


## Queries

**Task 5:** Calculate the total number of different drivers for each customer.

**Query:-**

```
SELECT
    CUSTOMER_ID
    , COUNT(DISTINCT(DRIVER_ID))
    AS
    TOTAL_NUMBER_OF_DRIVERS
FROM
    BOOKINGS_DETAIL
GROUP BY
    CUSTOMER_ID
ORDER BY
    CUSTOMER_ID;
```

**Output:**

 Add a name... Add a description...

13.21s Database cal

```
1 --Task 5: Calculate the total number of different drivers for each customer.
2 SELECT
3     CUSTOMER_ID
4     , COUNT(DISTINCT(DRIVER_ID)) AS TOTAL_NUMBER_OF_DRIVERS
5 FROM
6     BOOKINGS_DETAIL
7 GROUP BY
8     CUSTOMER_ID
9 ORDER BY
10    CUSTOMER_ID;
11
```

INFO : Map 1: 1/1 Reducer 2: 2/2 Reducer 3: 1/1  
INFO : Completed executing command(queryId=hive\_20230604180013\_8005cfde-1d20-4b03-9dc4-1d12ffd7fb2b); Time taken: 13.216 seconds  
INFO : OK

Query History Saved Queries Query Builder Results (100+)

	customer_id	total_number_of_drivers
1	10022393	1
2	10058402	1
3	10339567	1
4	10435129	1
5	10555335	1
6	10592274	1
7	10614890	1
8	10678994	1
9	11264797	1
10	11353346	1
11	11418437	1
12	11438890	1
13	11454977	1
14	11479815	1

### Validation:

```
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2020-11-17 12:23:06,034 Stage-1 map = 0%, reduce = 0%
2020-11-17 12:23:12,394 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 3.27 sec
2020-11-17 12:23:20,727 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 7.69 sec
MapReduce Total cumulative CPU time: 7 seconds 690 msec
Ended Job = job_1605615116654_0005
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 7.69 sec HDFS Read: 43007 HDFS Write: 11000 SUCCESS
Total MapReduce CPU Time Spent: 7 seconds 690 msec
OK
10022393      1
10058402      1
10339567      1
10435129      1
10555335      1
10592274      1
10614890      1
10678994      1
11264797      1
11353346      1
11418437      1
11438890      1
11454977      1
11479815      1
11518953      1
11580321      1
11596512      1
11608791      1
11655671      1
11757536      1
11764909      1
11860278      1
11981042      1
12106105      1
12142182      1
12312603      1
12334699      1
12367832      1
12856708      1
12885363      1
12913608      1
12914577      1
12966909      1
13015449      1
13229062      1
```




**Note: Expected output is exactly matching with validation document.**

**Task 6:** Calculate the total rides taken by each customer.

**Query:-**

```
SELECT
    CUSTOMER_ID
    , COUNT(BOOKING_ID) AS TOTAL_RIDES
FROM
    BOOKINGS_DETAIL
GROUP BY
    CUSTOMER_ID
ORDER BY
    CUSTOMER_ID;
```

## Output:

 Hive  

6.36s Data

```
1 --Task 6: Calculate the total rides taken by each customer.
2 SELECT
3     CUSTOMER_ID
4     , COUNT(BOOKING_ID) AS TOTAL_RIDES
5 FROM
6     BOOKINGS_DETAIL
7 GROUP BY
8     CUSTOMER_ID;
9 ORDER BY
10    CUSTOMER_ID;
11
```

INFO : Map 1: 1/1 Reducer 2: 2/2  
INFO : Completed executing command(queryId=hive\_20230604180210\_6c863202-4127-4cb2-99e5-d34bc232f2a3); Time taken: 6.35 seconds  
INFO : OK

Query History Saved Queries Query Builder Results (100+)

	customer_id	total_rides
1	10022393	1
2	10555335	1
3	10592274	1
4	10678994	1
5	11264797	1
6	11418437	1
7	11438890	1
8	11518953	1
9	11580321	1
10	11764909	1
11	11860278	1
12	12312603	1
13	12334699	1
14	12367832	1

[book/editor?type=hive](#)

## Validation:

```
Ended Job = job_1605615116654_0008
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 6.65 sec HDFS Read: 38721 HDFS Write: 11000 SUCCESS
Total MapReduce CPU Time Spent: 6 seconds 650 msec
OK
10022393      1
10058402      1
10339567      1
10435129      1
10555335      1
10592274      1
10614890      1
10678994      1
11264797      1
11353346      1
11418437      1
11438890      1
11454977      1
11479815      1
11518953      1
11580321      1
11596512      1
11608791      1
11655671      1
11757536      1
11764909      1
11860278      1
11981042      1
12106105      1
12142182      1
12312603      1
12312603      1
```

**Note: Expected output is exactly matching with validation document.**

**Task 7:** Find the total visits made by each customer on the booking page and the total 'Book Now' button presses. This can show the conversion ratio.

The booking page id is 'e7bc5fb2-1231-11eb-adc1-0242ac120002'.

The Book Now button id is 'fcb68aa-1231-11eb-adc1-0242ac120002'. You also need to calculate the conversion ratio as part of this task. Conversion ratio can be calculated as **Total 'Book Now' Button Press/Total Visits made by customer on the booking page**.

**Query:-**

```
SELECT
SUM(CASE WHEN PAGE_ID = 'e7bc5fb2-1231-11eb-adc1-0242ac120002' THEN 1 ELSE 0 END) AS
TOTAL_PAGE_VISITS,
SUM(CASE WHEN BUTTON_ID = 'fcb68aa-1231-11eb-adc1-0242ac120002' THEN 1 ELSE 0 END) AS
TOTAL_BUTTON_PRESSED,
ROUND(CAST(SUM(CASE WHEN BUTTON_ID = 'fcb68aa-1231-11eb-adc1-0242ac120002'
THEN 1 ELSE 0 END) AS FLOAT) /
CAST(SUM(CASE WHEN PAGE_ID = 'e7bc5fb2-1231-11eb-adc1-0242ac120002' THEN 1 ELSE
0 END) AS FLOAT), 4) AS CONVERSION_RATIO FROM
CLICKSTREAM_DATA;
```

## Output:-

The screenshot shows the Hive query execution interface. At the top, there's a header with the Hive logo, a name field, a description field, and some icons. Below the header, the query is displayed in a text area. The query is a SQL statement that calculates the conversion ratio based on page visits and button presses. The results are shown in a table below the query. The table has three columns: 'total\_page\_visits', 'total\_button\_pressed', and 'conversion\_ratio'. The results show 1014 total page visits, 999 total button presses, and a conversion ratio of 0.9852.

```
--Task 7: Find the total visits made by each customer on the booking page and the total 'Book Now' button presses. This can show the conversion ratio.
--The booking page id is 'e7bc5fb2-1231-11eb-adc1-0242ac120002'.
--The Book Now button id is 'fcba68aa-1231-11eb-adc1-0242ac120002'. You also need to calculate the conversion ratio as part of this task. Conversion ratio can be calculated as
SELECT
SUM(CASE WHEN PAGE_ID = 'e7bc5fb2-1231-11eb-adc1-0242ac120002' THEN 1 ELSE 0 END) AS TOTAL_PAGE_VISITS,
SUM(CASE WHEN BUTTON_ID = 'fcba68aa-1231-11eb-adc1-0242ac120002' THEN 1 ELSE 0 END) AS TOTAL_BUTTON_PRESSED,
ROUND(CAST(SUM(CASE WHEN BUTTON_ID = 'fcba68aa-1231-11eb-adc1-0242ac120002' THEN 1 ELSE 0 END) AS FLOAT) /
CAST(SUM(CASE WHEN PAGE_ID = 'e7bc5fb2-1231-11eb-adc1-0242ac120002' THEN 1 ELSE 0 END) AS FLOAT), 4) AS CONVERSION_RATIO
FROM CLICKSTREAM_DATA;
```

INFO : Map 1: 1/1 Reducer 2: 0(+1)/1  
INFO : Map 1: 1/1 Reducer 2: 1/1  
INFO : Completed executing command(queryId=hive\_20230605064221\_1f505f83-68e4-4dbc-8b83-f910b01c5ea4); Time taken: 5.829 seconds  
INFO : OK

	total_page_visits	total_button_pressed	conversion_ratio
1	1014	999	0.9852

## Validation:-

- When you run the query to get the conversion ratio, you should get the conversion ratio as **0.9688**.

**Note: Slightly difference in conversion ratio that is because ~16 event were captured from kafka stream.**

**Task 8:** Calculate the count of all trips done on black cabs.

## Query:-

```
SELECT
COUNT(BOOKING_ID) AS TOTAL_TRIPS_BY_BLACK_CABS
FROM
BOOKINGS_DETAIL
WHERE
CAB_COLOR = black;
```

## Output:

The screenshot shows the Hive query interface. At the top, there's a header with the Hive logo and options to 'Add a name...' and 'Add a description...'. Below this, the query is displayed in a text area:

```
--Task 8: Calculate the count of all trips done on black cabs.
SELECT
COUNT(BOOKING_ID) AS TOTAL_TRIPS_BY_BLACK_CABS
FROM
BOOKINGS_DETAIL
WHERE
CAB_COLOR = 'black';
```

Below the query, the execution status is shown: 'INFO : Map 1: 1/1 Reducer 2: 1/1', 'INFO : Completed executing command(queryId=hive\_20230604171120\_06f3d537-071d-4cae-8b5b-caf3ca6c4b34); Time taken: 6.432 seconds', and 'INFO : OK'. The results are displayed in a table with one row: 1 72.

total_trips_by_black_cabs	
1	72

## Validation:

- When you run the query to get the conversion ratio, you should get the conversion ratio as **0.9688**.
- Count of all trips done on black cabs - **72**.
- When you run the query to get the total amount of tips given date wise to all drivers by customers, you would get an output as shown below:


**Note: Number of trips are exactly matching with validation document.**


**Task 9:** Calculate the total amount of tips given date wise to all drivers by customers.

**Query:-**

```
SELECT
DATE(PICKUP_TIMESTAMP)
TRIP_DATE,
ROUND(SUM(TIP_AMOUNT),0)
AS TOTAL_TIP_AMOUNT
FROM
BOOKINGS_DETAIL
GROUP BY
DATE(PICKUP_TIMESTAMP)
ORDER BY
TRIP_DATE;
```

## Output:

 **Hive**

 **Add a name...** **Add a description...**

6.60s Database cabr

```
1 --Task 9: Calculate the total amount of tips given date wise to all drivers by customers.
2 SELECT
3     DATE(PICKUP_TIMESTAMP) TRIP_DATE
4     , ROUND(SUM(TIP_AMOUNT),0) AS TOTAL_TIP_AMOUNT
5 FROM
6     BOOKINGS_DETAIL
7 GROUP BY
8     DATE(PICKUP_TIMESTAMP)
9 ORDER BY
10    TRIP_DATE;
```

INFO : Map 1: 1/1 Reducer 2: 2/2 Reducer 3: 1/1

INFO : Completed executing command(queryId=hive\_20230604175432\_a2240e14-5494-48d1-ad8d-2bb17209c5b0); Time taken: 6.591 seconds

INFO : OK

applicatio

Query History

Saved Queries

Query Builder

Results (100+)

	trip_date	total_tip_amount
1	2020-01-01	59
2	2020-01-02	95
3	2020-01-03	11
4	2020-01-04	123
5	2020-01-05	134
6	2020-01-06	189
7	2020-01-07	148
8	2020-01-08	111
9	2020-01-09	48
10	2020-01-10	77
11	2020-01-11	81
12	2020-01-12	109
13	2020-01-14	142
14	2020-01-15	338
15	2020-01-16	155

Validation:

2020-01-01	59
2020-01-02	95
2020-01-03	11
2020-01-04	123
2020-01-05	134
2020-01-06	189
2020-01-07	148
2020-01-08	111
2020-01-09	48
2020-01-10	77
2020-01-11	81
2020-01-12	109
2020-01-14	142
2020-01-15	338
2020-01-16	155
2020-01-17	296
2020-01-18	240
2020-01-20	210
2020-01-21	5
2020-01-23	148
2020-01-24	472
2020-01-25	98
2020-01-26	209
2020-01-27	231
2020-01-28	567
2020-01-29	123
2020-01-30	112
2020-01-31	256
2020-02-01	317
2020-02-02	338
2020-02-03	191
2020-02-04	258
2020-02-05	212
2020-02-06	154
2020-02-07	91
2020-02-08	270

Note: Total amount of tips is exactly matching with validation document.




**Task 10:** Calculate the total count of all the bookings with ratings lower than 2 as given by customers in a particular month.

**Query:-**

```
SELECT
DATE_FORMAT(PICKUP_TIMESTAMP,
'yyyy-MM') TRIP_MONTH,
COUNT(BOOKING_ID) AS
NO_OF_BOOKINGS
FROM
BOOKINGS_DETAIL
WHERE RATING_BY_CUSTOMER < 2
GROUP BY
DATE_FORMAT(PICKUP_TIMESTAMP,
'yyyy-MM')
ORDER BY
TRIP_MONTH;
```

**Output:**

 Add a name... Add a description...

6.59s Database

```
1 --Task 10: Calculate the total count of all the bookings with ratings lower than 2 as given by customers in a particular month.
2
3 SELECT
4   DATE_FORMAT(PICKUP_TIMESTAMP, 'yyyy-MM') TRIP_MONTH
5   , COUNT(BOOKING_ID) AS NO_OF_BOOKINGS
6 FROM
7   BOOKINGS_DETAIL
8 WHERE
9   RATING_BY_CUSTOMER < 2
10 GROUP BY
11   DATE_FORMAT(PICKUP_TIMESTAMP, 'yyyy-MM')
12 ORDER BY
13   TRIP_MONTH;
```

INFO : Map 1: 1/1 Reducer 2: 2/2 Reducer 3: 1/1  
INFO : Completed executing command(queryId=hive\_20230604182704\_fc36cbb6-1096-4692-ad8f-93f842f89e84); Time taken: 6.583 seconds  
INFO : OK

Query History Saved Queries Query Builder Results (10)

	trip_month	no_of_bookings
1	2020-01	26
2	2020-02	16
3	2020-03	16
4	2020-04	21
5	2020-05	21
6	2020-06	14
7	2020-07	20
8	2020-08	32
9	2020-09	21
10	2020-10	15

### Validation:

```
Total MapReduce CPU Time Spent: 7 seconds 970 msec
OK
2020-01 26
2020-02 16
2020-03 16
2020-04 21
2020-05 21
2020-06 14
2020-07 20
2020-08 32
2020-09 21
2020-10 15
```

**Note: Count of bookings by month is exactly matching with validation document.**

**Task 11:** Calculate the count of total iOS users.

### Query:-

```
SELECT
    COUNT(DISTINCT(CUSTOMER_ID)) AS TOTAL_IOS_USERS
FROM
    CLICKSTREAM_DATA
WHERE
    OS_VERSION = 'iOS';
```

### Output:-

The screenshot shows the Hive query execution interface. At the top, there's a header with the Hive logo, a name field, a description field, and some icons. Below this, the query is displayed in a text area. To the right of the query, there's a status bar showing '8.30s', 'Database cabrides', 'Type text', and a settings icon. Below the query, there's a section for execution logs. The logs show the progress of the query execution, including the number of maps and reducers, and the time taken. At the bottom, there's a section for the results. The results are displayed in a table with one column, 'total\_ios\_users', and one row with the value '1515'.

```
1 --Task 11: Calculate the count of total iOS users.
2 SELECT
3     COUNT(DISTINCT(CUSTOMER_ID)) AS TOTAL_IOS_USERS
4 FROM
5     CLICKSTREAM_DATA
6 WHERE
7     OS_VERSION = 'iOS';
8
9
```

INFO : Map 1: 1/1 Reducer 2: 0/1 Reducer 3: 0/1  
INFO : Map 1: 1/1 Reducer 2: 2/2 Reducer 3: 0(+1)/1  
INFO : Map 1: 1/1 Reducer 2: 2/2 Reducer 3: 1/1  
INFO : Completed executing command(queryId=hive\_20230605062817\_49897e97-b2cd-4bf3-89f6-554554c04714); Time taken: 7.015 seconds  
INFO : OK

Query History Saved Queries Query Builder Results (1)

total_ios_users	
1	1515

**Validation:**

7. You should get the count of all iOS users as **1503**.

Note: 3004 event were captured from kafka stream where in validation documents its 2984 hence iOS user are slightly more in our analysis.