Εθνικό Μετσόβιο Πολυτεχνείο

Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

Προχωρημένα Θέματα Βάσεων Δεδομένων

Κυριακή Καρατζούνη el20634

Βικέντιος Βιτάλης el18803

Ομάδα 17

Αναφορά

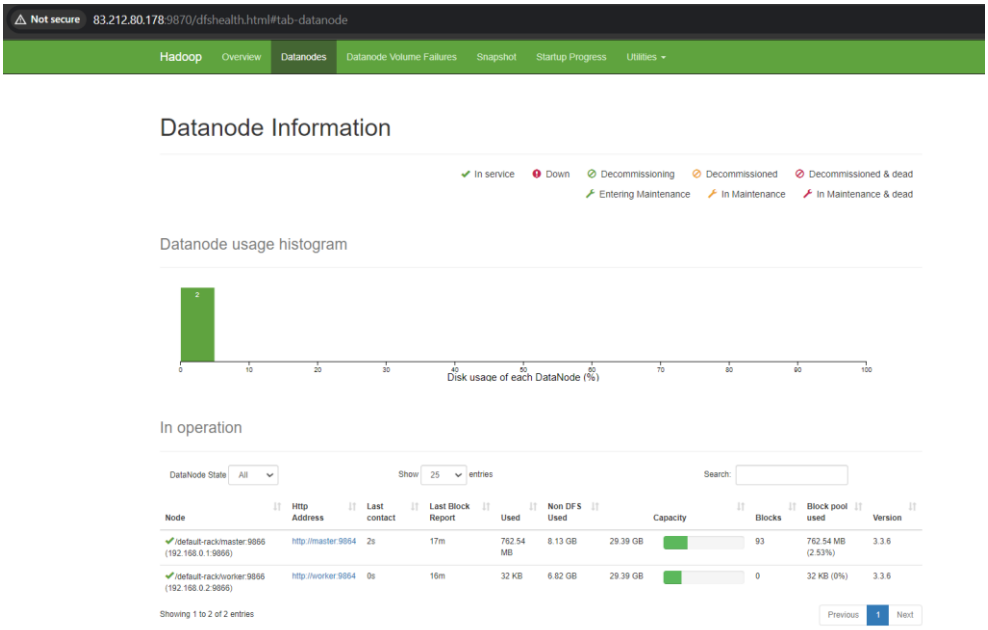
Github repository:
https://github.com/VikentiosVitalis/advanced_topics_in_database_systems

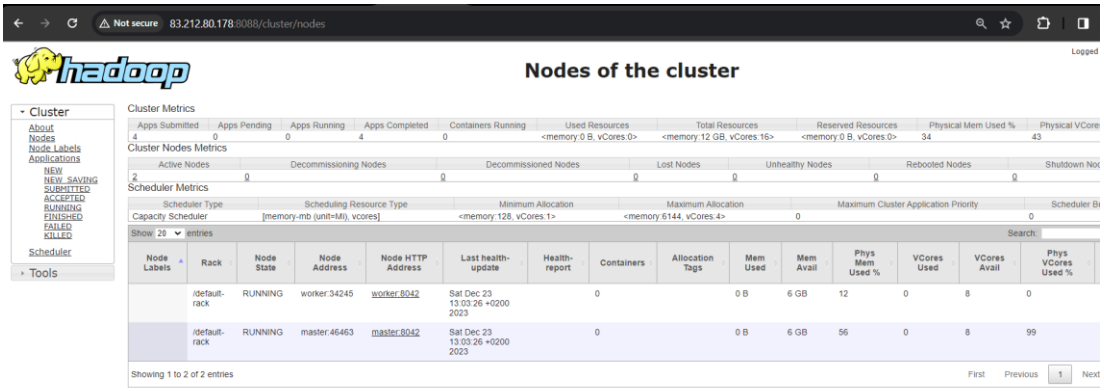
Ζητούμενο 1. Αρχικά δημιουργήσαμε στην υπηρεσία Okeanos Knossos ένα δίκτυο (cluster) 2 κόμβων σύμφωνα με τον εργαστηριακό οδηγό ‘’Advanced Topics in Database Systems: Lab guide.ipynb’’ κι εγκαταστήσαμε το λογισμικό και στους δύο κόμβους. Μέσω του WinSCP συνδεθήκαμε στον master node κι από την επιφάνεια εργασίας των Windows μεταφορτώσαμε τα σύνολα δεδομένων. Παρατίθονται τα UIs από τις υπηρεσίες HDFS, YARN και Spark History Server αντίστοιχα:

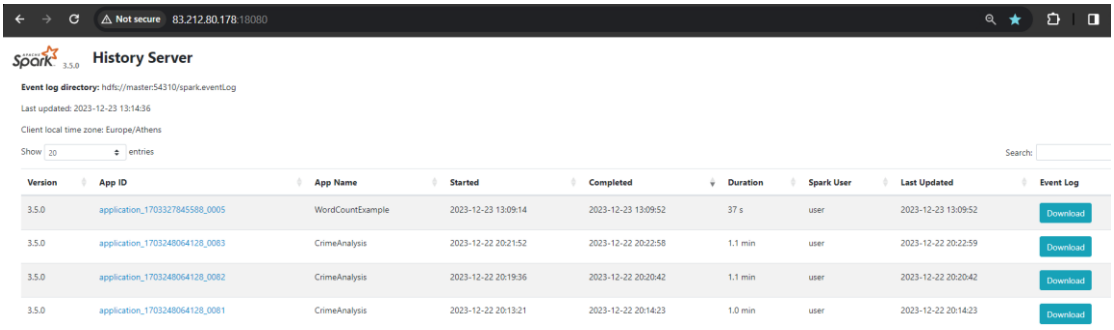
http://83.212.80.178:9870/dfshealth.html#tab-datanode

http://83.212.80.178:8088/cluster/nodes

http://83.212.80.178:18080/

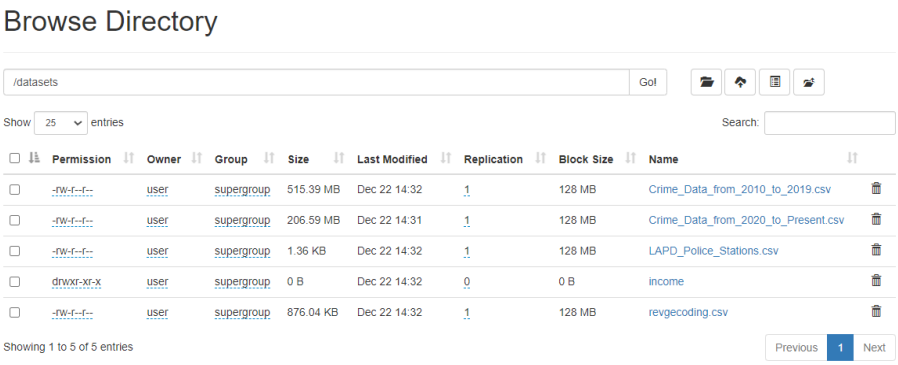






Με τις παρακάτω εντολές μεταφορτώνουμε τα αρχεία μας στην HDFS υπηρεσία που είναι διαθέσιμη μεταβένοντας στα Utilities > Browse the file system

- hadoop fs -mkdir hdfs://master:54310/datasets
- hadoop fs -mkdir hdfs://master:54310/datasets/income
- hadoop fs -put datasets/Crime_Data_from_2010_to_2019.csv hdfs://master:54310/datasets/.
- hadoop fs -put datasets/Crime_Data_from_2020_to_Present.csv hdfs://master:54310/datasets/.
- hadoop fs -put datasets/revgecoding.csv hdfs://master:54310/datasets/.
- hadoop fs -put datasets/LAPD_Police_Stations.csv hdfs://master:54310/datasets/.
- hadoop fs -put datasets/income/LA_income_2015.csv hdfs://master:54310/datasets/income/.
- hadoop fs -put datasets/income/LA_income_2017.csv hdfs://master:54310/datasets/income/.
- hadoop fs -put datasets/income/LA_income_2019.csv hdfs://master:54310/datasets/income/.
- hadoop fs -put datasets/income/LA_income_2021.csv hdfs://master:54310/datasets/income/.

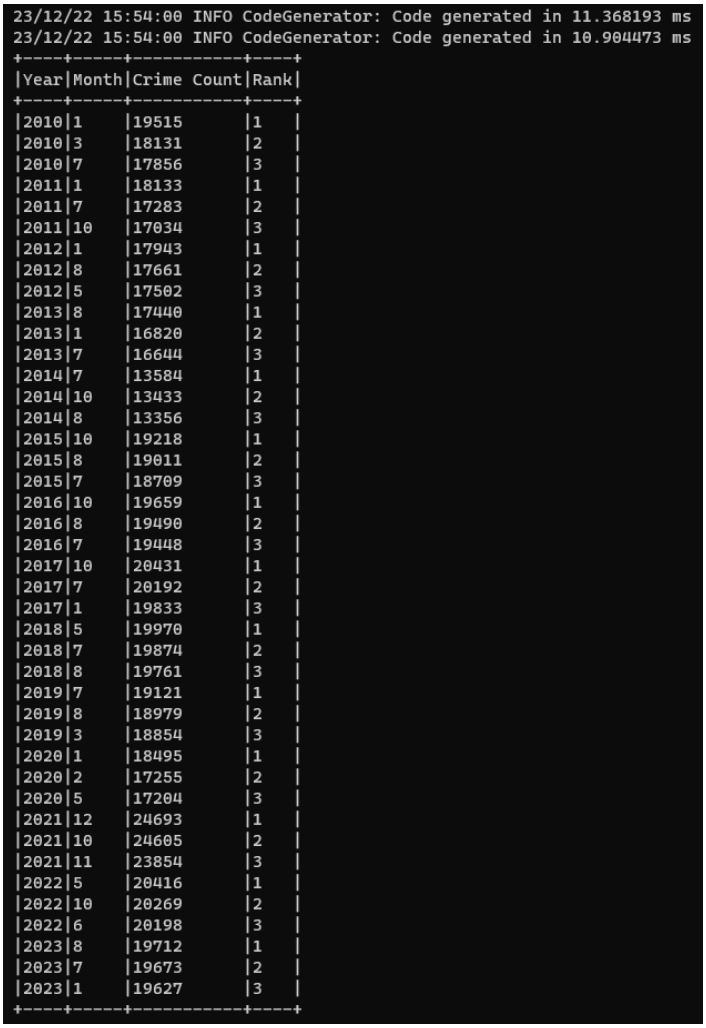


Ζητούμενο 2. Στο αρχείο queries/dataframe.py του github repository βρίσκεται η υλοποίηση του ερωτήματος σε python. Αρχικοποιούμε το Spark session, διαβάζουμε το αρχείο Crime_Data_from_2010_to_2019.csv, μετατρέπουμε τις στήλες στους αντίστοιχους τύπους δεδομένων, τυπώνουμε τους τύπους δεδομένων κάθε στήλης και τις συνολικές γραμμές. Παρακάτω φαίνεται η έξοδος στο Apache Spark περιβάλλον μετά την εκτέλεση του script μέσω spark-submit dataframe.py.

```
root
 |-- DR_NO: integer (nullable = true)
 |-- Date Rptd: date (nullable = true)
 |-- DATE OCC: date (nullable = true)
 |-- TIME OCC: integer (nullable = true)
 |-- AREA : integer (nullable = true)
 |-- AREA NAME: string (nullable = true)
 |-- Rpt Dist No: integer (nullable = true)
 |-- Part 1-2: integer (nullable = true)
 |-- Crm Cd: integer (nullable = true)
 |-- Crm Cd Desc: string (nullable = true)
 |-- Mocodes: string (nullable = true)
 |-- Vict Age: integer (nullable = true)
 |-- Vict Sex: string (nullable = true)
 |-- Vict Descent: string (nullable = true)
 |-- Premis Cd: integer (nullable = true)
 |-- Premis Desc: string (nullable = true)
 |-- Weapon Used Cd: integer (nullable = true)
 |-- Weapon Desc: string (nullable = true)
 |-- Status: string (nullable = true)
 |-- Status Desc: string (nullable = true)
 |-- Crm Cd 1: integer (nullable = true)
 |-- Crm Cd 2: integer (nullable = true)
 |-- Crm Cd 3: integer (nullable = true)
 |-- Crm Cd 4: integer (nullable = true)
 |-- LOCATION: string (nullable = true)
 |-- Cross Street: string (nullable = true)
 |-- LAT: double (nullable = true)
 |-- LON: double (nullable = true)
```

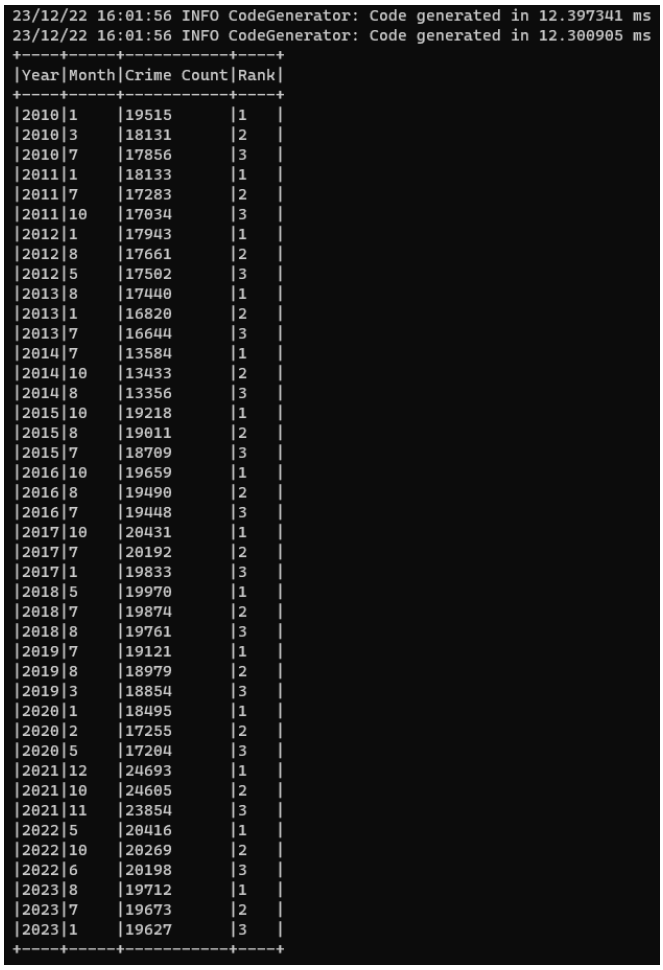
```
Total Number of Rows: 2135657
```

Ζητούμενο 3. Προκειμένου να υλοποιήσουμε το Query 1 με χρήση DataFrame, δημιουργούμε μια περίοδο Spark με 4 executors, δεικτοδοτούμε τα file paths με στόχο την ενοποίηση των δεδομένων εγκλημάτων για όλα τα διαθέσιμα έτη. Μετατρέπουμε την στήλη ‘DATE OCC’ σε τύπο δεδομένου datetime και εξάξουμε χρόνο και μήνα. Ομαδοποιούμε με βάση το πλήθος των εγκλημάτων και βρίσκουμε τους 3 μήνες με τα περισσότερα εγκλήματα για κάθε χρονιά. Τυπώνουμε τα αποτελέσματα και σταματάμε την περίοδο Spark. Εκτελούμε το Query μέσω της εντολής spark-submit q1df.py. Παρακάτω φαίνεται το αποτέλεσμα του Query1 χρησιμοποιώντας DataFrame μέσα από το Apache Spark περιβάλλον.



```
23/12/22 15:54:00 INFO CodeGenerator: Code generated in 11.368193 ms
23/12/22 15:54:00 INFO CodeGenerator: Code generated in 10.904473 ms
+----+-----+-----------+----+
|Year|Month|Crime Count|Rank|
+----+-----+-----------+----+
|2010|1    |19515      |1   |
|2010|3    |18131      |2   |
|2010|7    |17856      |3   |
|2011|1    |18133      |1   |
|2011|7    |17283      |2   |
|2011|10   |17034      |3   |
|2012|1    |17943      |1   |
|2012|8    |17661      |2   |
|2012|5    |17502      |3   |
|2013|8    |17440      |1   |
|2013|1    |16820      |2   |
|2013|7    |16644      |3   |
|2014|7    |13584      |1   |
|2014|10   |13433      |2   |
|2014|8    |13356      |3   |
|2015|10   |19218      |1   |
|2015|8    |19011      |2   |
|2015|7    |18709      |3   |
|2016|10   |19659      |1   |
|2016|8    |19490      |2   |
|2016|7    |19448      |3   |
|2017|10   |20431      |1   |
|2017|7    |20192      |2   |
|2017|1    |19833      |3   |
|2018|5    |19970      |1   |
|2018|7    |19874      |2   |
|2018|8    |19761      |3   |
|2019|7    |19121      |1   |
|2019|8    |18979      |2   |
|2019|3    |18854      |3   |
|2020|1    |18495      |1   |
|2020|2    |17255      |2   |
|2020|5    |17204      |3   |
|2021|12   |24693      |1   |
|2021|10   |24605      |2   |
|2021|11   |23854      |3   |
|2022|5    |20416      |1   |
|2022|10   |20269      |2   |
|2022|6    |20198      |3   |
|2023|8    |19712      |1   |
|2023|7    |19673      |2   |
|2023|1    |19627      |3   |
+----+-----+-----------+----+
```

Εν συνεχεία, έχουμε το Query 1 με χρήση SQL API. Δημιουργούμε περίοδο Spark με 4 executors, τοποθετούμε τα csv αρχεία σε Spark Data Frames, τα ενωποιούμε, μετατρέπουμε την στήλη ‘DATE OCC’ σε τύπο δεδομένου datetime, καταχωρούμε το Data Frame σε ένα προσωρινό SQL view, γράφουμε το SQL Query για την επεξεργασία δεδομένων, το εκτελούμε, τυπώνουμε το αποτέλεσμα και σταματάμε την περίοδο Spark. Παρακάτω φαίνεται το αποτέλεσμα του Query 1 χρησιμοποιώντας SQL API, μέσα από το Apache Spark περιβάλλον.



```
23/12/22 16:01:56 INFO CodeGenerator: Code generated in 12.397341 ms
23/12/22 16:01:56 INFO CodeGenerator: Code generated in 12.300905 ms
+----+-----+-----------+----+
|Year|Month|Crime Count|Rank|
+----+-----+-----------+----+
|2010|1    |19515      |1   |
|2010|3    |18131      |2   |
|2010|7    |17856      |3   |
|2011|1    |18133      |1   |
|2011|7    |17283      |2   |
|2011|10   |17034      |3   |
|2012|1    |17943      |1   |
|2012|8    |17661      |2   |
|2012|5    |17502      |3   |
|2013|8    |17440      |1   |
|2013|1    |16820      |2   |
|2013|7    |16644      |3   |
|2014|7    |13584      |1   |
|2014|10   |13433      |2   |
|2014|8    |13356      |3   |
|2015|10   |19218      |1   |
|2015|8    |19011      |2   |
|2015|7    |18709      |3   |
|2016|10   |19659      |1   |
|2016|8    |19490      |2   |
|2016|7    |19448      |3   |
|2017|10   |20431      |1   |
|2017|7    |20192      |2   |
|2017|1    |19833      |3   |
|2018|5    |19970      |1   |
|2018|7    |19874      |2   |
|2018|8    |19761      |3   |
|2019|7    |19121      |1   |
|2019|8    |18979      |2   |
|2019|3    |18854      |3   |
|2020|1    |18495      |1   |
|2020|2    |17255      |2   |
|2020|5    |17204      |3   |
|2021|12   |24693      |1   |
|2021|10   |24605      |2   |
|2021|11   |23854      |3   |
|2022|5    |20416      |1   |
|2022|10   |20269      |2   |
|2022|6    |20198      |3   |
|2023|8    |19712      |1   |
|2023|7    |19673      |2   |
|2023|1    |19627      |3   |
+----+-----+-----------+----+
```

Παρατηρώντας τους χρόνους εκτέλεσης, συμπεραίνουμε ότι οι υλοποίησεις DataFrame API και SQL API είναι πολύ κοντινές από άποψη αποδοτικότητας, με την DataFrame API να πετυχαίνει ελαφρώς καλύτερο χρόνο. Αυτό συμβαίνει διότι στην περίπτωση μας το σύνολο των δεδομένων επεξεργάζεται σχεδόν εφάμιλλα κι από τα δυο APIs.

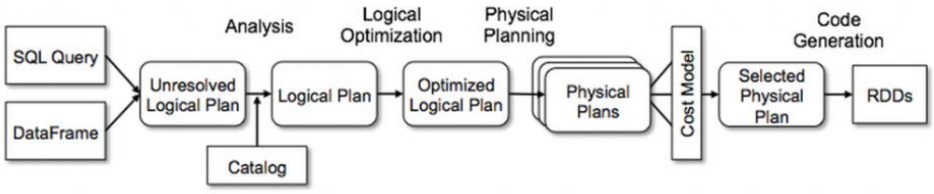
Ζητούμενο 4. Για την υλοποίηση του Query 2 χρησιμοποιώντας DataFrame δημιουργούμε μια περίοδο Spark, φτιάχνουμε μια συνάρτηση κατηγοριοποίησης της ημέρας σε Πρωϊ, Μεσημέρι, Απόγευμα και Βράδυ, καταχωρούμε την συνάρτηση που φτιάξαμε ως ορισμένη από τον χρήστη, διαβάζουμε τα αρχεία και τα ενωποιούμε, τα τοποθετούμε σε Data Frames, εφαρμόζουμε την συνάρτηση classify_time_segment. Παρακάτω φαίνεται η υλοποίηση του Query2 χρησιμοποιώντας DataFrame.

```
23/12/22 17:50:39 INFO CodeGenerator: Code generated in 23.693084 ms
23/12/22 17:50:39 INFO CodeGenerator: Code generated in 9.457702 ms
+-----------+-----------+
|Day Segment|Crime Count|
+-----------+-----------+
|      Night|     236730|
|    Evening|     186581|
|  Afternoon|     147622|
|    Morning|     123319|
|  Undefined|         85|
+-----------+-----------+
```

Εν συνεχεία, έχουμε το Query 2 με χρήση RDD API. Δημιουργούμε περίοδο Spark, φτιάχνουμε μια συνάρτηση κατηγοριοποίησης της ημέρας σε Πρωϊ, Μεσημέρι, Απόγευμα και Βράδυ, διαβάζουμε τα αρχεία σε RDDs και τα ενωποιούμε, χωρίζουμε κάθε γραμμή σε στήλες, εφαρμόζουμε τη συνάρτηση και φιλτράρουμε τα εγκλήματα δρόμου. Παρακάτω φαίνεται η υλοποίηση του Query2 χρησιμοποιώντας RDD API.

```
23/12/22 17:56:17 INFO CodeGenerator: Code generated in 16.85122 ms
23/12/22 17:56:17 INFO BlockManagerInfo: Removed broadcast_5_piece0 on
23/12/22 17:56:17 INFO BlockManagerInfo: Removed broadcast_5_piece0 on
+----------+----------+
|DaySegment|CrimeCount|
+----------+----------+
| Afternoon|    126104|
|     Night|    205343|
|   Morning|    107570|
|   Evening|    165412|
| Undefined|        75|
+----------+----------+
```

Παρατηρώντας τους χρόνους εκτέλεσης, όταν χρησιμοποιούμε τα DataFrame, ο κώδικας ειναι πιο αποδοτικός. Αυτό συμβαίνει επειδή τα DataFrames στο Spark είναι χτισμένα πάνω στη μηχανή Spark SQL, η οποία χρησιμοποιεί τον βελτιστοποιητή Catalyst. Επιπλέον, τα DataFrames βελτιστοποιούν τη χρήση μνήμης για δομημένα δεδομένα σε σύγκριση με τα RDDs. Αυτό έχει ως αποτέλεσμα καλύτερες επιδόσεις, ειδικά για μεγάλα σύνολα δεδομένων.



Ζητούμενο 5. Για την υλοποίηση του Query 3 δημιουργούμε περίοδο Spark με 2 executors, φορτώνουμε και διαβάζουμε τα αρχεία, μετατρέπουμε τα δεδομένα

εισοδήματος σε αριθμητικά αφού αφαιρέσουμε το σύμβολο του δολαρίου και τα κόμματα, φιλτράρουμε μόνο τα δεδομένα για το 2015 και αποκλείουμε τις περιπτώσεις χωρίς καταγωγή θύματος. Έπειτα, κάνουμε MAP τα LAT και LON σε ZIP Codes, εντοπίζουμε τα 3 ZIP Codes με το υψηλότερο και χαμηλότερο εισόδημα και δημιουργούμε μια συνάρτηση, ώστε να κάνουμε την αντιστοίχηση των γραμμάτων με τις καταγωγές. Εμφανίζουμε τα αποτελέσματα και επαναλαμβάνουμε την ίδια διαδικασία χρησιμοποιώντας 3 και 4 executors. Παρακάτω φαίνεται η έξοδος κι ο χρόνος του Query3 χρησιμοποιώντας 2 Spark executors.

```
23/12/23 16:16:41 INFO CodeGenerator: Code generated in 13.028926 ms
23/12/23 16:16:41 INFO CodeGenerator: Code generated in 26.231914 ms
+--------------------+-----+
|        Vict Descent|count|
+--------------------+-----+
|Hispanic/Latin/Me...| 1053|
|               White|  610|
|               Black|  349|
|               Other|  272|
|             Unknown|   71|
|          Other Asian|   46|
|              Korean|    4|
|American Indian/A...|    1|
|             Chinese|    1|
+--------------------+-----+
```

Παρακάτω φαίνεται η έξοδος κι ο χρόνος του Query3 χρησιμοποιώντας 3 Spark executors.

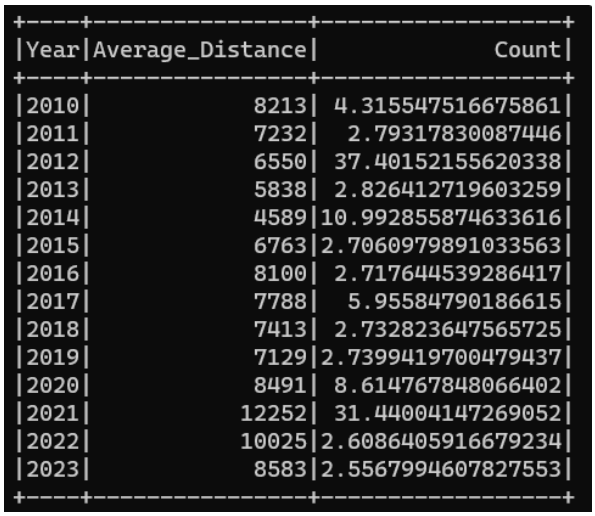
```
23/12/23 16:06:21 INFO CodeGenerator: Code generated in 9.0049 ms
23/12/23 16:06:21 INFO CodeGenerator: Code generated in 17.301408 ms
+--------------------+-----+
|        Vict Descent|count|
+--------------------+-----+
|Hispanic/Latin/Me...| 1053|
|               White|  610|
|               Black|  349|
|               Other|  272|
|             Unknown|   71|
|          Other Asian|   46|
|              Korean|    4|
|American Indian/A...|    1|
|             Chinese|    1|
+--------------------+-----+
```

Παρακάτω φαίνεται η έξοδος κι ο χρόνος του Query3 χρησιμοποιώντας 4 Spark executors.

```
23/12/22 16:59:10 INFO CodeGenerator: Code generated in 10.904093 ms
23/12/22 16:59:10 INFO CodeGenerator: Code generated in 8.980585 ms
+--------------------+-----+
|        Vict Descent|count|
+--------------------+-----+
|Hispanic/Latin/Me...| 1053|
|               White|  610|
|               Black|  349|
|               Other|  272|
|             Unknown|   71|
|         Other Asian|   46|
|              Korean|    4|
|             Chinese|    1|
|American Indian/A...|    1|
+--------------------+-----+
```

Παρατηρούμε ότι η υλοποίηση με 4 executors είναι πιο αποδοτική και γρηγορότερη, αφού όσο περισσότερους εκτελεστές έχουμε, τόσο περισσότερες εργασίες μπορούν να εκτελούνται παράλληλα.

Ζητούμενο 6. Για την υλοποίηση του Query 4a δημιουργούμε περίοδο Spark, φορτώνουμε, διαβάζουμε και ενώνουμε τα αρχεία, φιλτράρουμε τα δεδομένα, ώστε να περιλαμβάνονται μόνο περιστατικά που αφορούν πυροβόλα όπλα. Έπειτα, φορτώνουμε τα δεδομένα των αστυνομικών σταθμών και δημιουργούμε ένα λεξικό που να απεικονίζει τις περιοχές των αστυνομικών τμημάτων στις συντεταγμένες τους. Μετά, ορίζουμε τον τύπο Harvesine, ο οποίος υπολογίζει την απόσταση μεγάλου κύκλου μεταξύ δύο σημείων στην επιφάνεια της γης κι ορίζουμε μια συνάρτηση που χρησιμοποιεί τον τύπο Harvesine για τον υπολογισμό της απόστασης μεταξύ δύο συντεταγμένων. Τελικά, υπολογίζουμε την απόσταση από την τοποθεσία κάθε εγκλήματος που σχετίζεται με πυροβόλο όπλο έως το πλησιέστερο αστυνομικό τμήμα και εμφανίζουμε τα επιθυμητά αποτελέσματα. Παρακάτω φαίνεται η υλοποίηση του Query4a χρησιμοποιώντας DataFrame.



Year	Average_Distance	Count
2010	8213	4.315547516675861
2011	7232	2.79317830087446
2012	6550	37.40152155620338
2013	5838	2.826412719603259
2014	4589	10.992855874633616
2015	6763	2.7060979891033563
2016	8100	2.717644539286417
2017	7788	5.95584790186615
2018	7413	2.732823647565725
2019	7129	2.7399419700479437
2020	8491	8.614767848066402
2021	12252	31.44004147269052
2022	10025	2.6086405916679234
2023	8583	2.5567994607827553

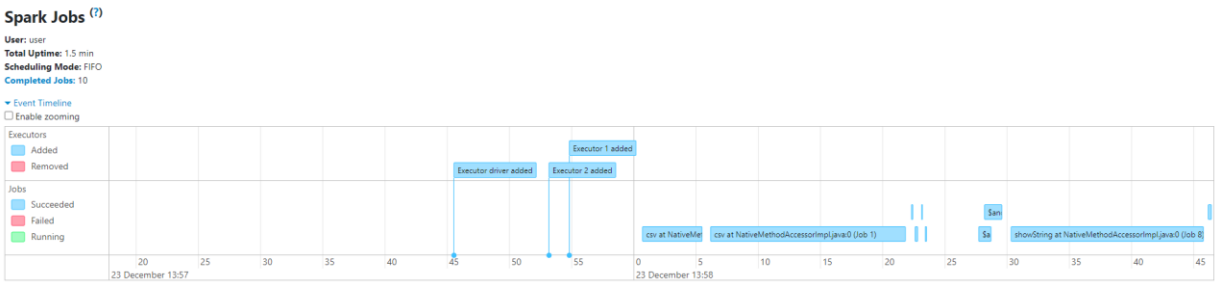
Αντίστοιχα, για την υλοποίηση του Query 4b δημιουργούμε περίοδο Spark, φορτώνουμε, διαβάζουμε και ενώνουμε τα αρχεία, φιλτράρουμε τα δεδομένα, ώστε να περιλαμβάνονται μόνο περιστατικά που αφορούν πυροβόλα όπλα. Η συνάρτηση haversine τροποποιείται για να χειρίζεται τιμές None και αποτελέσματα NaN (Not a Number). Χρησιμοποιούμε τη στήλη AREA NAME από το DataFrame firearm_crimes και τη στήλη DIVISION από το DataFrame police_stations για την ένωση. Ειδικότερα, μετατρέπουμε τη στήλη DIVISION σε κεφαλαία γράμματα για να ταιριάζει με τη μορφή του AREA NAME. Τώρα, ο υπολογισμός της απόστασης χρησιμοποιεί απευθείας το γεωγραφικό πλάτος (LAT) και το γεωγραφικό μήκος (LON) από το πλαίσιο δεδομένων firearm_crimes DataFrame και τις συντεταγμένες (Y, X) από το πλαίσιο δεδομένων police_stations DataFrame. Τα δεδομένα ομαδοποιούνται με βάση το 'AREA NAME' και τα αποτελέσματα εμφανίζονται με φθίνουσα σειρά. Παρακάτω φαίνεται η έξοδος του Query4b χρησιμοποιώντας DataFrame.

```
23/12/22 20:22:58 INFO CodeGenerator: Code generated in 13.393613 ms
23/12/22 20:22:58 INFO CodeGenerator: Code generated in 8.805028 ms
+-----------+----------------+-----+
|AREA NAME  |Average_Distance|Count|
+-----------+----------------+-----+
|77th Street|5.740           |16531|
|Southeast  |13.769          |12942|
|Newton     |9.875           |9601 |
|Southwest  |4.155           |8628 |
|Hollenbeck |15.014          |6101 |
|Harbor     |13.362          |5430 |
|Rampart    |4.102           |4983 |
|Mission    |7.547           |4455 |
|Olympic    |1.835           |4318 |
|Foothill   |3.811           |3881 |
|Northeast  |10.444          |3843 |
|Hollywood  |12.094          |3546 |
|Central    |4.771           |3460 |
|Wilshire   |13.358          |3420 |
|N Hollywood|NULL            |3341 |
|West Valley|17.096          |2784 |
|Van Nuys   |2.218           |2641 |
|Pacific    |13.265          |2641 |
|Devonshire |18.524          |2604 |
|Topanga    |3.487           |2307 |
|West LA    |NULL            |1509 |
+-----------+----------------+-----+
```

Ζητούμενο 7.

Στο Query 3 και στο Query 4b έχουμε joins, στα οποία προσθέτουμε την εντολή hint() για το εκάστοτε join, δηλαδή Broadcast, Merge, Shuffle Hash και Shuffle Replicate NI και την explain() προκειμένου να τυπωθεί ο τρόπος που οργανώνεται η εκτέλεση εσωτερικά του job. Λαμβάνουμε από το Spark UI το γραφικό και περιγραφικό πλάνο της οργάνωσης το οποίο για κάθε περίπτωση παραθέτουμε με εικόνες.

Q3 DataFrame Broadcast 2 Executors

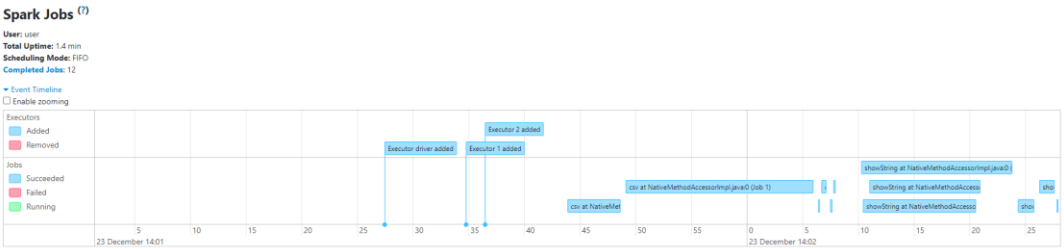


```
== Physical Plan ==
AdaptiveSparkPlan (53)
+- == Final Plan ==
   TakeOrderedAndProject (30)
   +- * HashAggregate (29)
      +- AQEShuffleRead (28)
         +- ShuffleQueryStage (27), Statistics(sizeInBytes=584.0 B, rowCount=16)
            +- Exchange (26)
               +- * HashAggregate (25)
                  +- * Project (24)
                     +- BatchEvalPython (23)
                        +- * Project (22)
                           +- * BroadcastHashJoin Inner BuildRight (21)
                              :- * Project (9)
                              :  +- * BroadcastHashJoin Inner BuildRight (8)
                              :     :- * Project (3)
                              :     :  +- * Filter (2)
                              :     :     +- Scan csv  (1)
                              :     +- BroadcastQueryStage (7), Statistics(sizeInBytes=6.0 MiB, rowCount=3.74E+4)
                              :        +- BroadcastExchange (6)
                              :           +- * Filter (5)
                              :              +- Scan csv  (4)
                              +- BroadcastQueryStage (20), Statistics(sizeInBytes=1024.3 KiB, rowCount=6)
                                 +- BroadcastExchange (19)
                                    +- Union (18)
                                       :- * Filter (13)
                                       :  +- TakeOrderedAndProject (12)
                                       :     +- * Project (11)
                                       :        +- Scan csv  (10)
                                       +- * Filter (17)
                                          +- TakeOrderedAndProject (16)
                                             +- * Project (15)
                                                +- Scan csv  (14)
```

```
+- == Initial Plan ==
   TakeOrderedAndProject (52)
   +- HashAggregate (51)
      +- Exchange (50)
         +- HashAggregate (49)
            +- Project (48)
               +- BatchEvalPython (47)
                  +- Project (46)
                     +- BroadcastHashJoin Inner BuildRight (45)
                        :- Project (36)
                        :  +- BroadcastHashJoin Inner BuildRight (35)
                        :     :- Project (32)
                        :     :  +- Filter (31)
                        :     :     +- Scan csv  (1)
                        :     +- BroadcastExchange (34)
                        :        +- Filter (33)
                        :           +- Scan csv  (4)
                        +- BroadcastExchange (44)
                           +- Union (43)
                              :- Filter (39)
                              :  +- TakeOrderedAndProject (38)
                              :     +- Project (37)
                              :        +- Scan csv  (10)
                              +- Filter (42)
                                 +- TakeOrderedAndProject (41)
                                    +- Project (40)
                                       +- Scan csv  (14)
```

```
23/12/23 13:58:46 INFO CodeGenerator: Code generated in 14.559453 ms
23/12/23 13:58:46 INFO CodeGenerator: Code generated in 15.720795 ms
```

Q3 DataFrame Merge 2 Executors

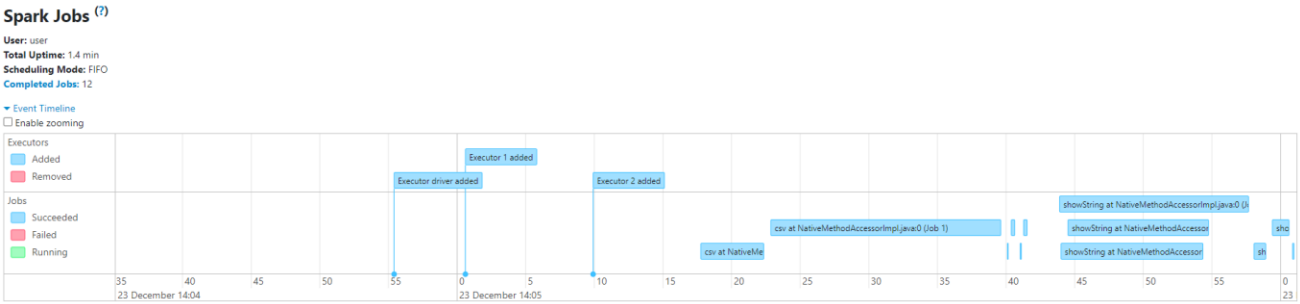



```
== Physical Plan ==
AdaptiveSparkPlan (71)
+- == Final Plan ==
   TakeOrderedAndProject (42)
   +- * HashAggregate (41)
      +- AQEShuffleRead (40)
         +- ShuffleQueryStage (39), Statistics(sizeInBytes=336.0 B, rowCount=9)
            +- Exchange (38)
               +- * HashAggregate (37)
                  +- * Project (36)
                     +- BatchEvalPython (35)
                        +- * Project (34)
                           +- * SortMergeJoin Inner (33)
                              :- * Sort (19)
                              :  +- AQEShuffleRead (18)
                              :     +- ShuffleQueryStage (17), Statistics(sizeInBytes=7.4 MiB, rowCount=1.95E+5)
                              :        +- Exchange (16)
                              :           +- * Project (15)
                              :              +- * SortMergeJoin Inner (14)
                              :                 :- * Sort (7)
                              :                 :  +- AQEShuffleRead (6)
                              :                 :     +- ShuffleQueryStage (5), Statistics(sizeInBytes=7.5 MiB, rowCount=1.96E+5)
                              :                 :        +- Exchange (4)
                              :                 :           +- * Project (3)
                              :                 :              +- * Filter (2)
                              :                 :                 +- Scan csv  (1)
                              :                 +- * Sort (13)
                              :                    +- AQEShuffleRead (12)
                              :                       +- ShuffleQueryStage (11), Statistics(sizeInBytes=1462.1 KiB, rowCount=3.74E+4)
                              :                          +- Exchange (10)
                              :                             +- * Filter (9)
                              :                                +- Scan csv  (8)
                              +- * Sort (32)
                                 +- AQEShuffleRead (31)
                                    +- ShuffleQueryStage (30), Statistics(sizeInBytes=96.0 B, rowCount=6)
                                       +- Exchange (29)
                                          +- Union (28)
                                             :- * Filter (23)
                                             :  +- TakeOrderedAndProject (22)
                                             :     +- * Project (21)
                                             :        +- Scan csv  (20)
                                             +- * Filter (27)
                                                +- TakeOrderedAndProject (26)
                                                   +- * Project (25)
                                                      +- Scan csv  (24)
```

```
+- == Initial Plan ==
   TakeOrderedAndProject (70)
   +- HashAggregate (69)
      +- Exchange (68)
         +- HashAggregate (67)
            +- Project (66)
               +- BatchEvalPython (65)
                  +- Project (64)
                     +- SortMergeJoin Inner (63)
                        :- Sort (53)
                        :  +- Exchange (52)
                        :     +- Project (51)
                        :        +- SortMergeJoin Inner (50)
                        :           :- Sort (46)
                        :           :  +- Exchange (45)
                        :           :     +- Project (44)
                        :           :        +- Filter (43)
                        :           :           +- Scan csv  (1)
                        :           +- Sort (49)
                        :              +- Exchange (48)
                        :                 +- Filter (47)
                        :                    +- Scan csv  (8)
                        +- Sort (62)
                           +- Exchange (61)
                              +- Union (60)
                                 :- Filter (56)
                                 :  +- TakeOrderedAndProject (55)
                                 :     +- Project (54)
                                 :        +- Scan csv  (20)
                                 +- Filter (59)
                                    +- TakeOrderedAndProject (58)
                                       +- Project (57)
                                          +- Scan csv  (24)
```

```
23/12/23 14:02:27 INFO CodeGenerator: Code generated in 14.869666 ms
23/12/23 14:02:28 INFO CodeGenerator: Code generated in 19.720457 ms
```

Q3 DataFrame Shuffle Hash 2 Executors

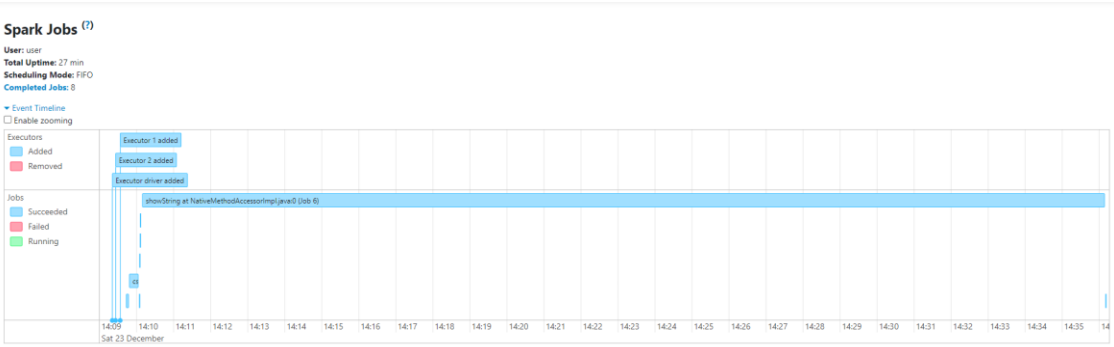



```
== Physical Plan ==
AdaptiveSparkPlan (63)
+- == Final Plan ==
   TakeOrderedAndProject (38)
   +- * HashAggregate (37)
      +- AQEShuffleRead (36)
         +- ShuffleQueryStage (35), Statistics(sizeInBytes=336.0 B, rowCount=9)
            +- Exchange (34)
               +- * HashAggregate (33)
                  +- * Project (32)
                     +- BatchEvalPython (31)
                        +- * Project (30)
                           +- * ShuffledHashJoin Inner BuildLeft (29)
                              :- AQEShuffleRead (16)
                              :  +- ShuffleQueryStage (15), Statistics(sizeInBytes=7.4 MiB, rowCount=1.95E+5)
                              :     +- Exchange (14)
                              :        +- * Project (13)
                              :           +- * ShuffledHashJoin Inner BuildLeft (12)
                              :              :- AQEShuffleRead (6)
                              :              :  +- ShuffleQueryStage (5), Statistics(sizeInBytes=7.5 MiB, rowCount=1.96E+5)
                              :              :     +- Exchange (4)
                              :              :        +- * Project (3)
                              :              :           +- * Filter (2)
                              :              :              +- Scan csv  (1)
                              :              +- AQEShuffleRead (11)
                              :                 +- ShuffleQueryStage (10), Statistics(sizeInBytes=1462.1 KiB, rowCount=3.74E+4)
                              :                    +- Exchange (9)
                              :                       +- * Filter (8)
                              :                          +- Scan csv  (7)
                              +- AQEShuffleRead (28)
                                 +- ShuffleQueryStage (27), Statistics(sizeInBytes=96.0 B, rowCount=6)
                                    +- Exchange (26)
                                       +- Union (25)
                                          :- * Filter (20)
                                          :  +- TakeOrderedAndProject (19)
                                          :     +- * Project (18)
                                          :        +- Scan csv  (17)
                                          +- * Filter (24)
                                             +- TakeOrderedAndProject (23)
                                                +- * Project (22)
                                                   +- Scan csv  (21)
```

```
+- == Initial Plan ==
   TakeOrderedAndProject (62)
   +- HashAggregate (61)
      +- Exchange (60)
         +- HashAggregate (59)
            +- Project (58)
               +- BatchEvalPython (57)
                  +- Project (56)
                     +- ShuffledHashJoin Inner BuildLeft (55)
                        :- Exchange (46)
                        :  +- Project (45)
                        :     +- ShuffledHashJoin Inner BuildLeft (44)
                        :        :- Exchange (41)
                        :        :  +- Project (40)
                        :        :     +- Filter (39)
                        :        :        +- Scan csv  (1)
                        :        +- Exchange (43)
                        :           +- Filter (42)
                        :              +- Scan csv  (7)
                        +- Exchange (54)
                           +- Union (53)
                              :- Filter (49)
                              :  +- TakeOrderedAndProject (48)
                              :     +- Project (47)
                              :        +- Scan csv  (17)
                              +- Filter (52)
                                 +- TakeOrderedAndProject (51)
                                    +- Project (50)
                                       +- Scan csv  (21)
```

```
23/12/23 14:06:01 INFO CodeGenerator: Code generated in 15.200458 ms
23/12/23 14:06:01 INFO CodeGenerator: Code generated in 22.011143 ms
```

Q3 DataFrame Shuffle Replicate Nl 2 Executors

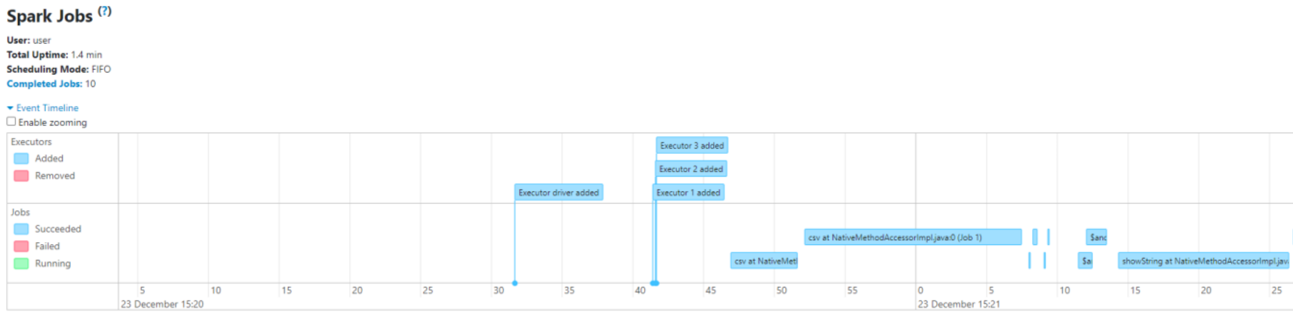



```
== Physical Plan ==
AdaptiveSparkPlan (47)
+- == Final Plan ==
   TakeOrderedAndProject (26)
   +- * HashAggregate (25)
      +- AQEShuffleRead (24)
         +- ShuffleQueryStage (23), Statistics(sizeInBytes=1016.0 B, rowCount=28)
            +- Exchange (22)
               +- * HashAggregate (21)
                  +- * Project (20)
                     +- BatchEvalPython (19)
                        +- * Project (18)
                           +- CartesianProduct Inner (17)
                              :- * Project (7)
                              :  +- CartesianProduct Inner (6)
                              :     :- * Project (3)
                              :     :  +- * Filter (2)
                              :     :     +- Scan csv  (1)
                              :     +- * Filter (5)
                              :        +- Scan csv  (4)
                              +- Union (16)
                                 :- * Filter (11)
                                 :  +- TakeOrderedAndProject (10)
                                 :     +- * Project (9)
                                 :        +- Scan csv  (8)
                                 +- * Filter (15)
                                    +- TakeOrderedAndProject (14)
                                       +- * Project (13)
                                          +- Scan csv  (12)
```

```
+- == Initial Plan ==
   TakeOrderedAndProject (46)
   +- HashAggregate (45)
      +- Exchange (44)
         +- HashAggregate (43)
            +- Project (42)
               +- BatchEvalPython (41)
                  +- Project (40)
                     +- CartesianProduct Inner (39)
                        :- Project (31)
                        :  +- CartesianProduct Inner (30)
                        :     :- Project (28)
                        :     :  +- Filter (27)
                        :     :     +- Scan csv  (1)
                        :     +- Filter (29)
                        :        +- Scan csv  (4)
                        +- Union (38)
                           :- Filter (34)
                           :  +- TakeOrderedAndProject (33)
                           :     +- Project (32)
                           :        +- Scan csv  (8)
                           +- Filter (37)
                              +- TakeOrderedAndProject (36)
                                 +- Project (35)
                                    +- Scan csv  (12)
```

```
23/12/23 14:36:09 INFO CodeGenerator: Code generated in 25.471172 ms
23/12/23 14:36:09 INFO CodeGenerator: Code generated in 18.46322 ms
```

Q3 DataFrame Broadcast 3 Executors

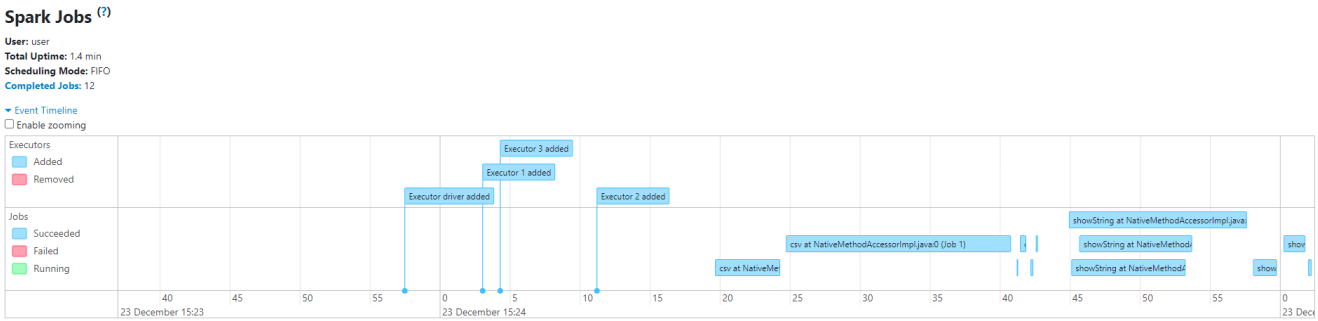



```
== Physical Plan ==
AdaptiveSparkPlan (53)
+- == Final Plan ==
   TakeOrderedAndProject (30)
   +- * HashAggregate (29)
      +- AQEShuffleRead (28)
         +- ShuffleQueryStage (27), Statistics(sizeInBytes=584.0 B, rowCount=16)
            +- Exchange (26)
               +- * HashAggregate (25)
                  +- * Project (24)
                     +- BatchEvalPython (23)
                        +- * Project (22)
                           +- * BroadcastHashJoin Inner BuildRight (21)
                              :- * Project (9)
                              :  +- * BroadcastHashJoin Inner BuildRight (8)
                              :     :- * Project (3)
                              :     :  +- * Filter (2)
                              :     :     +- Scan csv  (1)
                              :     +- BroadcastQueryStage (7), Statistics(sizeInBytes=6.0 MiB, rowCount=3.74E+4)
                              :        +- BroadcastExchange (6)
                              :           +- * Filter (5)
                              :              +- Scan csv  (4)
                              +- BroadcastQueryStage (20), Statistics(sizeInBytes=1024.3 KiB, rowCount=6)
                                 +- BroadcastExchange (19)
                                    +- Union (18)
                                       :- * Filter (13)
                                       :  +- TakeOrderedAndProject (12)
                                       :     +- * Project (11)
                                       :        +- Scan csv  (10)
                                       +- * Filter (17)
                                          +- TakeOrderedAndProject (16)
                                             +- * Project (15)
                                                +- Scan csv  (14)
```

```
+- == Initial Plan ==
   TakeOrderedAndProject (52)
   +- HashAggregate (51)
      +- Exchange (50)
         +- HashAggregate (49)
            +- Project (48)
               +- BatchEvalPython (47)
                  +- Project (46)
                     +- BroadcastHashJoin Inner BuildRight (45)
                        :- Project (36)
                        :  +- BroadcastHashJoin Inner BuildRight (35)
                        :     :- Project (32)
                        :     :  +- Filter (31)
                        :     :     +- Scan csv  (1)
                        :     +- BroadcastExchange (34)
                        :        +- Filter (33)
                        :           +- Scan csv  (4)
                        +- BroadcastExchange (44)
                           +- Union (43)
                              :- Filter (39)
                              :  +- TakeOrderedAndProject (38)
                              :     +- Project (37)
                              :        +- Scan csv  (10)
                              +- Filter (42)
                                 +- TakeOrderedAndProject (41)
                                    +- Project (40)
                                       +- Scan csv  (14)
```

```
23/12/23 15:21:27 INFO CodeGenerator: Code generated in 11.369322 ms
23/12/23 15:21:27 INFO CodeGenerator: Code generated in 11.966729 ms
```

Q3 DataFrame Merge 3 Executors

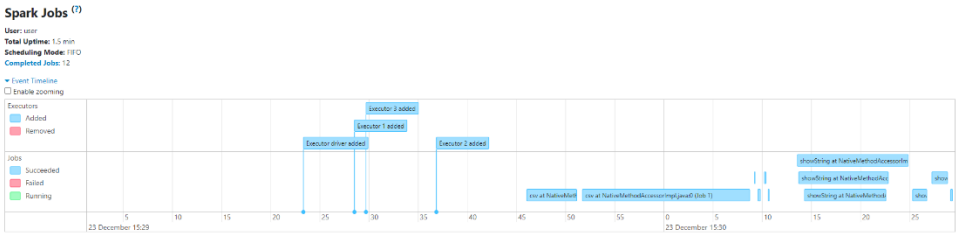



```
== Physical Plan ==
AdaptiveSparkPlan (71)
+- == Final Plan ==
   TakeOrderedAndProject (42)
   +- * HashAggregate (41)
      +- AQEShuffleRead (40)
         +- ShuffleQueryStage (39), Statistics(sizeInBytes=336.0 B, rowCount=9)
            +- Exchange (38)
               +- * HashAggregate (37)
                  +- * Project (36)
                     +- BatchEvalPython (35)
                        +- * Project (34)
                           +- * SortMergeJoin Inner (33)
                              :- * Sort (19)
                              :  +- AQEShuffleRead (18)
                              :     +- ShuffleQueryStage (17), Statistics(sizeInBytes=7.4 MiB, rowCount=1.95E+5)
                              :        +- Exchange (16)
                              :           +- * Project (15)
                              :              +- * SortMergeJoin Inner (14)
                              :                 :- * Sort (7)
                              :                 :  +- AQEShuffleRead (6)
                              :                 :     +- ShuffleQueryStage (5), Statistics(sizeInBytes=7.5 MiB, rowCount=1.96E+5)
                              :                 :        +- Exchange (4)
                              :                 :           +- * Project (3)
                              :                 :              +- * Filter (2)
                              :                 :                 +- Scan csv  (1)
                              :                 +- * Sort (13)
                              :                    +- AQEShuffleRead (12)
                              :                       +- ShuffleQueryStage (11), Statistics(sizeInBytes=1462.1 KiB, rowCount=3.74E+4)
                              :                          +- Exchange (10)
                              :                             +- * Filter (9)
                              :                                +- Scan csv  (8)
                              +- * Sort (32)
                                 +- AQEShuffleRead (31)
                                    +- ShuffleQueryStage (30), Statistics(sizeInBytes=96.0 B, rowCount=6)
                                       +- Exchange (29)
                                          +- Union (28)
                                             :- * Filter (23)
                                             :  +- TakeOrderedAndProject (22)
                                             :     +- * Project (21)
                                             :        +- Scan csv  (20)
                                             +- * Filter (27)
                                                +- TakeOrderedAndProject (26)
                                                   +- * Project (25)
                                                      +- Scan csv  (24)
```

```
+- == Initial Plan ==
   TakeOrderedAndProject (70)
   +- HashAggregate (69)
      +- Exchange (68)
         +- HashAggregate (67)
            +- Project (66)
               +- BatchEvalPython (65)
                  +- Project (64)
                     +- SortMergeJoin Inner (63)
                        :- Sort (53)
                        :  +- Exchange (52)
                        :     +- Project (51)
                        :        +- SortMergeJoin Inner (50)
                        :           :- Sort (46)
                        :           :  +- Exchange (45)
                        :           :     +- Project (44)
                        :           :        +- Filter (43)
                        :           :           +- Scan csv  (1)
                        :           +- Sort (49)
                        :              +- Exchange (48)
                        :                 +- Filter (47)
                        :                    +- Scan csv  (8)
                        +- Sort (62)
                           +- Exchange (61)
                              +- Union (60)
                                 :- Filter (56)
                                 :  +- TakeOrderedAndProject (55)
                                 :     +- Project (54)
                                 :        +- Scan csv  (20)
                                 +- Filter (59)
                                    +- TakeOrderedAndProject (58)
                                       +- Project (57)
                                          +- Scan csv  (24)
```

```
23/12/23 15:25:02 INFO CodeGenerator: Code generated in 32.901166 ms
23/12/23 15:25:02 INFO CodeGenerator: Code generated in 24.288173 ms
```

Q3 DataFrame Shuffle Hash 3 Executors

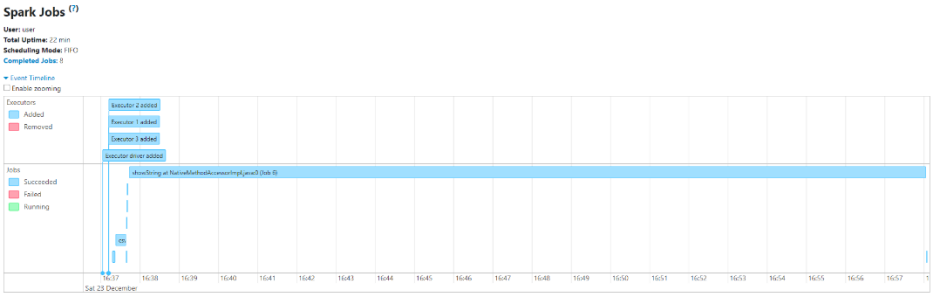



```
== Physical Plan ==
AdaptiveSparkPlan (63)
+- == Final Plan ==
   TakeOrderedAndProject (38)
   +- * HashAggregate (37)
      +- AQEShuffleRead (36)
         +- ShuffleQueryStage (35), Statistics(sizeInBytes=336.0 B, rowCount=9)
            +- Exchange (34)
               +- * HashAggregate (33)
                  +- * Project (32)
                     +- BatchEvalPython (31)
                        +- * Project (30)
                           +- * ShuffledHashJoin Inner BuildLeft (29)
                              :- AQEShuffleRead (16)
                              :  +- ShuffleQueryStage (15), Statistics(sizeInBytes=7.4 MiB, rowCount=1.95E+5)
                              :     +- Exchange (14)
                              :        +- * Project (13)
                              :           +- * ShuffledHashJoin Inner BuildLeft (12)
                              :              :- AQEShuffleRead (6)
                              :              :  +- ShuffleQueryStage (5), Statistics(sizeInBytes=7.5 MiB, rowCount=1.96E+5)
                              :              :     +- Exchange (4)
                              :              :        +- * Project (3)
                              :              :           +- * Filter (2)
                              :              :              +- Scan csv  (1)
                              :              +- AQEShuffleRead (11)
                              :                 +- ShuffleQueryStage (10), Statistics(sizeInBytes=1462.1 KiB, rowCount=3.74E+4)
                              :                    +- Exchange (9)
                              :                       +- * Filter (8)
                              :                          +- Scan csv  (7)
                              +- AQEShuffleRead (28)
                                 +- ShuffleQueryStage (27), Statistics(sizeInBytes=96.0 B, rowCount=6)
                                    +- Exchange (26)
                                       +- Union (25)
                                          :- * Filter (20)
                                          :  +- TakeOrderedAndProject (19)
                                          :     +- * Project (18)
                                          :        +- Scan csv  (17)
                                          +- * Filter (24)
                                             +- TakeOrderedAndProject (23)
                                                +- * Project (22)
                                                   +- Scan csv  (21)
```

```
+- == Initial Plan ==
   TakeOrderedAndProject (62)
   +- HashAggregate (61)
      +- Exchange (60)
         +- HashAggregate (59)
            +- Project (58)
               +- BatchEvalPython (57)
                  +- Project (56)
                     +- ShuffledHashJoin Inner BuildLeft (55)
                        :- Exchange (46)
                        :  +- Project (45)
                        :     +- ShuffledHashJoin Inner BuildLeft (44)
                        :        :- Exchange (41)
                        :        :  +- Project (40)
                        :        :     +- Filter (39)
                        :        :        +- Scan csv  (1)
                        :        +- Exchange (43)
                        :           +- Filter (42)
                        :              +- Scan csv  (7)
                        +- Exchange (54)
                           +- Union (53)
                              :- Filter (49)
                              :  +- TakeOrderedAndProject (48)
                              :     +- Project (47)
                              :        +- Scan csv  (17)
                              +- Filter (52)
                                 +- TakeOrderedAndProject (51)
                                    +- Project (50)
                                       +- Scan csv  (21)
```

```
23/12/23 15:30:29 INFO CodeGenerator: Code generated in 26.941863 ms
23/12/23 15:30:29 INFO CodeGenerator: Code generated in 41.049883 ms
```

Q3 DataFrame Shuffle Replicate Nl 3 Executors

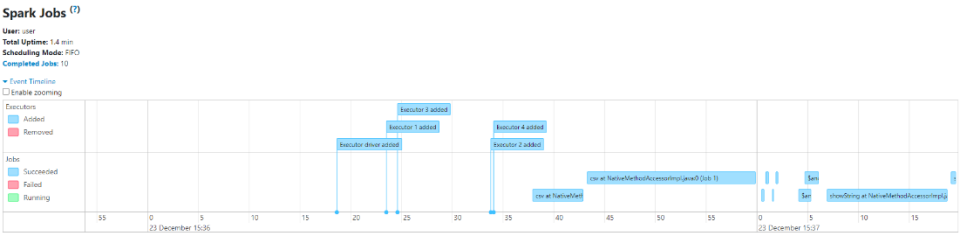



```
== Physical Plan ==
AdaptiveSparkPlan (47)
+- == Final Plan ==
   TakeOrderedAndProject (26)
   +- * HashAggregate (25)
      +- AQEShuffleRead (24)
         +- ShuffleQueryStage (23), Statistics(sizeInBytes=1016.0 B, rowCount=28)
            +- Exchange (22)
               +- * HashAggregate (21)
                  +- * Project (20)
                     +- BatchEvalPython (19)
                        +- * Project (18)
                           +- CartesianProduct Inner (17)
                              :- * Project (7)
                              :  +- CartesianProduct Inner (6)
                              :     :- * Project (3)
                              :     :  +- * Filter (2)
                              :     :     +- Scan csv  (1)
                              :     +- * Filter (5)
                              :        +- Scan csv  (4)
                              +- Union (16)
                                 :- * Filter (11)
                                 :  +- TakeOrderedAndProject (10)
                                 :     +- * Project (9)
                                 :        +- Scan csv  (8)
                                 +- * Filter (15)
                                    +- TakeOrderedAndProject (14)
                                       +- * Project (13)
                                          +- Scan csv  (12)
```

```
+- == Initial Plan ==
   TakeOrderedAndProject (46)
   +- HashAggregate (45)
      +- Exchange (44)
         +- HashAggregate (43)
            +- Project (42)
               +- BatchEvalPython (41)
                  +- Project (40)
                     +- CartesianProduct Inner (39)
                        :- Project (31)
                        :  +- CartesianProduct Inner (30)
                        :     :- Project (28)
                        :     :  +- Filter (27)
                        :     :     +- Scan csv  (1)
                        :     +- Filter (29)
                        :        +- Scan csv  (4)
                        +- Union (38)
                           :- Filter (34)
                           :  +- TakeOrderedAndProject (33)
                           :     +- Project (32)
                           :        +- Scan csv  (8)
                           +- Filter (37)
                              +- TakeOrderedAndProject (36)
                                 +- Project (35)
                                    +- Scan csv  (12)
```

```
23/12/23 16:58:03 INFO CodeGenerator: Code generated in 9.187842 ms
23/12/23 16:58:03 INFO CodeGenerator: Code generated in 15.405588 ms
```

Q3 DataFrame Broadcast 4 Executors

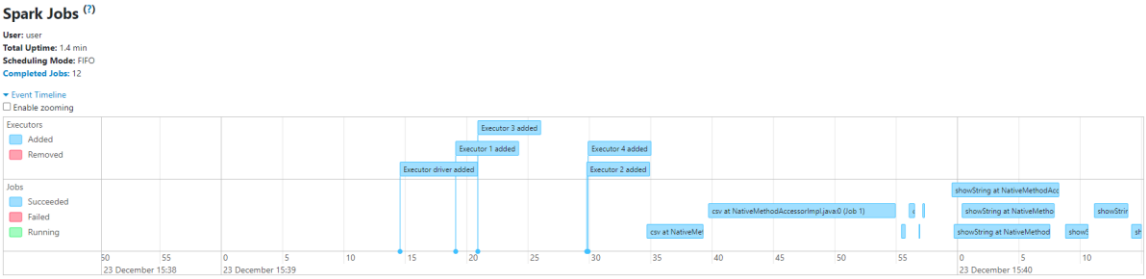



```
== Physical Plan ==
AdaptiveSparkPlan (53)
+- == Final Plan ==
   TakeOrderedAndProject (30)
   +- * HashAggregate (29)
      +- AQEShuffleRead (28)
         +- ShuffleQueryStage (27), Statistics(sizeInBytes=584.0 B, rowCount=16)
            +- Exchange (26)
               +- * HashAggregate (25)
                  +- * Project (24)
                     +- BatchEvalPython (23)
                        +- * Project (22)
                           +- * BroadcastHashJoin Inner BuildRight (21)
                              :- * Project (9)
                              :  +- * BroadcastHashJoin Inner BuildRight (8)
                              :     :- * Project (3)
                              :     :  +- * Filter (2)
                              :     :     +- Scan csv  (1)
                              :     +- BroadcastQueryStage (7), Statistics(sizeInBytes=6.0 MiB, rowCount=3.74E+4)
                              :        +- BroadcastExchange (6)
                              :           +- * Filter (5)
                              :              +- Scan csv  (4)
                              +- BroadcastQueryStage (20), Statistics(sizeInBytes=1024.3 KiB, rowCount=6)
                                 +- BroadcastExchange (19)
                                    +- Union (18)
                                       :- * Filter (13)
                                       :  +- TakeOrderedAndProject (12)
                                       :     +- * Project (11)
                                       :        +- Scan csv  (10)
                                       +- * Filter (17)
                                          +- TakeOrderedAndProject (16)
                                             +- * Project (15)
                                                +- Scan csv  (14)
```

```
+- == Initial Plan ==
   TakeOrderedAndProject (52)
   +- HashAggregate (51)
      +- Exchange (50)
         +- HashAggregate (49)
            +- Project (48)
               +- BatchEvalPython (47)
                  +- Project (46)
                     +- BroadcastHashJoin Inner BuildRight (45)
                        :- Project (36)
                        :  +- BroadcastHashJoin Inner BuildRight (35)
                        :     :- Project (32)
                        :     :  +- Filter (31)
                        :     :     +- Scan csv  (1)
                        :     +- BroadcastExchange (34)
                        :        +- Filter (33)
                        :           +- Scan csv  (4)
                        +- BroadcastExchange (44)
                           +- Union (43)
                              :- Filter (39)
                              :  +- TakeOrderedAndProject (38)
                              :     +- Project (37)
                              :        +- Scan csv  (10)
                              +- Filter (42)
                                 +- TakeOrderedAndProject (41)
                                    +- Project (40)
                                       +- Scan csv  (14)
```

```
23/12/23 15:37:19 INFO CodeGenerator: Code generated in 20.093437 ms
23/12/23 15:37:19 INFO CodeGenerator: Code generated in 24.642782 ms
```

Q3 DataFrame Merge 4 Executors

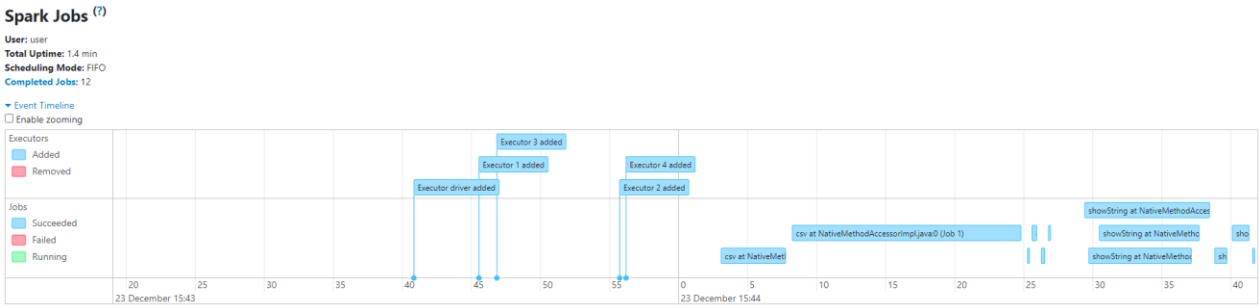



```
== Physical Plan ==
AdaptiveSparkPlan (71)
+- == Final Plan ==
   TakeOrderedAndProject (42)
   +- * HashAggregate (41)
      +- AQEShuffleRead (40)
         +- ShuffleQueryStage (39), Statistics(sizeInBytes=336.0 B, rowCount=9)
            +- Exchange (38)
               +- * HashAggregate (37)
                  +- * Project (36)
                     +- BatchEvalPython (35)
                        +- * Project (34)
                           +- * SortMergeJoin Inner (33)
                              :- * Sort (19)
                              :  +- AQEShuffleRead (18)
                              :     +- ShuffleQueryStage (17), Statistics(sizeInBytes=7.4 MiB, rowCount=1.95E+5)
                              :        +- Exchange (16)
                              :           +- * Project (15)
                              :              +- * SortMergeJoin Inner (14)
                              :                 :- * Sort (7)
                              :                 :  +- AQEShuffleRead (6)
                              :                 :     +- ShuffleQueryStage (5), Statistics(sizeInBytes=7.5 MiB, rowCount=1.96E+5)
                              :                 :        +- Exchange (4)
                              :                 :           +- * Project (3)
                              :                 :              +- * Filter (2)
                              :                 :                 +- Scan csv  (1)
                              :                 +- * Sort (13)
                              :                    +- AQEShuffleRead (12)
                              :                       +- ShuffleQueryStage (11), Statistics(sizeInBytes=1462.1 KiB, rowCount=3.74E+4)
                              :                          +- Exchange (10)
                              :                             +- * Filter (9)
                              :                                +- Scan csv  (8)
                              +- * Sort (32)
                                 +- AQEShuffleRead (31)
                                    +- ShuffleQueryStage (30), Statistics(sizeInBytes=96.0 B, rowCount=6)
                                       +- Exchange (29)
                                          +- Union (28)
                                             :- * Filter (23)
                                             :  +- TakeOrderedAndProject (22)
                                             :     +- * Project (21)
                                             :        +- Scan csv  (20)
                                             +- * Filter (27)
                                                +- TakeOrderedAndProject (26)
                                                   +- * Project (25)
                                                      +- Scan csv  (24)
```

```
+- == Initial Plan ==
   TakeOrderedAndProject (70)
   +- HashAggregate (69)
      +- Exchange (68)
         +- HashAggregate (67)
            +- Project (66)
               +- BatchEvalPython (65)
                  +- Project (64)
                     +- SortMergeJoin Inner (63)
                        :- Sort (53)
                        :  +- Exchange (52)
                        :     +- Project (51)
                        :        +- SortMergeJoin Inner (50)
                        :           :- Sort (46)
                        :           :  +- Exchange (45)
                        :           :     +- Project (44)
                        :           :        +- Filter (43)
                        :           :           +- Scan csv  (1)
                        :           +- Sort (49)
                        :              +- Exchange (48)
                        :                 +- Filter (47)
                        :                    +- Scan csv  (8)
                        +- Sort (62)
                           +- Exchange (61)
                              +- Union (60)
                                 :- Filter (56)
                                 :  +- TakeOrderedAndProject (55)
                                 :     +- Project (54)
                                 :        +- Scan csv  (20)
                                 +- Filter (59)
                                    +- TakeOrderedAndProject (58)
                                       +- Project (57)
                                          +- Scan csv  (24)
```

```
23/12/23 15:40:15 INFO CodeGenerator: Code generated in 12.739791 ms
23/12/23 15:40:15 INFO CodeGenerator: Code generated in 19.189621 ms
```

Q3 DataFrame Shuffle Hash 4 Executors

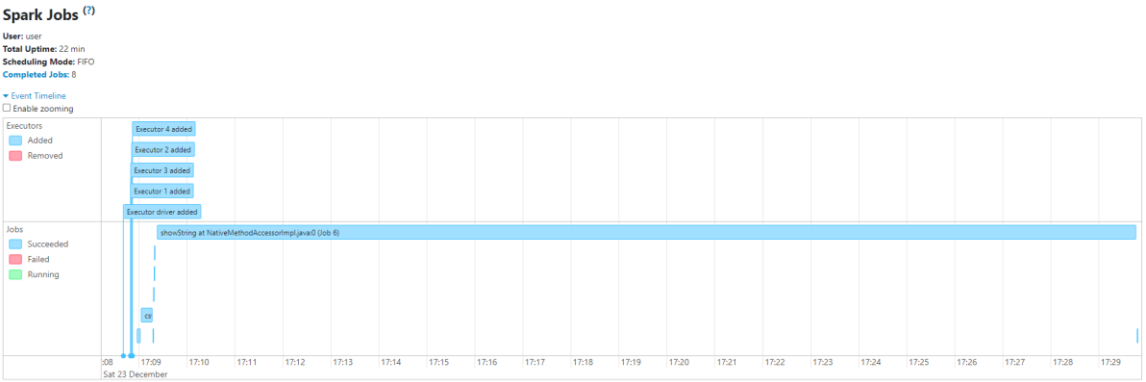



```
== Physical Plan ==
AdaptiveSparkPlan (63)
+- == Final Plan ==
   TakeOrderedAndProject (38)
   +- * HashAggregate (37)
      +- AQEShuffleRead (36)
         +- ShuffleQueryStage (35), Statistics(sizeInBytes=336.0 B, rowCount=9)
            +- Exchange (34)
               +- * HashAggregate (33)
                  +- * Project (32)
                     +- BatchEvalPython (31)
                        +- * Project (30)
                           +- * ShuffledHashJoin Inner BuildLeft (29)
                              :- AQEShuffleRead (16)
                              :  +- ShuffleQueryStage (15), Statistics(sizeInBytes=7.4 MiB, rowCount=1.95E+5)
                              :     +- Exchange (14)
                              :        +- * Project (13)
                              :           +- * ShuffledHashJoin Inner BuildLeft (12)
                              :              :- AQEShuffleRead (6)
                              :              :  +- ShuffleQueryStage (5), Statistics(sizeInBytes=7.5 MiB, rowCount=1.96E+5)
                              :              :     +- Exchange (4)
                              :              :        +- * Project (3)
                              :              :           +- * Filter (2)
                              :              :              +- Scan csv  (1)
                              :              +- AQEShuffleRead (11)
                              :                 +- ShuffleQueryStage (10), Statistics(sizeInBytes=1462.1 KiB, rowCount=3.74E+4)
                              :                    +- Exchange (9)
                              :                       +- * Filter (8)
                              :                          +- Scan csv  (7)
                              +- AQEShuffleRead (28)
                                 +- ShuffleQueryStage (27), Statistics(sizeInBytes=96.0 B, rowCount=6)
                                    +- Exchange (26)
                                       +- Union (25)
                                          :- * Filter (20)
                                          :  +- TakeOrderedAndProject (19)
                                          :     +- * Project (18)
                                          :        +- Scan csv  (17)
                                          +- * Filter (24)
                                             +- TakeOrderedAndProject (23)
                                                +- * Project (22)
                                                   +- Scan csv  (21)
```

```
+- == Initial Plan ==
   TakeOrderedAndProject (62)
   +- HashAggregate (61)
      +- Exchange (60)
         +- HashAggregate (59)
            +- Project (58)
               +- BatchEvalPython (57)
                  +- Project (56)
                     +- ShuffledHashJoin Inner BuildLeft (55)
                        :- Exchange (46)
                        :  +- Project (45)
                        :     +- ShuffledHashJoin Inner BuildLeft (44)
                        :        :- Exchange (41)
                        :        :  +- Project (40)
                        :        :     +- Filter (39)
                        :        :        +- Scan csv  (1)
                        :        +- Exchange (43)
                        :           +- Filter (42)
                        :              +- Scan csv  (7)
                        +- Exchange (54)
                           +- Union (53)
                              :- Filter (49)
                              :  +- TakeOrderedAndProject (48)
                              :     +- Project (47)
                              :        +- Scan csv  (17)
                              +- Filter (52)
                                 +- TakeOrderedAndProject (51)
                                    +- Project (50)
                                       +- Scan csv  (21)
```

```
23/12/23 15:44:41 INFO CodeGenerator: Code generated in 10.912193 ms
23/12/23 15:44:41 INFO CodeGenerator: Code generated in 14.007559 ms
```

Q3 DataFrame Shuffle Replicate Nl 4 Executors

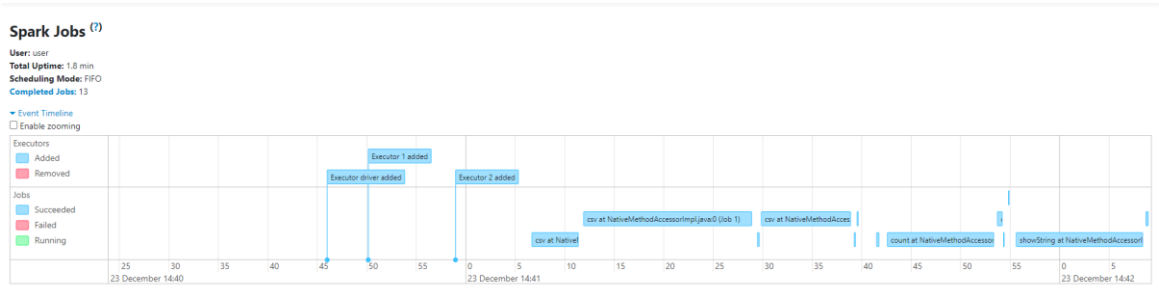



```
== Physical Plan ==
AdaptiveSparkPlan (47)
+- == Final Plan ==
   TakeOrderedAndProject (26)
   +- * HashAggregate (25)
      +- AQEShuffleRead (24)
         +- ShuffleQueryStage (23), Statistics(sizeInBytes=1016.0 B, rowCount=28)
            +- Exchange (22)
               +- * HashAggregate (21)
                  +- * Project (20)
                     +- BatchEvalPython (19)
                        +- * Project (18)
                           +- CartesianProduct Inner (17)
                              :- * Project (7)
                              :  +- CartesianProduct Inner (6)
                              :     :- * Project (3)
                              :     :  +- * Filter (2)
                              :     :     +- Scan csv  (1)
                              :     +- * Filter (5)
                              :        +- Scan csv  (4)
                              +- Union (16)
                                 :- * Filter (11)
                                 :  +- TakeOrderedAndProject (10)
                                 :     +- * Project (9)
                                 :        +- Scan csv  (8)
                                 +- * Filter (15)
                                    +- TakeOrderedAndProject (14)
                                       +- * Project (13)
                                          +- Scan csv  (12)
```

```
+- == Initial Plan ==
   TakeOrderedAndProject (46)
   +- HashAggregate (45)
      +- Exchange (44)
         +- HashAggregate (43)
            +- Project (42)
               +- BatchEvalPython (41)
                  +- Project (40)
                     +- CartesianProduct Inner (39)
                        :- Project (31)
                        :  +- CartesianProduct Inner (30)
                        :     :- Project (28)
                        :     :  +- Filter (27)
                        :     :     +- Scan csv  (1)
                        :     +- Filter (29)
                        :        +- Scan csv  (4)
                        +- Union (38)
                           :- Filter (34)
                           :  +- TakeOrderedAndProject (33)
                           :     +- Project (32)
                           :        +- Scan csv  (8)
                           +- Filter (37)
                              +- TakeOrderedAndProject (36)
                                 +- Project (35)
                                    +- Scan csv  (12)
```

```
23/12/23 17:29:48 INFO CodeGenerator: Code generated in 11.596965 ms
23/12/23 17:29:48 INFO CodeGenerator: Code generated in 17.431003 ms
```

Q4b DataFrame Broadcast

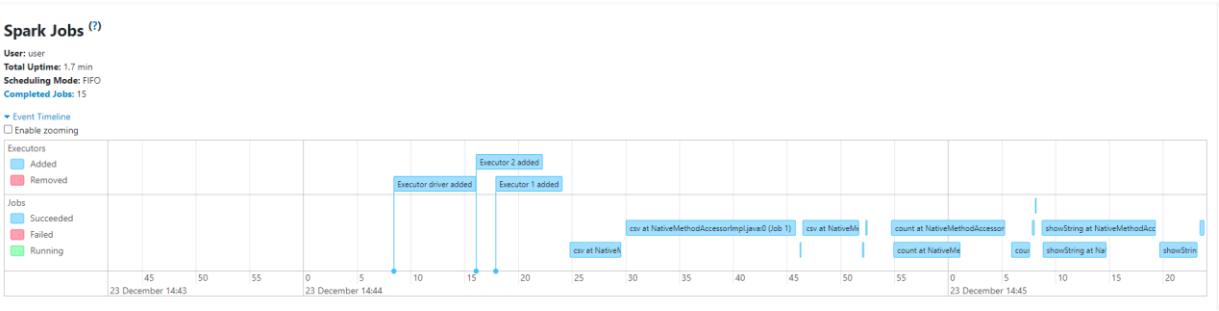



```
== Physical Plan ==
AdaptiveSparkPlan (37)
+- == Final Plan ==
   TakeOrderedAndProject (21)
   +- * HashAggregate (20)
      +- AQEShuffleRead (19)
         +- ShuffleQueryStage (18), Statistics(sizeInBytes=7.4 KiB, rowCount=147)
            +- Exchange (17)
               +- * HashAggregate (16)
                  +- * Project (15)
                     +- BatchEvalPython (14)
                        +- * Project (13)
                           +- * BroadcastHashJoin LeftOuter BuildRight (12)
                              :- Union (7)
                              :  :- * Project (3)
                              :  :  +- * Filter (2)
                              :  :     +- Scan csv  (1)
                              :  +- * Project (6)
                              :     +- * Filter (5)
                              :        +- Scan csv  (4)
                              +- BroadcastQueryStage (11), Statistics(sizeInBytes=4.0 MiB, rowCount=21)
                                 +- BroadcastExchange (10)
                                    +- * Filter (9)
                                       +- Scan csv  (8)
```

```
+- == Initial Plan ==
   TakeOrderedAndProject (36)
   +- HashAggregate (35)
      +- Exchange (34)
         +- HashAggregate (33)
            +- Project (32)
               +- BatchEvalPython (31)
                  +- Project (30)
                     +- BroadcastHashJoin LeftOuter BuildRight (29)
                        :- Union (26)
                        :  :- Project (23)
                        :  :  +- Filter (22)
                        :  :     +- Scan csv  (1)
                        :  +- Project (25)
                        :     +- Filter (24)
                        :        +- Scan csv  (4)
                        +- BroadcastExchange (28)
                           +- Filter (27)
                              +- Scan csv  (8)
```

```
23/12/23 14:42:08 INFO CodeGenerator: Code generated in 12.094038 ms
23/12/23 14:42:09 INFO BlockManagerInfo: Removed broadcast_15_piece0
23/12/23 14:42:09 INFO CodeGenerator: Code generated in 13.227107 ms
23/12/23 14:42:09 INFO BlockManagerInfo: Removed broadcast_16_piece0
```

Q4b DataFrame Merge

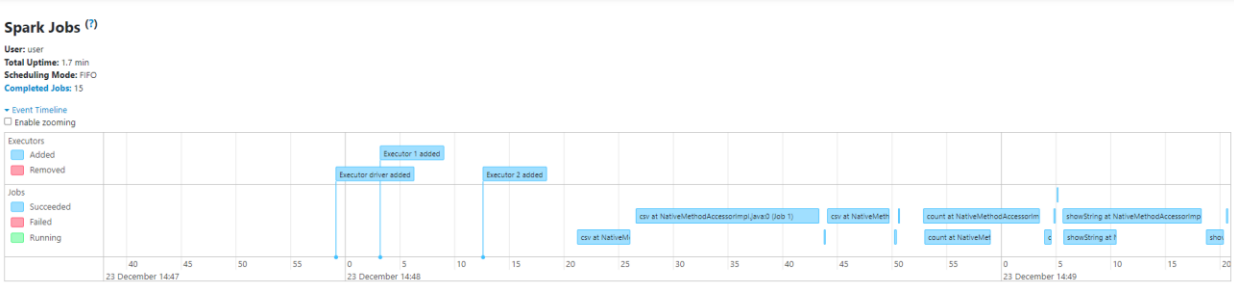



```
== Physical Plan ==
AdaptiveSparkPlan (46)
+- == Final Plan ==
   TakeOrderedAndProject (27)
   +- * HashAggregate (26)
      +- AQEShuffleRead (25)
         +- ShuffleQueryStage (24), Statistics(sizeInBytes=1080.0 B, rowCount=21)
            +- Exchange (23)
               +- * HashAggregate (22)
                  +- * Project (21)
                     +- BatchEvalPython (20)
                        +- * Project (19)
                           +- * SortMergeJoin LeftOuter (18)
                              :- * Sort (11)
                              :  +- AQEShuffleRead (10)
                              :     +- ShuffleQueryStage (9), Statistics(sizeInBytes=5.4 MiB, rowCount=1.09E+5)
                              :        +- Exchange (8)
                              :           +- Union (7)
                              :              :- * Project (3)
                              :              :  +- * Filter (2)
                              :              :     +- Scan csv  (1)
                              :              +- * Project (6)
                              :                 +- * Filter (5)
                              :                    +- Scan csv  (4)
                              +- * Sort (17)
                                 +- AQEShuffleRead (16)
                                    +- ShuffleQueryStage (15), Statistics(sizeInBytes=920.0 B, rowCount=21)
                                       +- Exchange (14)
                                          +- * Filter (13)
                                             +- Scan csv  (12)
```

```
+- == Initial Plan ==
   TakeOrderedAndProject (45)
   +- HashAggregate (44)
      +- Exchange (43)
         +- HashAggregate (42)
            +- Project (41)
               +- BatchEvalPython (40)
                  +- Project (39)
                     +- SortMergeJoin LeftOuter (38)
                        :- Sort (34)
                        :  +- Exchange (33)
                        :     +- Union (32)
                        :        :- Project (29)
                        :        :  +- Filter (28)
                        :        :     +- Scan csv  (1)
                        :        +- Project (31)
                        :           +- Filter (30)
                        :              +- Scan csv  (4)
                        +- Sort (37)
                           +- Exchange (36)
                              +- Filter (35)
                                 +- Scan csv  (12)
```

```
23/12/23 14:45:23 INFO CodeGenerator: Code generated in 21.806663 ms
23/12/23 14:45:23 INFO CodeGenerator: Code generated in 33.895207 ms
```

Q4b DataFrame Shuffle Hash

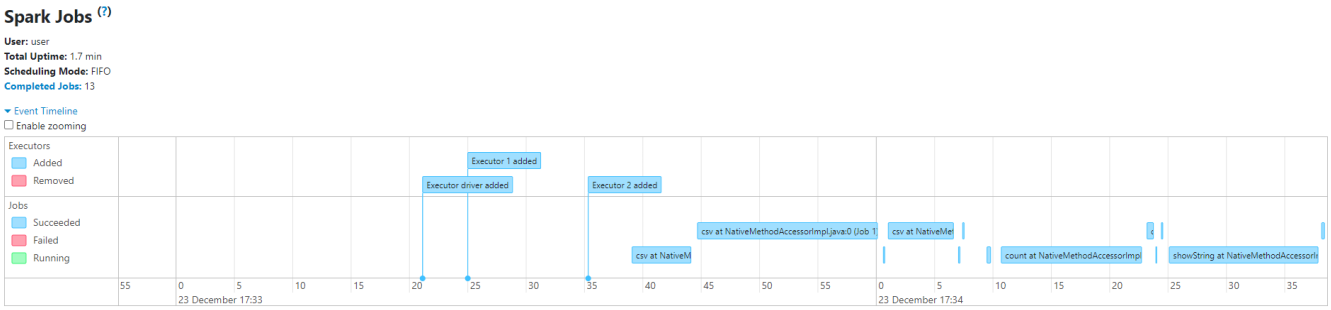



```
== Physical Plan ==
AdaptiveSparkPlan (42)
+- == Final Plan ==
   TakeOrderedAndProject (25)
   +- * HashAggregate (24)
      +- AQEShuffleRead (23)
         +- ShuffleQueryStage (22), Statistics(sizeInBytes=1080.0 B, rowCount=21)
            +- Exchange (21)
               +- * HashAggregate (20)
                  +- * Project (19)
                     +- BatchEvalPython (18)
                        +- * Project (17)
                           +- * ShuffledHashJoin LeftOuter BuildRight (16)
                              :- AQEShuffleRead (10)
                              :  +- ShuffleQueryStage (9), Statistics(sizeInBytes=5.4 MiB, rowCount=1.09E+5)
                              :     +- Exchange (8)
                              :        +- Union (7)
                              :           :- * Project (3)
                              :           :  +- * Filter (2)
                              :           :     +- Scan csv  (1)
                              :           +- * Project (6)
                              :              +- * Filter (5)
                              :                 +- Scan csv  (4)
                              +- AQEShuffleRead (15)
                                 +- ShuffleQueryStage (14), Statistics(sizeInBytes=920.0 B, rowCount=21)
                                    +- Exchange (13)
                                       +- * Filter (12)
                                          +- Scan csv  (11)
```

```
+- == Initial Plan ==
   TakeOrderedAndProject (41)
   +- HashAggregate (40)
      +- Exchange (39)
         +- HashAggregate (38)
            +- Project (37)
               +- BatchEvalPython (36)
                  +- Project (35)
                     +- ShuffledHashJoin LeftOuter BuildRight (34)
                        :- Exchange (31)
                        :  +- Union (30)
                        :     :- Project (27)
                        :     :  +- Filter (26)
                        :     :     +- Scan csv  (1)
                        :     +- Project (29)
                        :        +- Filter (28)
                        :           +- Scan csv  (4)
                        +- Exchange (33)
                           +- Filter (32)
                              +- Scan csv  (11)
```

```
23/12/23 14:49:20 INFO CodeGenerator: Code generated in 8.69088 ms
23/12/23 14:49:20 INFO CodeGenerator: Code generated in 9.390749 ms
```

Q4b DataFrame Shuffle Replicate Nl




```
== Physical Plan ==
AdaptiveSparkPlan (37)
+- == Final Plan ==
   TakeOrderedAndProject (21)
   +- * HashAggregate (20)
      +- AQEShuffleRead (19)
         +- ShuffleQueryStage (18), Statistics(sizeInBytes=7.4 KiB, rowCount=147)
            +- Exchange (17)
               +- * HashAggregate (16)
                  +- * Project (15)
                     +- BatchEvalPython (14)
                        +- * Project (13)
                           +- * BroadcastHashJoin LeftOuter BuildRight (12)
                              :- Union (7)
                              :  :- * Project (3)
                              :  :  +- * Filter (2)
                              :  :     +- Scan csv  (1)
                              :  +- * Project (6)
                              :     +- * Filter (5)
                              :        +- Scan csv  (4)
                              +- BroadcastQueryStage (11), Statistics(sizeInBytes=4.0 MiB, rowCount=21)
                                 +- BroadcastExchange (10)
                                    +- * Filter (9)
                                       +- Scan csv  (8)
```

```
+- == Initial Plan ==
   TakeOrderedAndProject (36)
   +- HashAggregate (35)
      +- Exchange (34)
         +- HashAggregate (33)
            +- Project (32)
               +- BatchEvalPython (31)
                  +- Project (30)
                     +- BroadcastHashJoin LeftOuter BuildRight (29)
                        :- Union (26)
                        :  :- Project (23)
                        :  :  +- Filter (22)
                        :  :     +- Scan csv  (1)
                        :  +- Project (25)
                        :     +- Filter (24)
                        :        +- Scan csv  (4)
                        +- BroadcastExchange (28)
                           +- Filter (27)
                              +- Scan csv  (8)
```

```
23/12/23 17:40:42 INFO CodeGenerator: Code generated in 13.193721 ms
23/12/23 17:40:42 INFO CodeGenerator: Code generated in 26.261717 ms
```

Broadcast Join: Αυτή η μέθοδος είναι ιδανική όταν ένα από τα σύνολα δεδομένων είναι πολύ μικρότερο από το άλλο. Το μικρότερο σύνολο δεδομένων μπορεί να χωρέσει στη μνήμη κάθε κόμβου. Ελαχιστοποιεί την ανακατανομή δεδομένων στο δίκτυο, επειδή το μικρότερο σύνολο δεδομένων μεταδίδεται σε όλους τους κόμβους. Αυτό οδηγεί σε σημαντική βελτίωση των επιδόσεων, ειδικά για μεγάλα σύνολα δεδομένων.

Merge Join: Η μέθοδος Join είναι γενικά καλή για μεγάλα σύνολα δεδομένων που είναι πολύ μεγάλα για να μεταδοθούν. Ταξινομεί τα σύνολα δεδομένων με βάση τα κλειδιά σύνδεσης και στη συνέχεια εκτελεί τη συγχώνευση. Αυτή η στρατηγική είναι αποδοτική για μεγάλα σύνολα δεδομένων, αλλά περιλαμβάνει την ανακατανομή των δεδομένων στο δίκτυο, η οποία μπορεί να είναι δαπανηρή.

Shuffle Hash Join: Χρήσιμη μέθοδος όταν και τα δύο σύνολα δεδομένων είναι μεγάλα αλλά εξακολουθούν να είναι αρκετά μικρά ώστε να χωράνε στη μνήμη όταν κατατμηθούν. Κατακερματίζει τα σύνολα δεδομένων και τα ανακατεύει στους κόμβους.

Shuffle and Replicate Nested Loop Join (Shuffle Replicate NL): Αυτή η μέθοδος ανακατεύει το ένα σύνολο δεδομένων και αναπαράγει το άλλο για κάθε διαχωριστικό. Είναι γενικά η λιγότερο αποδοτική στρατηγική σύνδεσης και χρησιμοποιείται μόνο για συγκεκριμένες περιπτώσεις όπου άλλες συνδέσεις δεν είναι εφαρμόσιμες.

Στη δική μας περίπτωση, παρατηρούμε από τα αποτελέσματα που λάβαμε ότι η Broadcast Join και η Merge Join είναι πιο αποδοτικές. Η Shuffle Hash είναι εμφανώς καλύτερη από την Shuffle Replicate NL. Αυτό συμβάινει λόγω του όγκου των δεδομένων που έχουμε να επεξεργαστούμε και του τρόπου επεξεργασίας τους κάθε φορά.