

Assignment # 2

Data Transformations Activity

Course name	Data Analysis Tools for Analytics
Course code	LA-DTA 1054
Course facilitator	Kenton White
Student name	Muhammad Qureshi
Student ID	c0813038
Date of Submission due	Nov 01, 2021

Question1.

Create a table called sales _genre

The screenshot shows the Apache Tez UI interface. At the top, there are tabs for QUERY, JOBS, TABLES, SAVED QUERIES, UDFs, and SETTINGS, along with a NOTIFICATIONS icon. Below the tabs, there are three open worksheets: Worksheet1*, Worksheet2*, and Worksheet3*. The Worksheet3* tab is active, showing the following SQL query:

```
1 create table sales_genre like sales
```

To the right of the query editor is a sidebar titled "default" which lists tables in the current database. The sidebar shows three tables: hivesampletable, sales, and sales_genre. A red arrow points from the cursor area towards the sales_genre table in the sidebar. At the bottom of the interface, there are buttons for Execute, Save As, Insert UDF, Visual Explain, and a RESULTS tab. The RESULTS tab is currently selected.

Schema:

```
1 create table sales_genre like sales
```

Load table sales into this table sales_genre

The screenshot shows a data processing interface with a query editor and a results table.

Query Editor:

```
1 create table sales_genre like sales
2
3 ROW FORMAT DELIMITED FIELDS TERMINATED BY ','
4   LINES TERMINATED BY '\n'
5 STORED AS TEXTFILE LOCATION'/sales'
6 TBLPROPERTIES (
7   "skip.header.line.count"="1");
8
9 select * from sales_genre limit 10;
```

Results Table:

genre.publisher	sales_genre.na_sales	sales_genre.eu_sales	sales_genre.jp_sales	sales_genre.other_sales	sales_genre.global_sales	sales_genre.critic_score	sales_genre.c
30	41	28	3	8	82.53	76	51
30	29	3	6	0	40.24	null	null
30	15	12	0	0	25.59	02	79

Schema:

```
1 create table sales_genre like sales
2
3 ROW FORMAT DELIMITED FIELDS TERMINATED BY ','
4   LINES TERMINATED BY '\n'
5 STORED AS TEXTFILE LOCATION'/sales'
6 TBLPROPERTIES (
7   "skip.header.line.count"="1");
8
9 select * from sales_genre limit 10;
```

Select this column: **Genre, Global_sale, Critic_score.**

The screenshot shows a Hive query editor interface. On the left, a code editor displays the following SQL-like code:

```
1 CREATE TABLE `sales_genre`(`genre` string,`global_sales` float,`critic_score` int)ROW FORMAT DELIMITED FIELDS TERMINATED BY ','LINES TERMINATED BY '\n'STORED AS TEXTFILE LOCATION'/sales_genre'TBLPROPERTIES (`skip.header.line.count`="1");select * from sales_genre limit 10;
```

On the right, a sidebar titled "default" shows a list of tables: "hivesampletable", "sales", and "sales genre". Below the sidebar, the results of the query are displayed in a table:

sales_genre.genre	sales_genre.global_sales	sales_genre.critic_score
Sports	82.50	76
Platform	40.24	null
Racing	35.52	82
Sports	32.77	80
Role-Playing	31.37	null
Puzzle	30.26	null

Schema:

```
1 CREATE TABLE `sales_genre`(`genre` string,`global_sales` float,`critic_score` int)ROW FORMAT DELIMITED FIELDS TERMINATED BY ','LINES TERMINATED BY '\n'STORED AS TEXTFILE LOCATION'/sales_genre'TBLPROPERTIES (`skip.header.line.count`="1");select * from sales_genre limit 10;
```



Round the data found in Global_sales

The screenshot shows the Apache Hive interface. In the top-left, there's a code editor with the following SQL query:

```
1 select round (global_sales,'0')
2 from sales_genre;
3 select * from sales_genre limit 10;
```

To the right of the code editor is a sidebar titled "default" which lists four tables: hivesampletable, sales, sales_genre (which is highlighted in yellow), and sales_genre1.

Below the code editor, there are several buttons: "Execute" (highlighted in green), "Save As", "Insert UDF", "Visual Explain", and "TEZ UI".

The main area displays the results of the query, with the title "RESULTS". The results are presented in a table:

sales_genre.genre	sales_genre.global_sales	sales_genre.critic_score
Sports	82	76
Platform	40	null
Racing	35	82
Sports	32	80
Role Playing	31	null
Puzzle	30	null

Schema:

```
1 select round (global_sales,'0')
2 from sales_genre;
3 select * from sales_genre limit 10;
```

Filter the data to only look at those items in the Critic_score column that are greater than 0

The screenshot shows a database interface with two main sections: a top section for viewing results and a bottom section for writing queries.

Top Section (RESULTS):

- Buttons: Execute, Save As, Insert UDF, Visual Explain.
- Panel: Filter columns (with a clear button).
- Table:

sales_genre.genre	sales_genre.global_sales	sales_genre.critic_score
Sports	82.53	76
Racing	35.52	82
Sports	32.77	80
Platform	29.8	89
Misc	28.92	98
Platform	28.32	87
Racing	23.21	91
Sports	22.7	80
Misc	21.01	61
Sports	21.79	80
Action	21.04	97
Action	20.81	95
Misc	20.15	77
Action	16.27	97
Action	16.15	95

Bottom Section (DATABASE):

- Buttons: Execute, Save As, Insert UDF, Visual Explain.
- Panel: Filter columns (with a clear button).
- Code Editor:

```
1 select
2 *
3 from
4 sales_genre
5 where
6 critic_score >= 0
7 ;
8
```
- Database Browser:
 - Selected Database: default
 - Tables:
 - Search Tables
 - hivesamplable
 - sales
 - sales_genre** (highlighted)
 - sales_genre1

Schema:

```
1 select
2 *
3 from
4 sales_genre
5 where
6 critic_score >= 0
7 ;
8
```

Order the data in the critic _ score column from highest to lowest.

The screenshot shows the Apache Tez UI interface. At the top, there is a database browser window with a query editor containing the following SQL:

```
1 select * from sales_genre order by critic_score DESC
```

The results pane displays the sorted data:

sales_genre.genre	sales_genre.global_sales	sales_genre.critic_score
Action	11.01	98
Sports	4.68	98
Fighting	0.34	98
Action	10.5	98
Action	16.27	97
Action	1.09	97
Action	21.04	97
Action	13.1	97
Platform	7.51	97
Shooter	6.43	97
Shooter	2.82	97
Sports	4.41	97
Action	5.48	97
Platform	11.35	97
Action	12.61	97
Shooter	1.82	96
Action	1.69	96
Action	1.59	96

Schema:

```
1
2 select * from sales_genre order by critic_score DESC
3
4
```