

Assignment - Data Aggregation Activity (10%)

- 1) Create two tables, one showing the number of salespeople with attrition and the other showing the number of salespeople with non-attrition.

Salespeople with attrition:

The image displays two screenshots of the Ambari Hive View 2.0 interface, showing the process of creating and executing Hive queries.

Top Screenshot: The interface shows the 'HIVE' section with the 'QUERY' tab selected. The 'DATABASE' dropdown is set to 'default'. The query editor contains the following SQL code:

```
1 CREATE TABLE attrition_yes
2 (
3   Attrition string,
4   Department string,
5   MonthlyIncome int
6 );
7
```

The 'Tables(2)' panel on the right shows the 'employee' and 'hivesampletable' tables.

Bottom Screenshot: The interface shows the 'HIVE' section with the 'QUERY' tab selected. The 'DATABASE' dropdown is set to 'default'. The query editor contains the following SQL code:

```
1 INSERT OVERWRITE TABLE attrition_yes
2 SELECT Attrition, Department, ROUND(MonthlyIncome,-3)
3 FROM employee
4 WHERE Attrition LIKE "Yes%" AND Department LIKE "Sales%";
5
```

The 'Tables(5)' panel on the right shows the 'attrition_all', 'attrition_no', and 'attrition_yes' tables.

I have assumed this activity picks up after the Data Transform Activity and have therefore rounded the Monthly Income data to the nearest thousandth.

Salespeople with non-attrition:

The screenshot shows the Ambari Hive View 2.0 interface. The top navigation bar includes a home icon, the text 'Hive View 2.0', the user 'gillianb', and a dropdown menu for 'admin'. A green notification bar at the top states 'Query has been submitted. (details)'. The main interface is divided into a left sidebar with navigation links (Dashboard, Services, HDFS, YARN, MapReduce2, Tez, Hive, Oozie, ZooKeeper, Ambari Metrics, WebHCat, Hosts, Alerts) and a central workspace. The workspace has tabs for QUERY, JOBS, TABLES, SAVED QUERIES, UDFs, and SETTINGS. The 'QUERY' tab is active, showing a worksheet named 'Worksheet1'. The 'DATABASE' dropdown is set to 'default'. The query editor contains the following SQL code:

```
1 CREATE TABLE attrition_no
2 (
3   Attrition string,
4   Department string,
5   MonthlyIncome int
6 );
7
```

Below the query editor are buttons for 'Execute', 'Save As', 'Insert UDF', and 'Visual Explain'. On the right, a 'Tables(2)' pane shows a search bar and a list of tables: 'employee' and 'hivesampletable'.

The screenshot shows the Ambari Hive View 2.0 interface after the query has been executed. The 'QUERY' tab is still active, and the query editor now contains the following SQL code:

```
1 INSERT OVERWRITE TABLE attrition_no
2 SELECT Attrition, Department, ROUND(MonthlyIncome,-3)
3 FROM employee
4 WHERE Attrition LIKE "%No%" AND Department LIKE "%Sales%";
5
6
```

The 'Execute' button is highlighted in green, indicating the query was successful. The right-hand 'Tables(5)' pane now shows a list of tables: 'attrition_all', 'attrition_no', and 'attrition_yes', in addition to the previously visible tables.

I have assumed this activity picks up after the Data Transform Activity and have therefore rounded the Monthly Income data to the nearest thousandth.

- 2) Create three statistics tables showing the monthly income for all salespeople, those with attrition and those without attrition.

All salespeople:

The screenshot shows the Ambari Hive View 2.0 interface. On the left is a sidebar with navigation links: Dashboard, Services (HDFS, YARN, MapReduce2), Tez, Hive, Oozie, ZooKeeper, Ambari Metrics, WebHcat, Hosts, Alerts, Cluster Admin, Stack and Versions, Service Accounts, and Service Auto Start. The main area is titled 'HIVE' and has tabs for QUERY, JOBS, TABLES, SAVED QUERIES, UDFs, and SETTINGS. The 'QUERY' tab is active, showing a worksheet with a SQL query:

```
1 SELECT Department, AVG(ROUND(MonthlyIncome,-3)) AS average_monthly_income, MIN(ROUND(MonthlyIncome,-3)) AS min_monthly_income, MAX(ROUND(MonthlyIncome,-3)) AS max_monthly_income
2 FROM employee
3 WHERE Department LIKE "%Sales%"
4 GROUP BY Department;
```

Below the query editor are buttons for 'Execute', 'Save As', 'Insert UDF', and 'Visual Explain'. The 'RESULTS' tab is active, showing a table with the following data:

department	average_monthly_income	min_monthly_income	max_monthly_income
Sales	6959.641255605381	1000	20000

On the right side of the interface, there is a 'default' database dropdown and a 'Tables(5)' list containing: attrition_all, attrition_no, attrition_yes, employee (highlighted), and hivesampletable.

I have assumed this activity picks up after the Data Transform Activity and have therefore rounded the Monthly Income data to the nearest thousandth.

Those with attrition:

The screenshot shows the Ambari Hive View 2.0 interface. The left sidebar contains navigation links for Dashboard, Services (HDFS, YARN, MapReduce2, Tez, Hive, Oozie, ZooKeeper, Ambari Metrics, WebHCat), Hosts, Alerts, Cluster Admin, Stack and Versions, Service Accounts, and Service Auto Start. The main panel is titled 'HIVE' and includes tabs for QUERY, JOBS, TABLES, SAVED QUERIES, UDFs, and SETTINGS. A 'Worksheet1' tab is active. The 'DATABASE' dropdown is set to 'default'. The query editor contains the following SQL:

```
1 SELECT Department, AVG(MonthlyIncome) AS average_monthly_income, MIN(MonthlyIncome) AS min_monthly_income, MAX(MonthlyIncome) AS max_monthly_income
2 FROM attrition_yes
3 GROUP BY Department;
```

Buttons for 'Execute', 'Save As', 'Insert UDF', and 'Visual Explain' are visible. Below the query, the 'RESULTS' tab is active, displaying a table with the following data:

department	average_monthly_income	min_monthly_income	max_monthly_income
Sales	5913.04347826087	1000	20000

On the right, a 'Tables(5)' list shows 'attrition_all', 'attrition_no', 'attrition_yes', 'employee' (highlighted), and 'hivesampletable'.

Those without attrition:

The screenshot shows the Ambari Hive View 2.0 interface with the same layout as the previous one. The 'DATABASE' dropdown is still 'default'. The query editor contains the following SQL:

```
1 SELECT Department, AVG(MonthlyIncome) AS average_monthly_income, MIN(MonthlyIncome) AS min_monthly_income, MAX(MonthlyIncome) AS max_monthly_income
2 FROM attrition_no
3 GROUP BY Department;
```

The 'RESULTS' tab is active, displaying a table with the following data:

department	average_monthly_income	min_monthly_income	max_monthly_income
Sales	7231.6384180790965	1000	20000

The 'Tables(5)' list on the right is identical to the previous screenshot, with 'employee' highlighted.

- 3) Create two tables (one for salespeople with attrition and one for salespeople without attrition) showing the monthly income and the count of salespeople making that amount of income.

Salespeople with attrition:

The screenshot displays the Ambari Hive View 2.0 interface. On the left is a dark sidebar with navigation links: Dashboard, Services (HDFS, YARN, MapReduce2, Tez, Hive, Oozie, ZooKeeper, Ambari Metrics, WebHcat), Hosts, Alerts, Cluster Admin, Stack and Versions, Service Accounts, and Service Auto Start. The main panel is titled 'HIVE' and includes tabs for QUERY, JOBS, TABLES, SAVED QUERIES, UDFs, and SETTINGS. A 'Worksheet1' tab is active. The 'DATABASE' dropdown is set to 'default'. The SQL query editor contains the following code:

```
1 SELECT MonthlyIncome AS monthlyincome, COUNT(MonthlyIncome) AS count
2 FROM attrition_yes
3 GROUP BY MonthlyIncome;
4
```

Below the query editor are buttons for 'Execute', 'Save As', 'Insert UDF', and 'Visual Explain'. The 'RESULTS' tab is selected, showing a table with two columns: 'monthlyincome' and 'count'. The results are as follows:

monthlyincome	count
1000	4
6000	10
8000	5
9000	3
10000	10
13000	2
19000	1
20000	1
2000	15
3000	13
4000	3
5000	15
7000	6
11000	2
14000	2

On the right side of the interface, a 'Tables(5)' panel is visible, listing the following tables: attrition_all, attrition_no, attrition_yes, employee (highlighted), and hivesampletable.

Salespeople without attrition:

The screenshot displays the Ambari Hive View 2.0 interface. On the left is a dark sidebar with navigation links for Dashboard, Services (HDFS, YARN, MapReduce2, Tez, Hive, Oozie, ZooKeeper, Ambari Metrics, WebHcat), Hosts, Alerts, and Cluster Admin. The main panel is titled 'HIVE' and includes tabs for QUERY, JOBS, TABLES, SAVED QUERIES, UDFs, and SETTINGS. A 'Worksheet1' tab is active. The 'DATABASE' dropdown is set to 'default'. The SQL query editor contains the following code:

```
1 SELECT monthlyincome AS monthlyincome, COUNT(monthlyincome) AS count
2 FROM attrition_no
3 GROUP BY MonthlyIncome;
4
```

Below the query editor are buttons for 'Execute', 'Save As', 'Insert UDF', and 'Visual Explain'. The 'RESULTS' tab is selected, showing a table with columns 'monthlyincome' and 'count'. The table contains 20 rows of data. To the right of the results is a 'Tables(5)' sidebar listing tables in the 'default' database: attrition_all, attrition_no, attrition_yes, employee (highlighted), and hivesampletable.

monthlyincome	count
2000	18
3000	24
4000	36
5000	77
7000	35
11000	9
14000	5
16000	5
18000	6
1000	2
6000	39
8000	25
9000	20
10000	24
12000	4
13000	6
15000	2
17000	9
19000	4
20000	4