Hive Assignment

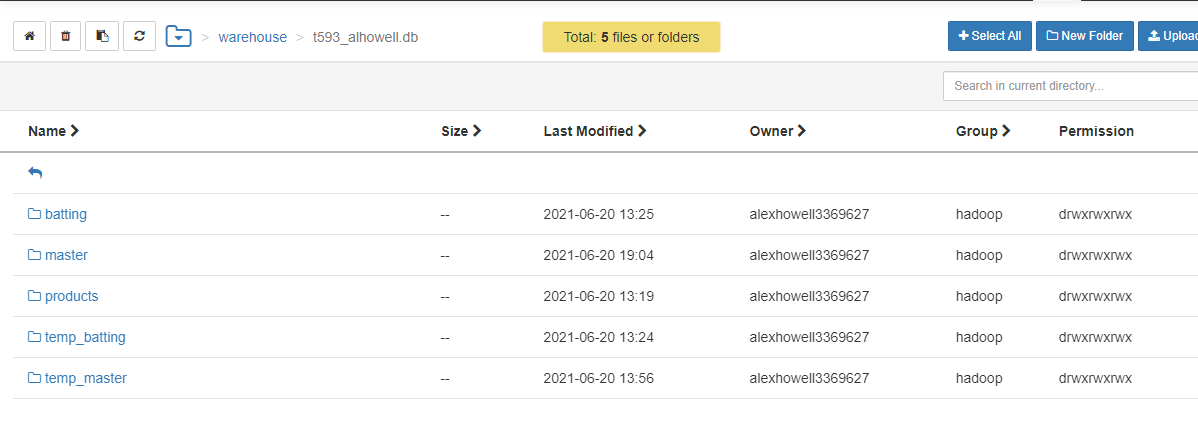
Team 5:

Alex Howell

Ulrich Wartes

# Question 1

Screenshot of directories:



# Question 2

## 2a

Query:

SELECT year, Max(runs), Min(runs), avg(runs)

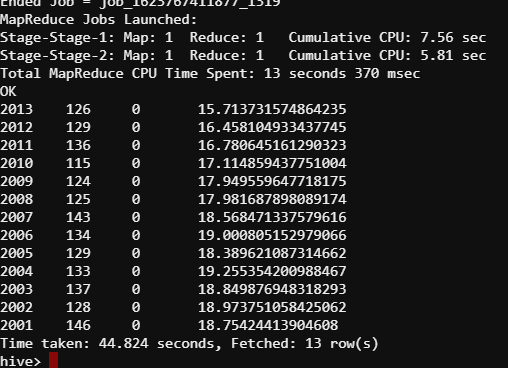
from batting

WHERE year > 2000

GROUP by year

Order by year desc;

Screenshot:



## 2b

Query:

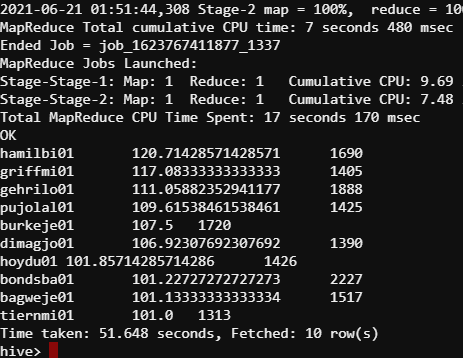
SELECT player\_id, avg(runs) as average, sum(runs) as total

From batting

group by player\_id

Order by average desc

LIMIT 10;

Screenshot:  


## 2c

Query:

SELECT m.fname, m.lname, avg(b.runs) as avgRuns, sum(b.runs) as sum\_runs

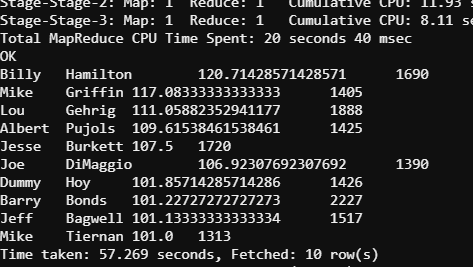
FROM batting b

JOIN master m ON m.player\_id = b.player\_id

GROUP BY b.player\_id, m.fname, m.lname

ORDER BY avgRuns DESC limit 10;

Screenshot:



# Question 3

## 3a

Query:

CREATE EXTERNAL TABLE IF NOT EXISTS

ad\_data(campaign\_id STRING,

ad\_date STRING,

ad\_time STRING,

ad\_keyword STRING,

ad\_display\_site STRING,

ad\_placement STRING,

ad\_was\_clicked INT,

ad\_cpc INT )

COMMENT 'hdfs ad\_data'

ROW FORMAT DELIMITED

FIELDS TERMINATED BY '\t'

LOCATION '/user/alexhowell3369627/Homework\_1/ad\_network\_data\_etl/step\_2';

Screenshot (of count query):

Text

Description automatically generated

## 3b

### Question 3 from Pig Assignment:

Query:

SELECT ad\_display\_site, SUM(ad\_cpc) as Total from ad\_data

WHERE ad\_was\_clicked=1

Group by ad\_display\_site

ORDER by Total ASC

LIMIT 3;

Screenshot:

Text

Description automatically generated with medium confidence

### Question 4 from Pig Assignment:

Query:

SELECT ad\_keyword, SUM(ad\_cpc) as total from ad\_data

WHERE ad\_was\_clicked=1

Group by ad\_keyword

ORDER by total DESC

LIMIT 5;

Screenshot:

Text

Description automatically generated

# Question 4

## Loading the data:

### Movies (I used the fixed file so that I could verify against excel)

CREATE TABLE movies (

movieId INT,

title STRING,

genres ARRAY <STRING>)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY ','

COLLECTION ITEMS TERMINATED BY '|';

LOAD DATA INPATH '/user/alexhowell3369627/movies\_fixed.csv' OVERWRITE INTO TABLE movies;

### Links

CREATE TABLE links (

movieid INT,

imdbid INT,

tmdbid INT)

ROW FORMAT DELIMITED FIELDS

TERMINATED BY ',';

LOAD DATA INPATH '/user/alexhowell3369627/links.csv' INTO TABLE links;

### Ratings

CREATE TABLE ratings (

userId INT,

movieId INT,

rating INT,

times INT)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY ',' ;

LOAD DATA INPATH '/user/alexhowell3369627/ratings.csv' OVERWRITE INTO TABLE ratings;

### Tags

CREATE TABLE tags (

userId INT,

movieId INT,

tag STRING,

times INT)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY ',' ;

LOAD DATA INPATH '/user/alexhowell3369627/tags.csv' OVERWRITE INTO TABLE tags;

## 4-1

Query (note: I alphabetized my list):

SELECT m.title FROM movies m

LATERAL VIEW EXPLODE(m.genres) t as genre

WHERE genre='Adventure'

ORDER by m.title ASC;

Screenshot:

Text

Description automatically generated

## 4-2

Query (I also ordered mine by most genres to fewest):

SELECT m.title, count(m.genres) as GenreCount FROM movies m

LATERAL VIEW EXPLODE(m.genres) t as genre

GROUP by m.title

HAVING GenreCount > 2

ORDER BY GenreCount DESC;

Screenshot:

Text

Description automatically generated

## 4-3

Query:

SELECT b.title, AVG(r.rating) as AverageRating

from ratings r

INNER JOIN

(SELECT \* FROM movies m

LATERAL VIEW EXPLODE(m.genres) t as genre

WHERE genre='Children') b

on r.movieid=b.movieid

GROUP by b.title

HAVING AverageRating > 3;

Screenshot:

Text

Description automatically generated

## 4-4

Query:

SELECT r.userid, AVG(r.rating) AverageRating from ratings r

GROUP by r.userid

HAVING AverageRating > 3;

Screenshot:

Text

Description automatically generated

## 4-5

Query:

SELECT m.title, AVG(r.rating) as AverageRating from movies m

INNER JOIN ratings r on m.movieid=r.movieid

WHERE m.title like '%Star Wars%'

GROUP BY m.title

ORDER BY AverageRating DESC

LIMIT 1;

Screenshot:

Text

Description automatically generated

## 4-6

Query:

SELECT AVG(r.rating) as AverageRating from movies m

INNER JOIN ratings r on m.movieid=r.movieid

WHERE m.title like '%2008%';

Screenshot:

Text

Description automatically generated