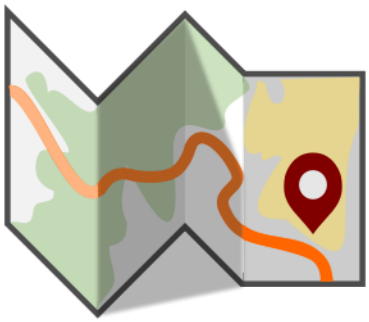
Noțiuni de bază despre programarea în Python

QGIS Tutorials and Tips



Author

Ujaval Gandhi

http://google.com/+UjavalGandhi

Translations by

Sorin Călinică

Noțiuni de bază despre programarea în Python

QGIS are o interfață de programare puternică, care vă îngăduie să extindeți funcționalitatea de bază a aplicației, precum și să scrieți script-uri pentru automatizarea sarcinilor. QGIS suportă limbajul de scripting popular, Python. Chiar dacă sunteți un începător, învățarea unor noțiuni despre Python și despre interfața de programare a QGIS vă permite să fiți mult mai productivi în munca dvs. Acest tutorial nu necesită cunoștințe de programare prealabile, având scopul de a oferi o introducere în scriptarea Python, în cadrul QGIS (PyQGIS).

Privire de ansamblu asupra activității

Vom încărca un strat vectorial de tip punct, reprezentând toate aeroporturile majore, folosind scriptarea Python pentru a crea un fișier text cu numele, codul, latitudinea și longitudinea pentru fiecare dintre aeroporturile stratului.

Obținerea datelor

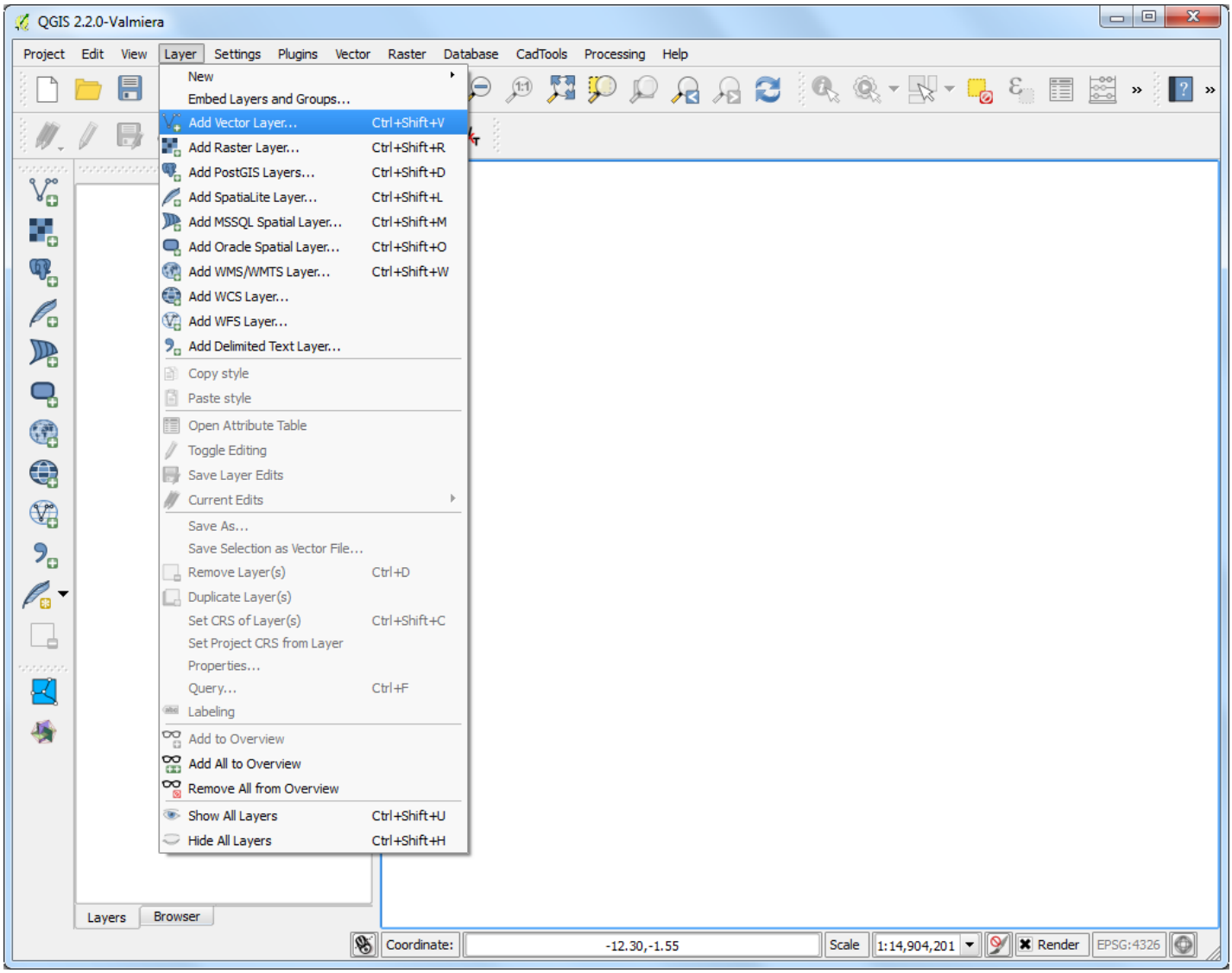
Vom folosi setul de date Airports de la Natural Earth.

Decărcați Fișierul shape de Aeroporturi.

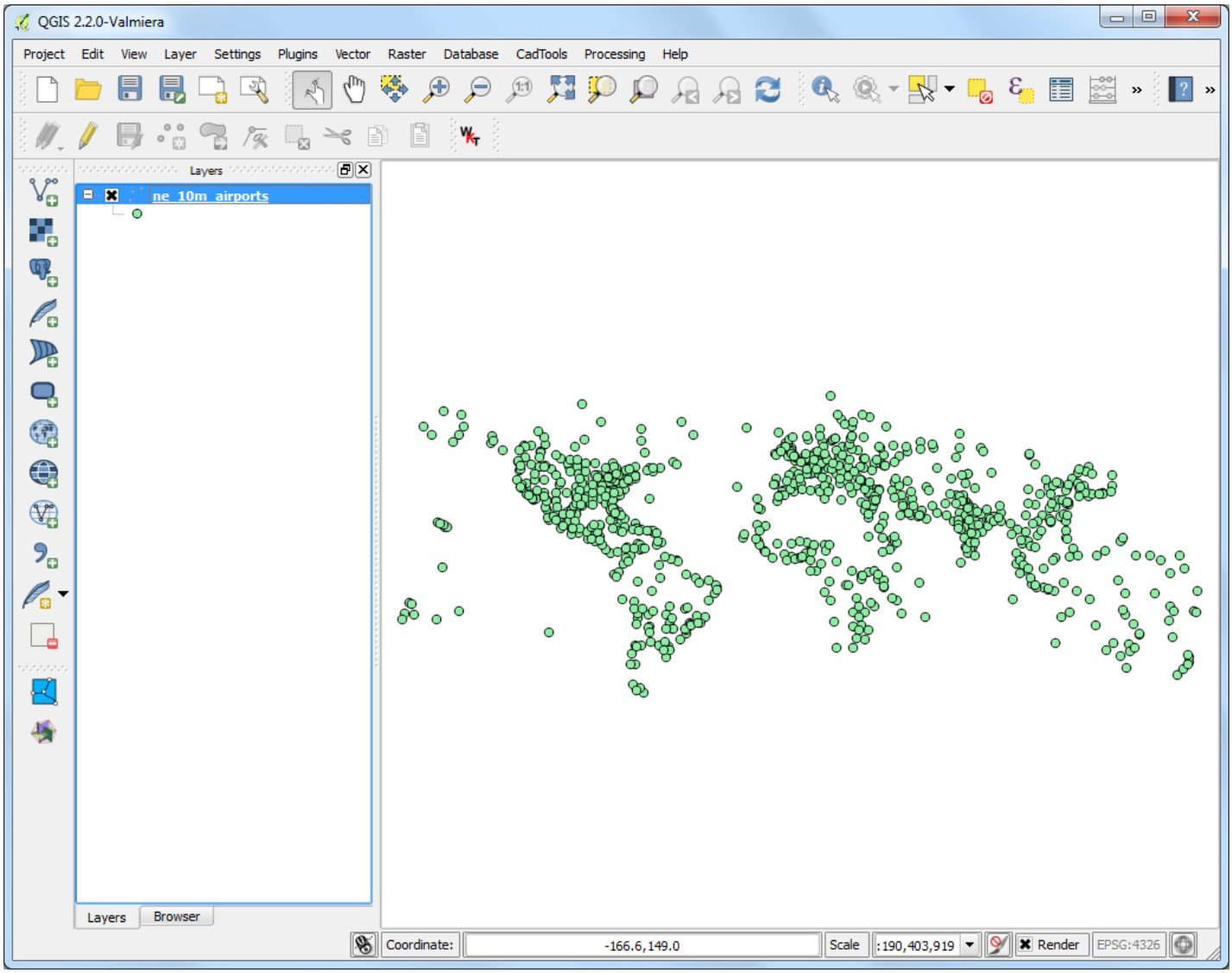
Sursa de date [NATURALEARTH]

Procedura

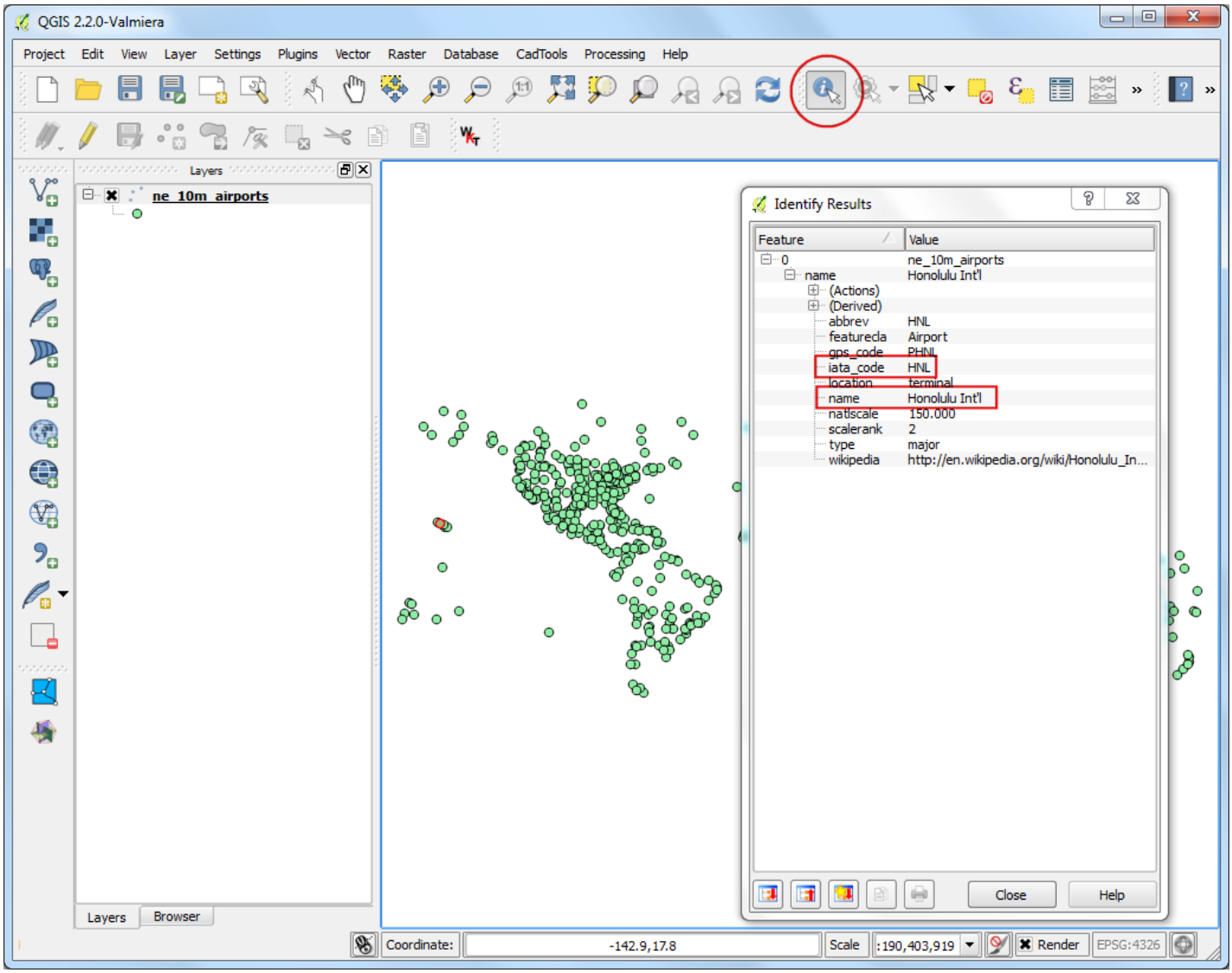
1. În QGIS, mergeți la Layers ‣ Add Vector Layer. Navigați la fișierul descărcat ***ne_10m_airports.zip*** și faceți clic pe Open. Selectați stratul ***ne_10m_airports.shp*** și faceți clic pe OK.



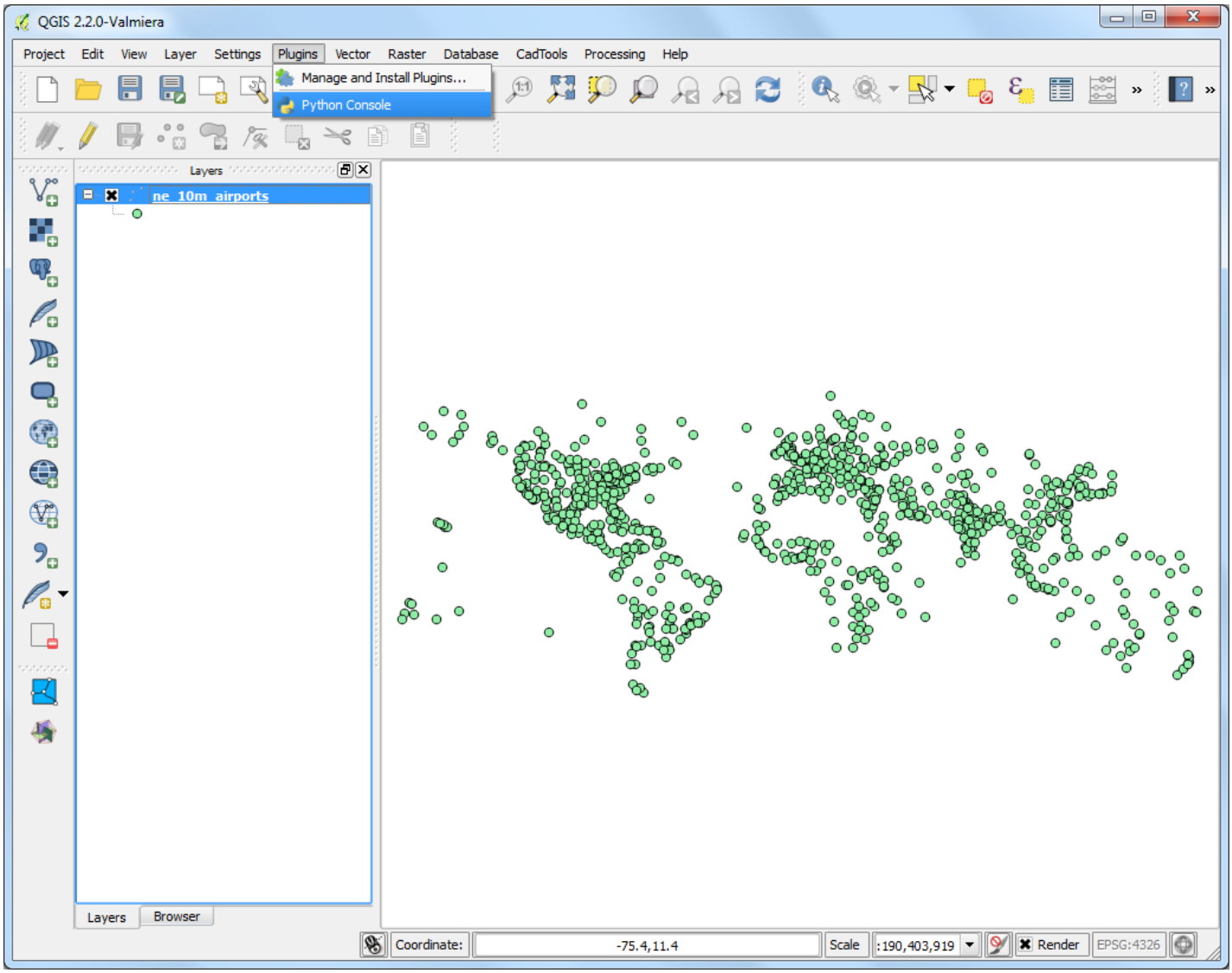
2. Veți vedea stratul ***ne_10m_airports*** încărcat în QGIS.



3. Selectați instrumentul Identify și faceți clic pe oricare dintre puncte, pentru a examina atributele disponibile. Veți vedea că numele aeroportului, precum și codul de 3 cifre al acestuia, sunt conținute în atributele ***name***, respectiv ***iata_code***.

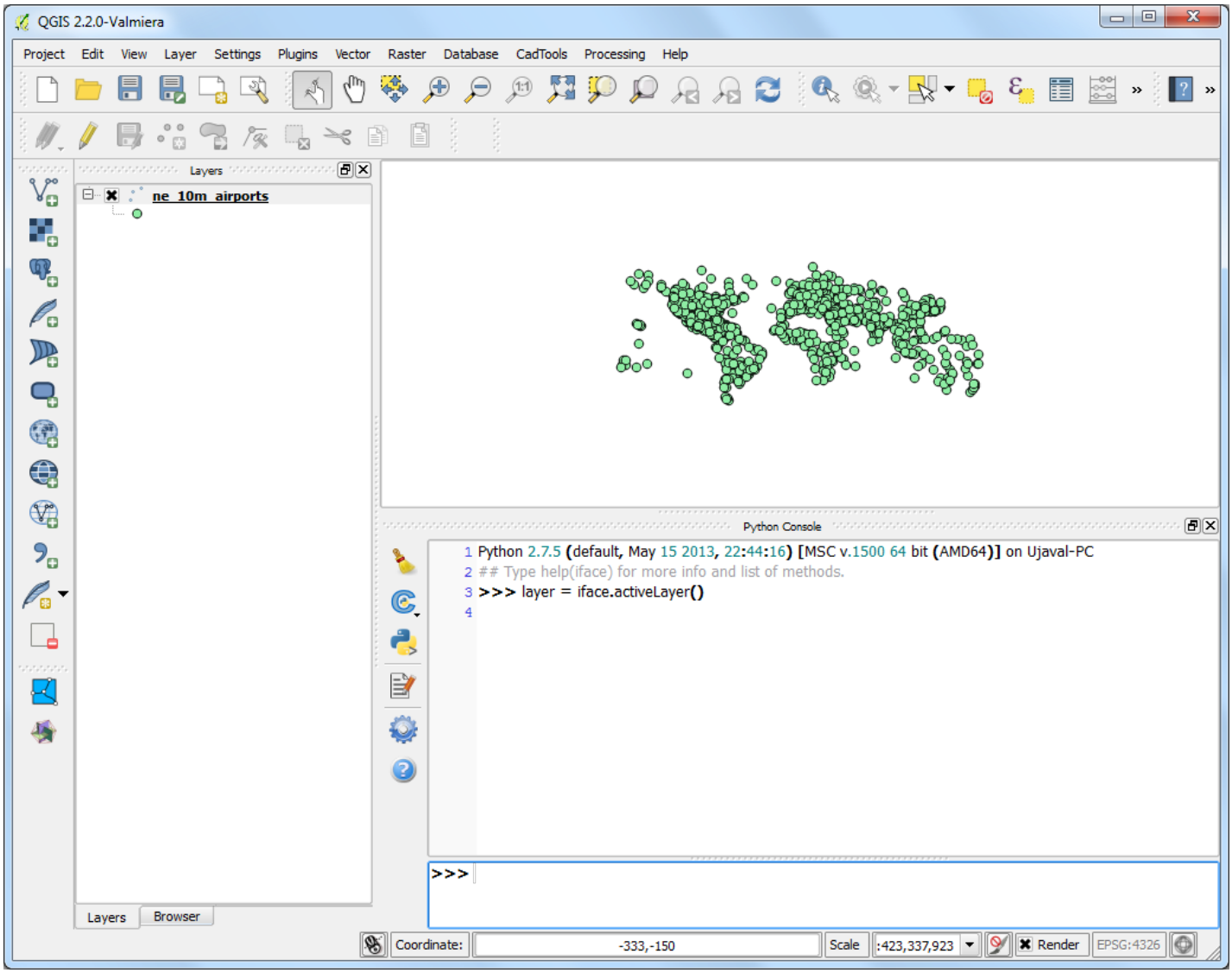


4. QGIS oferă o consolă încorporată, în care puteți introduce comenzi Python și să obțineți rezultatul. Această consolă este foarte bună pentru a învăța scriptarea și, de asemenea, pentru a procesa rapid datele. Deschideți Python Console din Plugins ‣ Python Console.



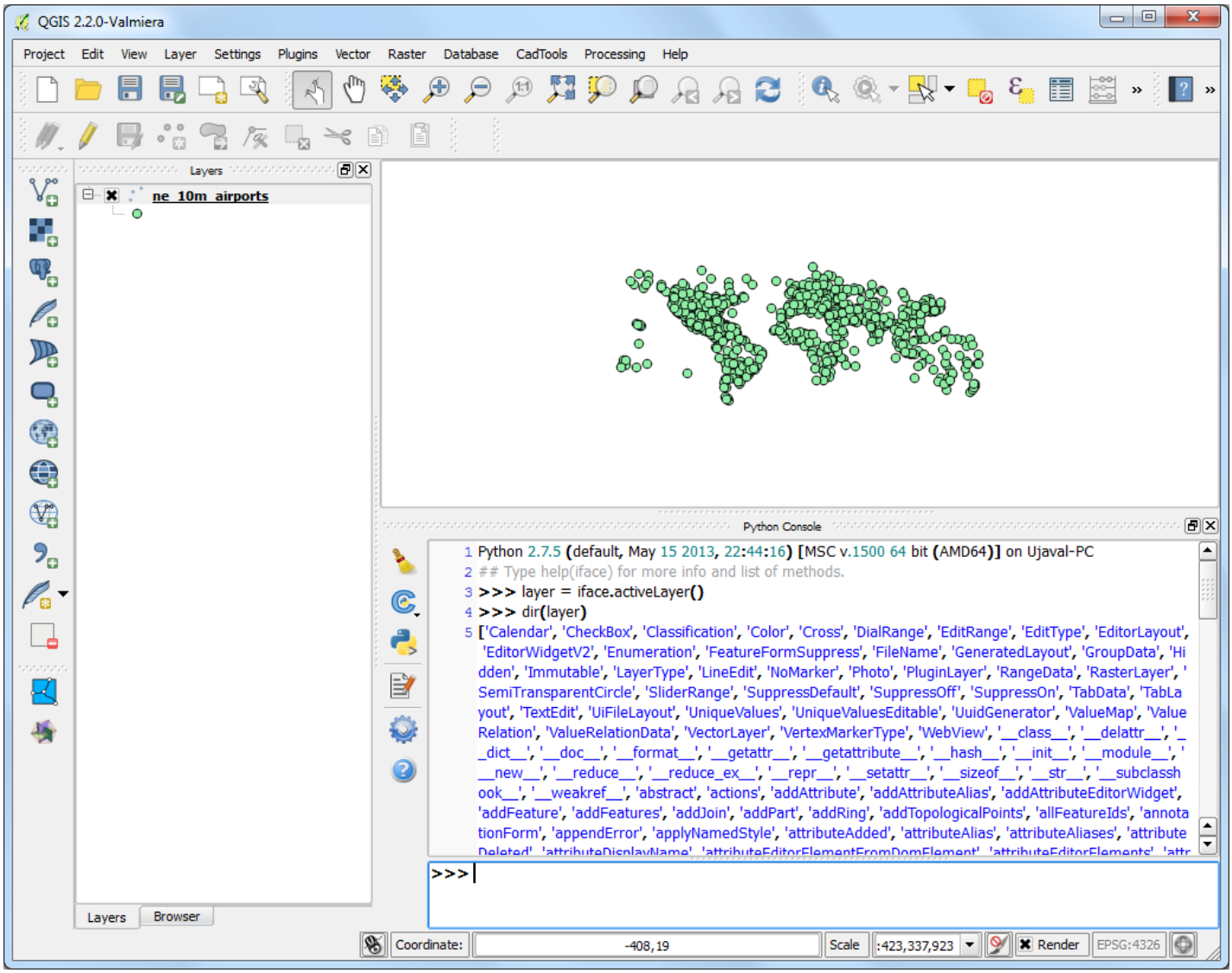
5. Veți vedea un nou panou, deschis în partea de jos a suportului de hărți al QGIS. Veți vedea un prompt cum ar fi **>>>**, în care aveți posibilitatea să tastați comenzi. Interacțiunea cu mediul QGIS, se face folosind variabila ***iface***. Pentru a accesa stratul din QGIS, activ în mod curent, aveți posibilitatea să tastați următoarele, apoi să apăsați Enter. Această comandă obține referința către stratul încărcat și o stochează în variabila ***layer***.

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



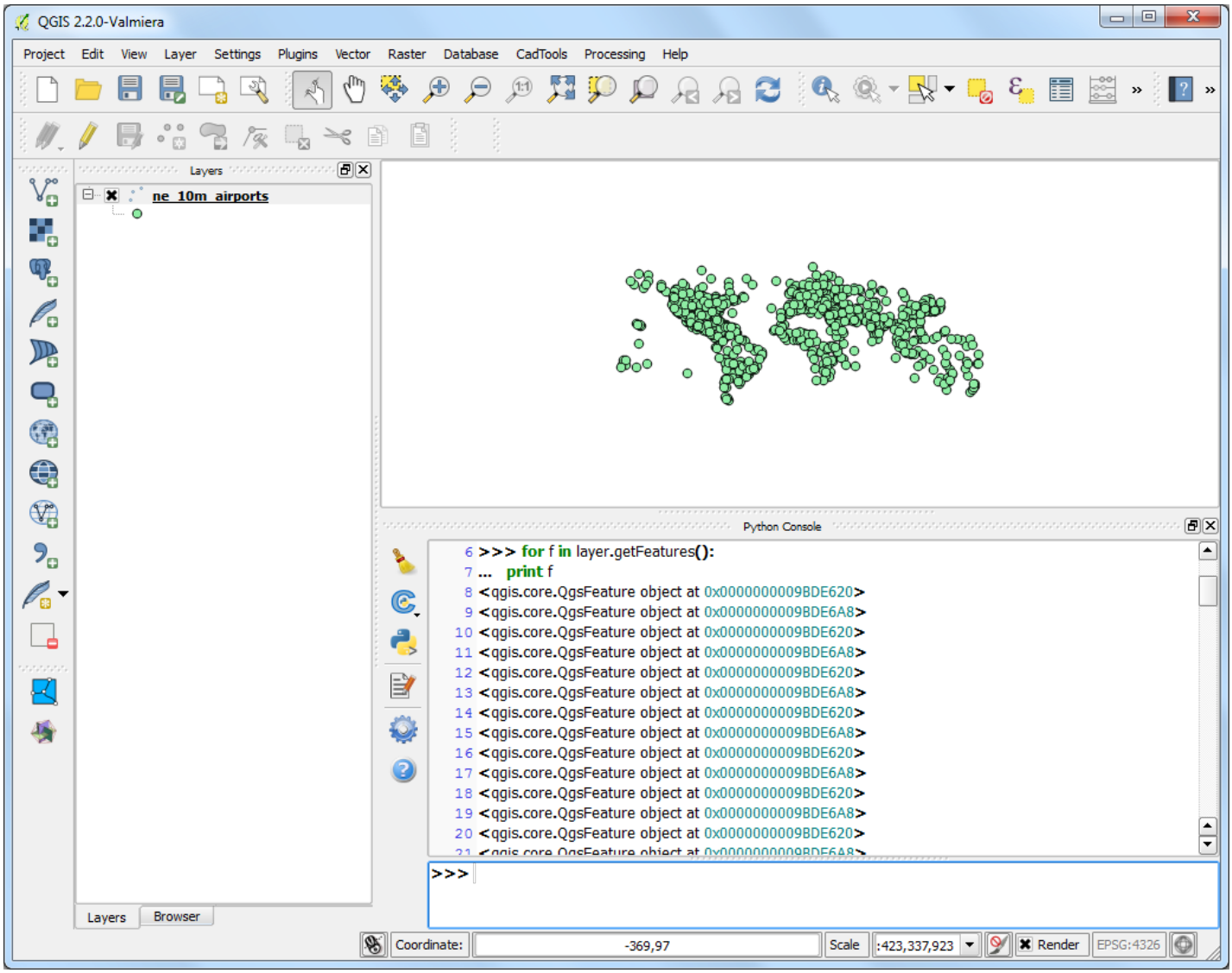
6. Există în Python o funcție utilă, numită ***dir()***, care afișează toate metodele disponibile pentru orice obiect. Acest lucru este folositor atunci când nu știți precis care funcții sunt disponibile pentru un anumit obiect. Executați următoarea comandă pentru a vedea ce operații putem executa cu variabila ***layer***.

```
dir(layer)
```



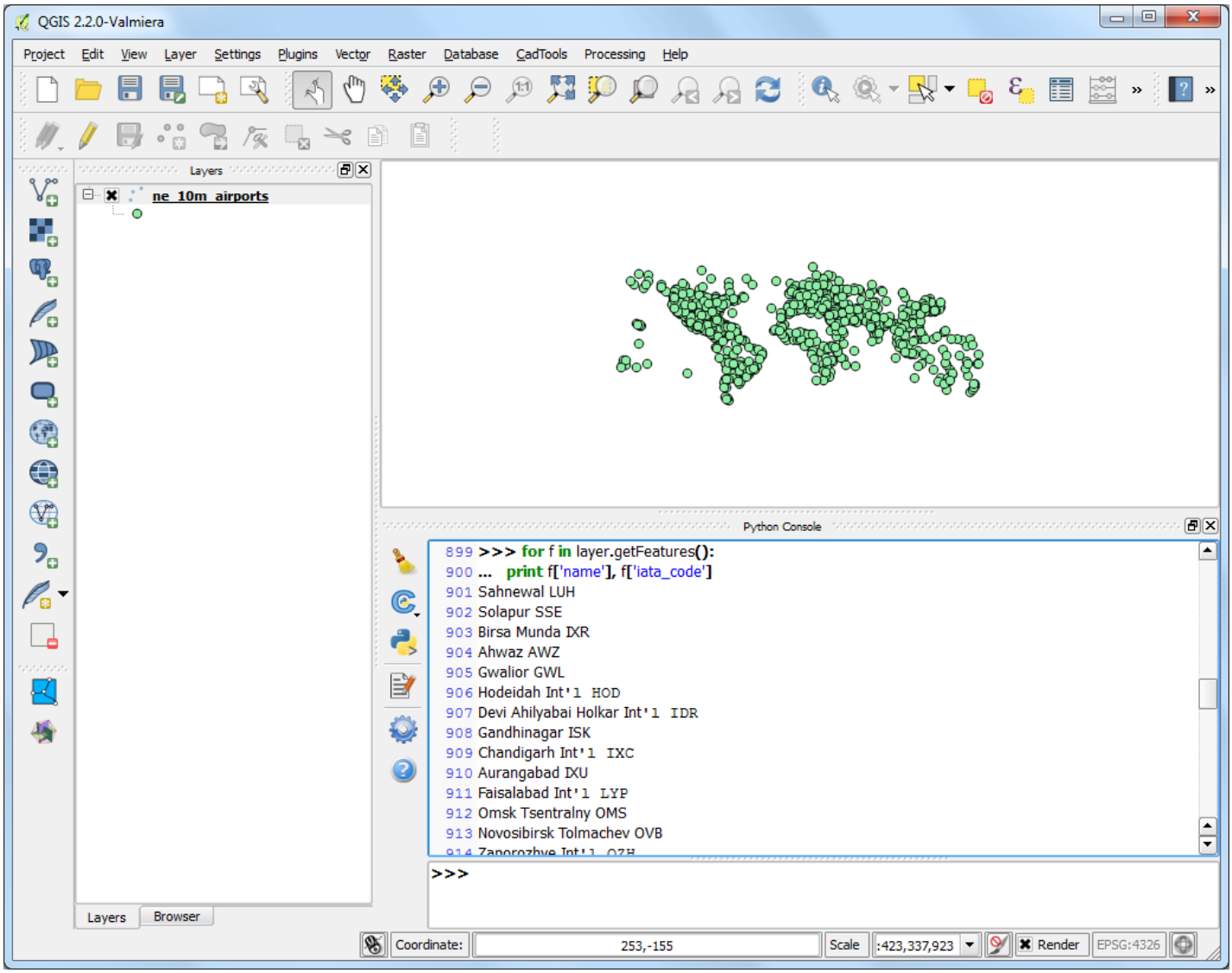
7. Veți vedea o listă lungă de funcții disponibile. Pentru moment, vom folosi o funcție numită ***getFeatures()***, care va obține referința către toate entitățile unui strat. În cazul nostru, fiecare entitate va fi un punct, reprezentând un aeroport. Puteți tasta următoarea comandă, pentru a parcurge fiecare dintre entitățile stratului curent. Asigurați-vă că ați adăugat 2 spații, înainte de a introduce a doua linie.

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



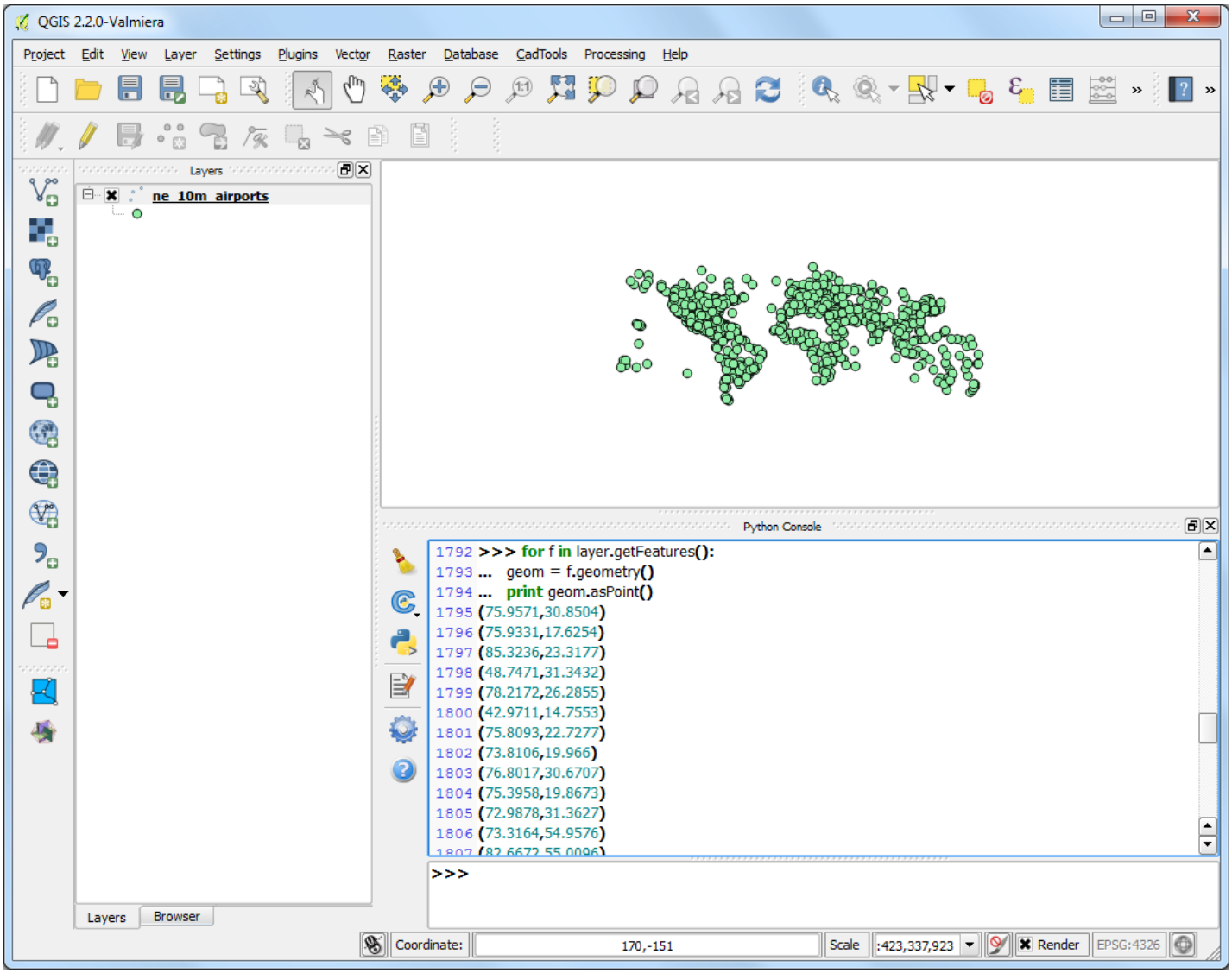
8. După cum veți vedea în rezultat, fiecare linie conține o referire la o entitate din cadrul stratului. Referința către entitate este stocată în variabila ***f***`. Putem folosi variabila ***f***` pentru a accesa atributele fiecărei entități. Introduceți următoarele, pentru a obține ***name*** și ***iata_code*** pentru fiecare entitate de tip aeroport.

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



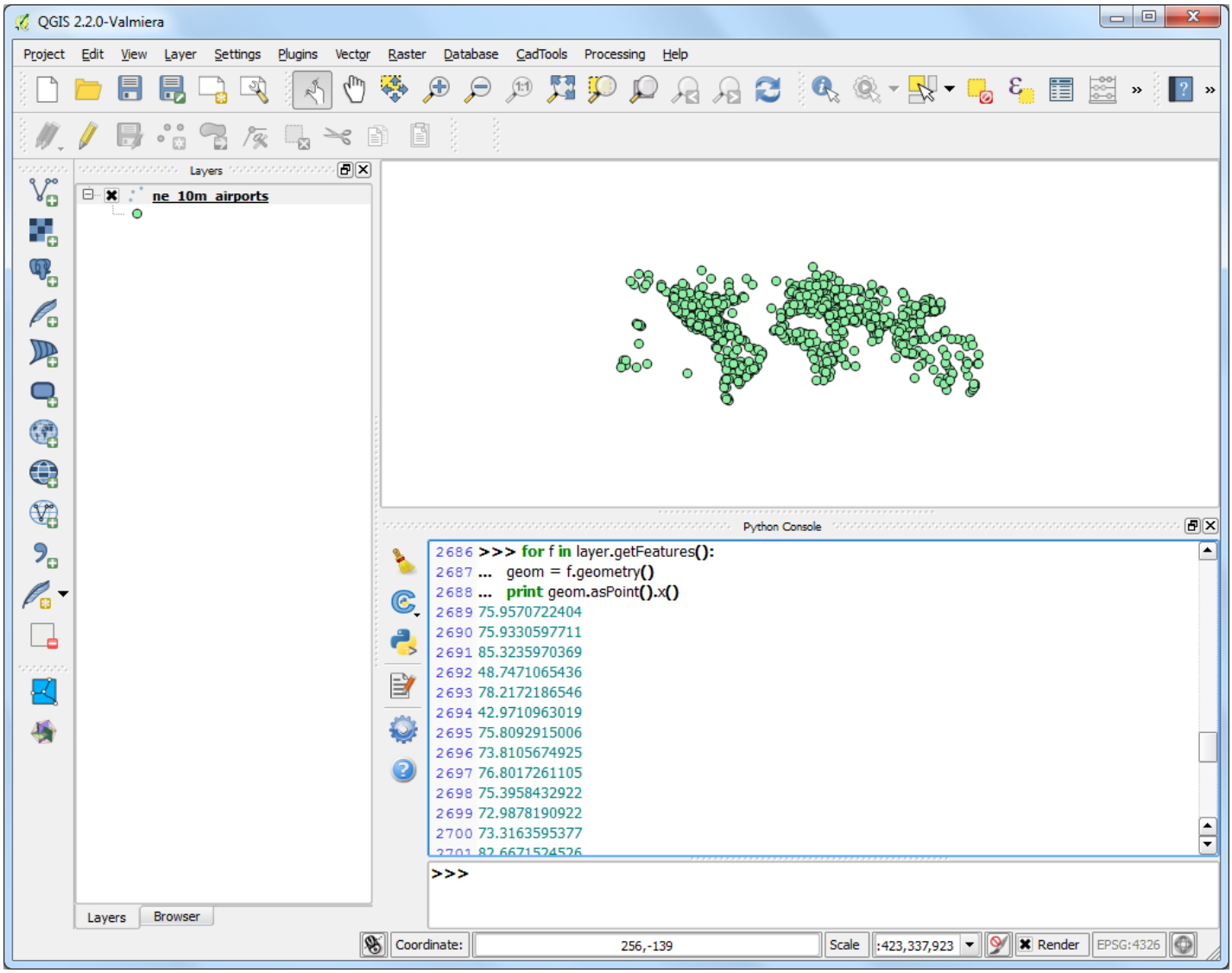
9. Deci, acum știți cum să accesați programatic atributul fiecărei entități dintr-un strat. În continuare, să vedem cum putem accesa coordonatele entităților. Coordonatele unei entități vectoriale pot fi accesate prin apelarea funcției ***geometry()***. Această funcție returnează un obiect de tip geometrie, pe care îl putem stoca în variabila ***geom***. Puteți rula funcția ***asPoint()*** pe obiectul respectiv, pentru a obține coordonatele x și y ale punctului. În cazul în care entitatea este o linie sau un poligon, puteți folosi funcțiile ***asPolyline()*** sau ***asPolygon()***. Introduceți următorul cod în linia de comandă și apăsați Enter, pentru a vedea coordonatele x și y ale fiecărei entități.

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



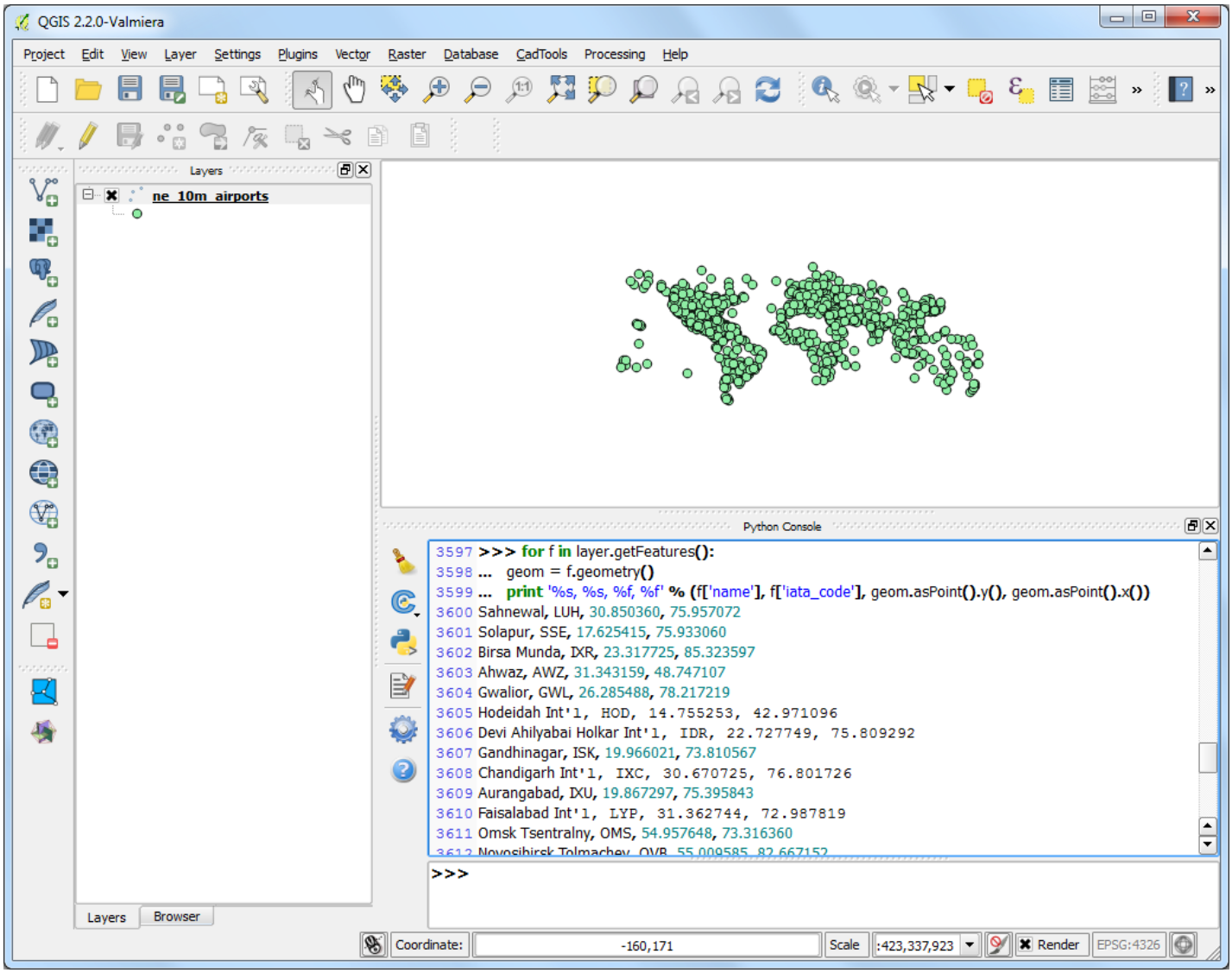
10. Cum procedăm dacă vrem să obținem numai coordonata ***x*** a entității? Vom apela funcția ***x()*** pentru obiectul punct, și îi vom obține coordonata x.

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



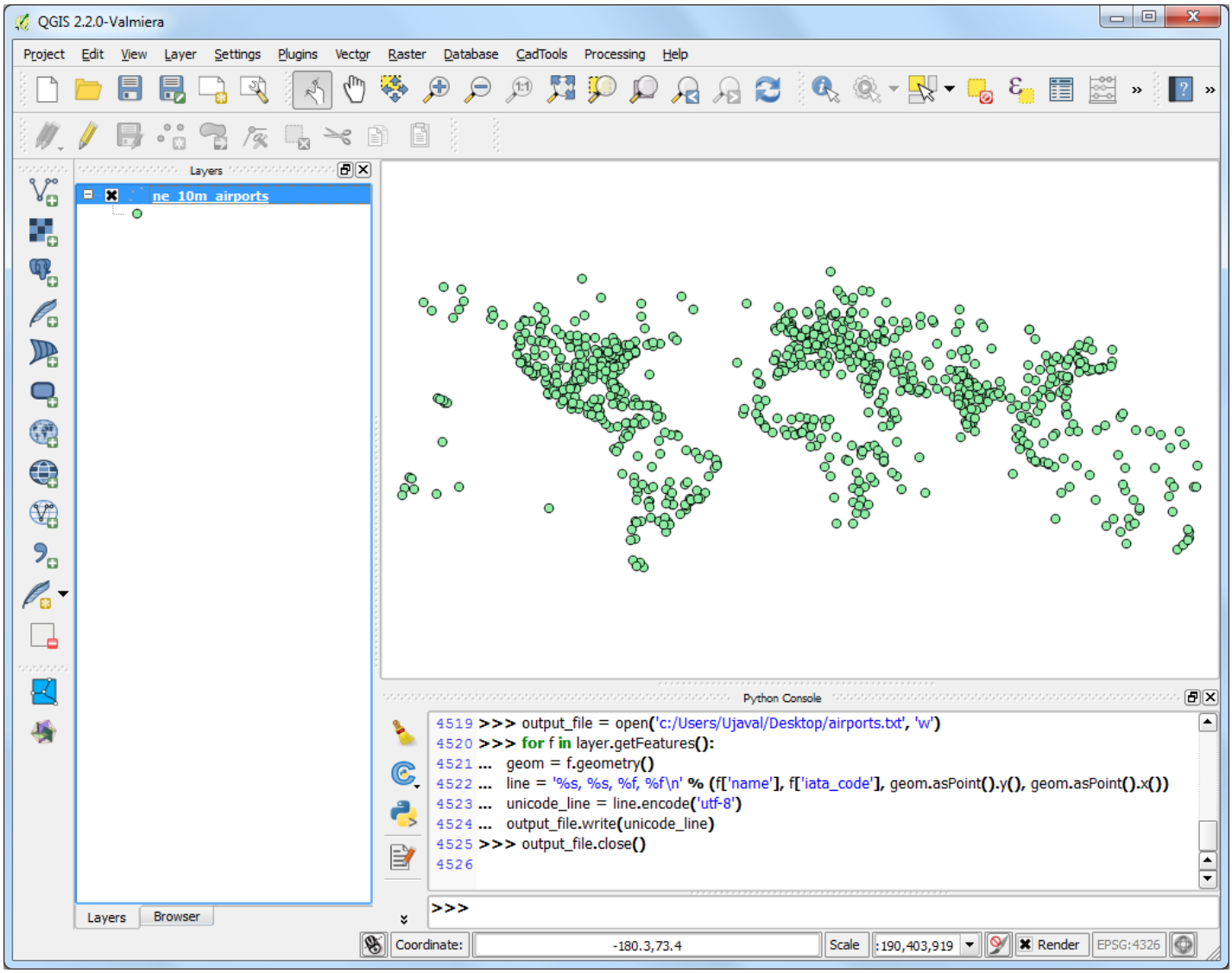
11. Acum avem toate piesele, pe care le putem îmbina pentru a genera rezultatul dorit. Introduceți codul de mai jos pentru a obține numele, iata_code, latitudinea și longitudinea fiecăreia dintre entitățile aeroport. Notațiile ***%s*** and ***%f*** sunt moduri de a formata șirurile și variabilele numerice.

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

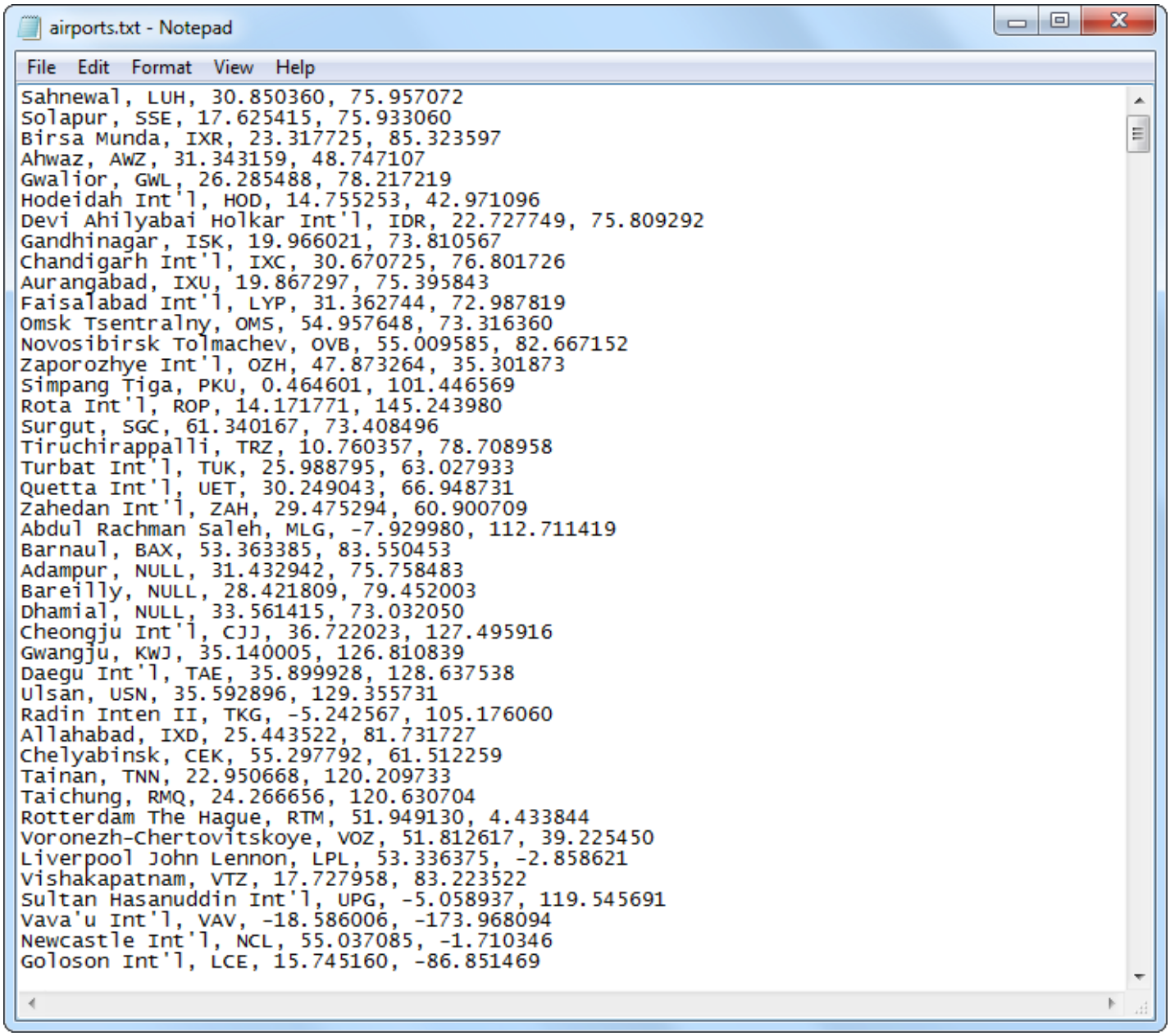


12. Puteți vedea rezultatul afișat în consolă. Un mod mai util de a stoca rezultatul ar fi un fișier. Puteți tasta codul de mai jos, pentru a crea un fișier în care veți înregistra rezultatul. Înlocuiți calea fișierului cu cea de pe sistemul propriu. Rețineți că vom adăuga ***\n*** în încheierea formatării liniei noastre. În acest mod, se va trece la o linie nouă, după adăugarea datelor pentru o entitate. De asemenea, trebuie să rețineți linia ***unicode_line = line.encode('utf-8')***. Deoarece stratul nostru conține unele entități cu caractere Unicode, nu putem face o simplă scriere într-un fișier text. Vom codifica textul folosind codificarea UTF-8 și apoi îl vom scrie în fișierul text.

```
output_file = open('c:/Users/Ujaval/Desktop/airports.txt', 'w')
for f in layer.getFeatures():
  geom = f.geometry()
  line = '%s, %s, %f, %f\n' % (f['name'], f['iata_code'],
          geom.asPoint().y(), geom.asPoint().x())
  unicode_line = line.encode('utf-8')
  output_file.write(unicode_line)
output_file.close()
```



13. Puteți merge la locația fișierului de ieșire specificat, pentru a deschide fișierul text. Veți vedea datele din fișierul shape de aeroporturi, pe care le-am extras folosind scriptarea Python.


airports.txt - Notepad
File Edit Format View Help
Sahnewal, LUH, 30.850360, 75.957072
Solapur, SSE, 17.625415, 75.933060
Birsa Munda, 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