به نام خدا

گزارش تمرین سری اول درس داده کاوی

فاطمه غلام زاده \_ 9531060

سوال 1 :

* از OLTP برای کاربردهای روزمره مثل ذخیره transaction ها استفاده می شود اما از OLAP برای پشتیبانی تصمیم استفاده می شود.
* OLTP کاربرد محور است اما OLAP موضوع محور است.
* دیتاهای موجود در OLTP بسیار جزئی و آپدیت شده هستند اما دیتاهای OLAP مربوط به زمان های طولانی (مثلا چند سال) هستند ، هم چنین به صورت تجمعی ( مثل اینکه چند بعد از دیتا را integrate کنیم) هستند.
* کاربران OLTP معمولا کارمندان یا مسئولان IT شرکت ها هستند اما با OLAP مدیران سازمان یا دیتاآنالیست ها کار می کنند.
* برای کار کردن با OLTP از transaction های ساده استفاده می شود اما برای OLAP کوئری های پیچیده .
* تعداد یوزر های OLTP از OLAP بیشتر است ( OLAP از مرتبه 100 اما OLTP از مرتبه 1000)
* سایز دیتابیس های OLAP از OLTP بیشتر است (OLTP در محدوده 100MB-GB اما OLAP در محدوده 100GB-TB)

سوال 2 :

برای انجام این سوال در phpmyadmin کوئری زدم تا جدول London12 ایجاد شود. ابتدا دو فایل داده شده را در قسمت import اضافه کردم بعد روی آن ها به صورت زیر کوئری زدم :

INSERT INTO london12 (country,gender,agegroup,sport,gold,silver,bronze)

SELECT country,gender,age,sport,gold,silver,bronze FROM Olympic

این کوئری ستون های country,gender,agegroup,sport,gold,silver,bronze را از جدول Olympic انتخاب می کند و در جدول London12 قرار می دهد.برای ایجاد ستون continent از inner join استفاده کردم . به صورت زیر :

UPDATE london12

INNER JOIN

countries\_by\_continent ON london12.country = countries\_by\_continent.country

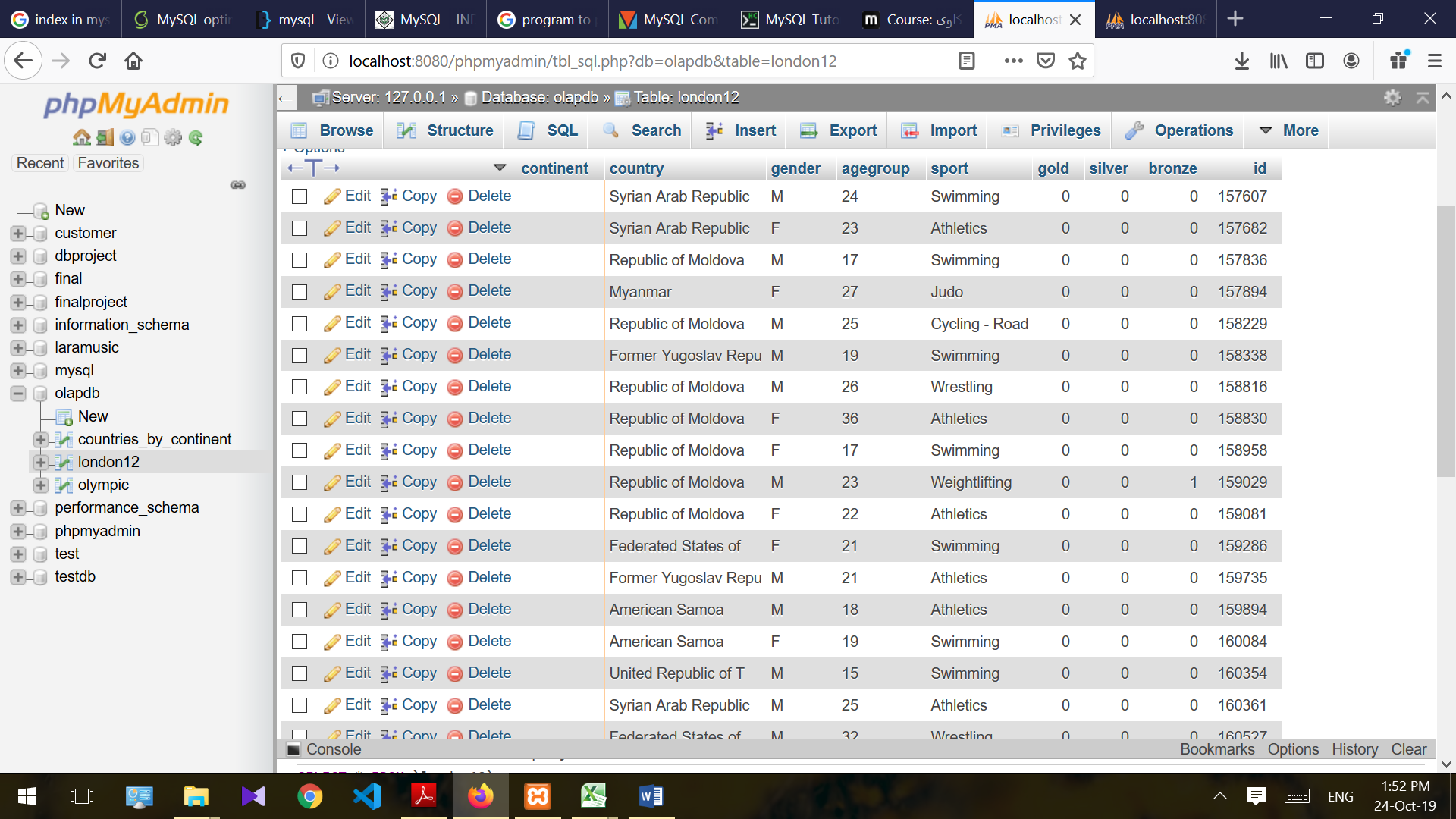
SET

london12.continent = countries\_by\_continent.continent;

بعد از انجام این عمل برای اینکه بفهمم چه کشور های قاره شان متفاوت در دو فایل ذخیره شده یک کوئری زدم و کشور هایی را که ستون قاره آن ها برابر '' بود پیدا کردم ، به این شکل :

SELECT \* FROM `london12` WHERE continent = ''

نتیجه این کوئری جدول زیر شد :



بعد جدول Olympic را با توجه به جدول countries\_by\_continent.country اصلاح کردم (یعنی نام کشور ها را در جدول Olympic با نامی که در فایل countries\_by\_continent.country آمده بود یکسان کردم )  
کوئری هایی که زدم به شکل زیر است :

UPDATE olympic

SET country = 'china'

WHERE country = "People's Republic of";

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

UPDATE olympic

SET country = 'Iran'

WHERE country = "Islamic Republic of";

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

UPDATE olympic

SET country = 'Syria'

WHERE country = "Syrian Arab Republic";

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

UPDATE olympic

SET country = 'Moldova'

WHERE country = "Republic of Moldova";

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

UPDATE olympic

SET country = 'Burma (Myanmar)'

WHERE country = "Myanmar";

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

UPDATE olympic

SET country = ' Macedonia '

WHERE country = "Former Yugoslav Repu";

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

UPDATE olympic

SET country = ' Samoa '

WHERE country = " American Samoa";

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

UPDATE olympic

SET country = ' Micronesia'

WHERE country = " Federated States of";

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

UPDATE olympic

SET country = ' Tanzania '

WHERE country = "United Republic of T";

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

UPDATE olympic

SET country = 'Brunei'

WHERE country = "Brunei Darussalam";

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

UPDATE olympic

SET country = 'Burkina'

WHERE country = "Burkina Faso";

در آخر برای اینکه این تغییرات روی london12 هم ایجاد شود این جدول را حذف کردم و دوباره 2 کوئری اول که برای ایجاد آن بود زدم تا ستون قاره به درستی ایجاد شود.سپس برای درست کردن ستون agegroup این 4 کوئری را زدم :

UPDATE london12

SET

agegroup ='A'

WHERE agegroup<20

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

UPDATE london12

SET

agegroup ='B'

WHERE (agegroup>= 20 and agegroup < 25 )

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

UPDATE london12

SET

agegroup ='C'

WHERE (agegroup>= 25 and agegroup < 30 )

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

UPDATE london12

SET

agegroup ='D'

WHERE (agegroup>=30)

سوال 3 : فایل Q3.py ضمیمه شده است . در واقع همه ترکیب های ممکن از مجموعه :

strings = [**"sport"**,**"agegroup"**,**"gender"**,**"country"**,**"continent"**]  
  
را تولید کردم و با آن ایندکس ایجاد کردم.

کد آن به صورت زیر است :

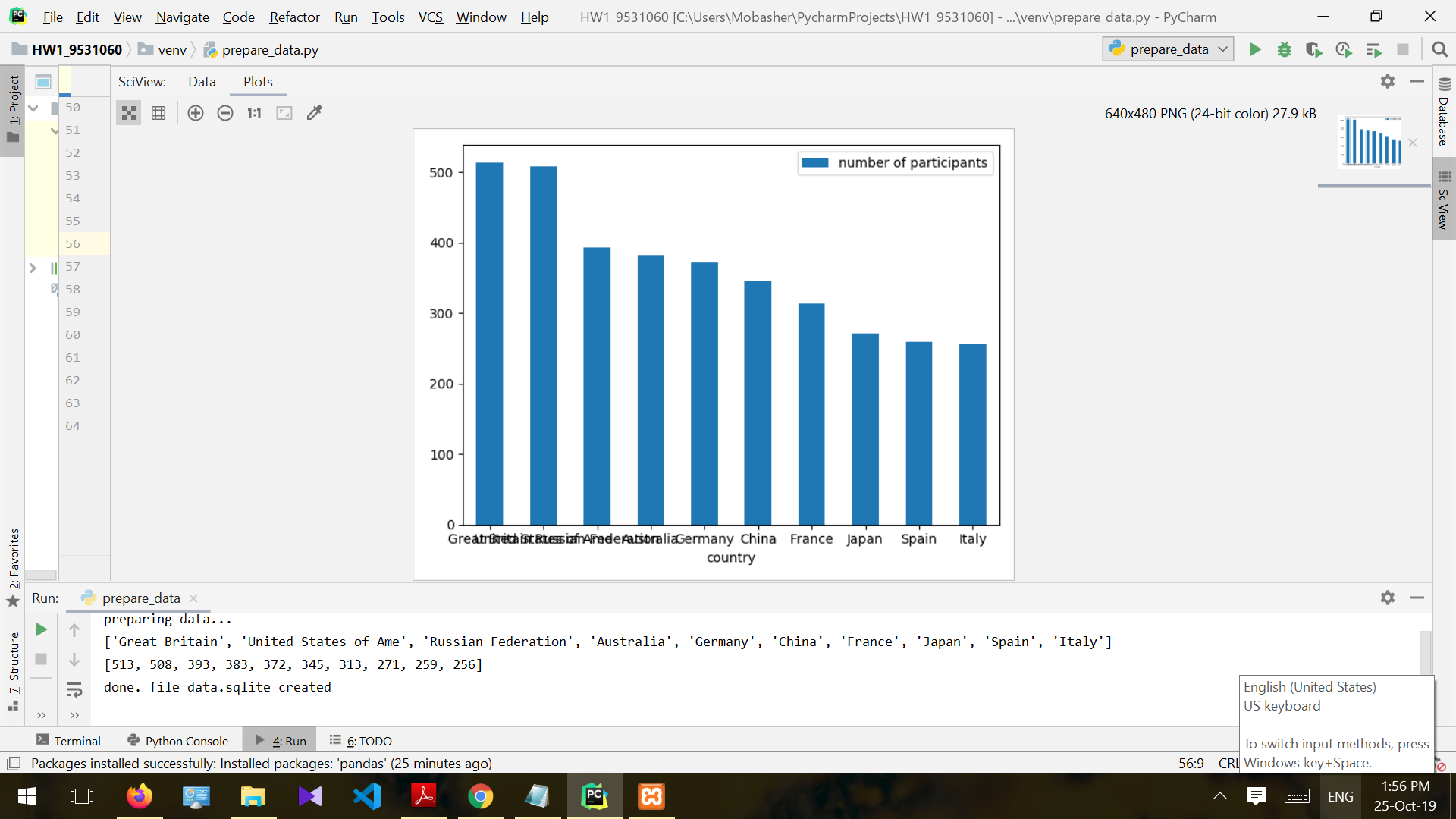
mydb = connector.connect(  
 host=**"localhost"**,  
 user=**"root"**,  
 passwd=**""**,  
 database=**"olapdb"**)  
  
mycursor = mydb.cursor()  
  
*################## Question 3 #########################*strings = [**"sport"**,**"agegroup"**,**"gender"**,**"country"**,**"continent"**]  
  
**def** convertTuple(tup):  
 str = **','**.join(tup)  
 **return** str  
k = 0;  
**for** i **in** range(1,5):  
 sList = list(itertools.combinations(strings, i))  
 **for** j **in** range(0,len(sList)):  
 *# print(convertTuple(sList.\_\_getitem\_\_(j)))* s = **"""CREATE INDEX h"""**+str(k)+**""" ON london12("""** + convertTuple(sList.\_\_getitem\_\_(j)) + **""");"""** k+=1  
 print(s)  
 mycursor.execute(s)

سوال 4 :

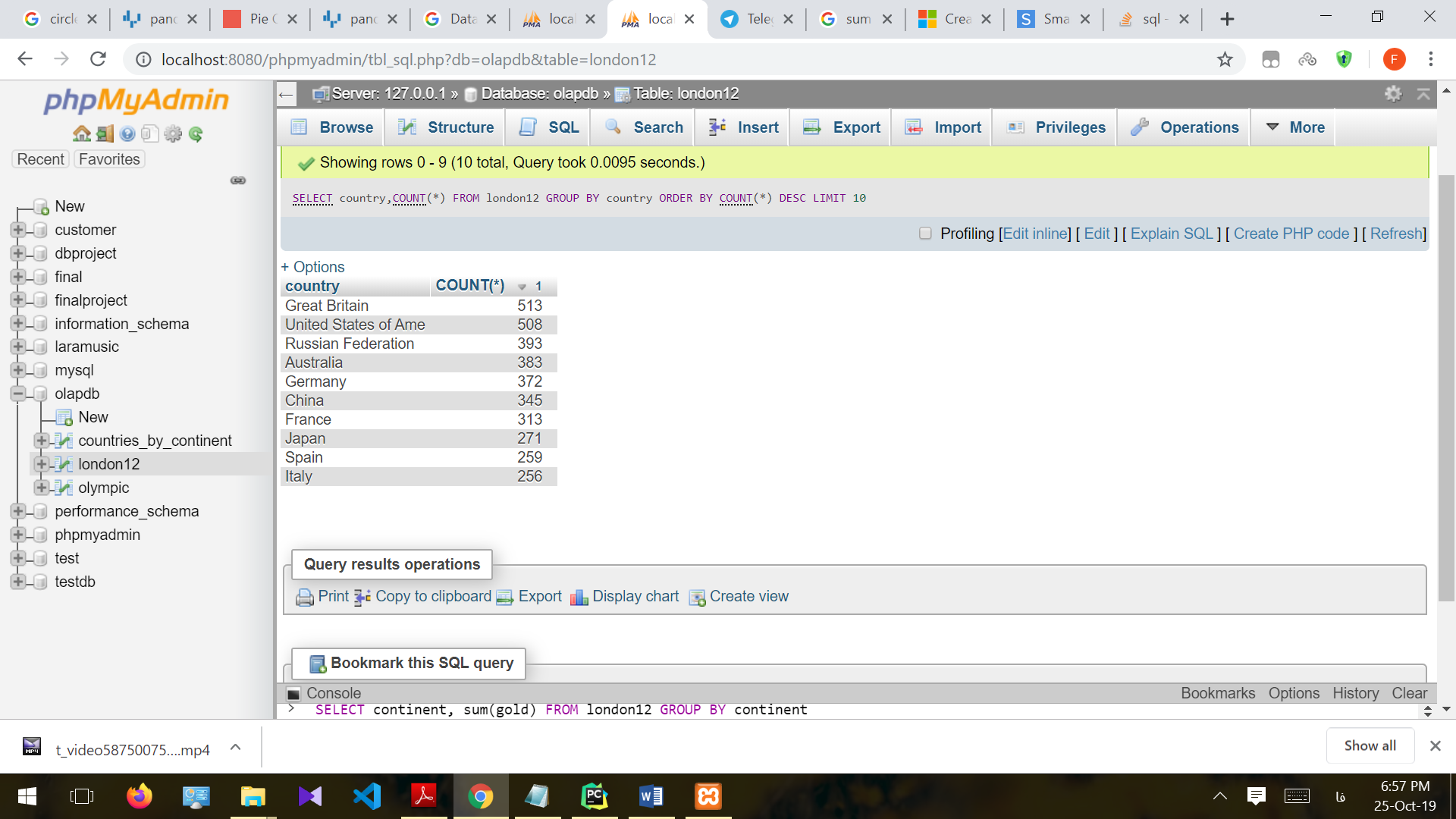
قسمت a ) فایل Q4\_a.pyضمیمه شده است..کد آن به صورت زیر است :

mydb = connector.connect(  
 host=**"localhost"**,  
 user=**"root"**,  
 passwd=**""**,  
 database=**"olapdb"**)  
  
mycursor = mydb.cursor()  
*################## Question 4 , part a #########################*query = **"""SELECT country,***COUNT***(***\****)  
FROM london12  
GROUP BY country  
ORDER BY** *COUNT***(***\****)  
DESC LIMIT 10  
;"""**mycursor.execute(query)  
records = mycursor.fetchall()  
  
s1 = []  
s2 = []  
  
**for** row **in** records:  
 s1.append(row[0])  
 s2.append(row[1])  
  
df = pd.DataFrame({**'country'**:s1, **'number of participants'**:s2})  
ax = df.plot.bar(x=**'country'**, y=**'number of participants'**, rot=0)  
plt.show()

خروجی کد به صورت زیر است :



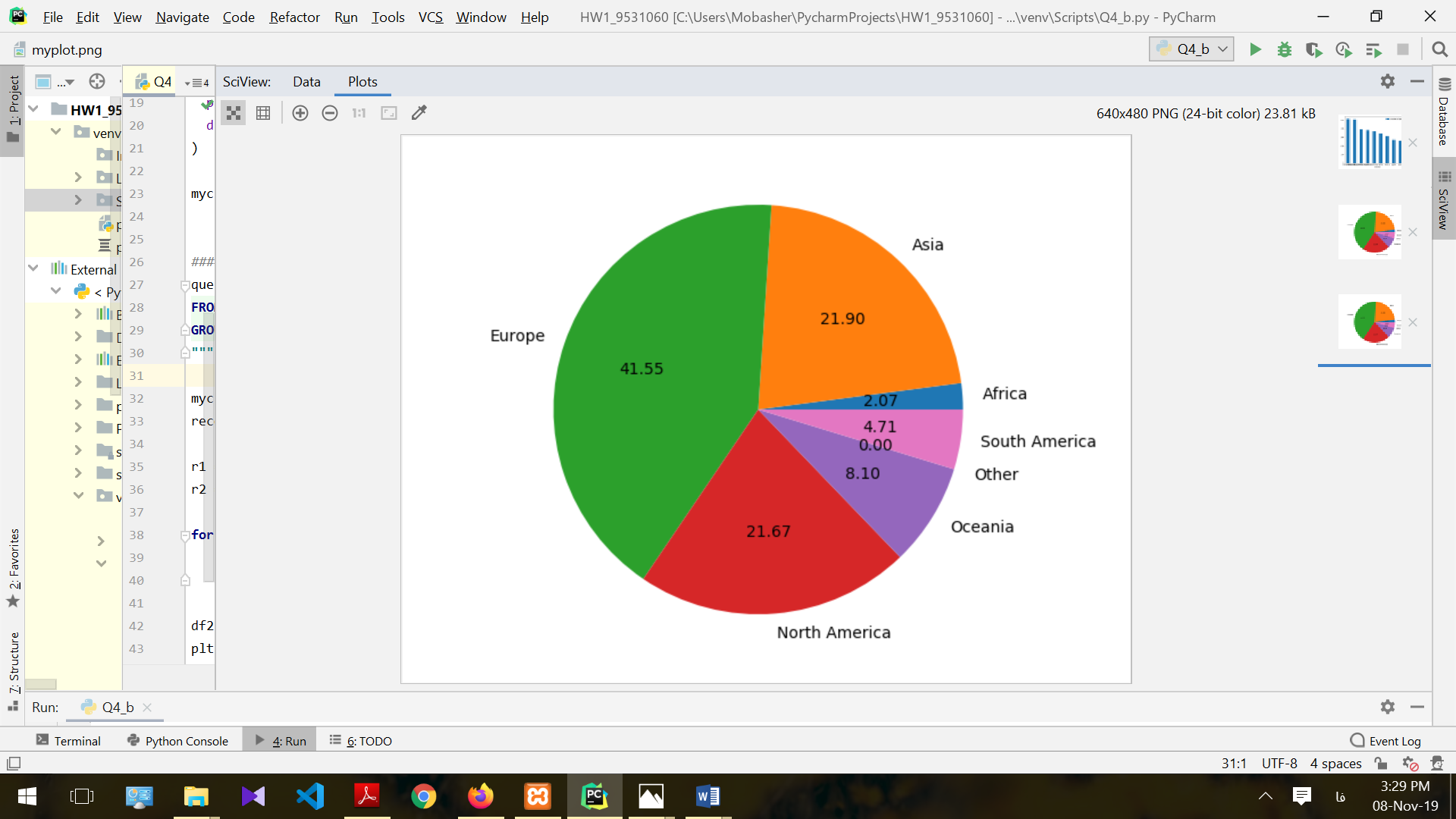
برای اطمینان با خروجی کوئری در mysql مقایسه کردم .خروجی کوئری هم به شکل زیر است :



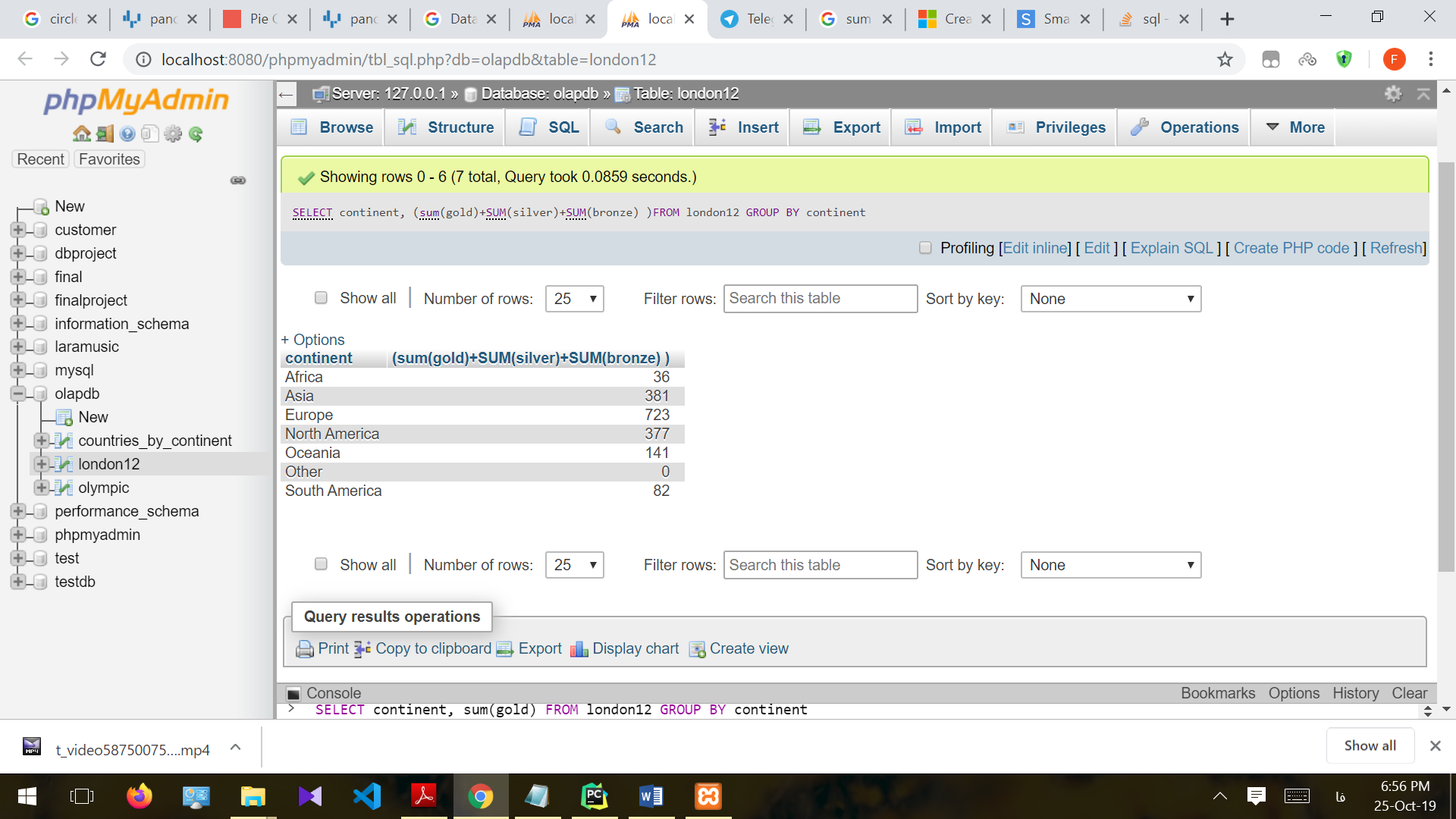
قسمت b )فایل Q4\_b.py ضمیمه شده است.کد آن به شکل زیر است :

query2 = **""" SELECT continent, (***SUM***(gold)+***SUM***(silver)+***SUM***(bronze))  
FROM london12  
GROUP BY continent ;  
"""**mycursor.execute(query2)  
records2= mycursor.fetchall()  
  
r1 = []  
r2 = []  
  
**for** row **in** records2:  
 r1.append(row[0])  
 r2.append(row[1])  
  
df2 = pd.DataFrame(records2,columns=[**'r1'**,**'r2'**])  
plt.pie(df2[**'r2'**],labels=df2[**'r1'**],autopct=**"%.2f"**)  
  
plt.show()

خروجی به صورت زیر خواهد شد :



خروجی کوئری در mysql نیز به شکل زیر است :

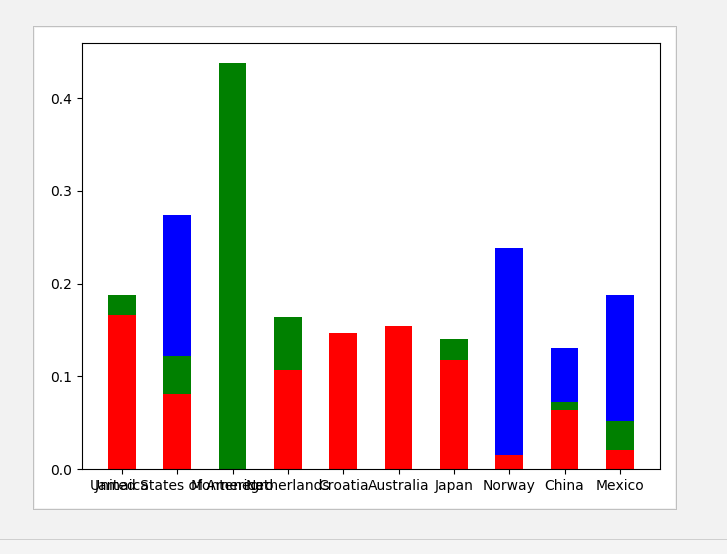


قسمت c )

فایل Q4\_c.py ضمیمه شده است. کد آن به صورت زیر است :

print(**"preparing data..."**)  
  
  
mydb = connector.connect(  
 host=**"localhost"**,  
 user=**"root"**,  
 passwd=**""**,  
 database=**"olapdb"**)  
  
mycursor = mydb.cursor()  
  
Q = **"""SELECT country,Big.gold\_sum/Big.co,Big.silver\_sum/Big.co,Big.bronze\_sum/Big.co,Big.s/Big.co as r  
 FROM (SELECTcountry,sum(gold)as gold\_sum,sum(silver)as silver\_sum,SUM(bronze)as bronze\_sum,COUNT(\*) as co,sum(gold+silver+bronze)as s  
 FROM london12  
 GROUP BY country  
 HAVING count(\*) >=30 ) as Big  
 ORDER BY r  
 DESC LIMIT 10 """**mycursor.execute(Q)  
records = mycursor.fetchall()  
array1 = []  
array2 = []  
array3 = []  
array4 = []  
*# print(records)***for** row **in** records:  
 array1.append(row[0])  
 array2.append(row[1])  
 array3.append(row[2])  
 array4.append(row[3])  
ax = plt.subplot(111)  
ax.bar(array1, array2, width=0.5, color=**'b'**, align=**'center'**)  
ax.bar(array1, array3, width=0.5, color=**'g'**, align=**'center'**)  
ax.bar(array1, array4, width=0.5, color=**'r'**, align=**'center'**)  
plt.show()

نمودار خروجی به شکل زیر است :



رنگ آبی : طلا  
رنگ قرمز : برنز

رنگ سبز : نقره

سوال 5 )

برای یافتن تعداد کل کوئری های ممکن 5 حالت در نظر میگیریم :

حال اول : فقط از drill down استفاده کنیم.

در این حالت فقط باید ترکیبات مختلف gender,sport,agegroup,continent را انتخاب کنیم و عملگر drill down را بر روی آن ها اعمال کنیم.پس تعداد حالت ها می شود :

0,4) + C(1,4) + C(2,4) + C(3,4) + C(4,4) = 1 + 4 + 6 + 4 + 1 = 16)C

حالت دوم : dice with 4 attributes

در این حالت همه ی متغیر ها در dice شرکت می کنند و برای drill down چیزی باقی نمی ماند بنابراین تعداد حالات ممکن برابر است با ضرب تعداد حالاتی که برای هر متغیر داریم :

30\*4\*2\*7=1680

حالت سوم : dice with 3 attributes

یک متغیر برای drill down و سه متغیر دیگر برای dice استفاده می شوند. متغیری که برای drill down استفاده می شود 2 حالت دارد و سه متغیری که برای dice استفاده می شوند به تعداد ضرب حالت هایشان :

(4\*7\*2\*2)+(30\*4\*7\*2)+(30\*4\*2\*2)+(30\*2\*7\*2)=3112

حالت چهارم : dice with 2 attributes  
دو متغیر برای drill down و دو متغیر دیگر برای dice استفاده می شوند. . دو متغیری که برای drill down استفاده می شود 4 حالت دارند و دو متغیری که برای dice استفاده می شوند به تعداد ضرب حالت هایشان :

(2\*4\*4)+(2\*30\*4)+(2\*7\*4)+(7\*4\*4)+(7\*30\*4)+(4\*30\*4)=1760

حالت پنجم : slice

در slice فقط روی یک متغیر شرط میگذاریم و سه متغیر دیگر در drill down شرکت می کنند که 8 حالت دارند :

(30\*8)+(4\*8)+(7\*8)+(2\*8)=344

تعداد کل کوئری ها جمع 5 حالت گفته شده است :

344 + 1760 + 3112 + 1680 + 16 = 6912

سوال 6 ) فایل Q6.py ضمیمه شده است.

قسمت a)

کوئری پاسخ به شکل زیر است :

Cut= 'continent=Africa' Drilldown = 'sport,gender,agegroup'

قسمت b) کوئری پاسخ به شکل زیر است :

Cut='sport = Swimming,agegroup=C,continent=North America,gender=M'

قسمت c) کوئری پاسخ به شکل زیر است :

Cut='gender=M,continent=South America'

قسمت d) کوئری پاسخ به شکل زیر است :

Cut='agegroup=B,continent=North America'