Ranking in Hive

...

Rank, dense rank, row number, partition by

IMPORTANT

Copyright Infringement and Illegal Content Sharing Notice

Let us first create the table and load the data

Note: We will be using rank_dataset.csv as our dataset for practicals.

```
create table rank_test(name string,score int) row format delimited fields terminated by ',' lines terminated by '\n' stored as textfile;

load data local inpath '/home/cloudera/Downloads/rank_dataset.csv' into table rank_test;
```

Screenshots of previous commands

```
hive> use trendytech;
OK
Time taken: 0.033 seconds
hive> create table rank_test(name string,score int) row format delimited fields ter
minated by ',' lines terminated by '\n'stored as textfile;
OK
Time taken: 0.06 seconds
hive>
```

```
[cloudera@quickstart Downloads]$ gedit rank_dataset.csv
[cloudera@quickstart Downloads]$ cat rank_dataset.csv
John,1500
Albert,1500
Mark,1000
Frank,1150
Loopa,1100
Lui,1300
John,1300
John,900
Lesa,1500
Lesa,900
Pars,800
leo,700
leo,1500
lock,650
Bhut,800
Lio,500
[cloudera@quickstart Downloads]$
```

```
hive> load data local inpath '/home/cloudera/Downloads/rank_dataset.csv' into table
 rank_test;
Loading data to table trendytech.rank_test
Table trendytech.rank_test stats: [numFiles=1, totalSize=153]
OK
Time taken: 0.719 seconds
hive>
```

Let us call the rank function on table

```
select name, score, rank() over(order by score desc) as ranking from rank_test;
```

leo	1500	1
Albert	1500	1
Lesa	1500	1
John	1500	1
John	1300	5
Lui	1300	5
Frank	1150	7
Loopa	1100	8
Mark	1000	9
John	900	10
Lesa	900	10
Pars	800	12
Bhut	800	12
leo	700	14
lock	650	15
Lio	500	16

Ranking based on descending order of score (column 2). Highest score will be rank 1

Ties are assigned same rank with next ranking skipped.

Dense rank

Dense Rank gives ranking. ranks are consecutive no ranks are skipped even in case of ties.

select name, score, dense_rank() over(order by score desc) as ranking from rank_test;

```
OK
leo     1500    1
Albert  1500    1
Lesa    1500    1
John    1500    1
John    1300    2
Lui     1300    2
Frank   1150    3
Loopa   1100    4
Mark    1000    5
John    900     6
Lesa    900     6
Pars    800     7
Bhut    800     7
leo     700     8
lock    650     9
Lio     500     10
Time taken: 20.895 seconds, Fetched: 16 row(s)
```

Row number

Row number assigns unique numbers to each row given the order by clause.

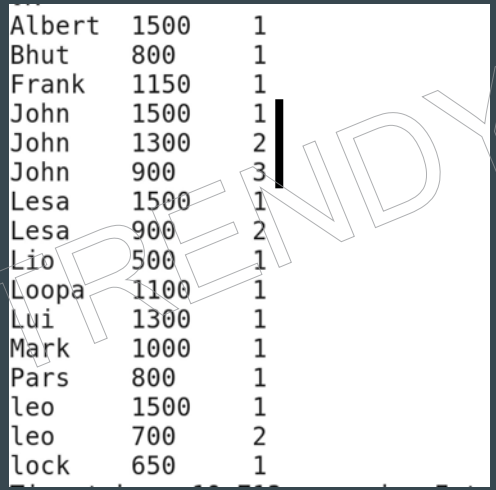
```
select name, score, row_number() over(order by score desc) as ranking from rank_test;
```

```
leo     1500    1
Albert  1500    2
Lesa    1500    3
John    1500    4
John    1300    5
Lui     1300    6
Frank   1150    7
Loopa   1100    8
Mark    1000    9
John    900     10
Lesa    900     11
Pars    800     12
Bhut    800     13
leo     700     14
lock    650     15
Lio     500     16
```

Row number with partition by clause

select name, score, row_number() over(partition by name order by score desc) as ranking from rank_test;

The above query first groups based on names and then do the ranking for each group.



Albert	1500	1
Bhut	800	1
Frank	1150	1
John	1500	1
John	1300	2
John	900	3
Lesa	1500	1
Lesa	900	2
Lio	500	1
Loopa	1100	1
Lui	1300	1
Mark	1000	1
Pars	800	1
leo	1500	1
leo	700	2
lock	650	1

We have learnt ranking in hive

Happy Learning!!!

Google REVIEWS

5 Star Google Rated
Big Data Course

LEARN FROM THE EXPERT

9108179578

Call for more details

Follow US

Trainer	Mr. Sumit Mittal
Phone	9108179578
Email	trendytech.sumit@gmail.com
Website	https://trendytech.in/courses/big-data-online-training/
LinkedIn	https://www.linkedin.com/in/bigdatabysumit/
Twitter	@BigdataBySumit
Instagram	bigdatabysumit
Facebook	https://www.facebook.com/trendytech.in/
Youtube	https://www.youtube.com/channel/UCbTggJVf0NDTfWX-C_gUGSg