

ΟΝΟΜ/ΜΟ: ΚΩΝΣΤΑΤΟΠΟΥΛΟΥ ΕΥΑΓΓΕΛΙΑ
ΑΜ:1059560 3^ο ΕΤΟΣ
E-MAIL: st1059560@ceid.upatras.gr

ΚΩΔΙΚΑΣ ΜΕ ΣΧΟΛΙΑ:

```
import matplotlib
import requests
import sqlite3
import xlrd
import os
import csv
import heapq
import numpy as np
import matplotlib.pyplot as plt
from matplotlib import style
style.use('fivethirtyeight')
matplotlib.rc('xtick', labelsiz=6)
```

#Script που κατεβάζει τα excel από το statistics.gr

x = [0, 1, 2, 3, 4] #url από τα οποία θα ανακτήσουμε τα excel (χρησιμοποιώ μόνο αυτά του 4ου τριμήνου που περιλαμβάνει τα μέσα μεταφοράς)

resp = [0, 1, 2, 3, 4] #με τη βοήθεια του list αυτού παίρνω τα περιεχόμενα των excel

output = [0, 1, 2, 3, 4]

```
x[
    0] =
"https://www.statistics.gr/el/statistics?p_p_id=documents_WAR_publicationsportlet_INSTANCE_VBZO
ni0vs5VJ&p_p_lifecycle=2&p_p_state=normal&p_p_mode=view&p_p_cacheability=cacheLevelPage&p_p_col
_id=column-
2&p_p_col_count=4&p_p_col_pos=2&documents_WAR_publicationsportlet_INSTANCE_VBZOni0vs5VJ_javax.
faces.resource=document&documents_WAR_publicationsportlet_INSTANCE_VBZOni0vs5VJ_ln=downloadRes
ources&documents_WAR_publicationsportlet_INSTANCE_VBZOni0vs5VJ_documentID=113865&documents_WA
R_publicationsportlet_INSTANCE_VBZOni0vs5VJ_locale=el"
x[
    1] =
"https://www.statistics.gr/el/statistics?p_p_id=documents_WAR_publicationsportlet_INSTANCE_VBZO
ni0vs5VJ&p_p_lifecycle=2&p_p_state=normal&p_p_mode=view&p_p_cacheability=cacheLevelPage&p_p_col
_id=column-
2&p_p_col_count=4&p_p_col_pos=2&documents_WAR_publicationsportlet_INSTANCE_VBZOni0vs5VJ_javax.
faces.resource=document&documents_WAR_publicationsportlet_INSTANCE_VBZOni0vs5VJ_ln=downloadRes
ources&documents_WAR_publicationsportlet_INSTANCE_VBZOni0vs5VJ_documentID=113886&documents_WA
R_publicationsportlet_INSTANCE_VBZOni0vs5VJ_locale=el"
x[
    2] =
"https://www.statistics.gr/el/statistics?p_p_id=documents_WAR_publicationsportlet_INSTANCE_VBZO
ni0vs5VJ&p_p_lifecycle=2&p_p_state=normal&p_p_mode=view&p_p_cacheability=cacheLevelPage&p_p_col
_id=column-
2&p_p_col_count=4&p_p_col_pos=2&documents_WAR_publicationsportlet_INSTANCE_VBZOni0vs5VJ_javax.
faces.resource=document&documents_WAR_publicationsportlet_INSTANCE_VBZOni0vs5VJ_ln=downloadRes
ources&documents_WAR_publicationsportlet_INSTANCE_VBZOni0vs5VJ_documentID=113905&documents_WA
R_publicationsportlet_INSTANCE_VBZOni0vs5VJ_locale=el"
x[
    3] =
"https://www.statistics.gr/el/statistics?p_p_id=documents_WAR_publicationsportlet_INSTANCE_VBZO
ni0vs5VJ&p_p_lifecycle=2&p_p_state=normal&p_p_mode=view&p_p_cacheability=cacheLevelPage&p_p_col
_id=column-
2&p_p_col_count=4&p_p_col_pos=2&documents_WAR_publicationsportlet_INSTANCE_VBZOni0vs5VJ_javax.
faces.resource=document&documents_WAR_publicationsportlet_INSTANCE_VBZOni0vs5VJ_ln=downloadRes
ources&documents_WAR_publicationsportlet_INSTANCE_VBZOni0vs5VJ_documentID=113925&documents_WA
R_publicationsportlet_INSTANCE_VBZOni0vs5VJ_locale=el"
x[
    4] =
"https://www.statistics.gr/el/statistics?p_p_id=documents_WAR_publicationsportlet_INSTANCE_VBZO
```

```
ni0vs5VJ&p_p_lifecycle=2&p_p_state=normal&p_p_mode=view&p_p_cacheability=cacheLevelPage&p_p_col_id=column-2&p_p_col_count=4&p_p_col_pos=2&_documents_WAR_publicationsportlet_INSTANCE_VBZ0ni0vs5VJ_javax.faces.resource=document&_documents_WAR_publicationsportlet_INSTANCE_VBZ0ni0vs5VJ_ln=downloadResources&_documents_WAR_publicationsportlet_INSTANCE_VBZ0ni0vs5VJ_documentID=198755&_documents_WAR_publicationsportlet_INSTANCE_VBZ0ni0vs5VJ_locale=el"
```

```
for i in range(5):  
    resp[i] = requests.get(x[i])
```

#για κάθε output[i] δημιουργώ και ένα .xls στο αρχείο του project στα οποία γράφω (wb) τα περιεχόμενα των excels

```
output[0] = open('2011_all_transport.xls', 'wb')  
output[1] = open('2012_all_transport.xls', 'wb')  
output[2] = open('2013_all_transport.xls', 'wb')  
output[3] = open('2014_all_transport.xls', 'wb')  
output[4] = open('2015_all_transport.xls', 'wb')
```

```
for i in range(5):  
    output[i].write(resp[i].content)  
    output[i].close()
```

#ανοίγω πρώτο workbook και πάω στο sheet του Δεκεμβρίου που περιέχει πληροφορίες για όλους τους μήνες του 2011

```
workbook = xlrd.open_workbook('2011_all_transport.xls')  
worksheet = workbook.sheet_by_index(11)
```

```
all_2011 = worksheet.cell(134, 6).value      #παίρνει την τιμή του κελιού που δείχνει το  
συνολικό πλήθος τουριστών για το 2011  
plane_2011 = worksheet.cell(134, 2).value    #παίρνει την τιμή του κελιού που δείχνει τους  
συνολικούς τουρίστες που ταξίδεψαν με αεροπλάνο το 2011  
train_2011 = worksheet.cell(134, 3).value    #παίρνει την τιμή του κελιού που δείχνει τους  
συνολικούς τουρίστες που ταξίδεψαν με τρένο το 2011  
boat_2011 = worksheet.cell(134, 4).value     #παίρνει την τιμή του κελιού που δείχνει τους  
συνολικούς τουρίστες που ταξίδεψαν με πλοίο το 2011  
car_2011 = worksheet.cell(134, 5).value      #παίρνει την τιμή του κελιού που δείχνει τους  
συνολικούς τουρίστες που ταξίδεψαν με αυτοκίνητο το 2011
```

```
worksheet = workbook.sheet_by_index(0)  
jan_tourists_2011 = worksheet.cell(65, 6).value #παίρνει την τιμή του κελιού που δείχνει  
τους συνολικούς τουρίστες που ταξίδεψαν τον Ιανουάριο του 2011
```

```
worksheet = workbook.sheet_by_index(1)  
feb_tourists_2011 = worksheet.cell(65, 6).value #παίρνει την τιμή του κελιού που δείχνει  
τους συνολικούς τουρίστες που ταξίδεψαν τον Φλεβάρη του 2011
```

```
worksheet = workbook.sheet_by_index(2)  
mar_tourists_2011 = worksheet.cell(65, 6).value #παίρνει την τιμή του κελιού που δείχνει  
τους συνολικούς τουρίστες που ταξίδεψαν τον Μάρτη του 2011
```

```
_1st_trimester_2011 = jan_tourists_2011 + feb_tourists_2011 + mar_tourists_2011 #υπολογίζω το  
πλήθος των τουριστών που ταξίδεψαν στην Ελλάδα το πρώτο τρίμηνο του 2011
```

```
worksheet = workbook.sheet_by_index(3)  
apr_tourists_2011 = worksheet.cell(65, 6).value #παίρνει την τιμή του κελιού που δείχνει  
τους συνολικούς τουρίστες που ταξίδεψαν τον Απρίλη του 2011
```

```
worksheet = workbook.sheet_by_index(4)  
may_tourists_2011 = worksheet.cell(65, 6).value #παίρνει την τιμή του κελιού που δείχνει  
τους συνολικούς τουρίστες που ταξίδεψαν τον Μάη του 2011
```

```
worksheet = workbook.sheet_by_index(5)  
jun_tourists_2011 = worksheet.cell(65, 6).value #παίρνει την τιμή του κελιού που δείχνει τους  
συνολικούς τουρίστες που ταξίδεψαν τον Ιούνιο του 2011
```

```
_2nd_trimester_2011 = apr_tourists_2011 + may_tourists_2011 + jun_tourists_2011 #υπολογίζω το  
πλήθος των τουριστών που ταξίδεψαν στην Ελλάδα το δεύτερο τρίμηνο του 2011
```

```
worksheet = workbook.sheet_by_index(6)  
jul_tourists_2011 = worksheet.cell(65, 6).value #παίρνει την τιμή του κελιού που δείχνει  
τους συνολικούς τουρίστες που ταξίδεψαν τον Ιούλη του 2011
```

```
worksheet = workbook.sheet_by_index(7)  
aug_tourists_2011 = worksheet.cell(65, 6).value #παίρνει την τιμή του κελιού που δείχνει  
τους συνολικούς τουρίστες που ταξίδεψαν τον Αύγουστο του 2011
```

```
worksheet = workbook.sheet_by_index(8)  
sep_tourists_2011 = worksheet.cell(65, 6).value #παίρνει την τιμή του κελιού που δείχνει  
τους συνολικούς τουρίστες που ταξίδεψαν τον Σεπτέμβρη του 2011
```

```
_3rd_trimester_2011 = jul_tourists_2011 + aug_tourists_2011 + sep_tourists_2011 #υπολογίζω το  
πλήθος των τουριστών που ταξίδεψαν στην Ελλάδα το τρίτο τρίμηνο του 2011
```

```
worksheet = workbook.sheet_by_index(9)  
oct_tourists_2011 = worksheet.cell(65, 6).value #παίρνει την τιμή του κελιού που δείχνει  
τους συνολικούς τουρίστες που ταξίδεψαν τον Οκτώβρη του 2011
```

```
worksheet = workbook.sheet_by_index(10)  
nov_tourists_2011 = worksheet.cell(65, 6).value #παίρνει την τιμή του κελιού που δείχνει  
τους συνολικούς τουρίστες που ταξίδεψαν τον Νοέμβρη του 2011
```

```
worksheet = workbook.sheet_by_index(11)  
dec_tourists_2011 = worksheet.cell(65, 6).value #παίρνει την τιμή του κελιού που δείχνει  
τους συνολικούς τουρίστες που ταξίδεψαν τον Δεκέμβρη του 2011
```

```
_4th_trimester_2011 = oct_tourists_2011 + nov_tourists_2011 + dec_tourists_2011 #υπολογίζω  
το πλήθος των τουριστών που ταξίδεψαν στην Ελλάδα το τέταρτο τρίμηνο του 2011
```

```
col = 6  
worksheet = workbook.sheet_by_index(11)
```

```
#όταν υπολογίζω το μέγιστο του column του excel που περιέχει το πλήθος των τουριστών ανά χώρα,  
δε θέλω να συμπεριλάβω τα κελιά με τα συνολικά πλήθη ανά ήπειρο:
```

```
Europe = [[worksheet.cell_value(r, col)] for r in range(76,108)] #τουρίστες από  
χώρες της Ευρώπης για το 2011
```

```
Asia = [[worksheet.cell_value(r, col)] for r in range(110,118)] #τουρίστες από  
χώρες της Ασίας για το 2011
```

```
Africa = [[worksheet.cell_value(r, col)] for r in range(120,122)] #τουρίστες από  
χώρες της Αφρικής για το 2011
```

```
America = [[worksheet.cell_value(r, col)] for r in range(124,129)] #τουρίστες από  
χώρες της Αμερικής για το 2011
```

```
Australia = [[worksheet.cell_value(r, col) for r in range(131,132)]] #τουρίστες από  
την Αυστραλία για το 2011
```

```
all2011 = Europe + Asia + Africa + America + Australia  
all0f2011 = []
```

```
#ενοποιημένη λίστα με το πλήθος τουριστών από χώρες των παραπάνω ηπείρων (all2011 --> λίστα από  
παραπάνω λίστες
```

```
#all0f2011 --> μία λίστα με στοιχεία των προηγούμενων λιστών)
```

```
for i in all2011:  
    all0f2011 += i
```

```
sum_2011 = sum(all0f2011) #σύνολο πλήθος τουριστών για το 2011 (θα το  
χρησιμοποιήσω αργότερα για το γράφημα που δείχνει το μερίδιο των χωρών με τους περισσότερους  
τουρίστες)
```

```
#επιστρέφει τα 3 μεγαλύτερα πλήθη τουριστών
```

```
top3of2011 = heapq.nlargest(3, all0f2011)
```

```
third_2011 = top3of2011[2]
```

```
second_2011 = top3of2011[1]
first_2011 = top3of2011[0]
maxoverall2011 = max(allOf2011)
```

```
#με αυτά τα 3 for σαρώνω τις γραμμές και στήλες του φύλλου του excel για να βρω σε ποια χώρα
αντιστοιχεί το μέγιστο πλήθος τουριστών
for row in range(worksheet.nrows):
    for col in range(worksheet.ncols):
        if worksheet.cell_value(row, col) == maxoverall2011:
            maxcountry2011 = worksheet.cell(row, 1).value #θέλω να μου επιστρέψει το
όνομα της χώρας που βρίσκεται στο column 1 και στο row που προέκυψε από το iteration
```

```
#αντίστοιχα για τη χώρα με το δεύτερο και τρίτο μεγαλύτερο πλήθος τουριστών
```

```
for row in range(worksheet.nrows):
    for col in range(worksheet.ncols):
        if worksheet.cell_value(row, col) == second_2011:
            secondcountry2011 = worksheet.cell(row, 1).value
```

```
for row in range(worksheet.nrows):
    for col in range(worksheet.ncols):
        if worksheet.cell_value(row, col) == third_2011:
            thirdcountry2011 = worksheet.cell(row, 1).value
```

```
#Επαναλαμβάνω ακριβώς την ίδια διαδικασία για τα έτη 2012-2015. Οι μεταβλητές που χρησιμοποιώ
έχουν τα ίδια ονόματα με αυτές του 2011 με μόνη διαφορά το έτος στο τέλος
```

```
workbook = xlrd.open_workbook('2012_all_transport.xls')
worksheet = workbook.sheet_by_index(11)
all_2012 = worksheet.cell(136, 6).value
plane_2012 = worksheet.cell(136, 2).value
train_2012 = worksheet.cell(136, 3).value
boat_2012 = worksheet.cell(136, 4).value
car_2012 = worksheet.cell(136, 5).value
```

```
worksheet = workbook.sheet_by_index(0)
jan_tourists_2012 = worksheet.cell(65, 6).value
```

```
worksheet = workbook.sheet_by_index(1)
feb_tourists_2012 = worksheet.cell(65, 6).value
```

```
worksheet = workbook.sheet_by_index(2)
mar_tourists_2012 = worksheet.cell(65, 6).value
```

```
_1st_trimester_2012 = jan_tourists_2012 + feb_tourists_2012 + mar_tourists_2012
```

```
worksheet = workbook.sheet_by_index(3)
apr_tourists_2012 = worksheet.cell(65, 6).value
```

```
worksheet = workbook.sheet_by_index(4)
may_tourists_2012 = worksheet.cell(65, 6).value
```

```
worksheet = workbook.sheet_by_index(5)
jun_tourists_2012 = worksheet.cell(65, 6).value
```

```
_2nd_trimester_2012 = apr_tourists_2012 + may_tourists_2012 + jun_tourists_2012
```

```
worksheet = workbook.sheet_by_index(6)
jul_tourists_2012 = worksheet.cell(65, 6).value
```

```
worksheet = workbook.sheet_by_index(7)
aug_tourists_2012 = worksheet.cell(65, 6).value
```

```
worksheet = workbook.sheet_by_index(8)
sep_tourists_2012 = worksheet.cell(65, 6).value
```

```
_3rd_trimester_2012 = jul_tourists_2012 + aug_tourists_2012 + sep_tourists_2012
```

```
worksheet = workbook.sheet_by_index(9)
oct_tourists_2012 = worksheet.cell(65, 6).value
```

```
worksheet = workbook.sheet_by_index(10)
nov_tourists_2012 = worksheet.cell(65, 6).value
```

```
worksheet = workbook.sheet_by_index(11)
dec_tourists_2012 = worksheet.cell(65, 6).value
```

```
_4th_trimester_2012 = oct_tourists_2012 + nov_tourists_2012 + dec_tourists_2012
```

```
col = 6
worksheet = workbook.sheet_by_index(11)
Europe = [[worksheet.cell_value(r, col)] for r in range(78,108)]
Asia = [[worksheet.cell_value(r, col)] for r in range(112,120)]
Africa = [[worksheet.cell_value(r, col)] for r in range(122,124)]
America = [[worksheet.cell_value(r, col)] for r in range(126,131)]
Australia = [[worksheet.cell_value(r, col)] for r in range(133,135)]
all2012 = Europe + Asia + Africa + America + Australia
all0f2012 = []
```

```
for i in all2012:
    all0f2012 += i
```

```
sum_2012 = sum(all0f2012)
top3of2012 = heapq.nlargest(3, all0f2012)
third_2012 = top3of2012[2]
second_2012 = top3of2012[1]
first_2012 = top3of2012[0]
maxoverall2012 = max(all0f2012)
```

```
for row in range(worksheet.nrows):
    for col in range(worksheet.ncols):
        if worksheet.cell_value(row, col) == maxoverall2012:
            maxcountry2012 = worksheet.cell(row, 1).value
```

```
for row in range(worksheet.nrows):
    for col in range(worksheet.ncols):
        if worksheet.cell_value(row, col) == second_2012:
            secondcountry2012 = worksheet.cell(row, 1).value
```

```
for row in range(worksheet.nrows):
    for col in range(worksheet.ncols):
        if worksheet.cell_value(row, col) == third_2012:
            thirdcountry2012 = worksheet.cell(row, 1).value
```

```
workbook = xlrd.open_workbook('2013_all_transport.xls')
worksheet = workbook.sheet_by_index(11)
all_2013 = worksheet.cell(136, 6).value
plane_2013 = worksheet.cell(136, 2).value
train_2013 = worksheet.cell(136, 3).value
boat_2013 = worksheet.cell(136, 4).value
car_2013 = worksheet.cell(136, 5).value
```

```
worksheet = workbook.sheet_by_index(0)
jan_tourists_2013 = worksheet.cell(64, 6).value
```

```
worksheet = workbook.sheet_by_index(1)
feb_tourists_2013 = worksheet.cell(64, 6).value
```

```
worksheet = workbook.sheet_by_index(2)
mar_tourists_2013 = worksheet.cell(64, 6).value
```

```
_1st_trimester_2013 = jan_tourists_2013 + feb_tourists_2013 + mar_tourists_2013
```

```
worksheet = workbook.sheet_by_index(3)
apr_tourists_2013 = worksheet.cell(64, 6).value
```

```
worksheet = workbook.sheet_by_index(4)
may_tourists_2013 = worksheet.cell(64, 6).value
```

```
worksheet = workbook.sheet_by_index(5)
jun_tourists_2013 = worksheet.cell(64, 6).value
```

```
_2nd_trimester_2013 = apr_tourists_2013 + may_tourists_2013 + jun_tourists_2013
```

```
worksheet = workbook.sheet_by_index(6)
jul_tourists_2013 = worksheet.cell(65, 6).value
```

```
worksheet = workbook.sheet_by_index(7)
aug_tourists_2013 = worksheet.cell(65, 6).value
```

```
worksheet = workbook.sheet_by_index(8)
sep_tourists_2013 = worksheet.cell(65, 6).value
```

```
_3rd_trimester_2013 = jul_tourists_2013 + aug_tourists_2013 + sep_tourists_2013
```

```
worksheet = workbook.sheet_by_index(9)
oct_tourists_2013 = worksheet.cell(65, 6).value
```

```
worksheet = workbook.sheet_by_index(10)
nov_tourists_2013 = worksheet.cell(65, 6).value
```

```
worksheet = workbook.sheet_by_index(11)
dec_tourists_2013 = worksheet.cell(65, 6).value
```

```
_4th_trimester_2013 = oct_tourists_2013 + nov_tourists_2013 + dec_tourists_2013
```

```
col = 6
worksheet = workbook.sheet_by_index(11)
Europe = [[worksheet.cell_value(r, col)] for r in range(78,111)]
Asia = [[worksheet.cell_value(r, col)] for r in range(113,121)]
Africa = [[worksheet.cell_value(r, col)] for r in range(123,125)]
America = [[worksheet.cell_value(r, col)] for r in range(127,132)]
Australia = [[worksheet.cell_value(r, col)] for r in range(134,135)]
all2013 = Europe + Asia + Africa + America + Australia
all0f2013 = []
```

```
for i in all2013:
    all0f2013 += i
```

```
sum_2013 = sum(all0f2013)
top3of2013 = heapq.nlargest(3, all0f2013)
third_2013 = top3of2013[2]
second_2013 = top3of2013[1]
first_2013 = top3of2013[0]
maxoverall2013 = max(all0f2013)
```

```
for row in range(worksheet.nrows):
    for col in range(worksheet.ncols):
        if worksheet.cell_value(row, col) == maxoverall2013:
            maxcountry2013 = worksheet.cell(row, 1).value
```

```
for row in range(worksheet.nrows):
    for col in range(worksheet.ncols):
        if worksheet.cell_value(row, col) == second_2013:
            secondcountry2013 = worksheet.cell(row, 1).value
```



```

for row in range(worksheet.nrows):
    for col in range(worksheet.ncols):
        if worksheet.cell_value(row, col) == third_2013:
            thirdcountry2013 = worksheet.cell(row, 1).value

workbook = xlrd.open_workbook('2014_all_transport.xls')
worksheet = workbook.sheet_by_index(11)
all_2014 = worksheet.cell(136, 6).value
plane_2014 = worksheet.cell(136, 2).value
train_2014 = worksheet.cell(136, 3).value
boat_2014 = worksheet.cell(136, 4).value
car_2014 = worksheet.cell(136, 5).value

worksheet = workbook.sheet_by_index(0)
jan_tourists_2014 = worksheet.cell(65, 6).value

worksheet = workbook.sheet_by_index(1)
feb_tourists_2014 = worksheet.cell(65, 6).value

worksheet = workbook.sheet_by_index(2)
mar_tourists_2014 = worksheet.cell(65, 6).value

_1st_trimester_2014 = jan_tourists_2014 + feb_tourists_2014 + mar_tourists_2014

worksheet = workbook.sheet_by_index(3)
apr_tourists_2014 = worksheet.cell(65, 6).value

worksheet = workbook.sheet_by_index(4)
may_tourists_2014 = worksheet.cell(65, 6).value

worksheet = workbook.sheet_by_index(5)
jun_tourists_2014 = worksheet.cell(65, 6).value

_2nd_trimester_2014 = apr_tourists_2014 + may_tourists_2014 + jun_tourists_2014

worksheet = workbook.sheet_by_index(6)
jul_tourists_2014 = worksheet.cell(65, 6).value

worksheet = workbook.sheet_by_index(7)
aug_tourists_2014 = worksheet.cell(65, 6).value

worksheet = workbook.sheet_by_index(8)
sep_tourists_2014 = worksheet.cell(65, 6).value

_3rd_trimester_2014 = jul_tourists_2014 + aug_tourists_2014 + sep_tourists_2014

worksheet = workbook.sheet_by_index(9)
oct_tourists_2014 = worksheet.cell(65, 6).value

worksheet = workbook.sheet_by_index(10)
nov_tourists_2014 = worksheet.cell(65, 6).value

worksheet = workbook.sheet_by_index(11)
dec_tourists_2014 = worksheet.cell(65, 6).value

_4th_trimester_2014 = oct_tourists_2014 + nov_tourists_2014 + dec_tourists_2014

col = 6
worksheet = workbook.sheet_by_index(11)
Europe = [[worksheet.cell_value(r, col)] for r in range(78,111)]
Asia = [[worksheet.cell_value(r, col)] for r in range(113,121)]
Africa = [[worksheet.cell_value(r, col)] for r in range(123,125)]
America = [[worksheet.cell_value(r, col)] for r in range(127,132)]
Australia = [[worksheet.cell_value(r, col)] for r in range(134,135)]
all2014 = Europe + Asia + Africa + America + Australia

```

```

allOf2014 = []

for i in all2014:
    allOf2014 += i

sum_2014 = sum(allOf2014)
top3of2014 = heapq.nlargest(3, allOf2014)
third_2014 = top3of2014[2]
second_2014 = top3of2014[1]
first_2014 = top3of2014[0]
maxoverall2014 = max(allOf2014)

for row in range(worksheet.nrows):
    for col in range(worksheet.ncols):
        if worksheet.cell_value(row, col) == maxoverall2014:
            maxcountry2014 = worksheet.cell(row, 1).value

for row in range(worksheet.nrows):
    for col in range(worksheet.ncols):
        if worksheet.cell_value(row, col) == second_2014:
            secondcountry2014 = worksheet.cell(row, 1).value

for row in range(worksheet.nrows):
    for col in range(worksheet.ncols):
        if worksheet.cell_value(row, col) == third_2014:
            thirdcountry2014 = worksheet.cell(row, 1).value

workbook = xlrd.open_workbook('2015_all_transport.xls')
worksheet = workbook.sheet_by_index(11)
all_2015 = worksheet.cell(136, 6).value
plane_2015 = worksheet.cell(136, 2).value
train_2015 = worksheet.cell(136, 3).value
boat_2015 = worksheet.cell(136, 4).value
car_2015 = worksheet.cell(136, 5).value

worksheet = workbook.sheet_by_index(0)
jan_tourists_2015 = worksheet.cell(66, 6).value

worksheet = workbook.sheet_by_index(1)
feb_tourists_2015 = worksheet.cell(66, 6).value

worksheet = workbook.sheet_by_index(2)
mar_tourists_2015 = worksheet.cell(66, 6).value

_1st_trimester_2015 = jan_tourists_2015 + feb_tourists_2015 + mar_tourists_2015

worksheet = workbook.sheet_by_index(3)
apr_tourists_2015 = worksheet.cell(66, 6).value

worksheet = workbook.sheet_by_index(4)
may_tourists_2015 = worksheet.cell(66, 6).value

worksheet = workbook.sheet_by_index(5)
jun_tourists_2015 = worksheet.cell(66, 6).value

_2nd_trimester_2015 = apr_tourists_2015 + may_tourists_2015 + jun_tourists_2015

worksheet = workbook.sheet_by_index(6)
jul_tourists_2015 = worksheet.cell(66, 6).value

worksheet = workbook.sheet_by_index(7)
aug_tourists_2015 = worksheet.cell(66, 6).value

worksheet = workbook.sheet_by_index(8)
sep_tourists_2015 = worksheet.cell(66, 6).value

```



```

_3rd_trimester_2015 = jul_tourists_2015 + aug_tourists_2015 + sep_tourists_2015

worksheet = workbook.sheet_by_index(9)
oct_tourists_2015 = worksheet.cell(66, 6).value

worksheet = workbook.sheet_by_index(10)
nov_tourists_2015 = worksheet.cell(66, 6).value

worksheet = workbook.sheet_by_index(11)
dec_tourists_2015 = worksheet.cell(66, 6).value

_4th_trimester_2015 = oct_tourists_2015 + nov_tourists_2015 + dec_tourists_2015

col = 6
worksheet = workbook.sheet_by_index(11)
Europe = [[worksheet.cell_value(r, col)] for r in range(77,111)]
Asia = [[worksheet.cell_value(r, col)] for r in range(113,121)]
Africa = [[worksheet.cell_value(r, col)] for r in range(123,125)]
America = [[worksheet.cell_value(r, col)] for r in range(127,132)]
Australia = [[worksheet.cell_value(r, col)] for r in range(134,135)]
all2015 = Europe + Asia + Africa + America + Australia
all0f2015 = []

for i in all2015:
    all0f2015 += i

sum_2015 = sum(all0f2015)
top3of2015 = heapq.nlargest(3, all0f2015)
third_2015 = top3of2015[2]
second_2015 = top3of2015[1]
first_2015 = top3of2015[0]
maxoverall2015 = max(all0f2015)

for row in range(worksheet.nrows):
    for col in range(worksheet.ncols):
        if worksheet.cell_value(row, col) == maxoverall2015:
            maxcountry2015 = worksheet.cell(row, 1).value

for row in range(worksheet.nrows):
    for col in range(worksheet.ncols):
        if worksheet.cell_value(row, col) == second_2015:
            secondcountry2015 = worksheet.cell(row, 1).value

for row in range(worksheet.nrows):
    for col in range(worksheet.ncols):
        if worksheet.cell_value(row, col) == third_2015:
            thirdcountry2015 = worksheet.cell(row, 1).value

conn = sqlite3.connect(':memory:') #ανοίγω μια σύνδεση σε μια βάση δεδομένων
που ενυπάρχει στη RAM αντί για τον δίσκο μου
cursor = conn.cursor() #δημιουργώ αντικείμενο κέρσορα

#δημιουργώ πίνακα allTourists που θα περιέχει τους συνολικούς τουρίστες που ταξίδεψαν στην
Ελλάδα την τετραετία 2011-2015
cursor.execute("""CREATE TABLE allTourists (
    tourists_2011 INTEGER,
    tourists_2012 INTEGER,
    tourists_2013 INTEGER,
    tourists_2014 INTEGER,
    tourists_2015 INTEGER)""")

#δημιουργώ πίνακα transportation που θα περιέχει τους συνολικούς τουρίστες που ταξίδεψαν ανά
μεταφορικό μέσο στην Ελλάδα την τετραετία 2011-2015

```

```

cursor.execute("""CREATE TABLE transportation (
    plane_2011 INTEGER,
    train_2011 INTEGER,
    boat_2011 INTEGER,
    car_2011 INTEGER,
    plane_2012 INTEGER,
    train_2012 INTEGER,
    boat_2012 INTEGER,
    car_2012 INTEGER,
    plane_2013 INTEGER,
    train_2013 INTEGER,
    boat_2013 INTEGER,
    car_2013 INTEGER,
    plane_2014 INTEGER,
    train_2014 INTEGER,
    boat_2014 INTEGER,
    car_2014 INTEGER,
    plane_2015 INTEGER,
    train_2015 INTEGER,
    boat_2015 INTEGER,
    car_2015 INTEGER
)""")

```

#δημιουργώ πίνακα Tourists_per_Trimester που θα περιέχει τους συνολικούς τουρίστες που ταξίδεψαν στην Ελλάδα την τετραετία 2011-2015 ανά τρίμηνο

```

cursor.execute("""CREATE TABLE Tourists_per_trimester (
    tri_1_2011 INTEGER,
    tri_2_2011 INTEGER,
    tri_3_2011 INTEGER,
    tri_4_2011 INTEGER,
    tri_1_2012 INTEGER,
    tri_2_2012 INTEGER,
    tri_3_2012 INTEGER,
    tri_4_2012 INTEGER,
    tri_1_2013 INTEGER,
    tri_2_2013 INTEGER,
    tri_3_2013 INTEGER,
    tri_4_2013 INTEGER,
    tri_1_2014 INTEGER,
    tri_2_2014 INTEGER,
    tri_3_2014 INTEGER,
    tri_4_2014 INTEGER,
    tri_1_2015 INTEGER,
    tri_2_2015 INTEGER,
    tri_3_2015 INTEGER,
    tri_4_2015 INTEGER)
""")

```

#δημιουργώ πίνακα Countries_with_most_tourists που θα περιέχει τη χώρα με τους περισσότερους τουρίστες που ταξίδεψαν στην Ελλάδα για κάθε τετραετία 2011-2015 (στο συγκεκριμένο υποερώτημα # ήθελα το csv αρχείο που θα προκύψει από αυτή τη βάση να είναι πιο περιεκτικό περιέχοντας μόνο τη χώρα με τους περισσότερους τουρίστες ανά έτος και να γίνω πιο ακριβής στα γραφήματα που θα εξάγω

με τα ποσοστά περισσότερων χωρών)

```

cursor.execute("""CREATE TABLE Countries_with_most_tourists (
    country_2011 STRING,
    country_2012 STRING,
    country_2013 STRING,
    country_2014 STRING,
    country_2015 STRING)""")

```

#εισάγω στον πρώτο πίνακα τις μεταβλητές all_201x που όρισα προηγουμένως με το συνολικό πλήθος τουριστών ανά έτος με τη συνάρτηση insert_date

```

def insert_date(all_2011, all_2012, all_2013, all_2014, all_2015):

```

```

with conn:
    cursor.execute("INSERT INTO allTourists VALUES(:2011, :2012, :2013, :2014, :2015)", \
                    {'2011': all_2011, '2012': all_2012, '2013': all_2013, '2014': all_2014,
                     '2015': all_2015})

insert_date(all_2011, all_2012, all_2013, all_2014, all_2015)

#ΔΗΜΙΟΥΡΓΙΑ PLOT 1

#δημιουργώ array με τις μεταβλητές all_201x για να τις τοποθετήσω ως τεταγμένες στο γραμμικό
plot που φτιάχνω
total_tourists = np.array([all_2011, all_2012, all_2013, all_2014, all_2015])
#οι τετμημένες του plot
years = [str(float(2011)), 2012, 2013, 2014, 2015]
plt.plot(years, total_tourists,color='magenta')
plt.title("Συνολικός αριθμός τουριστών στην τετραετία 2011 - 2015")
plt.xlabel("Έτος")
plt.ylabel("Πλήθος τουριστών (σε εκατομμύρια)")
plt.show()

#εισάγω στον δεύτερο πίνακα τις μεταβλητές plane/train/boat/car/_201x που όρισα προηγουμένως με
το συνολικό πλήθος τουριστών ανά μεταφορικό και έτος με τη συνάρτηση insert_transportation
def insert_transportation(plane_2011, train_2011, boat_2011, car_2011, plane_2012, train_2012,
boat_2012, car_2012,
                           plane_2013, train_2013, boat_2013, car_2013, plane_2014, train_2014,
boat_2014, car_2014,
                           plane_2015, train_2015, boat_2015, car_2015):
    with conn:
        cursor.execute(
            "INSERT INTO transportation VALUES(:2011_by_plane, :2011_by_train, :2011_by_boat,
:2011_by_car, :2012_by_plane, :2012_by_train, :2012_by_boat, :2012_by_car, :2013_by_plane,
:2013_by_train, :2013_by_boat, :2013_by_car, :2014_by_plane, :2014_by_train, :2014_by_boat,
:2014_by_car, :2015_by_plane, :2015_by_train, :2015_by_boat, :2015_by_car)", \
            {'2011_by_plane': plane_2011, '2011_by_train': train_2011, '2011_by_boat':
boat_2011,
            '2011_by_car': car_2011, '2012_by_plane': plane_2012, '2012_by_train': train_2012,
            '2012_by_boat': boat_2012, '2012_by_car': car_2012, '2013_by_plane': plane_2013,
            '2013_by_train': train_2013, '2013_by_boat': boat_2013, '2013_by_car': car_2013,
            '2014_by_plane': plane_2014, '2014_by_train': train_2014, '2014_by_boat':
boat_2014,
            '2014_by_car': car_2014, '2015_by_plane': plane_2015, '2015_by_train': train_2015,
            '2015_by_boat': boat_2015, '2015_by_car': car_2015})

insert_transportation(plane_2011, train_2011, boat_2011, car_2011, plane_2012, train_2012,
boat_2012, car_2012,
                       plane_2013, train_2013, boat_2013, car_2013, plane_2014, train_2014,
boat_2014, car_2014,
                       plane_2015, train_2015, boat_2015, car_2015)

#ΔΗΜΙΟΥΡΓΙΑ PLOT 2

#περνάω τις μεταβλητές plane/train/boat/car_201x σε array που θα αποτελέσουν τεταγμένες του bar
plot
height = np.array([plane_2011, train_2011, boat_2011, car_2011, plane_2012, train_2012,
boat_2012, car_2012,
                    plane_2013, train_2013, boat_2013, car_2013, plane_2014, train_2014,
boat_2014, car_2014,
                    plane_2015, train_2015, boat_2015, car_2015])

barlist = plt.bar(['Plane (2011)', 'Train (2011)', 'Boat (2011)', 'Car (2011)', 'Plane (2012)',
'Train (2012)', 'Boat (2012)', 'Car (2012)', 'Plane (2013)', 'Train (2013)', 'Boat (2013)', 'Car
(2013)', 'Plane (2014)', 'Train (2014)', 'Boat (2014)', 'Car (2014)', 'Plane (2015)', 'Train
(2015)', 'Boat (2015)', 'Car (2015)'],height)

```

#ορίζω διαφορετικό χρώμα για κάθε μεταφορικό μέσο

```
barlist[0].set_color('y')
barlist[4].set_color('y')
barlist[8].set_color('y')
barlist[12].set_color('y')
barlist[16].set_color('y')
barlist[1].set_color('g')
barlist[5].set_color('g')
barlist[9].set_color('g')
barlist[13].set_color('g')
barlist[17].set_color('g')
barlist[3].set_color('r')
barlist[7].set_color('r')
barlist[11].set_color('r')
barlist[15].set_color('r')
barlist[19].set_color('r')
```

```
plt.title("Αφίξεις τουριστών στην Ελλάδα ανά μέσο μεταφοράς για την τετραετία 2011-2015")
plt.xlabel("Μέσο μεταφοράς (έτος)")
plt.ylabel("Πλήθος τουριστών")
#το βάζω για να φαίνονται με ακρίβεια οι πολύ μικρές τιμές του τρένου σε σχέση με τα υπόλοιπα μεταφορικά μέσα
plt.yscale("log")
plt.show()
```

#εισάγω στον τρίτο πίνακα τις μεταβλητές xxx_trimester_201x που όρισα προηγουμένως με το συνολικό πλήθος τουριστών ανά τρίμηνο με τη συνάρτηση insert_trimesters

```
def insert_trimesters(trimester_1_2011, trimester_2_2011, trimester_3_2011, trimester_4_2011,
trimester_1_2012,
trimester_2_2012, trimester_3_2012, trimester_4_2012, trimester_1_2013,
trimester_2_2013,
trimester_3_2013, trimester_4_2013, trimester_1_2014, trimester_2_2014,
trimester_3_2014,
trimester_4_2014, trimester_1_2015, trimester_2_2015, trimester_3_2015,
trimester_4_2015):
    with conn:
        cursor.execute(
            "INSERT INTO Tourists_per_trimester VALUES(:1stTrimester2011, :2ndTrimester2011,
:3rdTrimester2011, :4thTrimester2011,:1stTrimester2012, :2ndTrimester2012, :3rdTrimester2012,
:4thTrimester2012,:1stTrimester2013, :2ndTrimester2013, :3rdTrimester2013,:4thTrimester2013,
:1stTrimester2014, :2ndTrimester2014, :3rdTrimester2014, :4thTrimester2014, :1stTrimester2015,
:2ndTrimester2015, :3rdTrimester2015, :4thTrimester2015)", \
            {'1stTrimester2011': trimester_1_2011, '2ndTrimester2011': trimester_2_2011,
'3rdTrimester2011': trimester_3_2011, '4thTrimester2011': trimester_4_2011,
'1stTrimester2012': trimester_1_2012, '2ndTrimester2012': trimester_2_2012,
'3rdTrimester2012': trimester_3_2012, '4thTrimester2012': trimester_4_2012,
'1stTrimester2013': trimester_1_2013, '2ndTrimester2013': trimester_2_2013,
'3rdTrimester2013': trimester_3_2013, '4thTrimester2013': trimester_4_2013,
'1stTrimester2014': trimester_1_2014, '2ndTrimester2014': trimester_2_2014,
'3rdTrimester2014': trimester_3_2014, '4thTrimester2014': trimester_4_2014,
'1stTrimester2015': trimester_1_2015, '2ndTrimester2015': trimester_2_2015,
'3rdTrimester2015': trimester_3_2015, '4thTrimester2015': trimester_4_2015})
```

```
insert_trimesters(_1st_trimester_2011, _2nd_trimester_2011, _3rd_trimester_2011,
_4th_trimester_2011,
_1st_trimester_2012, _2nd_trimester_2012, _3rd_trimester_2012,
_4th_trimester_2012,
_1st_trimester_2013, _2nd_trimester_2013, _3rd_trimester_2013,
_4th_trimester_2013,
_1st_trimester_2014, _2nd_trimester_2014, _3rd_trimester_2014,
_4th_trimester_2014,
_1st_trimester_2015, _2nd_trimester_2015, _3rd_trimester_2015,
_4th_trimester_2015)
```

#ΔΗΜΙΟΥΡΓΙΑ PLOT 3

```
#περνάω τις μεταβλητές _xxx_trimester_201x σε array για να τις κάνω τεταγμένες στο linear plot
μου
total_tourists_per_trimester = np.array([_1st_trimester_2011, _2nd_trimester_2011,
_3rd_trimester_2011, _4th_trimester_2011,
_1st_trimester_2012, _2nd_trimester_2012, _3rd_trimester_2012,
_4th_trimester_2012,
_1st_trimester_2013, _2nd_trimester_2013, _3rd_trimester_2013,
_4th_trimester_2013,
_1st_trimester_2014, _2nd_trimester_2014, _3rd_trimester_2014,
_4th_trimester_2014,
_1st_trimester_2015, _2nd_trimester_2015, _3rd_trimester_2015,
_4th_trimester_2015])

#οι τετμημένες του plot
time = ['JAN-MAR 2011', 'APR-JUN 2011', 'JUL-SEP 2011', 'OCT-DEC 2011', 'JAN-MAR 2012', 'APR-JUN
2012', 'JUL-SEP 2012', 'OCT-DEC 2012', 'JAN-MAR 2013', 'APR-JUN 2013', 'JUL-SEP 2013', 'OCT-DEC
2013', 'JAN-MAR 2014', 'APR-JUN 2014', 'JUL-SEP 2014', 'OCT-DEC 2014', 'JAN-MAR 2015', 'APR-JUN
2015', 'JUL-SEP 2015', 'OCT-DEC 2015']

plt.plot(time, total_tourists_per_trimester)
plt.title("Αφίξεις τουριστών στην Ελλάδα για την τετραετία 2011 - 2015 ανά τρίμηνο")
plt.xlabel("Τρίμηνο")
plt.ylabel("Πλήθος τουριστών (σε εκατομμύρια)")
plt.show()

#εισάγω στον πρώτο πίνακα τις μεταβλητές maxcountry201x που όρισα προηγουμένως με τη χώρα με το
μέγιστο πλήθος τουριστών ανά έτος με τη συνάρτηση insert_countries
def insert_countries(country_2011, country_2012, country_2013, country_2014, country_2015):
    with conn:
        cursor.execute(
            "INSERT INTO Countries_with_most_tourists VALUES
(:most_tourists_2011,:most_tourists_2012,:most_tourists_2013,:most_tourists_2014,:most_tourists
_2015)", \
            {'most_tourists_2011': country_2011, 'most_tourists_2012': country_2012,
'most_tourists_2013': country_2013,
'most_tourists_2014': country_2014, 'most_tourists_2015': country_2015})

insert_countries(maxcountry2011, maxcountry2012, maxcountry2013, maxcountry2014,
maxcountry2015)

#ΔΗΜΙΟΥΡΓΙΑ PIE CHARTS ΓΙΑ ΚΑΘΕ ΈΤΟΣ

#διαιρώ το πλήθος των τουριστών των τριών χωρών με τους περισσότερους τουρίστες με το συνολικό
πλήθος τουριστών του έτους για να βρω το μερίδιό τους (size) στο pie chart
third_2011_size = (third_2011/sum_2011)
second_2011_size = (second_2011/sum_2011)
first_2011_size = (first_2011/sum_2011)

#βρίσκω το ποσοστό που καταλαμβάνουν οι υπόλοιπες χώρες
rest_size_2011 = (1 - (third_2011_size + second_2011_size + first_2011_size))

third_2012_size = (third_2012/sum_2012)
second_2012_size = (second_2012/sum_2012)
first_2012_size = (first_2012/sum_2012)
rest_size_2012 = (1 - (third_2012_size + second_2012_size + first_2012_size))

third_2013_size = (third_2013/sum_2013)
second_2013_size = (second_2013/sum_2013)
first_2013_size = (first_2013/sum_2013)
rest_size_2013 = (1 - (third_2013_size + second_2013_size + first_2013_size))

third_2014_size = (third_2014/sum_2014)
```

```
second_2014_size = (second_2014/sum_2014)
first_2014_size = (first_2014/sum_2014)
rest_size_2014 = (1 - (third_2014_size + second_2014_size + first_2014_size))
```

```
third_2015_size = (third_2015/sum_2015)
second_2015_size = (second_2015/sum_2015)
first_2015_size = (first_2015/sum_2015)
rest_size_2015 = (1 - (third_2015_size + second_2015_size + first_2015_size))
```

```
#Φτιάχνω ένα figure για κάθε έτος
```

```
plt.figure(0)
labels = thirdcountry2011, secondcountry2011, maxcountry2011, 'Υπόλοιπες χώρες'
sizes = [third_2011_size, second_2011_size, first_2011_size, rest_size_2011]
colors = ['yellowgreen', 'lightcoral', 'lightskyblue', 'grey']
plt.pie(sizes, explode = (0, 0, 0.1, 0), labels=labels,
colors=colors,autopct='%1.1f%%',startangle=140)
plt.axis('equal')
plt.title('Χώρες καταγωγής με το μεγαλύτερο μερίδιο στις αφίξεις τουριστών στην Ελλάδα για το έτος 2011')
```

```
plt.figure(1)
labels = thirdcountry2012, secondcountry2012, maxcountry2012, 'Υπόλοιπες χώρες'
sizes = [third_2012_size, second_2012_size, first_2012_size, rest_size_2012]
colors = ['yellowgreen', 'lightcoral', 'lightskyblue', 'grey']
plt.pie(sizes, explode = (0, 0, 0.1, 0), labels=labels,
colors=colors,autopct='%1.1f%%',startangle=140)
plt.axis('equal')
plt.title('Χώρες καταγωγής με το μεγαλύτερο μερίδιο στις αφίξεις τουριστών στην Ελλάδα για το έτος 2012')
```

```
plt.figure(2)
labels = thirdcountry2013, secondcountry2013, maxcountry2013, 'Υπόλοιπες χώρες'
sizes = [third_2013_size, second_2013_size, first_2013_size, rest_size_2013]
colors = ['yellowgreen', 'lightcoral', 'lightskyblue', 'grey']
plt.pie(sizes, explode = (0, 0, 0.1, 0), labels=labels,
colors=colors,autopct='%1.1f%%',startangle=140)
plt.axis('equal')
plt.title('Χώρες καταγωγής με το μεγαλύτερο μερίδιο στις αφίξεις τουριστών στην Ελλάδα για το έτος 2013')
```

```
plt.figure(3)
labels = thirdcountry2014, secondcountry2014, maxcountry2014, 'Υπόλοιπες χώρες'
sizes = [third_2014_size, second_2014_size, first_2014_size, rest_size_2014]
colors = ['yellowgreen', 'lightcoral', 'lightskyblue', 'grey']
plt.pie(sizes, explode = (0, 0, 0.1, 0), labels=labels,
colors=colors,autopct='%1.1f%%',startangle=140)
plt.axis('equal')
plt.title('Χώρες καταγωγής με το μεγαλύτερο μερίδιο στις αφίξεις τουριστών στην Ελλάδα για το έτος 2014')
```

```
plt.figure(4)
labels = thirdcountry2011, secondcountry2011, maxcountry2011, 'Υπόλοιπες χώρες'
sizes = [third_2011_size, second_2011_size, first_2011_size, rest_size_2011]
colors = ['yellowgreen', 'lightcoral', 'lightskyblue', 'grey']
plt.pie(sizes, explode = (0, 0, 0.1, 0), labels=labels,
colors=colors,autopct='%1.1f%%',startangle=140)
plt.axis('equal')
plt.title('Χώρες καταγωγής με το μεγαλύτερο μερίδιο στις αφίξεις τουριστών στην Ελλάδα για το έτος 2015')
```

```
plt.show()
```



```
cursor = conn.cursor()
```

#Με τη βοήθεια του κέρσορα εκτελώ το query που θα μου επιστρέψει όλα τα records του πίνακα allTourists

```
cursor.execute('select * from allTourists')
```

#Γράφω τα records ("w") σε csv αρχείο

```
with open("Tourists_data.csv", "w", encoding = 'utf-16', newline='') as csv_file:
    csv_writer = csv.writer(csv_file, delimiter="\t")
    csv_writer.writerow([i[0] for i in cursor.description])
    csv_writer.writerows(cursor)
    dirpath = os.getcwd() + "/Tourists_data.csv"
```

#Με τη βοήθεια του κέρσορα εκτελώ το query που θα μου επιστρέψει όλα τα records του πίνακα transportation

```
cursor.execute('select * from transportation')
```

#Γράφω τα records ("w") σε csv αρχείο

```
with open("Transportation_data.csv", "w", encoding = 'utf-16', newline='') as csv_file:
    csv_writer = csv.writer(csv_file, delimiter="\t")
    csv_writer.writerow([i[0] for i in cursor.description])
    csv_writer.writerows(cursor)
    dirpath = os.getcwd() + "/Transportation.csv"
```

#Με τη βοήθεια του κέρσορα εκτελώ το query που θα μου επιστρέψει όλα τα records του πίνακα Tourists_per_Trimester

```
cursor.execute('select * from Tourists_per_trimester')
```

#Γράφω τα records ("w") σε csv αρχείο

```
with open("Tourists_per_trimester_data.csv", "w", encoding = 'utf-16', newline='') as csv_file:
    csv_writer = csv.writer(csv_file, delimiter="\t")
    csv_writer.writerow([i[0] for i in cursor.description])
    csv_writer.writerows(cursor)
    dirpath = os.getcwd() + "/Tourists_per_trimester.csv"
```

#Με τη βοήθεια του κέρσορα εκτελώ το query που θα μου επιστρέψει όλα τα records του πίνακα Countries_with_most_tourists

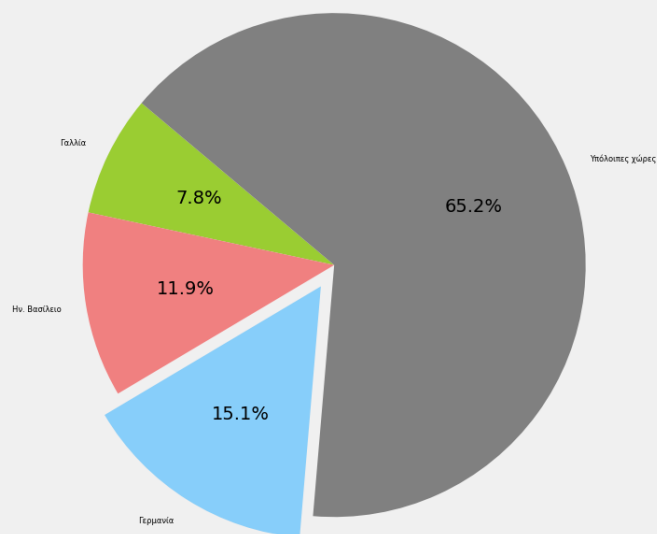
```
cursor.execute('select * from Countries_with_most_tourists')
```

#Γράφω τα records ("w") σε csv αρχείο

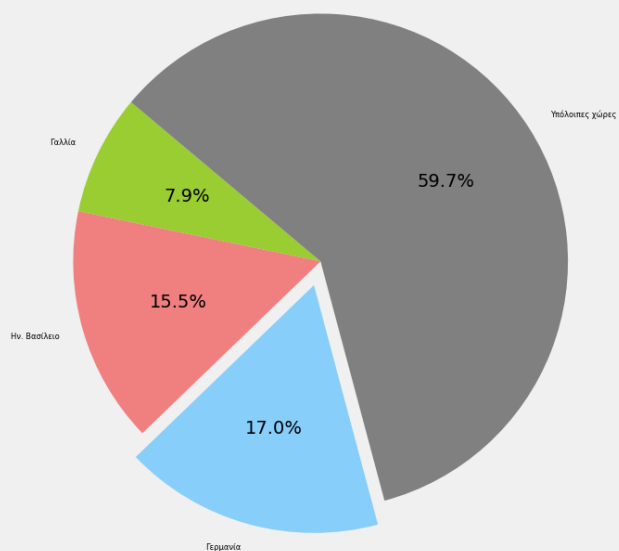
```
with open("Countries_with_most_tourists_data.csv", "w", encoding = 'utf-16', newline='') as csv_file:
    csv_writer = csv.writer(csv_file, delimiter="\t")
    csv_writer.writerow([i[0] for i in cursor.description])
    csv_writer.writerows(cursor)
    dirpath = os.getcwd() + "/Countries_with_most_tourists.csv"
```

ΓΡΑΦΗΜΑΤΑ:

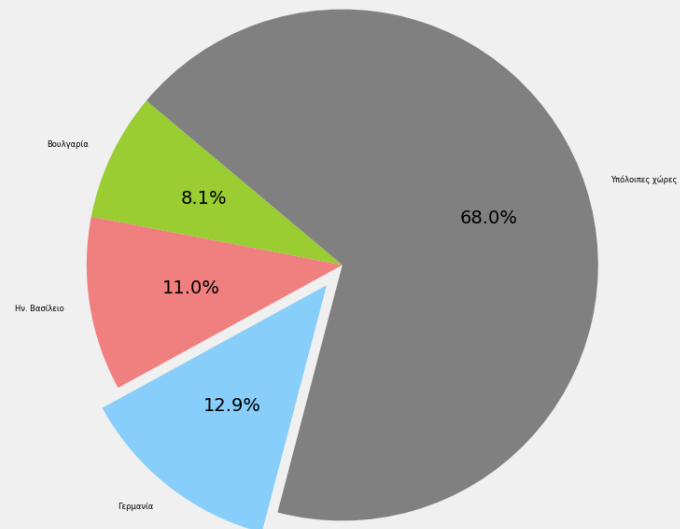
Χώρες καταγωγής με το μεγαλύτερο μερίδιο στις αφίξεις τουριστών στην Ελλάδα για το έτος 2011



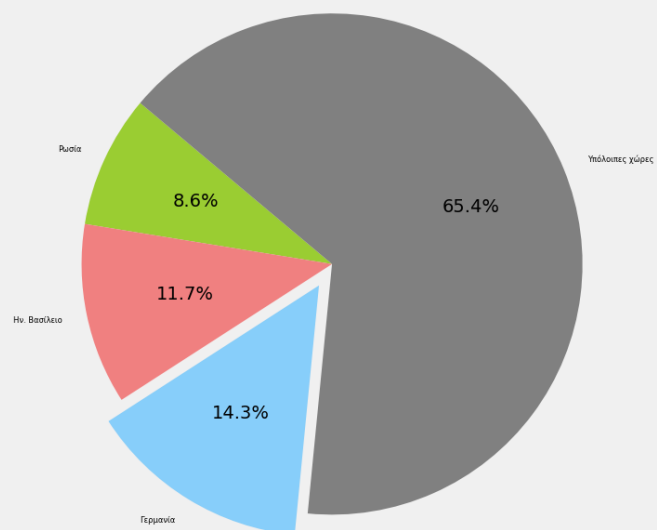
Χώρες καταγωγής με το μεγαλύτερο μερίδιο στις αφίξεις τουριστών στην Ελλάδα για το έτος 2012



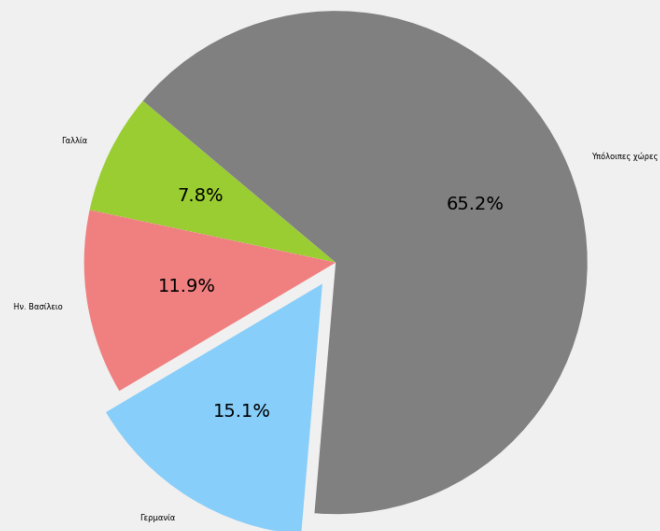
Χώρες καταγωγής με το μεγαλύτερο μερίδιο στις αφίξεις τουριστών στην Ελλάδα για το έτος 2014



Χώρες καταγωγής με το μεγαλύτερο μερίδιο στις αφίξεις τουριστών στην Ελλάδα για το έτος 2013

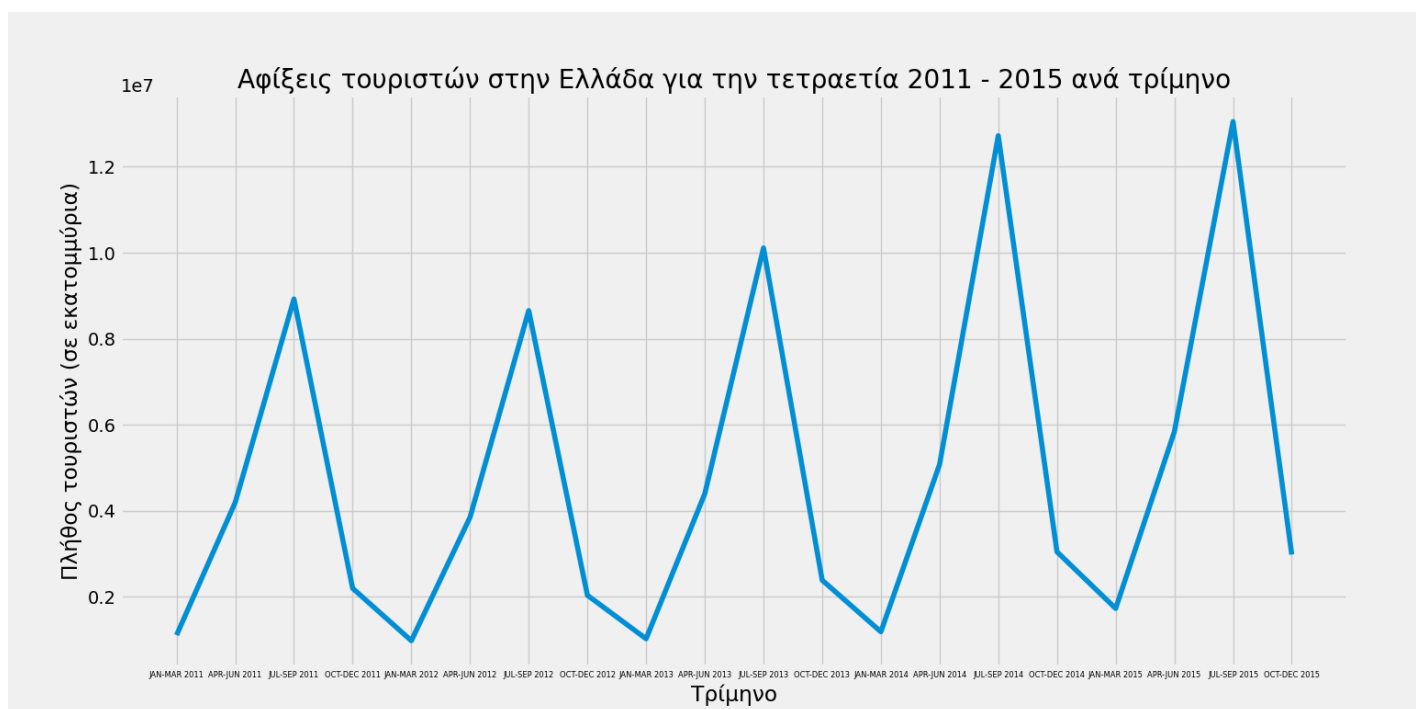
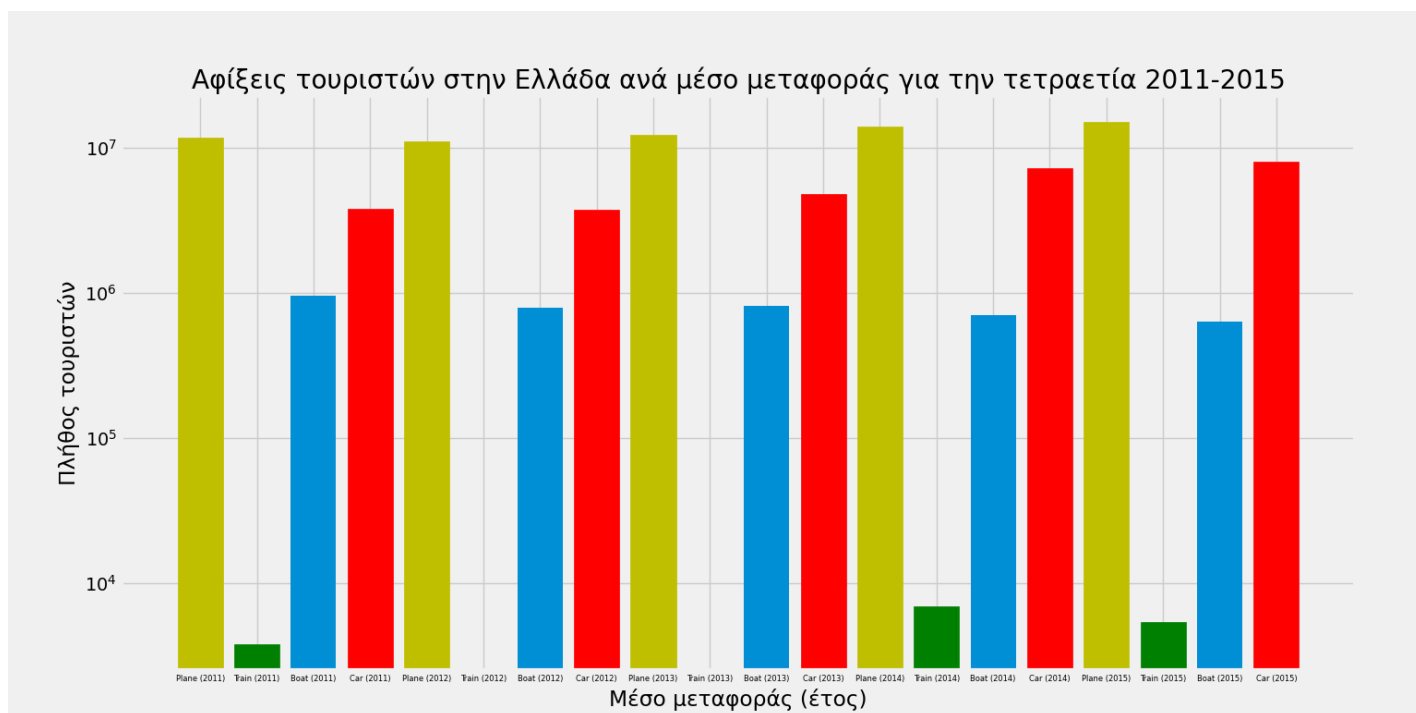


Χώρες καταγωγής με το μεγαλύτερο μερίδιο στις αφίξεις τουριστών στην Ελλάδα για το έτος 2015



Συνολικός αριθμός τουριστών στην τετραετία 2011 - 2015





Screenshots :

