Μάθημα:Διαχείριση μεγάλων δεδομένων - Big data management

1η εργασία

Κυριακόπουλος Βασίλης

ΑΜ 2022201800103

mail dit18103@uop.gr

Κυριακόπουλος Γιώργος

ΑΜ 2022201400112

mail dit14112@uop.gr

Σχόλια

Επειδή είναι πολλά τα αρχεία κώδικα το όνομα και ΑΜ έχουν μόνο τα αρχεία της main δηλαδή τα αρχεία με όνομα NewWordCount.java

Στο τρίτο ερώτημα υπολογίζονται και τέσσερις τύποι πελατών.

Ερώτημα 1

ΜΕΛΕΤΗ ΔΕΔΟΜΕΝΩΝ

Τα nominal δεδομένα είναι τα εξής:

```
Education, Marital_Status, Kidhome, Teenhome, Dt_Customer, Recency,
AcceptedCmp3, AcceptedCmp4, AcceptedCmp5,
AcceptedCmp1, AcceptedCmp2, Complain, Response
```

Τα numeric δεδομένα είναι τα εξής:

```
ID, Year_Birth, Income, Education, Marital_Status,
Recency, MntWines, MntFruits, MntMeatProducts, MntFishProducts,
MntSweetProducts, MntGoldProds, NumDealsPurchases, NumWebPurchases,
NumCatalogPurchases,
NumStorePurchases, NumWebVisitsMonth
```

ΠΡΟ-ΕΠΕΞΕΡΓΑΣΙΑ ΔΕΔΟΜΕΝΩΝ

Μέσα από τον κώδικα python εντοπίστηκαν nan τιμές στο csv αρχείο στην στήλη Income. Αυτές οι τιμές γίνονται 0(δεν συμμετέχουν στον υπολογισμό του μέσου ορού)συμμετέχουν σε μετέπειτα πράξεις .

Υπολογίζουμε τα outliers για τις στήλες που ζητούνται μέσω των τύπων:

Lower Bound: **(Q1 - 1.5 * IQR)**
Upper Bound: **(Q3 + 1.5 * IQR)**

Δεν διαγράφουμε τις ακραίες τιμές (outliers).Απλά δεν θα τις συμπεριλάβουμε υπόψιν αν υπολογίσουμε μέσο ορό για το Income ή Date_Birth.

Η ημερομηνίες χειρίζονται μέσα από το map reduce οπότε δεν τους κάνουμε προ επεξεργασία.

Συνοψίζοντας στο κομμάτι τις προ επεξεργασίας οι ενέργειες που έγιναν είναι οι τιμές Nan να γίνουν 0 στην στήλη Income, να αφαιρεθούν τα διπλότυπα(δεν υπάρχουν διπλότυπά σύμφωνα με το script), η διαγραφή κενών στηλών και η αλλαγή του delimiter από “;” σε “,”.Η διαδικασία αυτή γίνεται τρέχοντας το script preprocess.py.

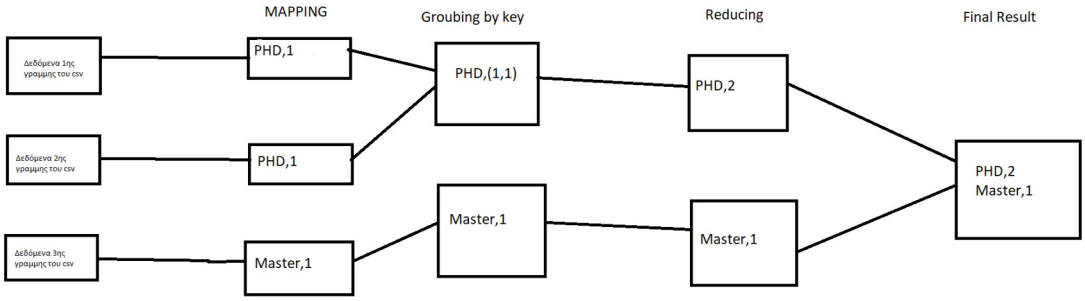
Ψευδό-κώδικας και σχηματική εκτέλεση

1ο ερώτημα

```
Map(){
        emit(education,1)
}

Reduce(key,list[]){
     count = 0
     for val in list:
          count=count+val
     emit(key,count)
}
```

Η λογική είναι στο map να στέλνουμε ως key το education ώστε να ομαδοποιήσει
με βάση το education και να στείλουμε την τιμή 1.
Στο reduce απλά προσθέτουμε τα στοιχεία της λίστας.
Έτσι στην έξοδο έχουμε το πλήθος των ατόμων ανά education



2ο ερώτημα

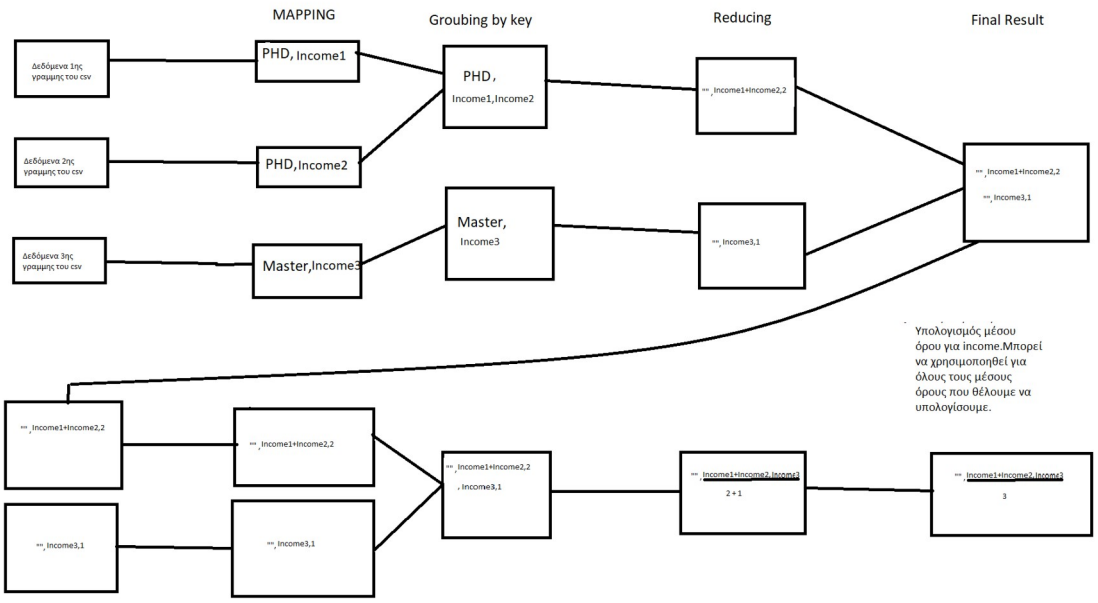
Δυο map reduce για υπολογισμό μέσου ορού για το κρασί.

```
Map(){
     Emit(education,mntwines)
}
Reduce(key,list[]){
     count = 0
     size = 0
     for val in list:
          sum=sum+val
          count=count+1

     emit(“”,(sum,count))
}
Map(){
     Emit(“”,(sum,count))
}
Reduce(key,list[]){
     count = 0
     size = 0
     for val in list:
          sum=sum+val
          count=count+1
```

```
     emit(“”,sum/count)
}
```

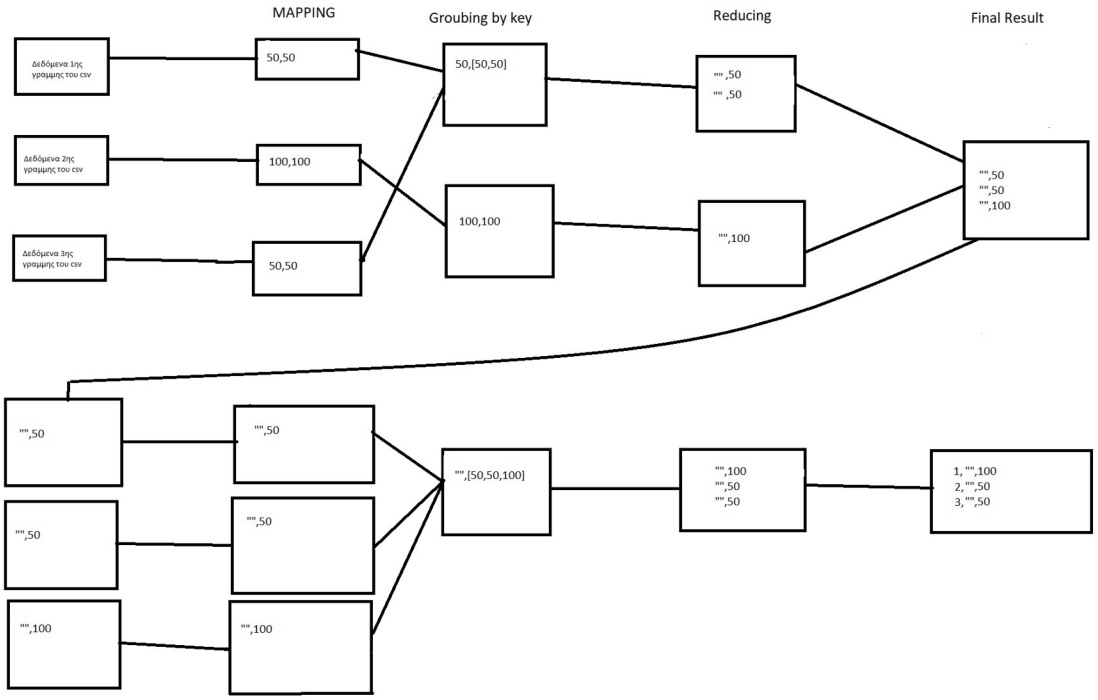
Γραφική απεικόνιση υπολογισμού μέσου όρου(για τρία στοιχεία).Το σχήμα είναι γενικό και αλλάζοντας την τιμή Income σε MntWines υπολογίζουμε τον επιθυμητό μέσο όρο




```
Map(){
     If(mntwines>mean*1.5):
          Emit(mntwines,
(id,datebirth,education,maritalstatus,income,mntwines))
}
Reduce(key,list[]){
     Sort(list)
     for i in list:
          emit(“”,
(i.id,i.datebirth,i.education,i.maritalstatus,i.income,i.mntwines))
}
```

```
Map(){
     Emit(mntwines,
(id,datebirth,education,maritalstatus,income,mntwines))
}
Reduce(key,list[]){
     Sort(list)
     Num = 1
     for i in list:
          emit(num,
(i.id,i.datebirth,i.education,i.maritalstatus,i.income,i.mntwines))
          num++
}
```

Σχηματική αναπαράστασή
Στο 1ο mapping περνάνε μόνο οι εγγραφές που ικανοποιούν την συνθήκη που ζητείται. Εδώ θεωρούμε πως όλες οι τιμές ικανοποιούν την συνθήκη για αυτό και περνάνε στο reduce
Επίσης για λόγους ευκολία ανάγνωσης περνάω ως τιμή μόνο το MntWines και όχι όλες τις τιμές που θέλει στην έξοδο το δεύτερο ερώτημα



Η λογική είναι στο 1ο map να στέλνουμε ως key το education(μπορούμε να χρησιμοποιήσουμε και κάποιο άλλο πεδίο για κλειδί δεν είναι υποχρεωτικό να είναι education) ώστε να ομαδοποιηθεί με βάση το education και την τιμή mntwines. Επιλέγουμε να στείλουμε την τιμή που μας ενδιαφέρει να υπολογίσουμε τον μέσο όρο που είναι το mntwines.Αν θέλαμε να βρούμε τον μέσο ορό για το income θα βάζαμε income.Στο 1ο reduce προσθέτουμε τις τιμές της λίστας, στέλνουμε ως key το κενό string και σαν δεδομένα το tuple(sum,count).
Στο δεύτερο map απλά στέλνουμε όλα τα στοιχεία της εξόδου του 1 reduce με ίδιο key.Αφού όλα έχουν το ίδιο key μαζεύονται όλα παρέα στο δεύτερο reduce οπού και προσθέτουμε όλα τα Sum= sum1+sum2+… και όλα τα Count=count1+count2+….Και στην έξοδο βγάζουμε το Mean = Sum/Count .

Αφού έχουμε υπολογίζει τον μέσο όρο τώρα κάνουμε 2 ακόμα map reduce για να εμφανίσουμε το τελικό αποτέλεσμα. Στο πρώτο map(έχουμε ως είσοδο τα δεδομένα) χρησιμοποιούμε ως key το mntwines στέλνουμε μόνο τα στοιχεία που έχουν mntwines>1.5*mean στο reduce.Στο reduce απλώς τα στέλνουμε στο δεύτερο map που αυτό με την σειρά του θα τα

στείλει στο δεύτερο reduce μαζεμένα όλα παρέα για να γίνει το sort.

3ο ερώτημα

Δυο map reduce για υπολογισμό μέσου ορού που ζητείται και αλλά δυο για τον υπολογισμό του μέσου όρου του income .
Έχοντας υπολογίσει τους μέσους όρους ακριβώς με τον ίδιο τρόπο με πριν κάνουμε άλλο ένα map reduce για το τελικό αποτέλεσμα.

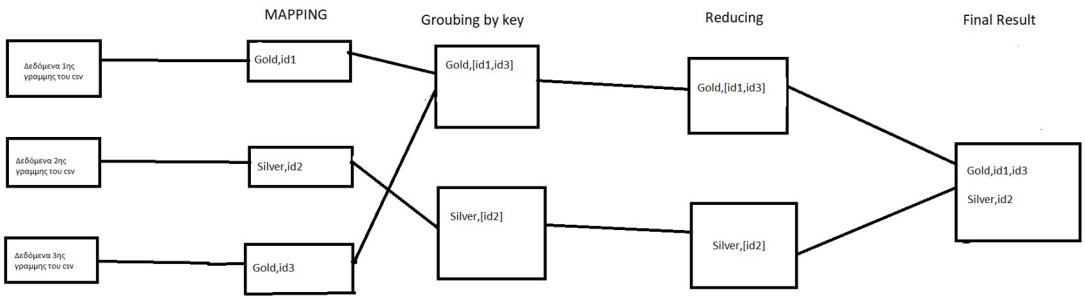
```
Map(){
      Sum = MntWines + MntFruits + MntMeatProducts +MntFishProducts + MntSweetProducts+
      MntGoldProds

      If(sum>1.5*mean_sum and date == 21 and income>69500):
            Emit(“Gold”,id)
      Else If(sum>1.5*mean_sum and date < 21 and income<69500):
            Emit(“Silver”,id)
      Else If(sum<0.25*mean_sum and date == 21 and
income<mean_income):
            Emit(“Bronze”,id)
      Else If(sum<0.25*mean_sum and date < 21 and
income<mean_income):
            Emit(“Paper”,id)

}
Reduce(key,list[]){
      text= “”;
      For i in list:
            text=text+i.id;
      emit(key,text);
}
```

Η λογική είναι ότι στο map ελέγχουμε αν ο πελάτης ανήκει σε κάποια κατηγορία αν ναι τότε στέλνουμε στο reduce το id του με key την κατηγορία που ανήκει .Στο reduce απλώς κάνουμε emit τα id που υπάρχουν στην λίστα.

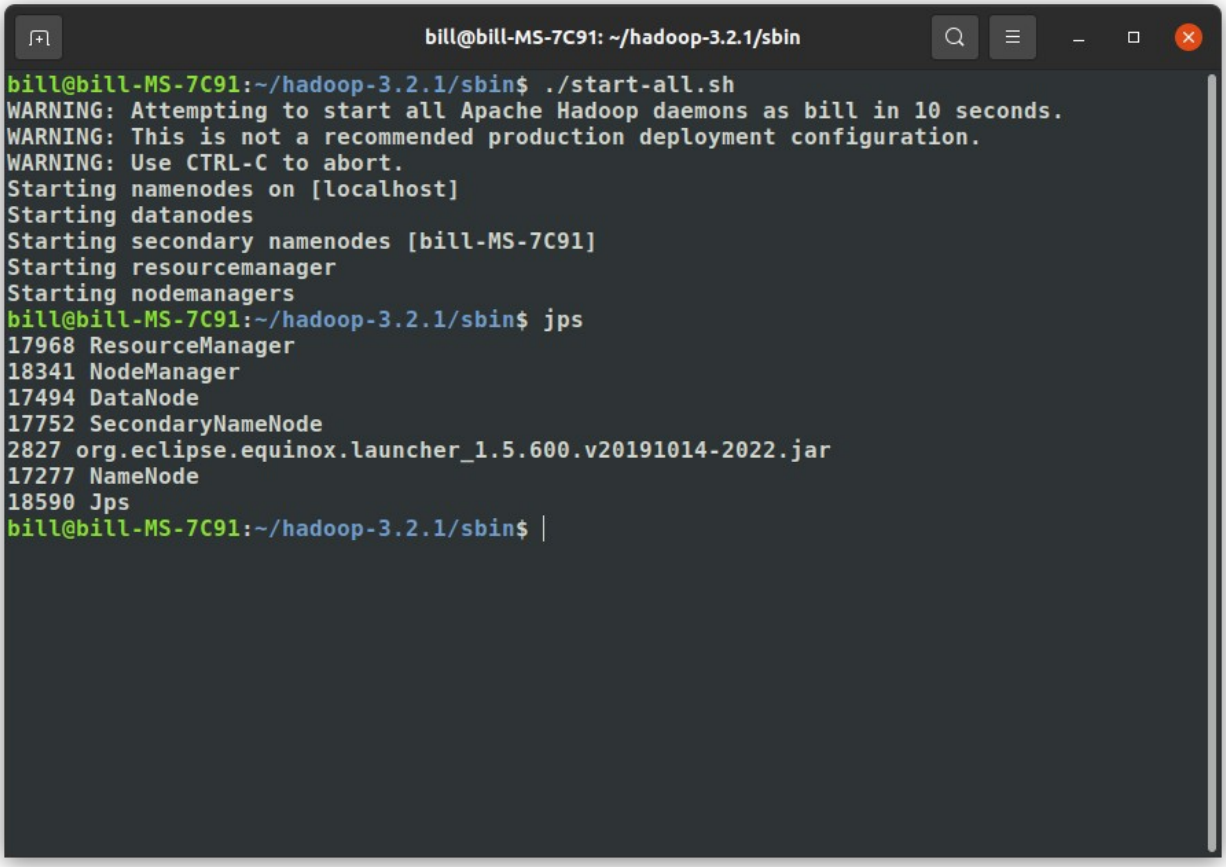
Σχηματικήαναπαράσταση είναι ίδια λογική με το 2ο ερώτημα χωρίς το δεύτερο map reduce.



Οδηγίες για την εκτέλεση του προγράμματος (user manual)

Δεδομένου ότι το Hadoop λειτουργεί .

Ενεργοποιούμαι το hadoop με την εντολή ./start-all.sh
δεδομένου ότι είμαστε στο φάκελο sbin του hadoop



Γράφουμε τον κώδικα στο eclipse δημιουργώντας ένα project και προσθέτουμε τα 3 jars που χρειαζόμαστε από το hadoop.

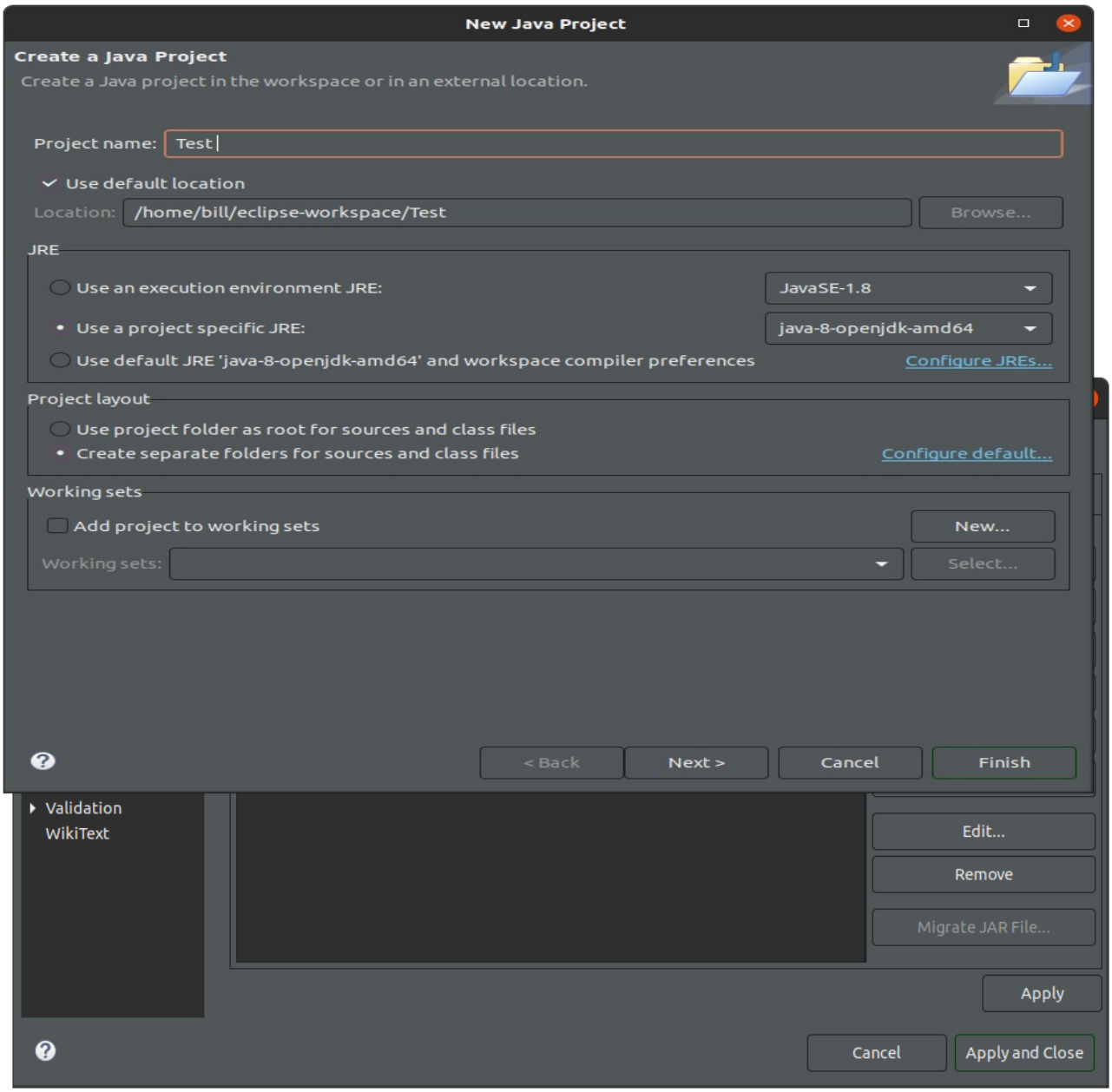
Για γίνει αυτό ακολουθούμαι τα εξής βήματα:
Πηγαίνουμε στο Project→Properties→Java Build Path→Libraries→Add External Jars

Αφου γράψουμε τον κώδικα , κάνουμε export το project σε jar file

ορίζοντας και την main κλάση μέσα απο το eclipse.

Για να εκτέλεσουμαι τον κώδικα μεσα απο το hadoop πρέπει να δώσουμε την εντολή

```
hadoop jar local/absolut/path/of/the/jar/wc1.jar
/hdfs/absolut/path/input1/ps2.csv     /r_ output1
```



Οπού το wc1.jar είναι το jar αρχείο του κώδικα και ps2.csv είναι το input αρχείο(σε αυτό το αρχείο έχει γίνει η προ-επεξεργασία) που έχουμε περάσει στο hdfs .Το /r_output1 είναι ο hdfs φάκελος που θα πάρουμε τα αποτελέσματα.
Αφού τελειώσει η εκτέλεση του map reduce μπορούμε να δούμε τα αποτελέσματα πηγαίνοντας στην διεύθυνση
http://localhost:9870/explorer.html#/

Επιλέγουμε το φάκελο του output και κατεβάζουμε το αρχείο part-r-00000 που έχει την έξοδο του hadoop.

Γραφική απεικόνιση

Τα αρχεία εξόδου για τα τρία ερωτήματα είναι
First_Query.txt
Second_Query4.txt , Third_Query5.txt
Τα αρχεία αυτά βρίσκονται στο φάκελο των αποτελεσμάτων
Για όλα τα αρχεία εξόδου η δομή τους είναι αναμενόμενη.
Επίσης υπάρχουν και αλλά αρχεία εξόδου που το καθένα είναι η έξοδος ενός map reduce.Αυτά τα αρχεία χρησιμοποιούνται για τον υπολογισμό του τελικού αρχείου.