ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ
ΕΡΓΑΣΤΗΡΙΟ ΥΠΟΛΟΓΙΣΤΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΕΞΑΜΗΝΙΑΙΑ ΕΡΓΑΣΙΑ

ΔΠΜΣ ΕΠΙΣΤΗΜΗ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ
ΑΚ. ΕΤΟΣ 2021 - 2022
ΔΙΑΧΕΙΡΙΣΗ ΔΕΔΟΜΕΝΩΝ ΜΕΓΑΛΗΣ ΚΛΙΜΑΚΑΣ

Χρήση του Apache Spark στις Βάσεις Δεδομένων

Στην παρούσα εργασία, θα γίνει χρήση του Apache Spark για τον υπολογισμό αναλυτικών ερωτημάτων πάνω σε αρχεία που περιγράφουν σύνολα δεδομένων. Το Apache Spark προσφέρει δύο βασικά APIs για την υλοποίηση αναλυτικών ερωτημάτων:

- RDD API
  - ✓ https://spark.apache.org/docs/2.4.4/rdd-programming-guide.html
- Dataframe API / Spark SQL
  - ✓ https://spark.apache.org/docs/2.4.4/sql-programming-guide.html

Στην χρήση του δευτέρου API στην εργασία συνίσταται η υλοποίηση με παραδοσιακή SQL όπως περιγράφεται στην ενότητα https://spark.apache.org/docs/2.4.4/sql-programming-guide.html#running-sql-queries-programmatically

Για τις ανάγκες της εργασίας θα χρησιμοποιηθεί ένα σύνολο δεδομένων από charts τραγουδιών του spotify, το οποίο μπορείτε να κατεβάσετε μέσω του ακόλουθου συνδέσμου:

http://www.cslab.ece.ntua.gr/~nprov/courses/spotify_data.tar.gz

Στο συμπιεσμένο αρχείο που κατεβάσατε, περιέχονται τέσσερα αρχεία κειμένου στην μορφή CSV, τα artists.csv, chart_artist_mapping.csv, charts.csv και regions.csv με συνολικό μέγεθος 2.1GB μετά την αποσυμπίεση. Για κάθε αρχείο, δίνονται ενδεικτικά στην συνέχεια κάποια rows του και μία σύντομη επεξήγηση της κάθε στήλης που υπάρχει μετά δύο διαχωριστικών («,») του CSV.

- Αρχείο charts.csv (2 GB)
  Στην θέση 0 και στην θέση 1 βρίσκεται το id και ο τίτλος του τραγουδιού αντίστοιχα. Η θέση 2 περιγράφει τη θέση του τραγουδιού στο αντίστοιχο chart (θέση 5) την αντίστοιχη ημέρα (θέση 3) για την κάθε χώρα, η οποία περιγραφέται από ένα αναγνωριστικό id (θέση 4). Οι θέσεις 6 και 7 περιγράφουν τη σχετική κίνηση του τραγουδιού στα charts και το πλήθος των streams του τραγουδιού αντίστοιχα. Σχόλια:
  - Τα διαθέσιμα charts είναι τα "top200" και τα "viral50".
  - Το πλήθος των streams είναι κενό στην περίπτωση των "viral50" charts.
  - Η σχετική κίνηση του τραγουδιού στα charts ορίζεται με τις τιμές NEW_ENTRY, MOVE_DOWN, MOVE_UP, SAME_POSITION

```
89017,Levels - Radio Edit,113,2018-04-29T00:00:00.000+03:00,46,top200,MOVE_DOWN,1449
56053,Fuego,28,2018-11-16T00:00:00.000+02:00,46,viral50,MOVE_UP,""
136298,Se Essa Bunda,23,2021-09-01T00:00:00.000+03:00,46,viral50,MOVE_UP,""
50717,Faded,48,2017-01-01T00:00:00.000+02:00,46,top200,MOVE_DOWN,2058
139601,Side To Side,124,2017-04-18T00:00:00.000+03:00,46,top200,NEW_ENTRY,1015
```

- Αρχείο chart_artist_mapping.csv (3.3 MB)
  Το αρχείο συνδέει κάθε τραγούδι από τα charts με έναν καλλιτέχνη. Στην θέση 0 δίνεται το αναγνωριστικό του τραγουδιού, ενώ στην θέση 1 δίνεται το id του καλλιτέχνη.

```
165257,54
38426,55
58125,56
139701,57
154369,58
16449,59
```

- Αρχείο regions.csv (4 KB)
  Το αρχείο περιγράφει τις χώρες για τις οποίες υπάρχουν διαθέσιμα charts, με την θέση 0 να δίνει το αναγνωριστικό της χώρας και την θέση 1 την λεκτική ονομασία της.

```
17,Egypt
19,Estonia
20,Finland
21,France
22,Germany
23,Greece
26,Hong Kong
```

- Αρχείο artists.csv (1.5 MB)
  Το αρχείο περιγράφει τους καλλιτέχνες, των οποίων τραγούδια βρίσκονται στα charts. Στην θέση 0 δίνεται το id του καλλιτέχνη και στην θέση 1 το όνομα του.

```
64321,Sam Robs
64322,Sam Rui
64323,Sam Ryder
64324,Sam Seg
64329,Sam Sillah
64330,Sam Smith
64331,Sam Sparro
64334.Sam The Kid
```

Στην εργασία θα υπολογιστούν τα αποτελέσματα για 6 ερωτήματα που παρουσιάζουν ενδιαφέρον από το διαθέσιμο σύνολο δεδομένων. Τα ερωτήματα παρουσιάζονται στον Πίνακας 1. και επιπλέον δίνονται Υποδείξεις για την σωστή υλοποίηση των ερωτημάτων στην σελίδα 6. Τα ζητούμενα του πρώτου μέρους είναι τα ακόλουθα:

Ζητούμενο 1 (5%): Φορτώστε τα 3 CSV αρχεία που σας δόθηκαν στο hdfs σε ένα φάκελο **files**.
Ζητούμενο 2 (5%): Όπως αναφέρθηκε τα δεδομένα σας δίνονται σε μορφή απλού κειμένου (csv). Παρόλα αυτά, είναι γνωστό ότι ο υπολογισμός ερωτημάτων αναλυτικής επεξεργασίας απευθείας πάνω σε αρχεία csv δεν είναι αποδοτικός. Για να βελτιστοποιηθεί η πρόσβαση των δεδομένων, παραδοσιακά οι βάσεις δεδομένων φορτώνουν τα δεδομένα σε ειδικά σχεδιασμένα binary formats. Παρ'ότι το Spark δεν είναι μια τυπική βάση δεδομένων, αλλά ένα σύστημα κατανεμημένης επεξεργασίας, για λόγους απόδοσης, υποστηρίζει κι αυτό μια παρόμοια λογική. Αντί να τρέξουμε τα ερωτήματά μας απευθείας πάνω στα csv αρχεία, μπορούμε να μετατρέψουμε πρώτα το dataset σε μια ειδική μορφή που:
- Έχει μικρότερο αποτύπωμα στη μνήμη και στον δίσκο και άρα βελτιστοποιεί το I/O (input/output) μειώνοντας τον χρόνο εκτέλεσης.
- Διατηρεί επιπλέον πληροφορία, όπως στατιστικά πάνω στο dataset, τα οποία βοηθούν στην πιο αποτελεσματική επεξεργασία του. Για παράδειγμα, αν ψάχνω σε ένα σύνολο δεδομένων τις τιμές που είναι μεγαλύτερες από 100 και σε κάθε block του dataset έχω πληροφορία για το ποια είναι η min και ποια η max τιμή, τότε μπορώ να παρακάμψω την επεξεργασία των blocks με max τιμή < 100 γλιτώνοντας έτσι χρόνο επεξεργασίας.

Το ειδικό format που χρησιμοποιούμε για να επιτύχουμε τα παραπάνω είναι το Apache Parquet. Όταν φορτώνουμε έναν πίνακα σε Parquet, αυτός μετατρέπεται κι αποθηκεύεται σε ένα columnar format που βελτιστοποιεί το I/O και τη χρήση της μνήμης κι έχει τα χαρακτηριστικά που αναφέραμε. Περισσότερες πληροφορίες σχετικά με το Parquet μπορείτε να βρείτε εδώ. Από άποψη κώδικα, η μετατροπή ενός dataset σε Parquet είναι ιδιαίτερα απλή. Παραδείγματα και πληροφορίες για το πώς διαβάζω και γράφω Parquet αρχεία μπορείτε να βρείτε εδώ και εδώ.
Στο συγκεκριμένο ερώτημα, ζητείται να μετατρέψετε κάθε ένα csv που υπάρχει στο hdfs σε Parquet μορφή, διαβάζοντας κάθε CSV σε dataframe **με σωστό σχήμα δεδομένων (με ποια παράμετρο πρέπει να διαβαστεί το csv για να έχει το σωστό σχήμα?)** και αποθηκεύοντας το στη συνέχεια σε parquet μορφή πίσω στο hdfs (συμβουλευτείτε και τις παραπάνω οδηγίες). Τελικά θα πρέπει να υπάρχουν 6 αρχεία στο hdfs, 3 CSV και 3 parquet.

Ζητούμενο 3 (80% Μονάδες): Για κάθε ερώτημα του Πίνακα 1 υλοποιείστε μία λύση με το RDD API και μία με Spark SQL, η οποία θα μπορεί να διαβάζει είτε αρχεία CSV χρησιμοποιώντας το option inferSchema είτε αρχεία Parquet. Οι μονάδες κατανέμονται ως εξής:

- Q1 – 9%
- Q2 – 9%
- Q3 – 12%
- Q4 – 15%
- Q5 – 15%
- Q6 – 20%

Ζητούμενο 4 (5% Μονάδες): Να εκτελεστούν οι υλοποιήσεις του ζητούμενου 3 για κάθε query. Συγκεκριμένα, θέλουμε τα αποτελέσματα και τους χρόνους εκτέλεσης από τις 3 ακόλουθες περιπτώσεις:

1. Map Reduce Queries – RDD API
2. Spark SQL με είσοδο το csv αρχείο (συμπεριλάβετε infer schema)
3. Spark SQL με είσοδο το parquet αρχείο

Δώστε τους χρόνους εκτέλεσης σε ένα ραβδόγραμμα, ομαδοποιημένους ανά Ερώτημα Σχολιάστε τα αποτελέσματα σε κάθε query. Για να λάβετε σωστά τους χρόνους εκτέλεσης, φροντίστε να κάνετε write το αποτέλεσμα του κάθε query σε csv στο hdfs (**σε ένα φάκελο outputs**), καθώς το spark έχει lazy evaluation, υπολογίζει ότι χρειάζεται και επομένως για αντικειμενικές μετρήσεις χρειάζεται να ζητηθεί ολόκληρο το αποτέλεσμα. Τι παρατηρείται με τη χρήση του parquet ? Γιατί δεν χρησιμοποιείται το infer schema?

Ζητούμενο 5 (5%): Το SparkSQL έχει υλοποιημένα δύο είδη ερωτημάτων συνένωσης στο DataFrame API. Συγκεκριμένα, με βάση τη δομή των δεδομένων και των υπολογισμών που θέλουμε καθώς και τις ρυθμίσεις του χρήστη, πραγματοποιεί από μόνο του κάποιες βελτιστοποιήσεις στην εκτέλεση του ερωτήματος χρησιμοποιώντας έναν βελτιστοποιητή ερωτημάτων (query optimizer), κάτι που όλες οι βάσεις δεδομένων έχουν. Μια τέτοια βελτιστοποίηση είναι ότι επιλέγει αυτόματα την υλοποίηση που θα χρησιμοποιήσει για ένα ερώτημα join λαμβάνοντας υπόψη το μέγεθος των δεδομένων και πολλές φορές αλλάζει και την σειρά ορισμένων τελεστών προσπαθώντας να μειώσει τον συνολικό χρόνο εκτέλεσης του ερωτήματος. Περισσότερες πληροφορίες για τις ρυθμίσεις βελτιστοποίησης του SparkSQL υπάρχουν εδώ.

Κάντε download το script που βρίσκεται εδώ, συμπληρώστε τις <> ώστε να μπορείτε να απενεργοποιήσετε την επιλογή του join από το βελτιστοποιητή. Εκτελέστε το συμπληρωμένο script και στο φακέλο που βρίσκονται τα αρχεία του hdfs θα βρείτε τα αποτελέσματα του benchmarking στο αρχείο "join_experiment.csv". Παρουσιάστε τα αποτελέσματα με την μορφή ενός ραβδογράμματος. Σχολιάστε τα αποτελέσματα των χρόνων εκτέλεσης σύμφωνα με τα πλάνα εκτέλεσης που παράγει ο βελτιστοποιητής στην κάθε περίπτωση. Τα πλάνα εκτέλεσης θα τυπωθούν στην οθόνη κατά την εκτέλεση του script.

Πίνακας 1: Τα ερωτήματα που ζητείται να υλοποιηθούν.

# Ερωτήματος	Λεκτική Περιγραφή
Q1	Ποιο είναι το συνολικό πλήθος των streams που έχουν καταγραφεί για το τραγούδι με τίτλο "Shape of You", σύμφωνα με τα top200 charts? Ως αποτελέσμα να δωθεί μόνο ένας αριθμός με το πλήθος
Q2	Για κάθε chart, να βρεθεί το τραγούδι με τον μεγαλύτερο μέσο χρόνο παραμονής (δείτε «Υποδείξεις») στην πρώτη θέση. Ως αποτέλεσμα, αναμένονται δύο γραμμές, μία για κάθε chart στην μορφή: *όνομα_chart, όνομα_τραγουδιού, μέσος_χρόνος_παραμονής_θέση#1* Αναμενόμενο αποτέλεσμα στο viral50 chart viral50,Calma - Remix,24.985507
Q3	Από τα top200 charts, να βρεθεί για κάθε μήνα της κάθε χρονιάς, το μέσο ημερήσιο πλήθος streams του τραγουδιού που βρίσκεται στην θέση 1 (δείτε «Υποδείξεις»), ταξινομημένα ως προς την χρονιά και τον μήνα. Ως αποτέλεσμα, αναμένεται μία γραμμή για κάθε μέρα κάθε μήνα από τα διαθέσιμα charts, η οποία θα είναι στη μορφή: *χρονιά, μήνας, μέσο_ημερήσιο_πλήθος_streams_θέση#1* Οι δύο πρώτες αναμενόμενες γραμμές του αποτελέσματος είναι: 2017,1,7618611.064516129 2017,2,8876450.785714285
Q4	Από τα viral50 charts, βρείτε για κάθε χώρα το (ή τα σε περίπτωση ισοψηφίας) τραγούδια με το μεγαλύτερο πλήθος παραμονής στο charts. Ταξινομείστε τα αποτελέσματα σας ως προς το όνομα της χώρας και το όνομα του τραγουδιού. Ως αποτέλεσμα δώστε μία γραμμή για κάθε τραγούδι κάθε χώρας στην μορφή : *χώρα, id_τραγουδιού, όνομα_τραγουδιού, πλήθος_παραμονής_στο_viral50* Ενδεικτικά αποτελέσματα: France,26355,Calma - Remix,189 Germany,131808,Roses - Imanbek Remix,225 Greece,36928,De Me Theloun,211 Greece,143230,Someone You Loved,211
Q5	Σύμφωνα με τα top200, βρείτε σε κάθε χρονιά τον καλλιτέχνη με το μεγαλύτερο μέσο πλήθος streams. Ταξινομείστε ως προς τη χρονιά. Ως αποτελέσμα, δώστε για κάθε χρονιά μία γραμμή στην εξής μορφή *χρονιά, όνομα_καλλιτέχνη, μέσο_πλήθος_streams* Ενδεικτικά αποτελέσματα: 2017,Ed Sheeran,62263262.666667 2018,Post Malone,68126958.681159
Q6	Για την Ελλάδα, βρείτε για κάθε χρονιά και chart τον καλλιτέχνη (ή τους καλλιτέχνες) που έχει (έχουν) παραμείνει διαδοχικές ημέρες περισσότερες φορές στο #1 κάποιο από τα τραγούδια του. Ταξινομείστε ως προς το chart και τη χρονιά. Ως αποτελέσμα δώστε μία γραμμή για κάθε καλλιτέχνη κάθε χρονιάς στη μορφή: όνομα_chart, χρονιά, όνομα καλλιτέχνη, ημέρες_διαδοχικής_παραμονής_#1 Ενδεικτικά αποτελέσματα: viral50,2017,21 Savage,13 viral50,2017,Post Malone,13 viral50,2018,Gigi D'Agostino,29 viral50,2018,Dynoro,29

Υποδείξεις:

1. Για τα ερωτήματα q1-q3 δεν χρειάζονται joins παρά μόνο ο πίνακας charts.
2. Στο RDD API για το charts αρχείο, χωρίστε σε tuple δεδομένων την κάθε γραμμής του αρχείου μέσω του μετασχηματισμού :

   map(lambda x : list(csv.reader([x], delimiter=',', quotechar='"'))[0])

   καθώς κάποιοι τίτλοι τραγουδιών περιέχουν «,», το οποίο πρέπει να αγνοηθεί όταν θα χωριστεί η γραμμή σε πεδία.
3. **Query 2:** Βρείτε τον μέσο χρόνο παραμονής ως εξής: Βρείτε για κάθε τραγούδι σε κάθε χώρα και chart το πλήθος παραμονής του στην θέση 1, και ο μέσος χρόνος παραμονής στη θέση 1, είναι το μέσο πλήθος ως προς τις χώρες για κάθε chart, δηλαδή το σύνολο των παραμονών στην πρώτη θέση ενός τραγουδιού ως προς τις χώρες διαιρεμένο με το πλήθος των χωρών. Μην μετρήσετε τις χώρες στον οποίων τα charts εμφανίζεται ένα τραγούδι, αλλά διαιρέστε με το συνολικό πλήθος των διαθέσιμων χωρών, που είναι 69 (δεν χρειάζεται να το μετρήσετε με κώδικα).
4. **Query 3:** Ως μέσο ημερήσιο πλήθος streams τραγουδιών στη θέση 1, θεωρούμε ότι σε κάθε μέρα βρίσκουμε το σύνολο των streams που αντιστοιχούν στη θέση 1, ανεξαρτήτως χώρας, και στην συνέχεια, βρίσκουμε το μέσο όρο ως προς τις ημέρες του μήνα.
5. **Query 5:** Ως μέσο πλήθος streams ενός καλλιτέχνη, θεωρούμε το συνολικό άθροισμα των streams τραγουδιών που συμμετείχε στην χρονιά σε όλες τις χώρες, διαιρεμένο με το πλήθος των χωρών, που δίνεται ότι είναι 69.
6. **Query 6:** Εκμεταλλευτείτε την μεταβλητή που περιγράφει την κίνηση του τραγουδιού στα charts για να λύσετε το query.

Σχόλια αναφορικά με την εργασία

1. Η εργασία να εκπονηθεί σε ομάδες το πολύ των **2 ατόμων.**
2. Ως ημερομηνία παράδοσης της εργασίας ορίζεται η **Κυριακή 24 Ιουλίου 2022**.
3. Η εργασία αποτελεί το **40%** του συνολικού βαθμού του μαθήματος. Για να υπολογιστεί ο βαθμός της εργασίας, η κάθε ομάδα θα πρέπει να περάσει επιτυχώς την **υποχρεωτική** προφορική εξέταση στο αντικείμενο της εργασίας. Η εξέταση θα γίνει μετά την παράδοση της εργασίας και θα αναρτηθεί σχετικό πρόγραμμα αφού ολοκληρωθεί η υποβολή των εργασιών. Σημειώνεται επίσης ότι η παράδοση και η εξέταση της εργασίας είναι υποχρεωτικά ώστε να προκύψει προβιβάσιμος βαθμός στο μάθημα συνολικά.
4. Απορίες για την εργασία θα γίνονται στο forum που δημιουργήθηκε στην σελίδα του μαθήματος στο helios.
5. Ως παραδοτέο να υποβληθεί ένα **zip** αρχείο με όνομα τους ΑΜ των μελών της ομάδας χωρισμένα με κάτω παύλα (ή το ΑΜ του φοιτητή σε περίπτωση μονομελούς ομάδας), π.χ. 00000000_00000001.zip ή 00000000.zip (ανάλογα με το πλήθος των ατόμων της ομάδας). Το συμπιεσμένο αρχείο θα περιέχει τα ακόλουθα
   I. η public_ip του okeanos σε ένα αρχείο με όνομα **ip** (στόχος είναι να ελεγχθεί ότι υπάρχουν τα αρχεία της εργασίας στο hdfs). Στο hdfs υπενθυμίζεται ότι θα πρέπει να υπάρχουν δύο φακέλοι, ένας με όνομα **files** και ένας με όνομα **outputs** που ο πρώτος θα περιέχει τα αρχεία που σας δόθηκαν και ο δεύτερος τα αποτελέσματα των ερωτημάτων που υλοποιήσατε.
   II. μία σύντομη αναφορά (**αυστηρά με όσα ζητούνται στην εκφώνηση**) σε **pdf** η οποία θα περιέχει αποκλειστικά της απαντήσεις στις ερωτήσεις που παρατίθενται, τα σχετικά διαγράμματα και ψευδοκώδικα σε Map Reduce που να περιγράφει τις υλοποιήσεις σας για το κάθε ερώτημα με το RDD API.
   III. ένα φάκελο output με τα αποτελέσματα των ερωτημάτων τόσο μέσω του Map Reduce όσο και μέσω του SparkSQL.
   IV. ένα φάκελο code που θα περιέχει όλους τους κώδικες που έχετε υλοποιήσει, όπως και τον δοσμένο κώδικα για το μέρος 2 συμπληρωμένο με τις κατάλληλες τιμές.