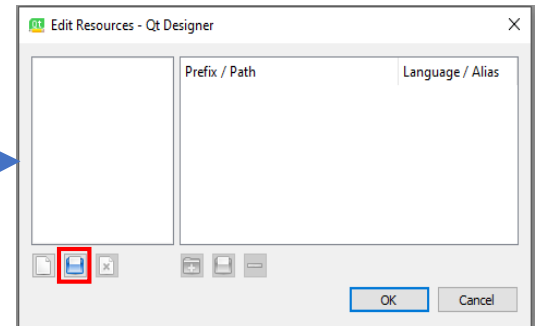
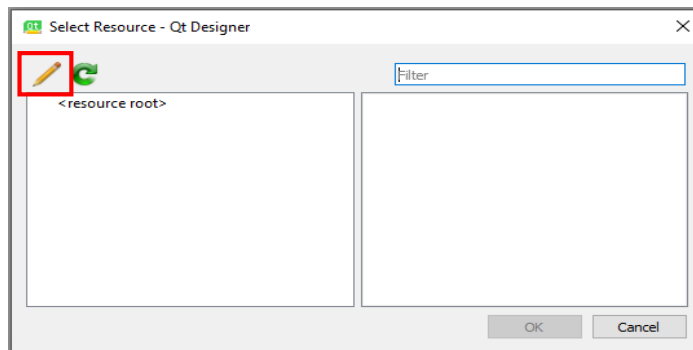
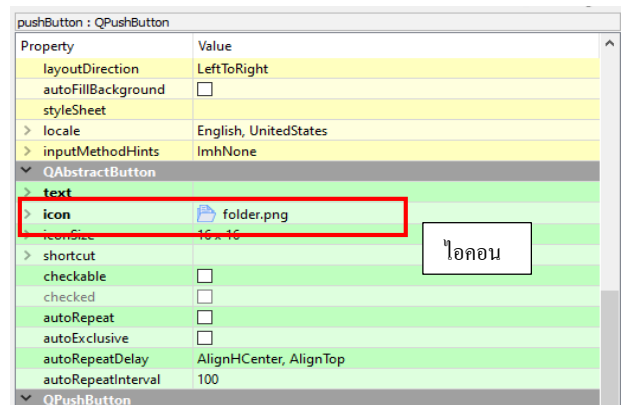
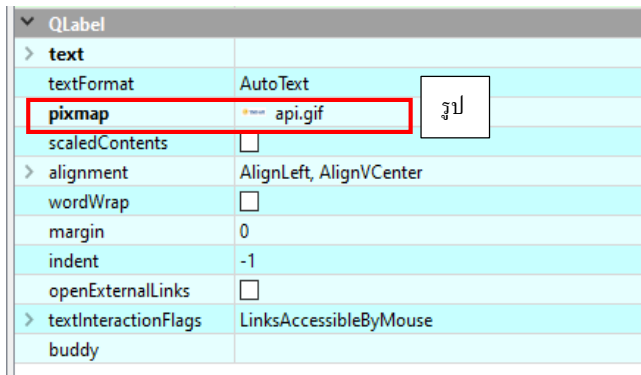
- คลิกที่ Dialog แล้วแก้ไขชื่อของ

Dialog ในช่อง window Title

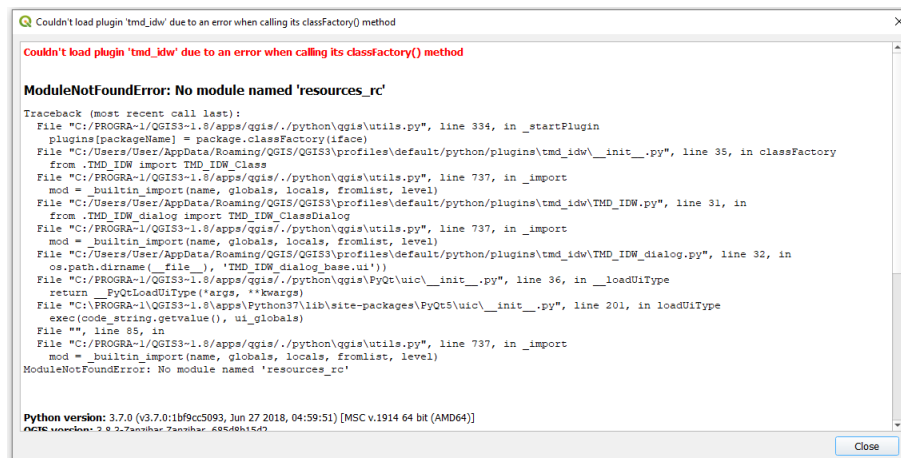




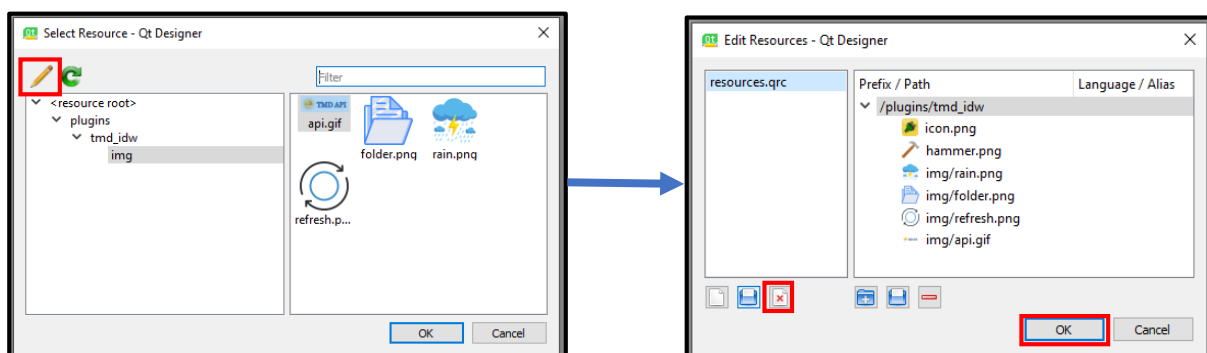
### 2.3.2 การแก้ไขรูปและแก้ไขไอคอน



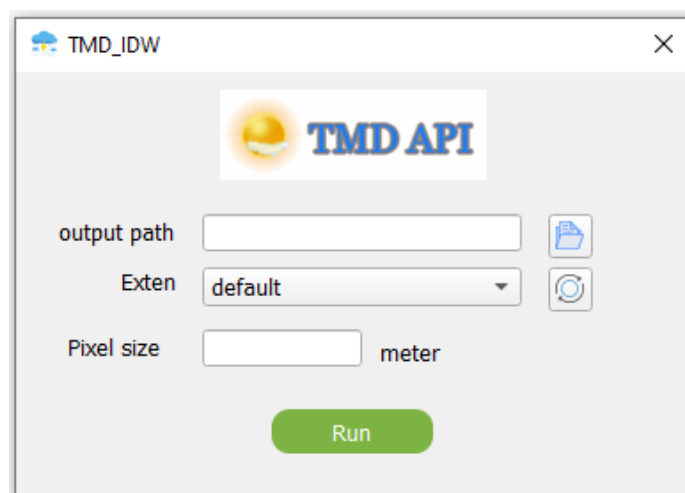
- เมื่อกด  reload ปลั๊กอินจะเกิด error



- แก้ข้อผิดพลาดด้วยการกดแก้ไขแล้ว remove ไฟล์ resources.qrc ออกแล้วทำการบันทึกอีกครั้ง ตามภาพ



- แล้วทำการ reload ปลั๊กอิน เมื่อเปิดก็จะได้ปลั๊กอินตามที่เรากำหนด



### 3.แก้ไขโค้ดของปลั๊กอินให้ปลั๊กอินทำงานตามที่เรต้องการ

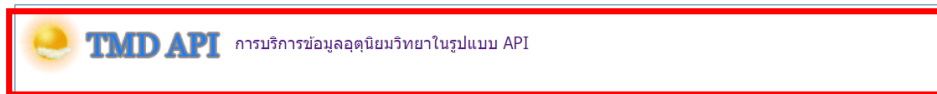
#### 3.1 การรวบรวมโค้ดย่อยๆของปลั๊กอิน


































##### 3.1.1 การเข้าถึงข้อมูลจากกรมอุตุนิยมวิทยาผ่านระบบ TMD API

เข้าเว็บไซต์ <https://www.tmd.go.th/>



- เลือก TMD API



No.	API Name	Version	Description	References
1	WeatherWarningNews	1.00	ข่าวเตือนภัยสภาพอากาศ ข่าวสารภูมิอากาศ ข่าวการติดตามสภาพอากาศร้าย	  
2	DailySeismicEvent	1.00	รายงานผลการเกิดแผ่นดินไหวในประเทศไทย ภูมิภาคใกล้เคียง และข้อมูลการเกิดแผ่นดินไหวทั่วโลก	  
3	ThailandClimateNormal	1.00	รายงานค่าสถิติภูมิอากาศ ค่าปกติ ประเทศไทย 2524-2553 Thailand Climate Standard Normal value 1981-2010	  
4	WeatherToday	2.00	ผลการตรวจวัดลักษณะอากาศรายวัน เวลา 07.00 น. ประกอบด้วยข้อมูล ณ เวลา 07.00 น. และค่าทางสถิติของ 24 ชั่วโมงที่ผ่านมาแสดงข้อมูลทุกวันเวลา 07.00 น.	  
5	Weather3Hours	2.00	ผลการตรวจวัดลักษณะอากาศราย 3 ชั่วโมงทุกสถานี ซึ่งเป็นการตรวจวัดโดยเจ้าหน้าที่ เวลาตรวจวัด ประกอบด้วย 0100 0400 0700 1000 1300 1600 1900 2200 น. แสดงข้อมูลทุกสถานี(122-125 สถานี)	  
6	HeatIndex	1.00	ผลการคำนวณดัชนี Heat Index จากแบบตรวจวัดสภาพอากาศอัตโนมัติทุกสถานีทั่วประเทศไทย แสดงผลการคำนวณในทุก 10 นาที	  
7	ThailandMonthlyRainfall	1.00	ค่าปริมาณฝนสะสมรายเดือนและจำนวนวันที่ฝนตกของสถานีอุตุนิยมวิทยาในประเทศไทย จำแนกรายปี สามารถกำหนดปีที่ต้องการได้	  
8	Station	1.00	สถานีอุตุนิยมวิทยา ที่ติดตั้งเครื่องมือตรวจอากาศทั้งในส่วนกลางและส่วนภูมิภาค ประกอบด้วยสถานีที่ตรวจด้วยเจ้าหน้าที่ และสถานีที่ตรวจด้วยเครื่องมืออัตโนมัติ	  
9	WeatherForecastDaily	1.00	ผลการพยากรณ์อากาศสำหรับประเทศไทยประจำวันละ 4 เวลา 06.00 น, 12.00 น, 17.00 น, 23.00 น ข้อมูลปรับปรุงทุกช่วงเวลาตามที่กำหนด	  
10	WeatherForecast7Days	1.00	ผลการพยากรณ์อากาศสำหรับประเทศไทยล่วงหน้า(ราย 7 วัน) แยกรายจังหวัด	  
11	WeatherForecast7DaysByRegion	1.00	ผลการพยากรณ์อากาศล่วงหน้า(ราย 7 วัน) จำแนกรายภาค ประกอบด้วยภาคเหนือ ภาคตะวันออกเฉียงเหนือ ภาคกลาง ภาคตะวันออก ภาคใต้ฝั่งตะวันออก และภาคใต้ฝั่งตะวันตก	  

- เลือกไอคอน หลอดไฟเพื่อดูเอกสารอ้างอิงและคำแนะนำการใช้งาน TMD API
- API ที่ใช้ครั้งนี้คือ Weather Today <http://data.tmd.go.th/api/WeatherToday/V1/?type=json>

### 3.1.2 การเรียกใช้ API ผ่าน Python Requests Library

```
Python Console
TMD_API.py
1 #import requests
2 import requests
3
4 # get api
5 response = requests.get('http://data.tmd.go.th/api/WeatherToday/V1/?type=json')
6 data=(response.json())
7
8 #check status_code
9 -if response.status_code == 200:
10     print(len(data['Stations'])) # check data from api
11
12 -else:
13     print('requests Fail')
14
15
```


import requests แล้ว ดึงข้อมูลจาก API ของกรมอุตุนิยม โดยใช้ method get ให้ผลลัพธ์ที่ได้เท่ากับ response เช็ค status\_code ว่า API ใช้งานได้ไหม ดูรูปแบบของข้อมูลจาก API แล้วให้ข้อมูลที่ได้อ่านเก็บไว้ในตัวแปร data

### 3.1.3 การเขียนไฟล์ CSV จาก ข้อมูล TMD API

```
Python Console
Untitled-0
Write_CSV_file.py
1 #import csv
2 import csv
3
4 #output path
5 output = 'D:/test.csv'
6
7 #write csv file
8 -with open(output, 'w', encoding="utf-8") as file:
9     file.write('Province,StationNameTh,Latitude,Longitude,Date,Rainfall\n')
10     for line in data['Stations']:
11         print(line)
12         try:
13             file.write(str(line["StationNameEng"]) + ',' + str(line["StationNameTh"]) +
14                 ',' + str(line["Latitude"]["Value"]) + ',' + str(line["Longitude"]["Value"]) +
15                 ',' + str(line["Observe"]["Time"]) + ',' + str(line["Observe"]["Rainfall"]["Value"]) + ',' + str('\n'))
16         except:
17             pass
18
19 file.close()
20
```

Import csv เพื่อใช้ในการเขียนไฟล์ csv หลังจากนั้นทำการเขียนไฟล์โดยเริ่มจาก ชื่อฟิลด์ก่อนในบรรทัดแรกของไฟล์ csv และบรรทัดต่อไปคือข้อมูลของแต่ละฟิลด์ โดยการวนลูปในตัวแปรที่ชื่อว่า data ซึ่งมีข้อมูลจากการใช้ API

### 3.1.4 การนำเข้าข้อมูลจากไฟล์ CSV เป็นจุด (point)



```
1
2 url = "file:///D:/test.csv?encoding=UTF-8&delimiter,&xField=longitude&yField=latitude&crs=epsg:4326"
3
4 #Make a vector layer
5 TMD_layer=QgsVectorLayer(url,"TMD-data","delimitedtext")
6
7 #Add CSV data....
8 QgsProject.instance().addMapLayer(TMD_layer)
```

เลือกที่อยู่ของไฟล์ csv กำหนดคุณสมบัติต่างๆแล้วทำการดึงเข้ามาเป็นข้อมูลจุด

### 3.1.5 เปลี่ยนระบบพิกัดจาก ระบบพิกัดภูมิศาสตร์ เป็นระบบพิกัด UTM zone 47 เพื่อให้ง่ายต่อการคำนวณ Pixel size



```
1 #save for chang CRS 4326 to 32647
2 layer = iface.activeLayer()
3
4 exp_crs = QgsCoordinateReferenceSystem(32647, QgsCoordinateReferenceSystem.EpsgCrsId)
5 ly_csv = QgsVectorLayer(layer.source(), layer.name(), 'delimitedtext')
6 writer = QgsVectorFileWriter.writeAsVectorFormat(ly_csv, 'D:/test_chengeCRS.shp', 'UTF-8', exp_crs, 'ESRI Shapefile', False)
7
8
9
10
```

ในระบบพิกัดภูมิศาสตร์ (4326) ระยะห่างระหว่างเส้นละติจูด 1 องศา คิดเป็นระยะทางบนผิวพิภพ ประมาณ 111 กิโลเมตร (69 ไมล์) จะยากในการคำนวณ Pixel size ของผลลัพธ์ จึงเปลี่ยนเป็นเป็นระบบพิกัด UTM zone 47 (32647)

### 3.1.6 Interpolation (IDW)

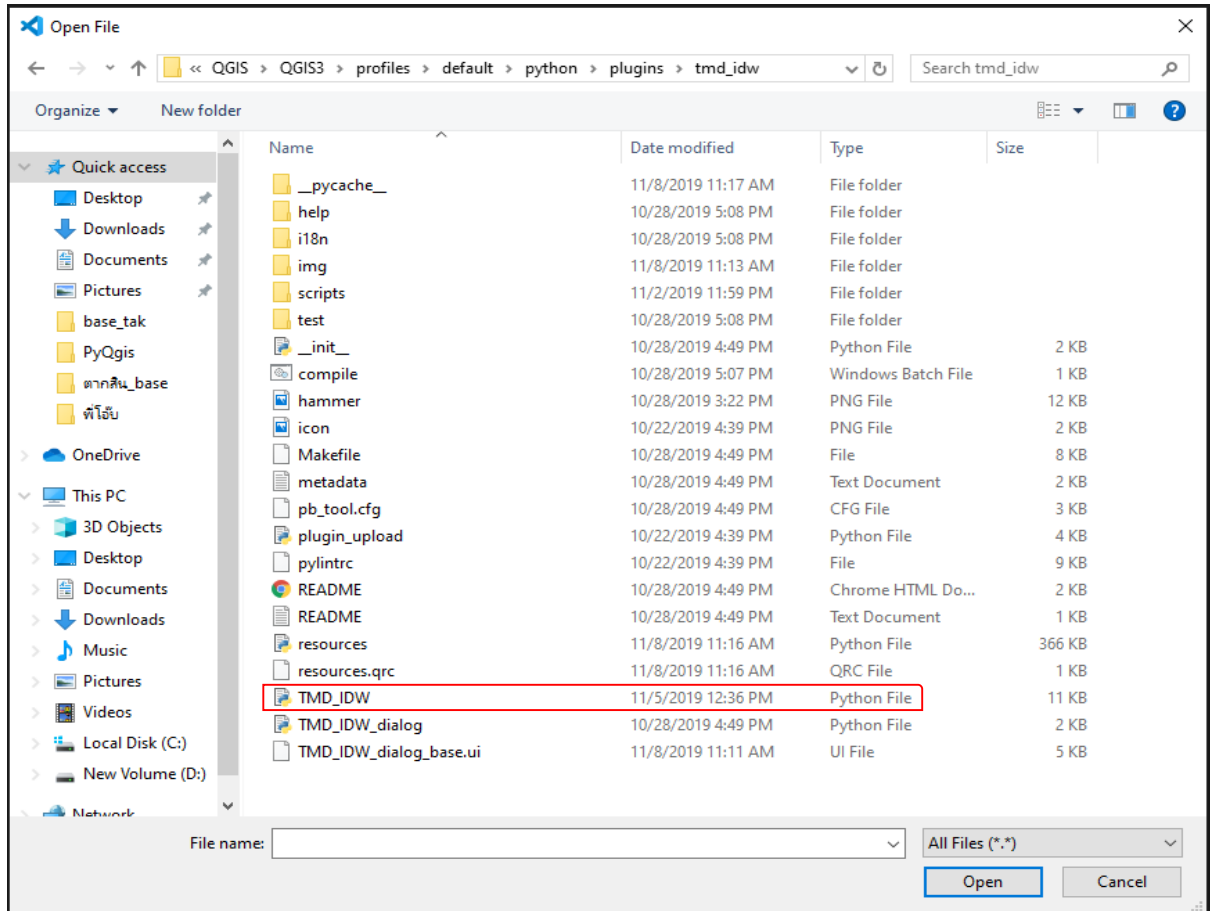


```
Python Console
1 layer = iface.activeLayer()
2 layer_data = QgsInterpolator.LayerData()
3 layer_data.source = layer
4 layer_data.zCoordInterpolation = False
5 layer_data.interpolationAttribute = 5
6 layer_data.sourceType = QgsInterpolator.SourcePoints
7
8 tin_interpolator = QgsIDWInterpolator([layer_data])
9 tin_interpolator.setDistanceCoefficient(2)
10
11 export_path = r"D:\\123.tiff"
12
13
14 #extent and col rows
15 rect = layer.extent()
16 res = 1000
17 ncol = int((rect.xMaximum() - rect.xMinimum()) / res)
18 nrow = int((rect.yMaximum() - rect.yMinimum()) / res)
19
20 output = QgsGridFileWriter(tin_interpolator, export_path, rect, ncol, nrow)
21 output.writeFile()
22
23 rlayer = iface.addRasterLayer(export_path, "interpolation_output")
24
25
26
```

การประมาณค่าเชิงพื้นที่ โดยวิธีการ IDW เป็นการประมาณค่าบนสมมติฐานที่ว่าจุดที่ยังไม่ทราบค่านั้นควรมีอิทธิพลจากจุดควบคุมที่อยู่ใกล้มากกว่าจุดควบคุมที่อยู่ไกล โดยในไฟล์ csv ฟิลด์ที่จะใช้คือฟิลด์ rainfall ซึ่งจะอยู่ในฟิลด์ที่ 6 แต่ในโปรแกรมจะเริ่มต้นจาก 0 เลขจะเป็นฟิลด์ที่ 5 ค่า Distance coefficient ใช้ 2 เป็นค่าเริ่มต้น ค่าขอบเขต ( extent ) ใช้ขอบเขตของข้อมูลที่เลือก และ Pixel size 1000 คือ 1000 เมตร

### 3.2 แก้ไขโค้ดในปลั๊กอิน (เปิดไฟล์ TMD\_IDW.py จากการดาวน์โหลด ประกอบ)

- เลือกแก้ไขไพธอนไฟล์ ตามชื่อที่เราตั้งตอนสร้างปลั๊กอิน



- การ import Library เพิ่มเติม ( บรรทัดที่ 28 - 41)

```

22  """
23
24  from qgis.PyQt.QtCore import QSettings, QTranslator, QCoreApplication
25  from qgis.PyQt.QtGui import QIcon
26  from qgis.PyQt.QtWidgets import QAction, QFileDialog, QMessageBox
27
28  import os.path, requests, csv, datetime
29  from qgis.core import *
30  from qgis.analysis import *
31
32
33  # Initialize Qt resources from file resources.py
34  from .resources import *
35  # Import the code for the dialog
36  from .TMD_IDW_dialog import TMD_IDW_ClassDialog
37
38
39  pixel_size=[]
40
41  date = str(datetime.date.today())
42  #date=date.replace("-", '_')
43
44

```

QFileDialog	ใช้ในการเลือกที่อยู่เพื่อเก็บข้อมูล
QMessageBox	ใช้ในการแจ้งข้อความเพื่อแจ้งเตือน
requests	ใช้ในการเรียก API และ
csv	ใช้ในการเขียนไฟล์ csv
datetime	ใช้ในการดูเวลาปัจจุบันเพื่อตั้งเป็นชื่อไฟล์
from qgis.core import *	เพื่อสามารถใช้งานเครื่องมือหลักของโปรแกรม QGIS
from qgis.analysis import *	เพื่อสามารถใช้งานเครื่องมือด้านการวิเคราะห์ของโปรแกรม QGIS
- การประกาศตัวแปร โกลบอล (Global Variable) ไว้นอกฟังก์ชันเพื่อที่จะสามารถใช้ได้ในทุกฟังก์ชัน	
pixel_size = [ ]	เพื่อเก็บค่า pixel size ของผลลัพธ์
date = str(datetime.date.today())	เพื่อใช้เป็นชื่อไฟล์ (ข้อมูลที่ใช้จากกรมอุตุนิยมวิทยาเป็นรายวัน)



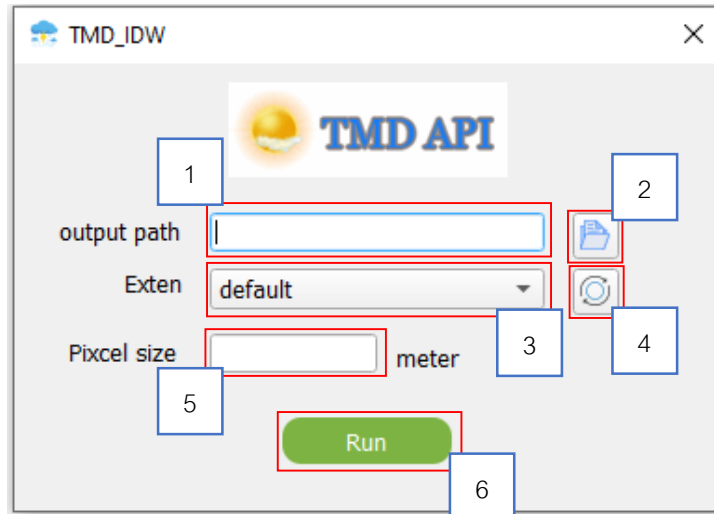
- การแก้ไขไอคอนของปลั๊กอินบนโปรแกรม QGIS จากไอคอน 🌿 เป็นไอคอนใหม่ (บรรทัดที่ 177)  
เริ่มจากนำไอคอนที่ต้องการใช้มาใส่ในไฟล์เดสก์ทอปปลั๊กอิน แล้วทำตามหัวข้อ 2.3 ในการแก้ไขไฟล์ resources.qrc แล้ว run compile.bat

```
173  
174 def initGui(self):  
175     """Create the menu entries and toolbar icons inside the QGIS GUI."""  
176  
177     icon_path = './plugins/tmd_idw/img/rain.png'  
178     self.add_action(  
179         icon_path,  
180         text=self.tr(u'TMD IDW'),  
181         callback=self.run,  
182         parent=self.iface.mainWindow())  
183  
184     # will be set False in run()  
185     self.first_start = True  
186  
187
```

จากนั้น ในฟังก์ชัน initGui มากำหนด icon path ใหม่ให้ตรงกับไฟล์ไอคอนใหม่ของเรา

แล้วกด  reload ปลั๊กอินใหม่ ไอคอนก็จะเปลี่ยนเป็นไอคอนใหม่





1	Object Name = output_path	2	object Name = openBrowse	3	object Name = exten
4	Object Name = refresh	5	object Name = pixel_size	6	object Name = run

```

197
198     def run(self):
199         """Run method that performs all the real work"""
200
201         # Create the dialog with elements (after translation) and keep r
202         # Only create GUI ONCE in callback, so that it will only load wh
203         if self.first_start == True:
204             self.first_start = False
205             self.dlg = TMD_IDW_ClassDialog()
206             self.dlg.run.clicked.connect(self.TMD_to_CSV)
207             self.dlg.openBrowse.clicked.connect(self.OpenBrowse)
208             self.append_layer_name()
209             self.dlg.refresh.clicked.connect(self.append_layer_name)
210

```

เมื่อกด	6	ให้ทำงานฟังก์ชัน TMD_to_CSV
เมื่อกด	2	ให้ทำงานฟังก์ชัน OpenBrowse
เมื่อกดเปิดปลั๊กอิน		ให้ทำงานฟังก์ชัน append_layer_name ทันที
เมื่อกด	4	ให้ทำงานฟังก์ชัน append_layer_name

## อธิบายการทำงานของฟังก์ชัน (ดูข้อมูลจากที่ดาวน์โหลดประกอบ)

ฟังก์ชัน TMD\_to\_CSV ( บรรทัดที่ 224 - 286)

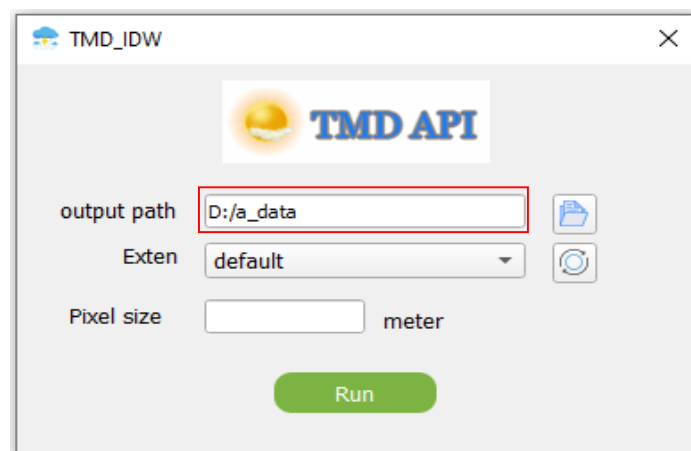
- เช็คว่าค่าช่อง  และ  เป็นค่าว่างไหมถ้าเป็นค่าว่างให้แจ้ง QMessageBox แจ้งเตือน (บรรทัดที่ 226 - 229)
- เช็คว่าค่าในช่อง  โดยดูจากค่าที่ถูก return มาถ้าไม่เป็นคำว่า 'number' ให้แจ้ง QMessageBox แจ้งเตือน ( บรรทัดที่ 230)
- เรียกใช้ API แล้วดูว่า API ใช้ได้ไหมโดยดูจาก status\_code ถ้าไม่เท่ากับ 200 ให้แจ้ง QMessageBox แจ้งเตือน ( บรรทัดที่ 232 - 234)
- เช็คว่ามีข้อมูลของสถานีใน API ไหมถ้าไม่มีให้แจ้ง QMessageBox แจ้งเตือน ( บรรทัดที่ 237)
- นำผลลัพธ์ที่ได้จาก API มาเขียนเป็น csv ไฟล์ จะบันทึกไปยังที่อยู่ตาม  โดยกำหนดชื่อเป็น 'Rain\_IDW\_date(วัน/เดือน/ปี).csv' โดยค่าจากตัวแปร date จะเป็นวัน/เดือน/ปี ปัจจุบัน ( บรรทัดที่ 240 - 253)
- นำข้อมูลจาก csv ไฟล์ บันทึกข้อมูลใหม่เป็นข้อมูลจุด ( point ) จะบันทึกไปยังที่อยู่ตาม  โดยกำหนดชื่อเป็น 'Rain\_IDW\_date(วัน/เดือน/ปี).shp' โดยค่าจากตัวแปร date จะเป็นวัน/เดือน/ปี ปัจจุบัน และแปลงระบบพิกัดจากระบบพิกัดภูมิศาสตร์ (4326) ที่ได้จาก csv ไฟล์ เป็นระบบพิกัด UTM (32647) ( บรรทัดที่ 255 - 263)
- นำเข้าข้อมูลจุดที่แปลงพิกัดเป็นระบบพิกัด UTM (32647) แล้วในโปรแกรม QGIS ( บรรทัดที่ 266 - 267)
- เมื่อผ่านขั้นตอนทั้งหมดให้ทำงานฟังก์ชัน IDW ต่อ ( บรรทัดที่ 269)
- บรรทัดที่ 272 – 286 คือการให้แจ้ง QMessageBox เมื่อไม่เข้าเงื่อนไขต่างๆ

### ฟังก์ชัน IDW ( บรรทัดที่ 289 - 229)

- กำหนดค่าต่างๆในการทำ IDW ค่า ( บรรทัดที่ 292 – 299 )
- กำหนดที่เก็บผลลัพธ์ โดยจะบันทึกไปยังที่อยู่ตาม 1 กำหนดชื่อเป็น ‘Rain\_IDW\_date(วัน/เดือน/ปี).tiff ( บรรทัดที่ 301-302)
- เลือกค่าขอบเขตของผลลัพธ์ โดยถ้าค่าในช่อง 3 เป็น ‘default’ ค่า Exten จากข้อมูลที่เรานำมาทำ IDW หรือถ้าค่าไม่เท่ากับ ‘default’ ให้ใช้ค่า Exten จากชื่อข้อมูลที่เราเลือก ( บรรทัดที่ 305-321)
- ทำการประมวลผล และนำผลลัพธ์เข้ามาโปรแกรม QGIS ( บรรทัดที่ 323 - 328)

### ฟังก์ชัน OpenBrowse ( บรรทัดที่ 333 - 336)

เมื่อเริ่มทำงานจะมี Dialog เด้งขึ้นมาเพื่อให้เลือกโฟลเดอร์เพื่อเก็บข้อมูล เมื่อเลือกเสร็จจะเก็บที่อยู่ของโฟลเดอร์ไว้ใน 1 เป็นข้อความ



### ฟังก์ชัน append\_layer\_name ( บรรทัดที่ 338 - 343)

การนำชื่อของข้อมูลที่เปิดในโปรแกรม QGIS มาใส่ใน 3 เพื่อใช้เป็นค่าขอบเขต (exten) ของผลลัพธ์จากการทำ IDW โดย default คือเอาค่าขอบเขตข้อมูลนั้นๆเลย

ฟังก์ชัน `check_type` ( บรรทัดที่ 348 - 358)

การเช็คข้อมูลในช่อง 

5
---

 ที่ไว้ระบุค่า Pixel size เป็นตัวเลขหรือไม่ ถ้าเป็นตัวเลขให้ return ค่า  
กลับเป็น 'number' ถ้าไม่ใช่ตัวเลขให้ return ค่ากลับเป็น 'no number'