

2015年10月12日

FOSS4G 2015 Tokyo コアデイセッション

@東京大学駒場第Ⅱキャンパス

意外に簡単!?

QGISプラグイン開発

(Python Plugin)



合同会社 緑 I T 事務所

代表 小池 隆

プラグイン開発の背景

➡ GEO × LOD (Linked Open Data)

➡ 「ここペディア」

現在地の
周辺地図

海上防災基地
横浜海上保安部

新港 (一)

大桟橋

海岸通

道駅

東京藝大

日横浜正金銀行本店

横浜税関

横浜税関監視部分庁

山下公園

日本大通り駅

県民ホール

開港記念会館

日本大通

山

11:39

kokopedia.midoriit.com

[1] 横浜開港資料館
横浜開港資料館（よこはまかいこうしりょうかん）は、神奈川県横浜市中区の山下公園の近くにある資料館。
<http://www.kaikou.city.yokohama.jp/>

[2] 横浜赤レンガ倉庫
横浜赤レンガ倉庫（よこはまあかレンガそうこ、Yokohama Red Brick Warehouse）は、神奈川県横浜市中区新港一丁目の横浜港にある歴史的建築物である。明治政府によって保税倉庫として建設され、当時の正式名称は新港埠頭保税倉庫。みなとみらい地区内の2街区にあたる。2号館は1911年（明治44年）、1号館は1913年（大正2年）に竣工。保税倉庫としての役割は1989年（平成元年）までに終え、しばらく放置されていた。2002年（平成14年）に、1号館は展示スペース、ホールなどの文化施設、2号館は商業施設となり、付近一帯は広場と公園を備える赤レンガパークとして整備され、横浜みなとみらい21地区の代表的な観光施設となっている。第45回BCS賞（2004年）受賞。2010年、日本初の「ユネスコ文化遺産保全のためのアジア太平洋遺産賞」優秀賞受賞。本項では赤レンガ倉庫と共に新港埠頭の歴史についても記述する。
<http://www.yokohama-akarenga.jp/index.html>

[3] 日本郵船歴史博物館
日本郵船歴史博物館（にっぽんゆうせんれきしはくぶつかん）は、日本郵船の企業博物館である。
<http://www.nyk.com/rekishi/>

[4] 安藤百福発明記念館
安藤百福発明記念館（あんどうもふくはつめい

地図範囲内の
Wikipedia記事

LOD (Linked Open Data) とは

- Web上でデータを公開・共有し、データ同士をつなげる技術
- データを「主語 述語 目的語」で表現
 - 緯度経度の述語
 - geo:lat
 - geo:long
- ウィキペディアの記事をLOD化して公開する「DBpedia」が有名

DBpedia Japanese

About: 建長寺

ja.dbpedia.org/page/建長寺

DBpedia Japanese ホーム SPARQL Endpoint Faceted Browser iSPARQL ダウンロード

About: 建長寺

An Entity of Type : 立地, from Named Graph : <http://ja.dbpedia.org>, within Data Space : ja.dbpedia.org

建長寺(けんちょうじ)は、神奈川県鎌倉市山ノ内にある禅宗の寺院で、臨済宗建長寺派の大本山である。山号を巨福山(こぶくさん)と称し、寺号は詳しくは建長興国禅寺(けんちょうこくぜんじ)という。鎌倉時代の建長5年(1253年)の創建で、本尊は地藏菩薩、開基(創立者)は鎌倉幕府第5代執権北条時頼、開山(初代住職)は南宋の禅僧蘭溪道隆で、第二世は同じく南宋の兀庵普寧である。鎌倉五山の第一位。境内は「建長寺境内」として国の史跡に指定されている。

Raw Data ▾

Property	Value
dbpedia-owl:abstract	建長寺(けんちょうじ)は、神奈川県鎌倉市山ノ内にある禅宗の寺院で、臨済宗建長寺派の大本山である。山号を巨福山(こぶくさん)と称し、寺号は詳しくは建長興国禅寺(けんちょうこくぜんじ)という。鎌倉時代の建長5年(1253年)の創建で、本尊は地藏菩薩、開基(創立者)は鎌倉幕府第5代執権北条時頼、開山(初代住職)は南宋の禅僧蘭溪道隆で、第二世は同じく南宋の兀庵普寧である。鎌倉五山の第一位。境内は「建長寺境内」として国の史跡に指定されている。
dbpedia-owl:address	神奈川県鎌倉市山ノ内8
dbpedia-owl:thumbnail	http://commons.wikimedia.org/wiki/Special:FilePath/Kencho-ji_108599196_ac1ef7203f_o.jpg?width=300

SPARQL Endpoint

➡ APIによるアクセス

Virtuoso SPARQL Query Editor

Default Data Set Name (Graph IRI)
http://ja.dbpedia.org

Query Text
select distinct * where { <http://ja.dbpedia.org/resource/東京都> ?p ?o . }

クエリ言語 (SPARQL) を用いた問合せ

Results Format: HTML

Execution timeout: milliseconds (values less than 1000 are ignored)

Options: of void variables

(The result can only be sent if it is saved on the server, see [details](#))

JSON

Run Query Reset

Copyright © 2015 [OpenLink Software](#)
Virtuoso version 07.10.3211 on Linux (x86_64-unknown-linux-gnu), Single Server Edition

PediaLayerプラグイン

- 指定した範囲内の事物に関する情報をDBpediaのSPARQL Endpointから取得
→ ベクタレイヤ作成

属性テーブル - pedialayer :: 総地物数: 477, フィルター数: 477, 選...

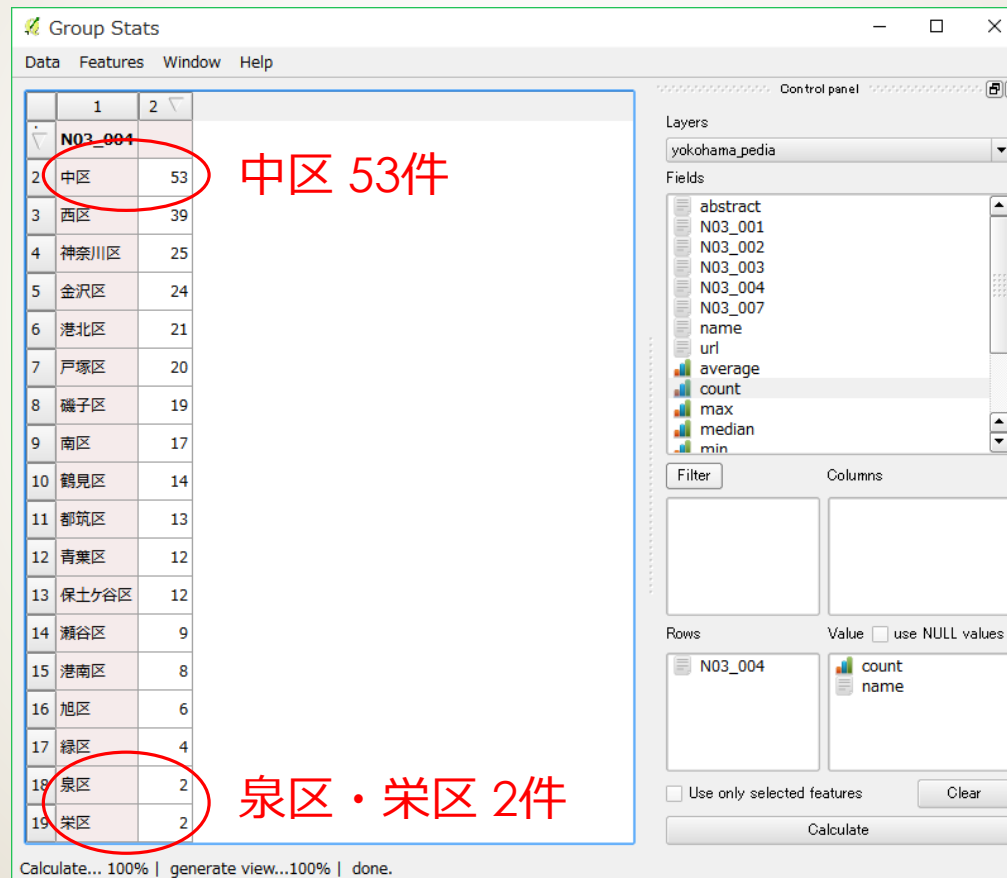
	name	url	abstract
0	地球シミュレータ	http://ja.wikipedia.org/wiki/地球シミュレータ	地球シミュレータ(ちきゅうしミュレータ)
1	鶴岡八幡宮	http://ja.wikipedia.org/wiki/鶴岡八幡宮	鶴岡八幡宮(つるがおか ぼちまんぐ)
2	横浜国立大学	http://ja.wikipedia.org/wiki/横浜国立大学	横浜国立大学(よこはまこくりつだいがく)
3	神奈川大学	http://ja.wikipedia.org/wiki/神奈川大学	神奈川大学(かながわだいがく、英…)
4	国道16号	http://ja.wikipedia.org/wiki/国道16号	国道16号(こくどう16ごう)は、神奈…
5	関東学院大学	http://ja.wikipedia.org/wiki/関東学院大学	関東学院大学(かんとがくいんだいがく)
6	日吉 (横浜市)	http://ja.wikipedia.org/wiki/日吉_(横浜市)	日本 > 神奈川県 > 横浜市 > 日吉
7	金沢文庫	http://ja.wikipedia.org/wiki/金沢文庫	金沢文庫(かねざわぶんこ、かなざわぶんこ)
8	音楽区 (横浜市)	http://ja.wikipedia.org/wiki/音楽区_(横浜市)	音楽区(あおぼろ)は、横浜市を構成する行政区画の一つである。
9	横浜中華街	http://ja.wikipedia.org/wiki/横浜中華街	横浜中華街(よこはまちゅうがわい)は、神奈川県横浜市中区にあり、中華系住民の集まる地区である。
10	横浜開港資料館	http://ja.wikipedia.org/wiki/横浜開港資料館	横浜開港資料館(よこはまかいはつしりょうかん)
11	カトリック山手教会	http://ja.wikipedia.org/wiki/カトリック山手教会	カトリック山手教会(カトリックやまてきょうかい)
12	マリノスタウン	http://ja.wikipedia.org/wiki/マリノスタウン	マリノスタウン(Marinos Town)は、神奈川県横浜市中区にある、横浜F・Cのホームスタジアムである。
13	元町 (横浜市)	http://ja.wikipedia.org/wiki/元町_(横浜市)	元町(もとまち、Motomachi、Yokohama)は、神奈川県横浜市中区にある町である。
14	和光大学	http://ja.wikipedia.org/wiki/和光大学	和光大学(わこうだいがく、英語: Wasei University)
15	戸塚宿	http://ja.wikipedia.org/wiki/戸塚宿	戸塚宿(とつかしゆく、とつかしゆく)は、神奈川県横浜市戸塚区にある宿場町である。
16	横浜赤レンガ倉庫	http://ja.wikipedia.org/wiki/横浜赤レンガ倉庫	横浜赤レンガ倉庫(よこはまあかレンガぐら)は、神奈川県横浜市中区にある歴史的建造物である。
17	玉川大学	http://ja.wikipedia.org/wiki/玉川大学	玉川大学(たまがわだいがく、英語: Tamagawa University)
18	程ヶ谷宿	http://ja.wikipedia.org/wiki/程ヶ谷宿	程ヶ谷宿(ほどがやしゆく、ほどがやしゆく)
19	フェリス学院大学	http://ja.wikipedia.org/wiki/フェリス学院大学	フェリス学院大学(フェリスがくいんだいがく)
20	ブリキのおもちゃ博物館	http://ja.wikipedia.org/wiki/ブリキのおもちゃ博物館	ブリキのおもちゃ博物館は、神奈川県横浜市にある博物館である。

取得する結果の最大数 999

OK キャンセル

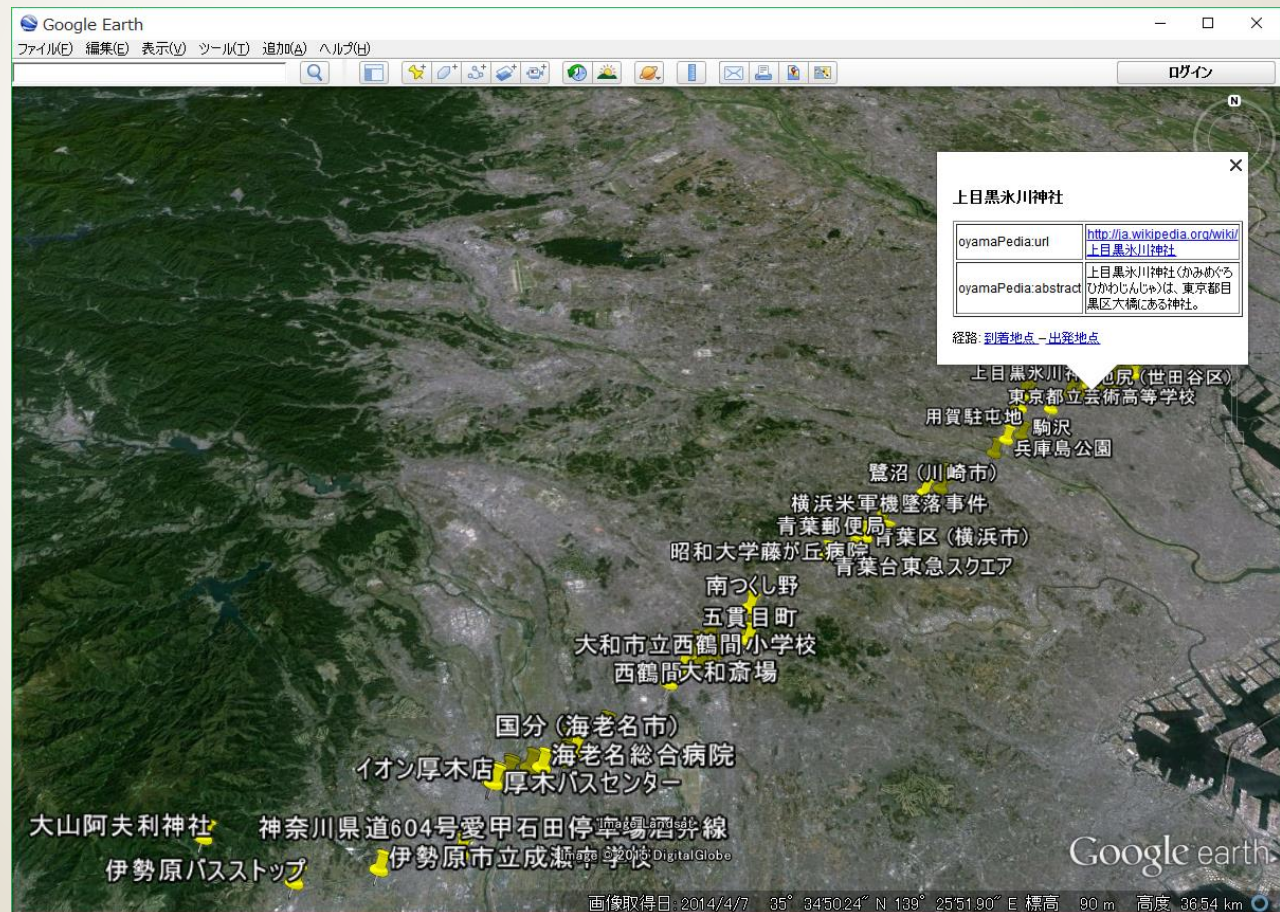
利用例①

➡ 横浜市の区毎のウィキペディア記事集計



利用例②

- ➡ 大山街道（赤坂～丹沢・大山）沿いのウィキペディア記事を抽出してKML作成



(Python Plugin)

プラグイン開発に必要なもの

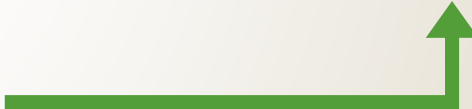
- ➡ QGIS本体
- ➡ Pythonの開発環境 (QGISにバンドル)
 - ➡ QGISのbinフォルダにPATHを通す
 - ➡ またはOSGeo4W Shellを利用 (Windowsの場合)
- ➡ QtのGUI開発環境「Qt Creator」
- ➡ 「Plugin Builder」プラグイン
 - ➡ 「Plugin Reloader」もあると便利
- ➡ makeコマンド (GnuWin32等)

開発者向け情報源

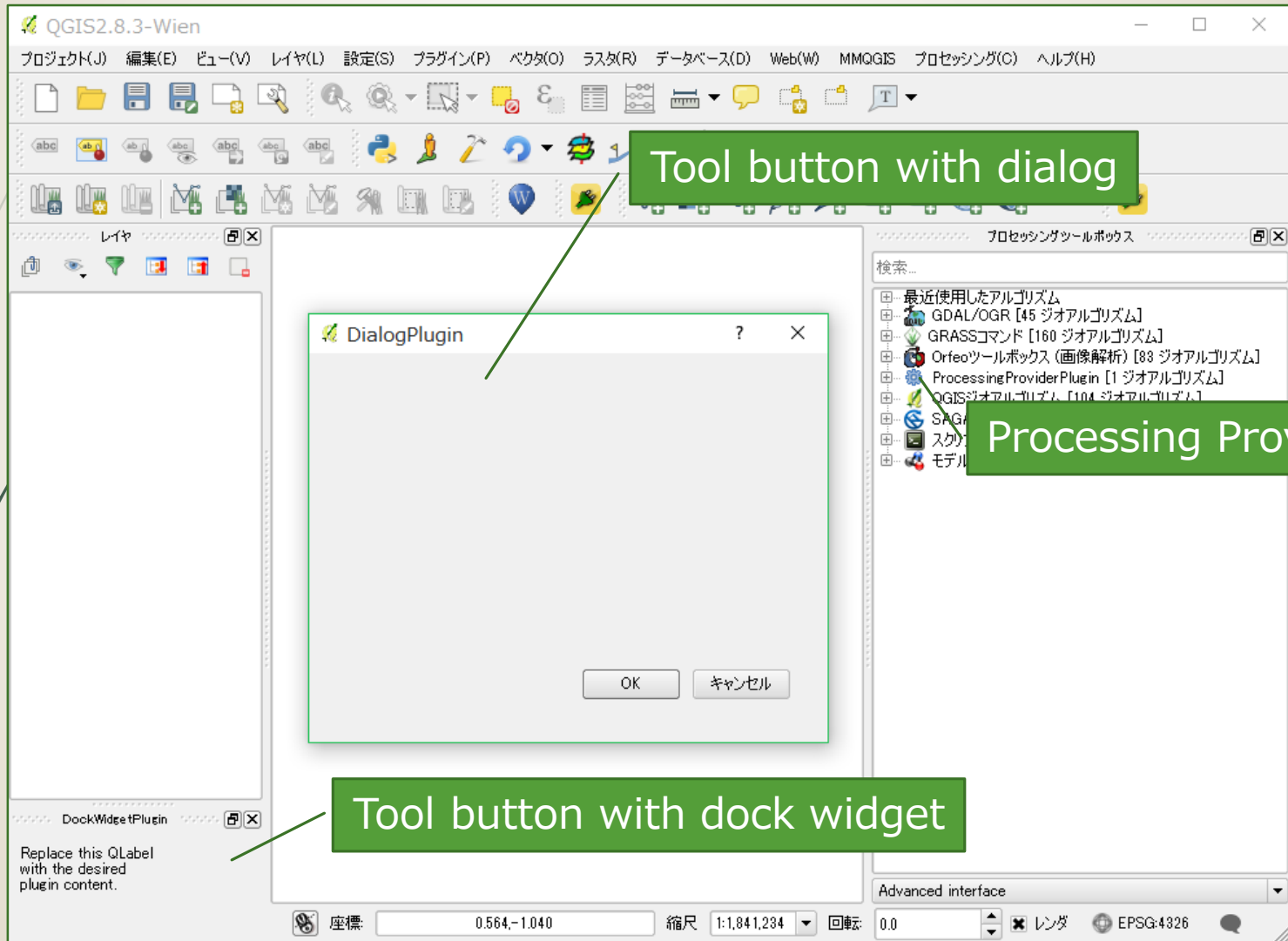
The screenshot shows a web browser window displaying the PyQGIS Developer Cookbook page. The browser's address bar shows the URL `docs.qgis.org/2.8/ja/docs/pyqgis_developer_cookbook/index.html`. The page header includes the QGIS logo with version 2.8, a language dropdown menu set to '日本語', and navigation links. The main content area features a '内容リスト' (Table of Contents) on the left and a 'PyQGIS 開発者用 Cookbook' section on the right. The 'PyQGIS 開発者用 Cookbook' section contains a list of topics:

- はじめに
 - Run Python code when QGIS starts
 - Python コンソール
 - Python プラグイン
 - Python アプリケーション
- Loading Projects
- レイヤのロード
 - ベクターレイヤ
 - ラスターレイヤ
 - マップレイヤレジストリ
- ラスターレイヤを使う
 - レイヤについて
 - 描画スタイル
 - レイヤの更新
 - 値の検索
- ベクターレイヤを使う
 - 属性情報の検索
 - フィーチャの選択
 - ベクターレイヤの反復処理

プラグイン開発～公開の流れ

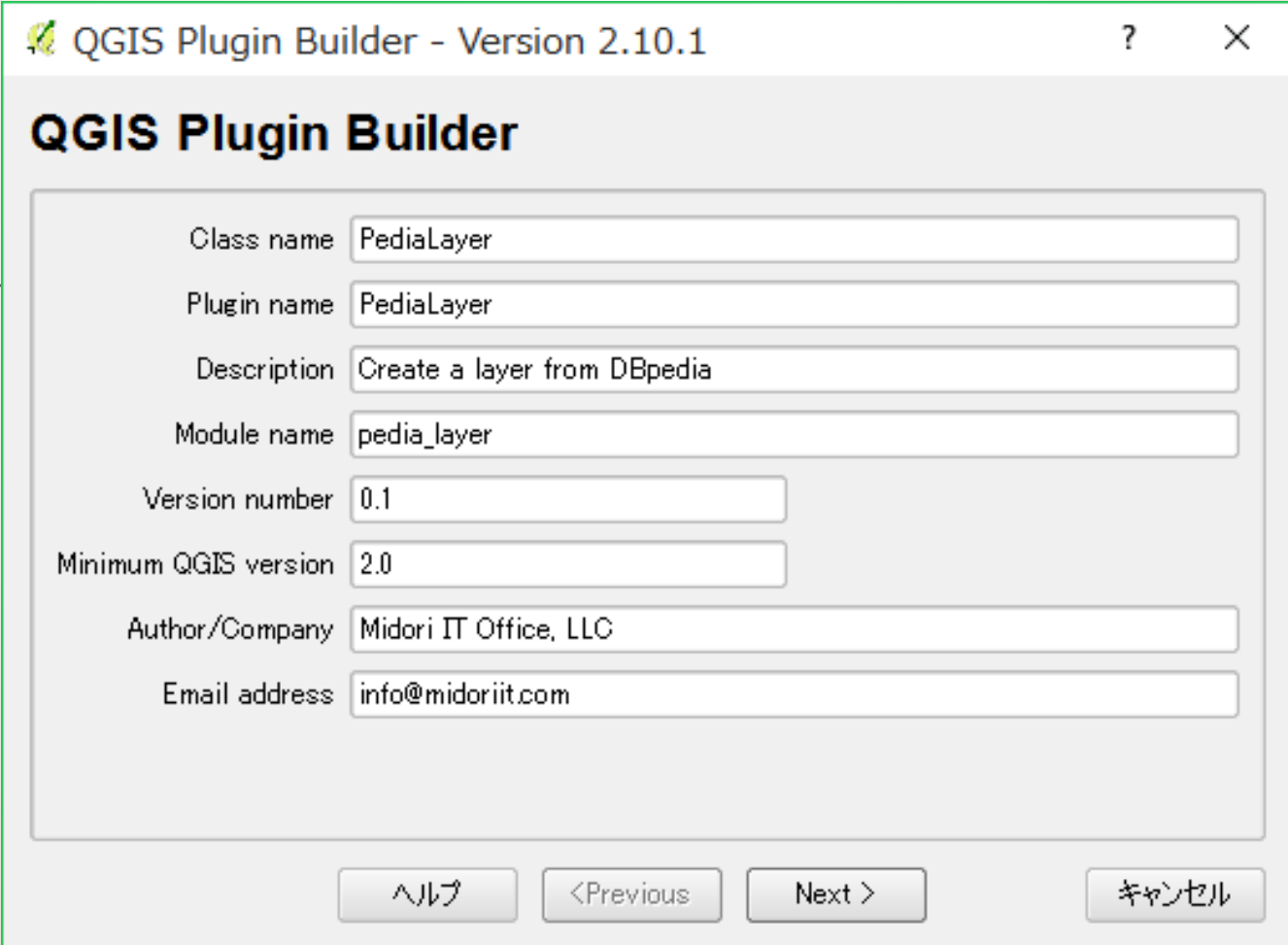
- Plugin Builderプラグインのウィザードで作成
プラグインのテンプレートを作成
- テンプレートに機能を付加
 - GUIの編集
 - Pythonプログラミング
 - makeとインストール
- プラグインの公開  本日はここまで
 - zipパッケージ作成
 - plugins.xmlファイルの作成

プラグインのテンプレートの種類



ウィザード① 基本情報

➡ プラグイン名/クラス名など



QGIS Plugin Builder - Version 2.10.1

QGIS Plugin Builder

Class name

Plugin name

Description

Module name

Version number

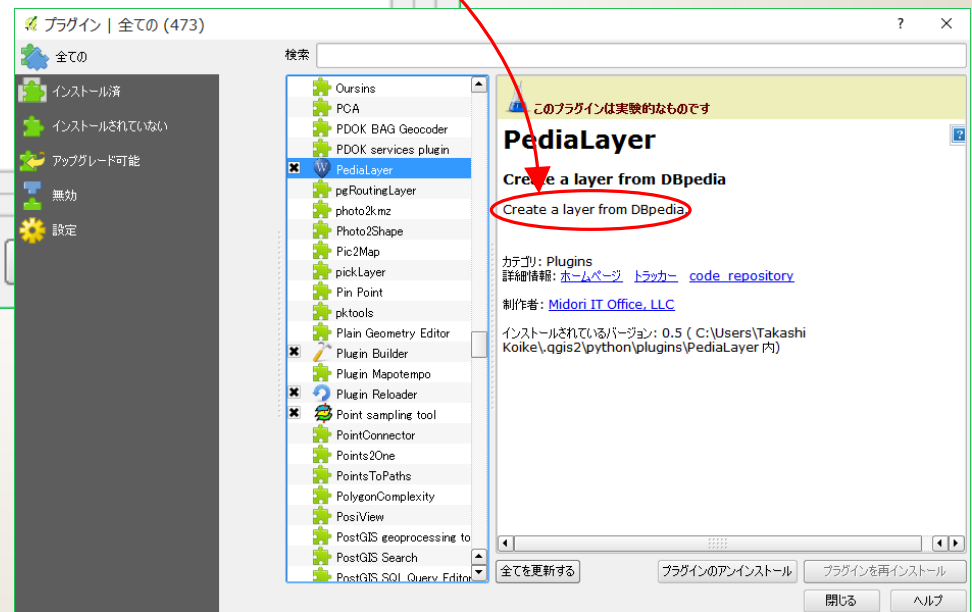
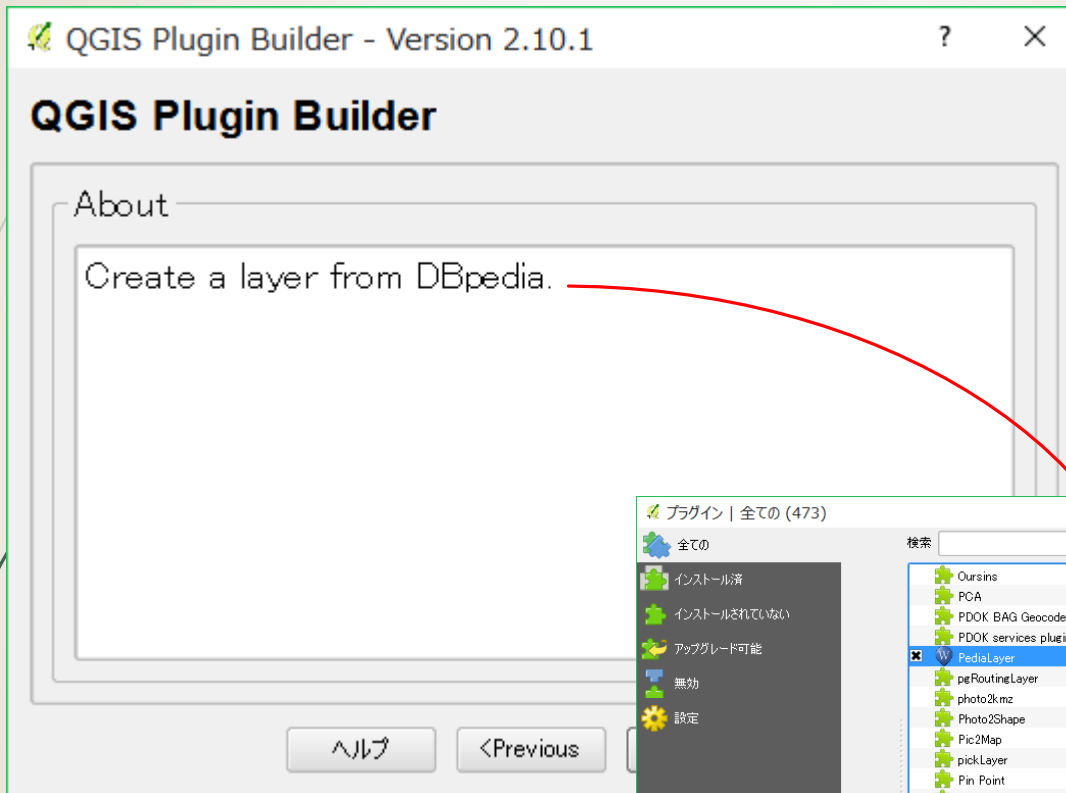
Minimum QGIS version

Author/Company

Email address

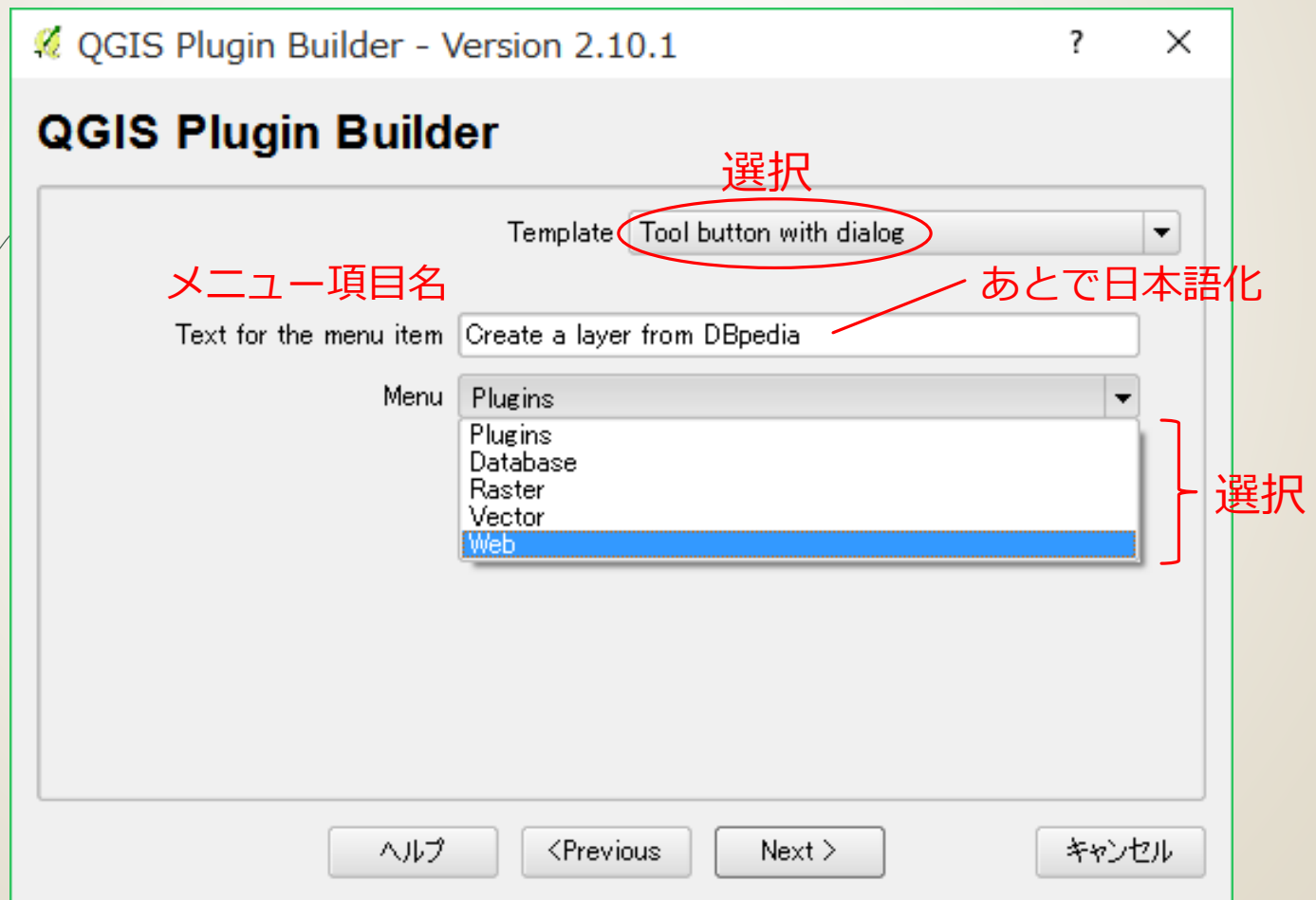
ヘルプ <Previous Next > キャンセル

ウィザード② プラグインの説明



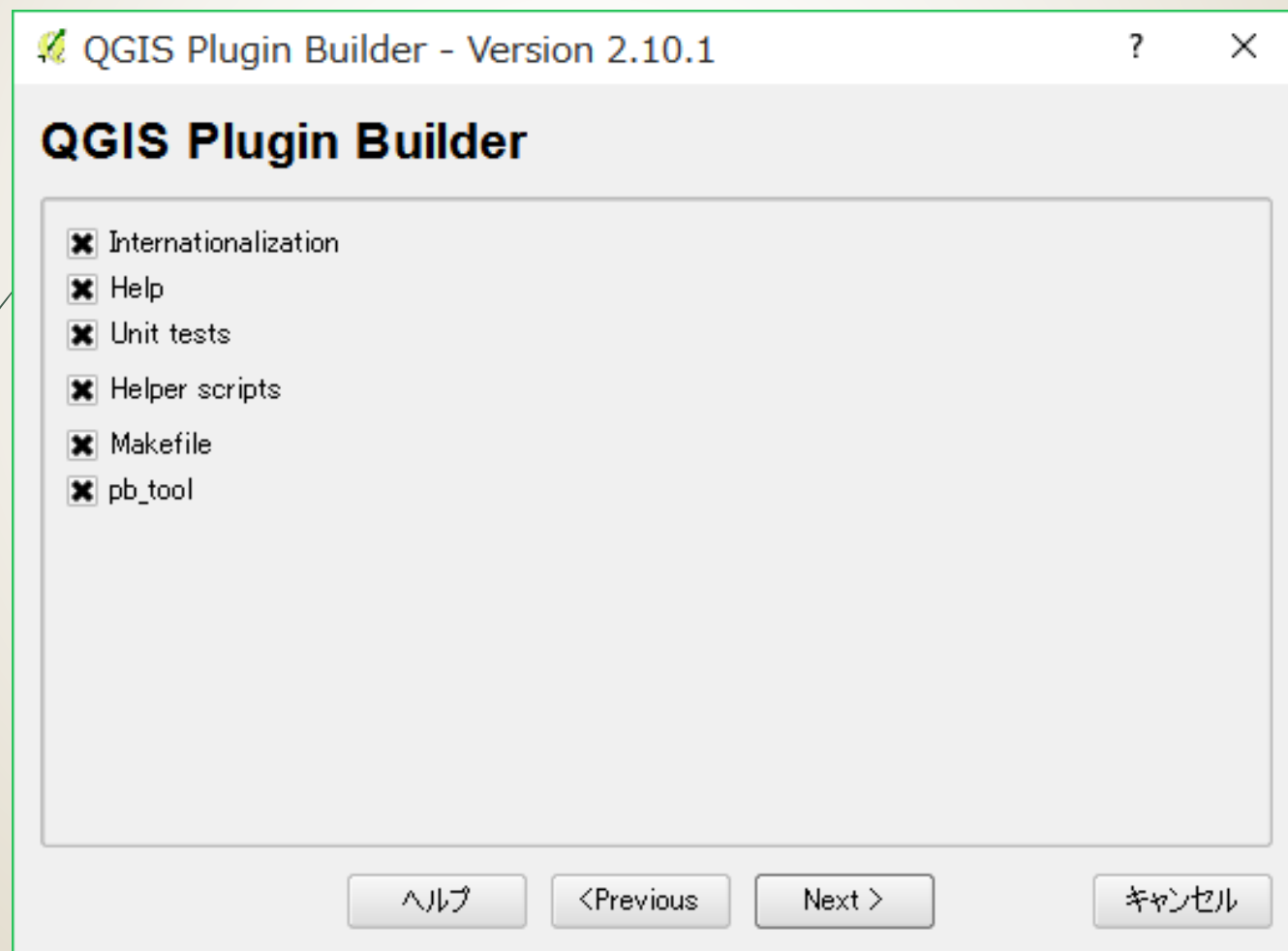
ウィザード③ テンプレートの選択

- ➡ 「Tool button with dialog」を選択



ウィザード④ 出力の選択

➡ とりあえず全部



ウィザード⑤ プラグイン公開先

- ➡ Gitリポジトリ等を入力
 - ➡ この時点では存在しなくても大丈夫

QGIS Plugin Builder - Version 2.10.1

QGIS Plugin Builder

Publication (mandatory Items)

Bug tracker

Repository

Publication (recommended Items)

Home page

Tags ...

Flag the plugin as experimental

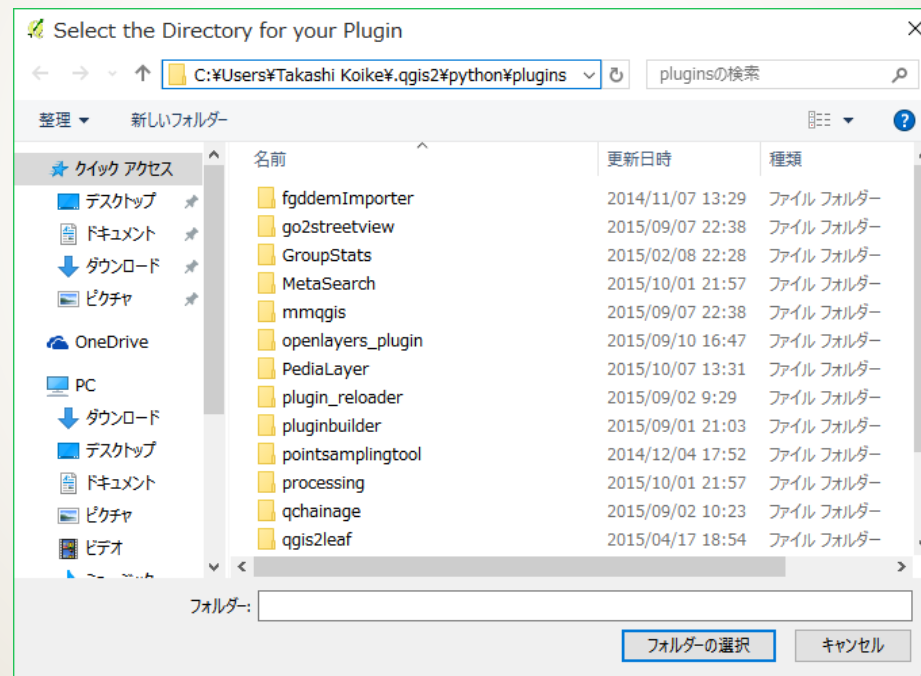
必須

実験的なプラグインではチェックを付ける

ヘルプ <Previous Next > キャンセル

ウィザード⑥ ファイル出力先

- ➡ プラグインフォルダを指定すると便利
- ➡ 更新をPlugin Reloaderで即反映可



C:\Users<ユーザ名>\qgis2\python\plugins

(Windowsの場合)

ウィザード⑦ 確認

Plugin Builder Results

Plugin Builder Results

Congratulations! You just built a plugin for QGIS!

Your plugin **PediaLayer** was created in:
C:\Users\takashi koike\.qgis2\python\plugins\PediaLayer

Your QGIS plugin directory is located at:
C:/Users/Takashi Koike/.qgis2/python/plugins


What's Next

1. In your plugin directory, compile the resources file using pyrcc4 (simply run **make** if you have automake or use **pb_tool**)
2. Test the generated sources using **make test** (or run tests from your IDE)
3. Copy the entire directory containing your new plugin to the QGIS plugin directory (see Notes below)
4. Test the plugin by enabling it in the QGIS plugin manager
5. Customize it by editing the implementation file **pedia_layer.py**
6. Create your own custom icon, replacing the default **icon.png**
7. Modify your user interface by opening **pedia_layer_dialog_base.ui** in Qt Designer

Notes:

- You can use the **Makefile** to compile and deploy when you make changes. This requires GNU make (gmake). The **Makefile** is ready to use, however you will have to edit it to add additional Python source files, dialogs, and translations.
- You can also use **pb_tool** to compile and deploy your plugin. Tweak the **pb_tool.cfg** file included with your plugin as you add files. Install **pb_tool** using *pip* or *easy_install*. See http://loc8.cc/pb_tool for more information.

For information on writing PyQGIS code, see http://loc8.cc/pyqgis_resources for a list of resources.

 ©2011-2015 GeoApt LLC - geoapt.com

OK

Plugin Builderの主な出力

➡ 「Tool button with dialog」 の場合

ファイル名	説明
Makefile	makeコマンドでビルドに使用
icon.png	デフォルトのアイコン 適宜置き換える
resources.qrc	このファイルの編集は不要 makeするとpyrcc4コマンドで resources.pyが作成される
<モジュール名>.py	プログラムのテンプレート 編集して機能を実装する
<モジュール名>_dialog.py	ダイアログ表示のコード このファイルの編集は不要
<モジュール名>_dialog_base.ui	ダイアログのGUI定義 Qt Creatorで編集する

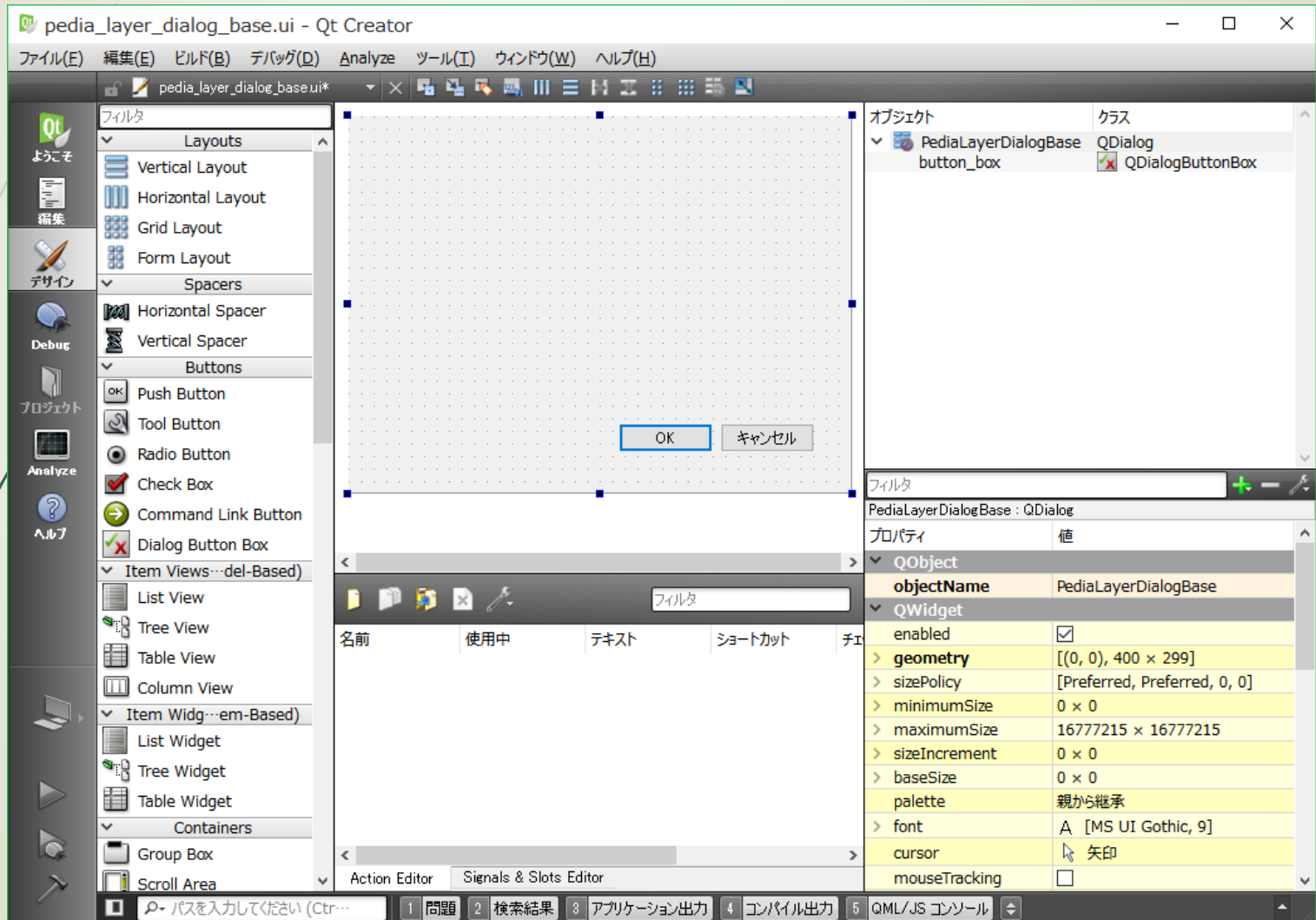
※選択したテンプレートにより、出力されるファイルは異なる

metadata.txt

- ➡ ウィザードで入力した、プラグインのメタ情報を保持（編集不要、修正可）

```
[general]
name=PediaLayer
qgisMinimumVersion=2.0
description=Create a layer from DBpedia
version=0.5
author=Midori IT Office, LLC
email=info@midoriit.com
about=Create a layer from DBpedia.
tracker=https://github.com/midoriit/PediaLayer/issues
repository=https://github.com/midoriit/PediaLayer
experimental=True
```

Qt CreatorによるGUI編集 (前)



Qt CreatorによるGUI編集 (後)

The screenshot shows the Qt Creator IDE with a dialog box design in progress. The design view shows a dialog box with a title bar, a search filter input field, and a 'Max number of results' label with a spin box set to 100. A red arrow points to the text 'Max number of results' with the note '文字列はあとで日本語化' (The string will be localized later). The widget palette on the left lists various Qt widgets like Layouts, Spacers, Buttons, and Item Views. The object inspector on the right shows the hierarchy of the dialog box, including the 'PediaLayerDialogBase' widget and its properties like 'windowModality', 'enabled', 'geometry', 'sizePolicy', 'minimumSize', 'maximumSize', 'sizeIncrement', 'baseSize', 'palette', 'font', and 'cursor'.

名前	使用中	テキスト	ショートカット	チェ...
フィルタ				

オブジェクト	クラス
PediaLayerDialogBase	QDialog
button_box	QDialogButtonBox
groupBox	QGroupBox
comboBox	QComboBox
radioButton_l	QRadioButton
radioButton_m	QRadioButton
label_2	QLabel
spinBox	QSpinBox
buttonGroup	QButtonGroup

プロパティ	値
QObject	
objectName	PediaLayerDialogBase
QWidget	
windowModality	NonModal
enabled	<input checked="" type="checkbox"/>
geometry	[(0, 0), 400 × 207]
sizePolicy	[Preferred, Preferred, 0, 0]
minimumSize	0 × 0
maximumSize	16777215 × 16777215
sizeIncrement	0 × 0
baseSize	0 × 0
palette	親から継承
font	A [MS UI Gothic, 9]
cursor	矢印

機能の実装① importの追加

- `from qgis.core import *`
 - QGISのコアAPIを使用するため
- `from qgis.gui import *`
 - QGISのGUI操作APIを使用するため
- `import json, urllib, urllib2`
 - DBpediaからデータを取得するため
- `from PyQt4.QtCore import *`
 - Qtのさまざまな機能を使用するため

機能の実装② runメソッド

➡ 初期化処理とイベント処理の追加

```
def run(self):
```

```
    ダイアログの初期化処理
```

```
    # show the dialog
```

```
    self.dlg.show()
```

```
    # Run the dialog event loop
```

```
    result = self.dlg.exec_()
```

```
    # See if OK was pressed
```

```
    if result:
```

```
        イベント処理：ベクタレイヤを追加
```

黒字のコードは
Plugin Builderが
生成したコード

ダイアログの初期化処理

```
self.dlg.comboBox.clear()
```

```
layers = self.iface.legendInterface().layers()
```

```
layer_list = []
```

QGISからすべてのレイヤを取得

```
for layer in layers:
```

ベクタレイヤまたはラスタレイヤのみ

```
    if isinstance(layer, QgsVectorLayer) or ¥  
        isinstance(layer, QgsRasterLayer):
```

```
        layer_list.append(layer.name())
```

```
self.dlg.comboBox.addItem(layer_list)
```

イベント処理の概要

- マップキャンバスまたは選択されたレイヤから矩形範囲を取得
- DBpediaから情報を取得するためのSPARQLクエリを作成
- クエリを発行し、JSON形式で結果取得
- ベクタレイヤを新規作成
- 取得したJSONデータからfeatureを作成し、レイヤに追加

<http://midoriit.com/2015/09/qgisプラグインの開発2.html>

SPARQLクエリ

```
SELECT distinct ?name ?abstract ?lat ?lon ?url
WHERE {
  ?s rdfs:label ?name ;
  dbpedia-owl:abstract ?abstract ;
  foaf:isPrimaryTopicOf ?url ;
  geo:lat ?lat ;
  geo:long ?lon .
  FILTER ( ?lon > xMin && ?lon < xMax &&
           ?lat > yMin && ?lat < yMax )
  FILTER ( LANG(?name)='ja' &&
           LANG(?abstract)='ja' )
} LIMIT n
```

青字は述語

範囲指定

ベクタレイヤと地物の追加

```
newLayer = QgsVectorLayer(...)
newLayer.setProviderEncoding("UTF-8")
QgsMapLayerRegistry.instance().addMapLayer(...)
newLayer.startEditing()
newLayer.addAttribute(QgsField("name", ...))
newLayer.addAttribute(QgsField("url", ...))
newLayer.addAttribute(QgsField("abstract", ...))
for item in list:
    feature = QgsFeature(newLayer.pendingFields())
    feature.setGeometry(QgsGeometry.fromPoint(...))
    feature.setAttribute("name", ...)
    feature.setAttribute("url", ...)
    feature.setAttribute("abstract", ...)
    newLayer.addFeature(feature)
newLayer.commitChanges()
newLayer.updateExtents()
```

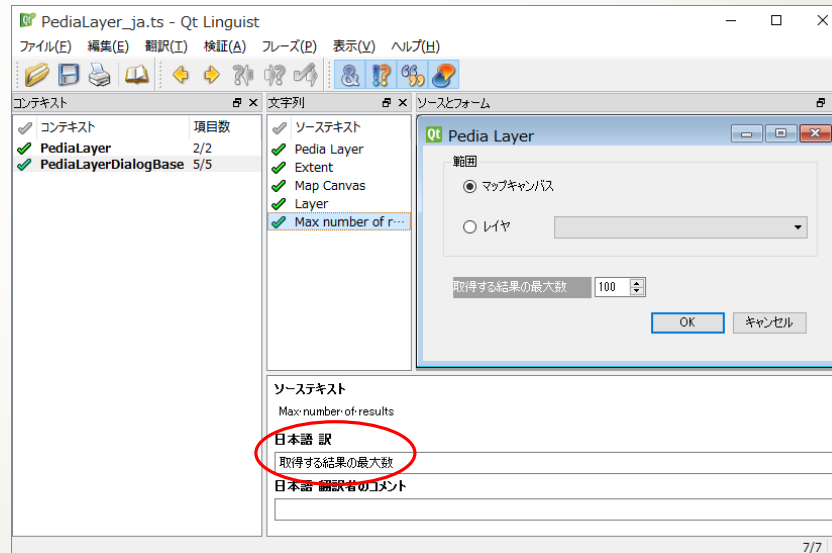
レイヤ追加

属性追加

地物追加

Qtによる国際化対応

- ➡ `pylupdate4`で、ソースファイルから翻訳すべき文字列を抽出
- ➡ `linguist`で日本語訳の文字列を入力



- ➡ `lrelease`で.qmファイルを作成

makeとインストール

➤ make

- resources.pyを作成するため、1度は必ず実行
 - icon.pngファイルを更新したら、make cleanしてからmake
- 「プラグインの管理とインストール」でプラグインを有効化
- Plugin Reloaderを使ってReloadすれば、プラグイン修正後のQGISの再起動は不要

まとめ

- ➡ Plugin BuilderでQGISプラグインの開発は大幅に省力化
- ➡ 豊富なPythonライブラリの利用
- ➡ 開発者向け情報が充実
- ➡ プラグインを作って公開しましょう

Let's
Try!

PediaLayerプラグイン

プラグインリポジトリ

<http://midoriit.com/qgis/plugins.xml>

ソースコード

<https://github.com/midoriit/PediaLayer>

ご清聴ありがとうございました

midoriit.com