

```

# CHALLENGE 4: OpenStreetMap API solution.
import numpy as np
import matplotlib.pyplot as plt
from matplotlib import cm
from geopy import distance
import requests, json

def roq(opq): # Request Overpass Query(OverPass Query)
    overpass_url = "http://overpass-api.de/api/interpreter"
    response = requests.get(overpass_url, params={'data': opq})
    if response.status_code is not 200:
        print('Error! Code:',response.status_code)
    return response.json()['elements']

def plotRelGeom(e):
    for w in e[0]['members']:
        lat = []; lon = []
        if w['type']=='way':
            for n in w['geometry']:
                lat.append(n['lat']); lon.append(n['lon'])
            plt.plot(lon,lat,'k',linewidth=2)

def plotGeomSet(opq,e,clr='k',th=0.8):
    opqLen = 0
    for w in e:
        lat = []; lon = []
        if w['type']=='way':
            for n in w['geometry']:
                lat.append(n['lat']); lon.append(n['lon'])
            plt.plot(lon,lat,color=clr,linewidth=th)
        for i in range(len(lat)-1):
            p0 = (lat[i],lon[i]);p1 = (lat[i+1],lon[i+1]);
            opqLen = opqLen + distance.distance( p0,p1 ).km
    return opqLen

# Task 1: get a list of cities in the UK
opq = '[out:json];area["ISO3166-1"="GB"];(node[place="city"](area));out;'
e = roq(opq)

# Task 2: Plot Oxford City outline
opqOx = """"[out:json];area["ISO3166-1"="GB"];
rel(area)[name="Oxford"][type=boundary]; out geom;""""
plotRelGeom(roq(opqOx))

# Task 3 + Extension: plot road types and their respective lengths.
roadTypes = ['trunk','primary','secondary','tertiary','unclassified',
             'residential','cycleway']

opqLen = []
for rType in roadTypes:
    # by inspecting the Oxford area, we find the area code is 3600394037
    print(rType)
    opq = "[out:json];way[highway="+rType+"](area:3600394037); out geom;";
    opqLen.append( plotGeomSet(opq,roq(opq)) )
plt.axis('equal'); plt.grid(); plt.axis('off'); plt.show()

```