

```

#coding: utf-8

s1 = 'SCENARIO 1: CONFIGURATION'

s2 = 'SCENARIO 2: QUERY'

s3 = 'SCENARIO 3: JSON'

s4 = 'SCENARIO 4: GEODATA: POINT IN POLYGON'

s5 = 'SCENARIO 5: DATA VISUALIZATION'

# -----BASICS-----

s1

help('modules')

import cx_Oracle

con = cx_Oracle.connect('oe/oe@localhost/orcl')

print con.version

s2

cur = con.cursor()

cur.execute('select * from customers where rownum <= 6')

for row in cur: print row

cur.fetchall()

person = '188'

cur.execute('select * from customers where customer_id='+person)

person = {'customer_id':188}

cur.execute('select * from customers where customer_id=:customer_id',person)

cur.fetchall()

# -----JSON-----

s3

cur.execute('select * from json where rownum<5')

for row in cur: print row

cur.execute('select json_value(doc, \''$.rating\') from json')

cur.execute('select j.doc.rating, count(*) from json j group by j.doc.rating')

# -----GEO-----

s4

res = cur.execute('select * from table( sdo_pointinpolygon(cursor(select
p.geometry.sdo_point.x lon, p.geometry.sdo_point.y lat, poi_id id, name from
world_sample.airports p), sdo_geometry (2003,8307,null,mdsys.sdo_elem_info_array(1, 1003,
3),mdsys.sdo_ordinate_array(6.0, 47.0, 15.0, 55.0)),0.05))')

for r in res: print r

s5

# build the basemap

from mpl_toolkits.basemap import Basemap

```

```

import matplotlib.pyplot as plt

import numpy as np

## specifying the projection to Mercator, central point of
## the projection, the image resolution
## and the projection region
map = Basemap(projection='merc', lat_0 = 51, lon_0 = 10, resolution = 'h', area_thresh =
0.1, llcrnrlon=6, llcrnrlat=47, urcrnrlon=15, urcrnrlat=55)

## drawing the map
map.drawcoastlines()
map.drawcountries()
map.fillcontinents(color = 'green')
map.drawmapboundary()

# Plot points
res = cur.execute('select * from table( sdo_pointinpolygon(cursor(select
p.geometry.sdo_point.x lon, p.geometry.sdo_point.y lat, poi_id id, name from
world_sample.airports p), sdo_geometry (2003,8307,null,mdsys.sdo_elem_info_array(1, 1003,
3),mdsys.sdo_ordinate_array(6.0, 47.0, 15.0, 55.0)),0.05))')

row = res.fetchone()

lats = []
lons = []

while row:
    lons.append(row[0])
    lats.append(row[1])
    row = res.fetchone()

x,y = map(lons,lats)
map.plot(x,y,'wo',markersize=5)
plt.show()

```