COMP30520 Practical 6

[redacted]

November 2020

Exercise 1

Exercise 1 - Question 1

```
movie_data = LOAD ’hdfs://localhost:8020/user/Movie_Log’
    USING PigStorage(’,’)
    AS (id: int,
    rate_movie: chararray, rating: int,
    completed_movie: chararray, genre: chararray,
    pause_movie: chararray, start_movie: chararray,
    browse_movie: chararray, list_movie: chararray,
    search_movie: chararray, login: chararray,
    logout: chararray, incomplete_movie: chararray,
    purchase_movie: chararray);

DUMP movie_data;
```

Exercise 1 - Question 2

```
rate_group = GROUP movie_data BY rating;
DUMP rate_group
```

Exercise 1 - Question 3

```
DESCRIBE rate_group;
```

The DESCRIBE function shows use the Schema used for the Data we Describe. This Schema shows what the group is made of and the datatypes of the variables inside the Schema.

Exercise 1 - Question 4

```
rate_group = FOREACH rate_group GENERATE group AS rating, movie_data.rate_movie;
```

Exercise 1 - Question 5

```
grpd = GROUP movie_data BY genre;
max_rtd = FOREACH grpd {
        ordrd = ORDER movie_data BY rating DESC;
        top = LIMIT ordrd 1;
        GENERATE top;
}
DUMP max_rtd;
```

Exercise 1 - Question 6

```
selected_clicks = FOREACH movie_data GENERATE start_movie, browse_movie, completed_movie,
    purchased_movie;
```

Exercise 2

Exercise 2 - Question 1

```
student_data = LOAD 'hdfs://localhost:8020/user/students.csv'
    USING PigStorage(',')
    AS (id: int, first_name: chararray,
        last_name: chararray, student_id: int,
        email: chararray, stage: int,
        degree: chararray);

DESCRIBE student_data
```

Exercise 2 - Question 2

```
students_details = LOAD 'hdfs://localhost:8020/user/students.csv' USING PigStorage(',');
students_details = FOREACH students_details generate $3 as student_id,
                                        $1 as first_name,
                                        $2 as last_name,
                                        $4 as email,
                                        $5 as stage,
                                        $6 as degree;

STORE renamed INTO 'hdfs://localhost:8020/pig_Output' USING PigStorage(',');
```

```
shell> hadoop fs -getmerge /user/pigoutput ./students_details.csv
```

Exercise 2 - Question 3

```
student_data = LOAD 'hdfs://localhost:8020/user/students_attendance.csv'
    USING PigStorage(',')
    AS (id: int, student_id: int, hours_attended: int);

DESCRIBE student_data;

/*find all students that have attended*/
SA_details = FILTER student_data BY hours_attended > 1;
STORE SA_details INTO 'hdfs://localhost:8020/pig_OutputSA' USING PigStorage(',');
```

```
shell> hadoop fs -getmerge /user/pigoutput ./students_attendance.csv
```

Exercise 2 - Question 4

```
attendance = LOAD 'hdfs://localhost:8020/user/students_attendance.csv' USING PigStorage(',')  AS (id: int, student_id: int, hours_attended: int);
data = GROUP attendance BY student_id;
result = FOREACH data GENERATE group, SUM(attendance.hours_attended);
DUMP result;
```

Exercise 2 - Question 5

```
student_data = LOAD ’hdfs://localhost:8020/user/students.csv’
    USING PigStorage(’,’)
    AS (id: int, first_name: chararray,
        last_name: chararray, student_id: int,
        email: chararray, stage: int,
        degree: chararray);

attendance = LOAD ’hdfs://localhost:8020/user/students_attendance.csv’
    USING PigStorage(’,’)
    AS (id: int, student_id: int, hours_attended: int);

joined = JOIN student_data BY student_id LEFT OUTER, attendance BY student_id;

result = FOREACH joined GENERATE student_data::student_id AS student_id,
    student_id::first_name AS name,
    attendance::hours_attended AS hours_attended;
```

Exercise 2 - Question 6

```
DUMP result;
```

And

```
ILLUSTRATE result;
```