Συστήματα Ανάλυσης και Διαχείρισης Μεγάλων δεδομένων

Project 2

Ελένη Τράμπαρη Λάρδα

A.M.: f3312217

Σύντομη αναφορά

Το πρόγραμμα γράφτηκε σε *Python* και βρίσκεται στο αρχείο *reports.exe*.

Αρχικά, για το πρόγραμμα ακολούθησα τα παρακάτω βήματα:

1. Διάβασα τα αρχεία .csv που μας δόθηκαν (data frames).
2. Έλεγξα ότι τα δεδομένα των αρχείων έχουν περαστεί σωστά (οι στήλες τους) και με τος σωστούς τύπους της κάθε στήλης (πχ String, Integer).
3. Μετέτρεψα τα data frames που δημιουργήθηκαν σε sql tables ώστε να μπορώ να θέτω sql ερωτήματα σε αυτά ως πίνακες.

Στο 1ο ερώτημα:

```
# 1. Να παράγει αναφορά με τον συνολικό αριθμό των προϊόντων ανα τμήμα της μορφής «Ονομα_τμήματος, Αριθμός_προϊόντων».
#Η αναφορά να είναι ταξινομημένη με το όνομα του τμήματος σε αλφαβητική σειρά.

print("1st Query ")
first = spark.sql("select d.department, count(product_id) as sumCount from productsTable as p,departmentsTable as d \
                   where p.department_id=d.department_id group by d.department order by d.department")
first.show(21) #there are 21 departments
```

Στη μέθοδο *.show()* έβαλα ως παράμετρο το 21, καθώς εμφανίζονται μόνο οι πρώτες 20 γραμμές του αποτελέσματος by default και έχουμε 21 τμήματα το παντοπωλείο και θα μπορούσαν να εμφανίζονται όλα.

Ο τρόπος με τον οποίο φτιαχνόταν η αναφορά είναι:

```
#create a report file
q1=first.toPandas()
q1.to_csv("data/reports_plots/q1.csv",index=False)
```

Όπου μετατρέπεται το Spark Data Frame σε Pandas Data Frame και έπειτα τα αποτελέσματα αυτά του sql ερωτήματος μεταφέρονται σε ένα *.csv* αρχείο.

Με αντίστοιχο τρόπο έγιναν και τα επόμενα ερωτήματα.

Στο 3ο ερώτημα:

```
#3. Να παραγει αναφορά με τα προϊόντα ανα τμήμα που δεν έχουν παραγγελθεί παραπάνω από μία φορά από κανέναν πελάτη.
#Η αναφορά πρέπει να έχει την μορφή «Ονομα_τμήματος, Κωδικός_προϊόντος, ονομα_Προϊόντος» και να είναι ταξινομημένη
#αλφαβητικά με το όνομα του τμήματος και το όνομα του προϊόντος.


print("3rd Query")
third = spark.sql("select d.department, p.product_id ,p.product_name \
                   from departmentsTable as d,productsTable as p,order_productsTable as o \
                   where d.department_id=p.department_id and p.product_id=o.product_id \
                   group by department,p.product_id ,product_name \
                   having sum(reordered)=0\
                   order by d.department,p.product_name")
third.show()
```

Στο ερώτημα αυτό χρησιμοποιείται το *having sum(reordered)=0* ώστε να εμφανιστούν τα προϊόντα που δεν έχουν παραγγελθεί πάνω από μια φορά (reordered=0) αλλά και **από κανένα πελάτη**.

Στο 4ο ερώτημα:

```
#4. Να παράγει αναφορά με το όνομα του προϊόντος κάθε τμήματος που έχει παραγγλεθεί επανελειμμένα τις περισσότερες
#φορές με την εξής μορφή: «Ονομα_τμήματος, Ονομα_προϊόντος, Φορές_που_έχει_παραγγελθεί».
#Η αναφορά να είναι ταξινομημένη αλφαβητικά με το όνομα του τμήματος.

print("4th Query")
forth= spark.sql("select department,product_name,count(reordered) as times_ordered \
                  from departmentsTable as d, productsTable as p, order_productsTable as o\
                  where d.department_id=p.department_id and p.product_id=o.product_id and reordered=1 \
                  group by department,product_name\
                  order by department, times_ordered desc")
#forth.show()

#εντολή για αφαίρεση duplicates ώστε να βρούμε το μοναδικό product_name (το max)
forth_2 = forth.drop_duplicates(subset=['department'])
forth_2.show()
```

Στο sql ερώτημα εμφανίζονται τα προϊόντα για κάθε department που έχουν παραγγελθεί στο παρελθόν από πελάτη. Εμφανίζονται ταξινομημένα με φθίνουσα σειρά με βάση το πλήθος των φορών που έχουν γίνει reordered.

Επειδή ζητείτε το προϊόν που έχει γίνει τις περισσότερες φορές reordered εκτελώ την εντολή *.drop_duplicates* με βάση το department ώστε να μου εμφανίζει μόνο την πρώτη εγγραφή από κάθε department (δηλαδή το προϊόν που έχει παραγγελθεί επανειλημμένα τις περισσότερες φορές).

Στο 5ο ερώτημα:

```
#5. Να παράγει αναφορά της μορφής «Όνομα διαδρόμου, Ποσοστό», με το ποσοστό των προϊόντων κάθε διαδρόμου που έχουν
#τοποθετηθεί πρώτα στο καλάθι κάποιας παραγγελίας. Η αναφορά να είναι ταξινομημένη αλφαβητικά με το όνομα του διαδρόμου.

print("5th Query")

print("Ypologismos arithmiti")
#arithmitis pososTou = proionta kathe diadromoy poy exoyn toopothetithei prwta sto kalathi
temp1 = spark.sql("select aisle_id,count(*) as count_products\
                  from (select o.product_id,a.aisle_id \
                      from aislesTable as a,order_productsTable as o,productsTable as p\
                      where a.aisle_id=p.aisle_id and p.product_id=o.product_id and add_to_cart_order=1\
                      group by o.product_id,a.aisle_id order by a.aisle_id)\
                  group by aisle_id order by aisle_id")
temp1.show()
temp1.createOrReplaceTempView("arithmitis")

print("Ypologismos paronomasti")
#paronomastis pososTOy = #proionta kathe diadromoy
temp2=spark.sql("select aisle_id,count(product_id) as total_count_products from productsTable group by aisle_id order by aisle_id")
temp2.show()
temp2.createOrReplaceTempView("paronomastis")

print("Teliko apotelesma 5th query")
#pososto = arithmitis/paronomastis
fifth = spark.sql("select a.aisle,  round(100*count_products/total_count_products,2) as products_aisle_OneTimeOrdered  \
                from arithmitis as ar,paronomastis as pa,aislesTable as a\
                where pa.aisle_id=ar.aisle_id and ar.aisle_id=a.aisle_id \
                order by a.aisle")
fifth.show()

#create a report file
q5=fifth.toPandas()
q5.to_csv("/Users/elena/Desktop/project2Dataset/data/reports_plots/q5.csv",index=False)
```

Για το 5ο ερώτημα χρειάστηκε να κάνω 3 διαφορετικά sql ερωτήματα ώστε να υπολογίσω πρώτα τον αριθμητή και τον παρονομαστή του ποσοστού που μας ζητείται να υπολογίσουμε (ποσοστό προϊόντων κάθε διαδρόμου που έχουν τοποθετηθεί πρώτα στο καλάθι κάποιας παραγγελίας).

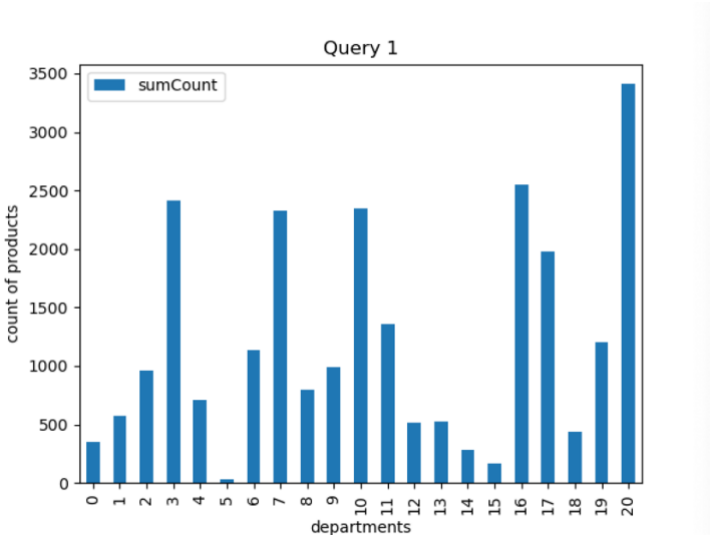
Στο 6ο ερώτημα:

Τα γραφήματα που μας ζητούνται στο 6ο ερώτημα έγιναν με αυτό το τρόπο:

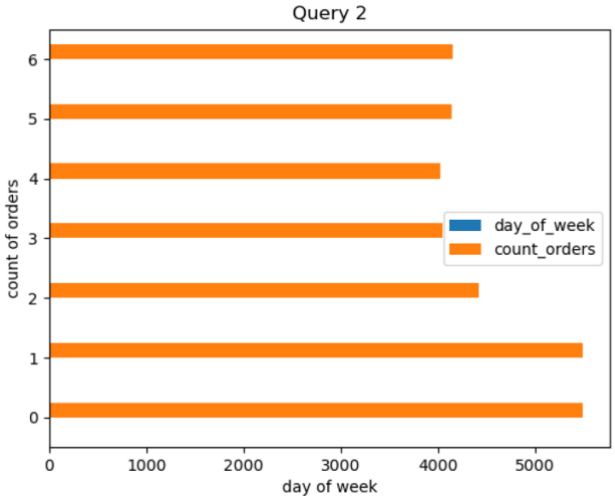
```
#6. Να δημιουργεί κατάλληλα γραφήματα (π.χ. Histogram, Pie chart) για την παρουσίαση των περιεχομένων της πρώτης
#και της δεύτερης αναφοράς (βλέπε 1 και 2).
```

```
#create plot for 1st report
q1.plot(kind="bar")
plt.xlabel("departments")
plt.ylabel("count of products")
plt.title("Query 1")
plt.savefig("data/reports_plots/q1_plot.png")
```

Και αποθηκεύτηκαν (με τη μέθοδο *savefig()*) σε αρχεία *.png* στον φάκελο reports_plots.



Για τα 21 departments (0-20) βλέπουμε το πλήθος των προϊόντων τους.



Για τις 7 ημέρες τις εβδομάδας (0-6) βλέπουμε το πλήθος των παραγγελιών.