

# Getting Started with Python Programming

## QGIS Tutorials and Tips



Author

Ujaval Gandhi

<http://google.com/+UjavalGandhi>

Translations by

SongHyun Choi

□ □ □ □ □ □ □ □ □ □

**QGIS**은 오픈 소스 지리정보시스템 소프트웨어로, 다양한 데이터 형식과 레이어를 처리할 수 있는 강력한 도구입니다. QGIS는 GIS 데이터를 시각화하고 분석하는 데 사용되며, 다양한 플랫폼에서 실행 가능합니다. QGIS는 GIS 분야에서 가장 인기 있는 소프트웨어 중 하나이며, 많은 전문가와 학생이 이를 사용하여 프로젝트를 수행합니다.

□ □ □ □

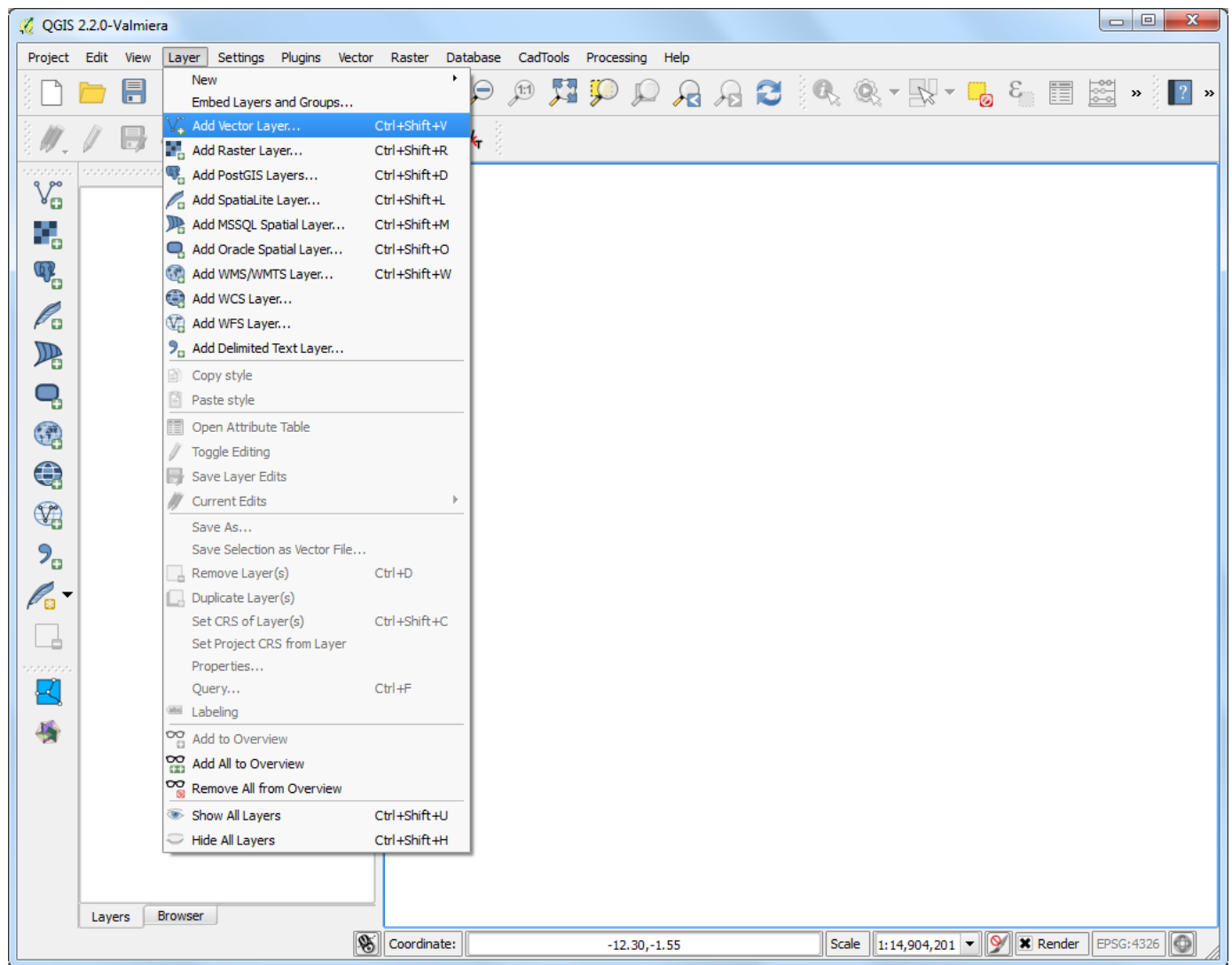
[illegible]

Natural Earth [Airports](#) □□□□ □□□□.

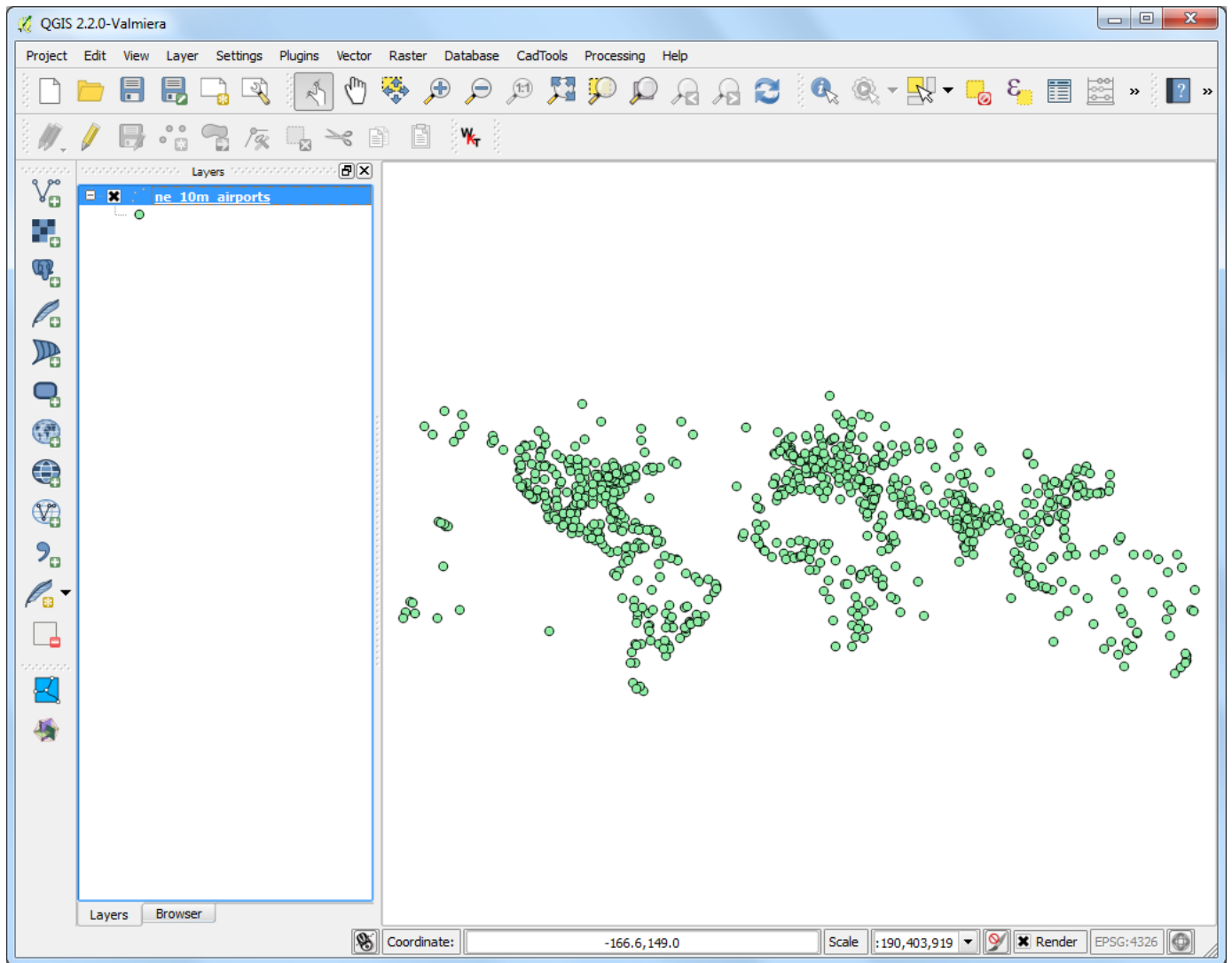
```
`Airports` shapefile <http://www.naturalearthdata.com/http://www.naturalearthdata.com/download/10m/cultural/ne\_10m\_airports.zip>`_ 00000 00000.
```

□□□ □□: [NATURALEARTH]

```
1. QGIS --> File --> Open --> ne_10m_airports.zip
Vector Layer --> Add
:guilabel: Open --> ne_10m_airports.shp
:guilabel: OK
```



2. QGIS `ne\_10m\_airports` 。



3. Identify the fields in the 'ne\_10m\_airports' layer. The fields are 'name' and 'iata\_code'.





```
5. QGIS 0000 0000 0000 0000 0000 00 00 0000. 0000 000 0 00 0000
>>>`■ ■■ ■■■■■ ■■■ ■■■■. QGIS■■■ ■■■■■■■■ `iface 000 00000
000. QGIS00 00 0000 0000 0000 0000 000 00 00000 00
Enter`0 0000. 0 0000 00000 00 0000 0000 0000 0000 `layer` 000 00000.
```

```
layer = iface.activeLayer()
```



6. `dir()` returns a list of attributes and methods for the active layer. This list can be used to check if a specific attribute or method exists in the layer object.

```
dir(layer)
```



7. `getFeatures()` returns a list of features. Each feature is a dictionary with keys for the feature's attributes and values for the attribute data. The first key is always 'geometry'. The other keys are the names of the attributes. The values are the values of the attributes. For example, if a feature has two attributes, 'name' and 'population', the dictionary would look like this:

```
{
  'geometry': {
    'x': 100,
    'y': 100
  },
  'name': 'New York',
  'population': 1000000
}
```

```
for f in layer.getFeatures():
    print f
```





8. `for f in layer.getFeatures():`  
 `print f['name'], f['iata_code']`

```

for f in layer.getFeatures():
    print f['name'], f['iata_code']

```



```

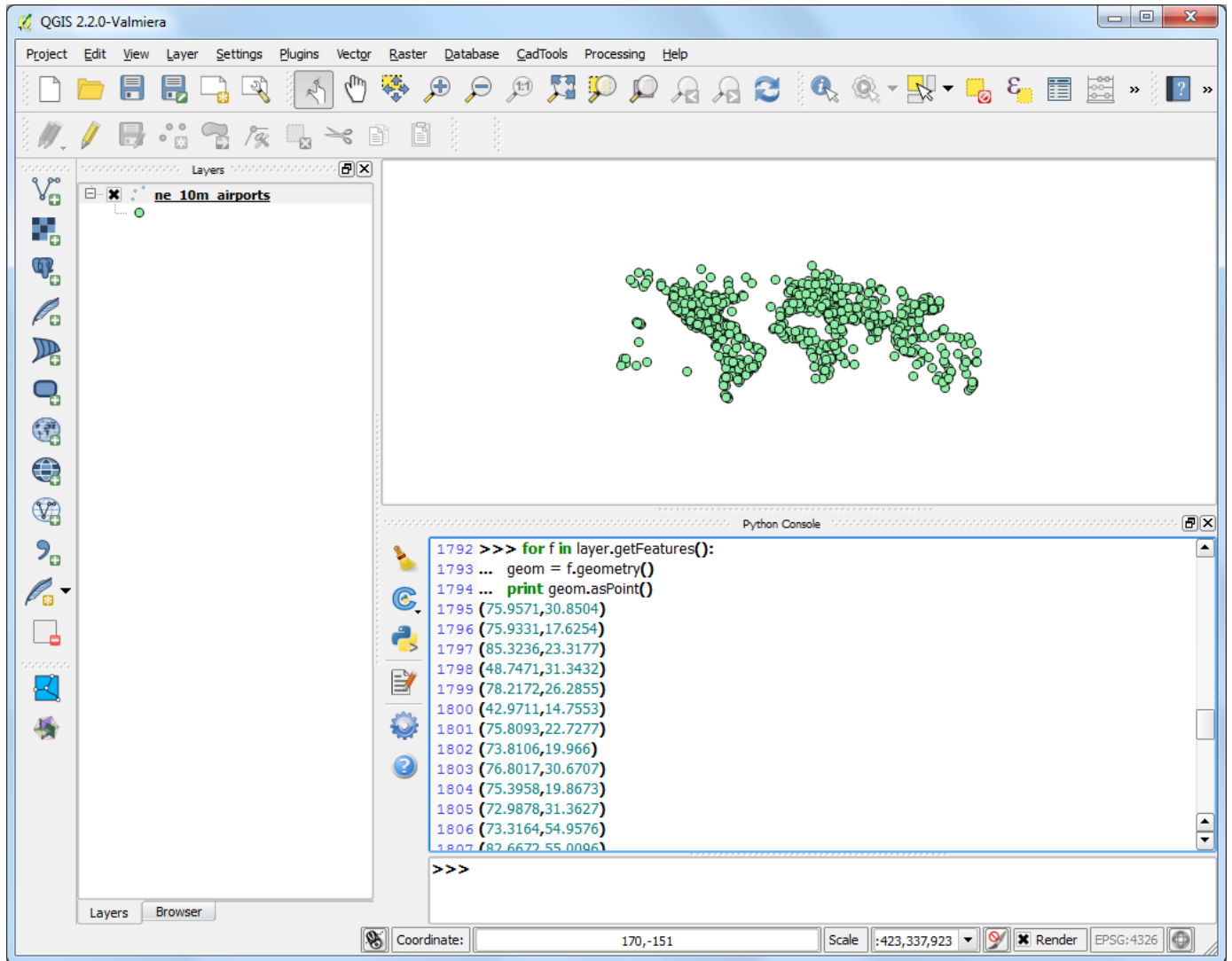
for f in layer.getFeatures():
    geom = f.geometry()
    print geom.asPoint()

```

```

for f in layer.getFeatures():
    geom = f.geometry()
    print geom.asPoint()

```



10. `for f in layer.getFeatures():`  
`geom = f.geometry()`  
`print geom.asPoint().x()`

```

for f in layer.getFeatures():
    geom = f.geometry()
    print geom.asPoint().x()

```



11. `for f in layer.getFeatures():`  
`geom = f.geometry()`  
`print '%s, %s, %f, %f' % (f['name'], f['iata_code'],`  
`geom.asPoint().y(), geom.asPoint().x())`

```

for f in layer.getFeatures():
    geom = f.geometry()
    print '%s, %s, %f, %f' % (f['name'], f['iata_code'],
        geom.asPoint().y(), geom.asPoint().x())
  
```





airports.txt - Notepad

File Edit Format View Help

```
Sahnewal, LUH, 30.850360, 75.957072
Solapur, SSE, 17.625415, 75.933060
Birsamunda, IXR, 23.317725, 85.323597
Ahwaz, AWZ, 31.343159, 48.747107
Gwalior, GWL, 26.285488, 78.217219
Hodeidah Int'l, HOD, 14.755253, 42.971096
Devi Ahilyabai Holkar Int'l, IDR, 22.727749, 75.809292
Gandhinagar, ISK, 19.966021, 73.810567
Chandigarh Int'l, IXC, 30.670725, 76.801726
Aurangabad, IXU, 19.867297, 75.395843
Faisalabad Int'l, LYP, 31.362744, 72.987819
Omsk Tsentralny, OMS, 54.957648, 73.316360
Novosibirsk Tolmachev, OVB, 55.009585, 82.667152
Zaporozhye Int'l, OZH, 47.873264, 35.301873
Simpang Tiga, PKU, 0.464601, 101.446569
Rota Int'l, ROP, 14.171771, 145.243980
Surgut, SGC, 61.340167, 73.408496
Tiruchirappalli, TRZ, 10.760357, 78.708958
Turbat Int'l, TUK, 25.988795, 63.027933
Quetta Int'l, UET, 30.249043, 66.948731
Zahedan Int'l, ZAH, 29.475294, 60.900709
Abdul Rachman Saleh, MLG, -7.929980, 112.711419
Barnaul, BAX, 53.363385, 83.550453
Adampur, NULL, 31.432942, 75.758483
Bareilly, NULL, 28.421809, 79.452003
Dhamial, NULL, 33.561415, 73.032050
Cheongju Int'l, CJJ, 36.722023, 127.495916
Gwangju, KWJ, 35.140005, 126.810839
Daegu Int'l, TAE, 35.899928, 128.637538
Ulsan, USN, 35.592896, 129.355731
Radin Inten II, TKG, -5.242567, 105.176060
Allahabad, IXD, 25.443522, 81.731727
Chelyabinsk, CEK, 55.297792, 61.512259
Tainan, TNN, 22.950668, 120.209733
Taichung, RMQ, 24.266656, 120.630704
Rotterdam The Hague, RTM, 51.949130, 4.433844
Voronezh-Chertovitskoye, VOZ, 51.812617, 39.225450
Liverpool John Lennon, LPL, 53.336375, -2.858621
Vishakapatnam, VTZ, 17.727958, 83.223522
Sultan Hasanuddin Int'l, UPG, -5.058937, 119.545691
Vava'u Int'l, VAV, -18.586006, -173.968094
Newcastle Int'l, NCL, 55.037085, -1.710346
Goloson Int'l, LCE, 15.745160, -86.851469
```