

# Data Analysis on a Local Database(MySQL)

## Agenda:

- 1) MySQL workbench Installations
- 2) Analysis on local Database

## 1) Installation steps of MySQL

### Step 1:

Go to the MySQL Workbench Download Page

Visit the [official MySQL Workbench download page](#) to download the necessary software.

### Step 2:

Locate the Download Option for Windows

On the right side of the page, scroll down to find the Download option. Select the appropriate installer for Windows based on your system requirements.

The screenshot shows the MySQL Workbench download page for version 8.0.40. At the top, there are tabs for 'General Availability (GA) Releases' (highlighted in orange), 'Archives', and a help icon. Below the tabs, it says 'MySQL Workbench 8.0.40'. A dropdown menu labeled 'Select Operating System:' shows 'Microsoft Windows' selected. Under 'Recommended Download:', there is a section for 'MySQL Installer for Windows' with the subtext 'All MySQL Products. For All Windows Platforms. In One Package.' It includes a thumbnail of the MySQL logo and a 'Go to Download Page >' button. Below this, under 'Other Downloads:', there is a table for 'Windows (x86, 32 & 64-bit), MSI Installer'. The table columns are 'Name', 'Version', 'File Size', and 'Download'. The entry shows 'Windows (x86, 64-bit), MSI Installer' (mysql-workbench-community-8.0.40-winx64.msi), '8.0.40', '41.9M', and a 'Download' button. To the right of the download button, there is a link 'MD5: aa104de4958333d26844b239e9a279c0 | Signature'.

### Step 3:

Locate the Installer in Your Local Folder

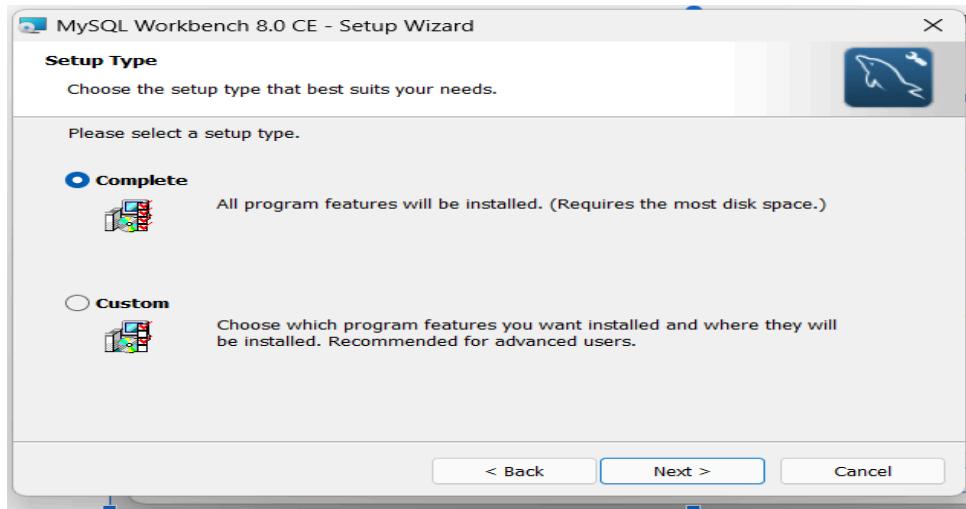
After the download is complete, navigate to your Downloads folder (or the location where the file was saved). Double-click the installer to begin the installation process on your local machine.



#### Step 4:

Download and Configure SQL Using the Installation Wizard

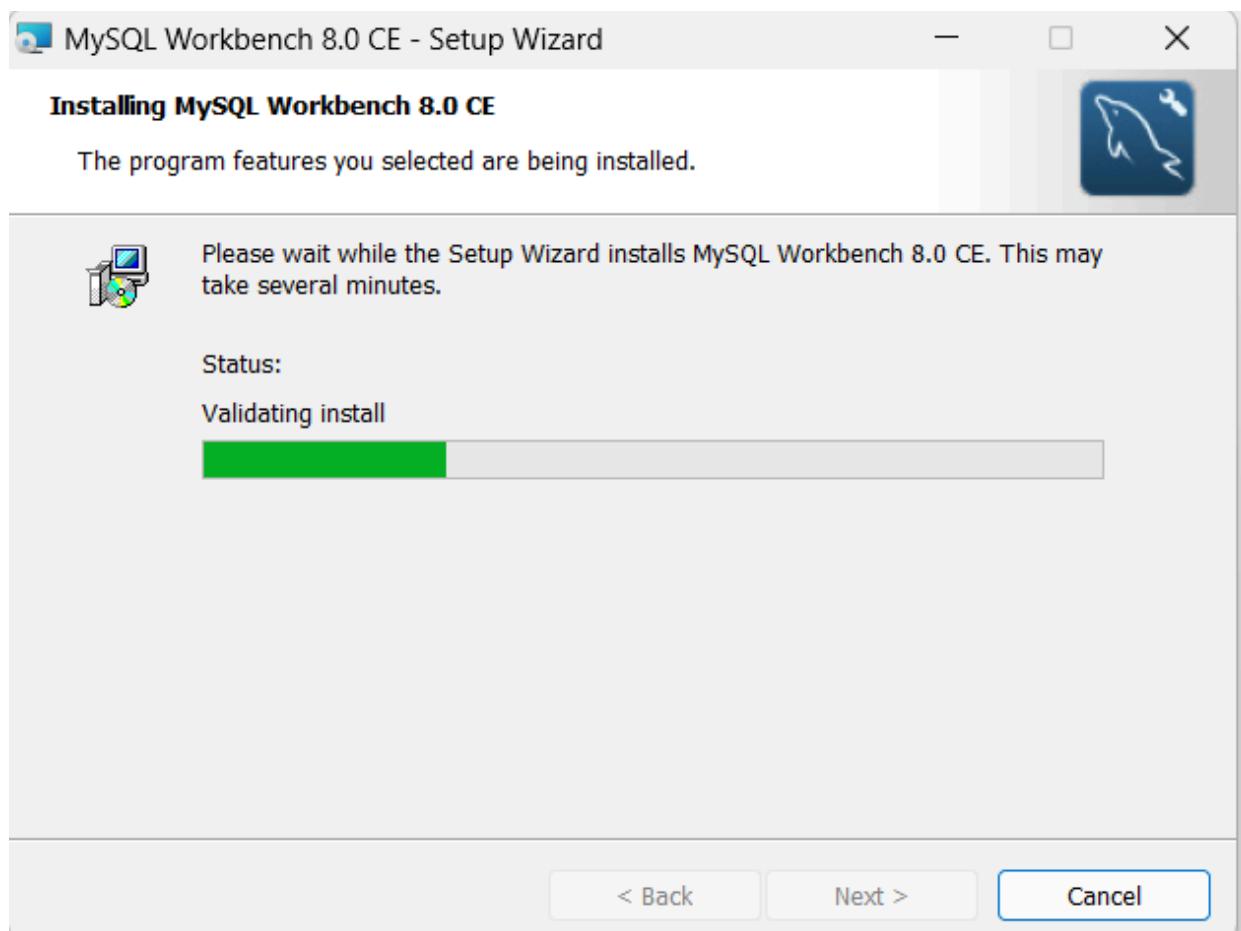
Launch the SQL installer and proceed with the installation wizard. When prompted, select the Custom installation option to tailor the installation according to your needs.



#### Step 5:

Ready to Install and Begin Installation

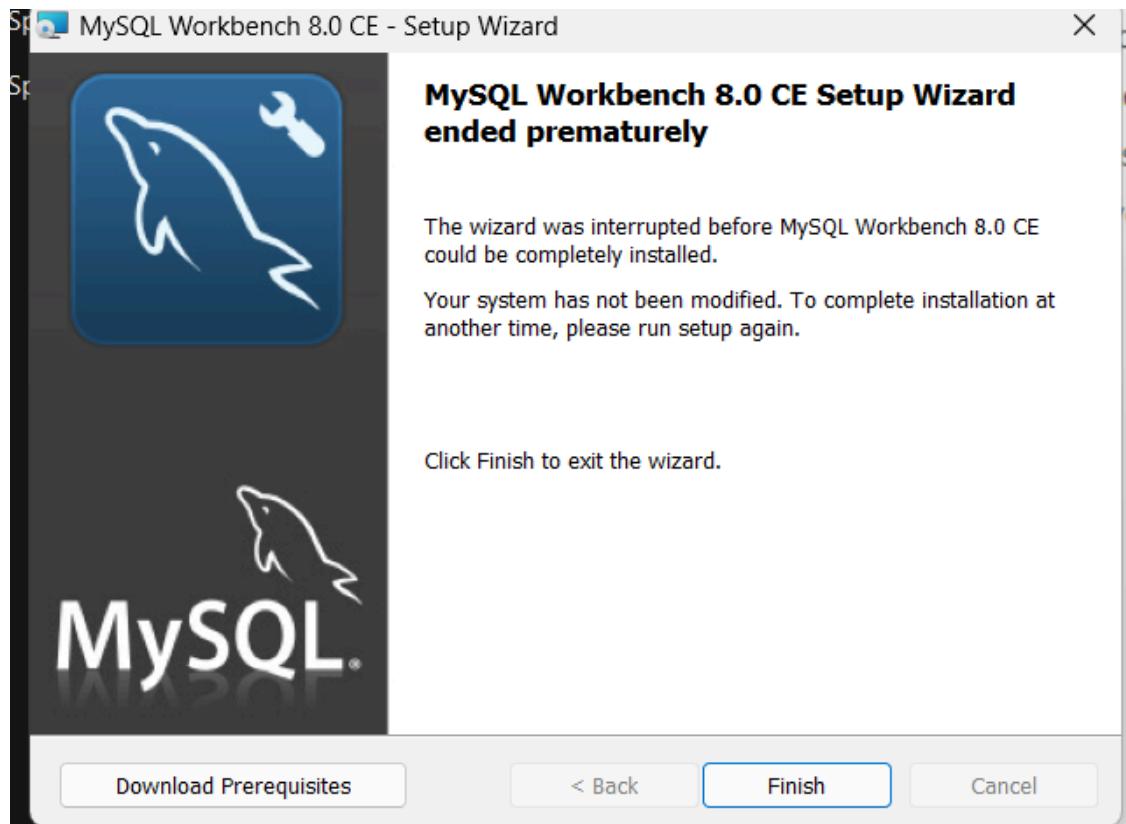
Once all configurations are set, the wizard will display a summary. Click Install to start the installation process. The program will now begin installing on your system.



## Step 6:

### Complete Installation and Address Server Issues

Once installation is complete, confirm that MySQL is installed successfully. If you encounter any server issues, use the MySQL Installer to troubleshoot and resolve them.



## 2) Basic Data Analytics Using Mysql on Indian Personal Finance and Spending Habits

### About dataset

This dataset is from Kaggle and provides financial information for 3,000 Indian individuals. It offers insights into their income, expenses, and potential savings. By analyzing this data, we can better understand Indian consumers' financial habits and preferences.

### Importing dataset into Database

#### -- Creating Database to import data in it

```
create database Finance;
```

#### -- create table before import dataset into database

```
CREATE TABLE FinancialData (
    Income FLOAT,
    Age INT,
    Dependents INT,
    Occupation VARCHAR(50),
    City_Tier VARCHAR(10),
    Rent FLOAT,
    Loan_Repayment FLOAT,
    Insurance FLOAT,
    Groceries FLOAT,
    Transport FLOAT,
    Eating_Out FLOAT,
    Entertainment FLOAT,
    Utilities FLOAT,
    Healthcare FLOAT,
    Education FLOAT,
    Miscellaneous FLOAT,
    Desired_Savings_Percentage FLOAT,
    Desired_Savings FLOAT,
    Disposable_Income FLOAT,
    Potential_Savings_Groceries FLOAT,
    Potential_Savings_Transport FLOAT,
    Potential_Savings_Eating_Out FLOAT,
    Potential_Savings_Entertainment FLOAT,
    Potential_Savings_Utilities FLOAT,
    Potential_Savings_Healthcare FLOAT,
    Potential_Savings_Education FLOAT,
    Potential_Savings_Miscellaneous FLOAT);
```

Select \* from finance.financialdata

The screenshot shows a database query results grid titled 'Result Grid'. The query executed was 'select \* from finance.financialdata;'. The grid displays 2999 rows of data across 15 columns: Income, Age, Dependents, Occupation, City\_Tier, Rent, Loan\_Repayment, Insurance, Groceries, Transport, Eating\_Out, Entertainment, Utilities, Healthcare, and Educa. The data includes various demographic and financial details for each individual, such as age ranges (e.g., 18-24, 25-34, etc.) and income levels (e.g., <10k, 10k-20k, etc.). The 'Result Grid' tab is selected in the top right corner, along with other tabs for 'Form Editor', 'Field Types', 'Query Stats', and 'Execution Plan'.

Income	Age	Dependents	Occupation	City_Tier	Rent	Loan_Repayment	Insurance	Groceries	Transport	Eating_Out	Entertainment	Utilities	Healthcare	Educa
44637.2	49	0	Self_Employed	Tier_1	13391.2	0	2206.49	6658.77	2636.97	1651.8	1536.18	2911.79	1546.91	0
26858.6	34	2	Retired	Tier_2	5371.72	0	869.523	2818.44	1543.02	649.378	1050.24	1626.14	1137.35	1551.7
50367.6	35	1	Student	Tier_3	7555.14	4612.1	2201.8	6313.22	3221.4	1513.81	1723.31	3368.46	2178.52	3160.0
101456	21	0	Self_Employed	Tier_3	15218.3	6809.44	4889.42	14690.1	7106.13	5040.25	2858.19	6128.55	4571.12	0
24875.3	52	4	Professional	Tier_2	4975.06	3112.61	635.907	3034.33	1276.16	692.827	660.187	1092.69	1169.1	1445.2
24875.6	57	4	Self_Employed	Tier_2	4975.12	0	1038.23	3250.07	1760.16	1049.07	751.017	1024.31	1022.3	2003.8
106120	53	0	Self_Employed	Tier_2	21223.9	0	4360.2	12790.4	6345.74	4390.91	2626.77	6202.55	3807.1	0
55431.3	40	1	Professional	Tier_2	11086.3	8307.7	1755.44	8194.48	3353	1741.91	1219.68	2361.98	2574.45	5520.4
20606.7	32	4	Retired	Tier_1	6182.02	3562.49	1018.59	3066.87	1170.79	688.06	418.234	921.233	830.237	1939.2
46304.8	30	1	Student	Tier_2	9260.96	0	2130.25	5065.63	2500.05	2040.31	1016.61	2945.43	1649.76	3085.9
20706.8	31	0	Student	Tier_3	3106.02	0	1031.78	3037.27	1459.28	451.411	785.29	1548.07	819.099	0
20668.6	31	4	Professional	Tier_2	4133.71	0	754.449	2597.4	1123.04	878.88	643.151	1408.9	1019.25	1506.2
36407.2	54	3	Retired	Tier_1	10922.2	4583.25	970.787	4685.67	2503.09	1328.96	822.883	2335.94	1153.75	1940.1
6492.02	22	3	Retired	Tier_3	973.802	0	184.243	955.499	494.824	309.24	229.826	471.87	222.648	590.0
7547.85	39	4	Self_Employed	Tier_2	1509.57	0	330.189	1032.73	568.236	266.438	277.466	510.595	235.712	521.9
19132.1	45	2	Student	Tier_2	3826.42	2525.41	590.193	2515.72	1164.07	661.751	660.749	1159.58	919.434	1442.1
13342.2	62	4	Retired	Tier_2	2668.44	2540.48	352.886	1920.1	961.38	622.272	331.705	985.635	531.557	1043.2
38574.6	21	4	Professional	Tier_2	7714.93	0	1838.34	5126.82	2614.48	1267.55	1802.74	2081.39	1196.04	3475.9
14509.1	55	4	Student	Tier_1	4352.74	829.441	438.597	1506.25	811.703	515.584	414.234	766.343	656.65	1449.9
6200.57	46	n	Professional	Tier_2	1022.5	0	717.505	1502.51	552.101	296.06	407.009	712.511	205.552	0

## Basic SQL Analysis

### Record Count:

-- total count for all records

```
select count(*) from financialdata ;
```

The screenshot shows a database query results grid titled 'Result Grid'. The query executed was 'select count(\*) from financialdata ;'. The grid displays 1 row with 1 column, showing the count as 2999. The 'Result Grid' tab is selected in the top right corner, along with other tabs for 'Form Editor', 'Field Types', 'Query Stats', and 'Execution Plan'.

count(*)
2999

-- Count the number of individuals aged 50 or above

```
SELECT COUNT(*) FROM financialdata WHERE Age >= 50;
```

```
2
3      -- Count the number of individuals aged 50 or above
4 •  SELECT COUNT(*) FROM financialdata WHERE Age >= 50;
5
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

COUNT(*)
940

## Aggregate Calculations

```
-- avg salary
```

```
select avg(income) from finance.financialdata;
```

```
6      -- avg salary
7 •  select avg(income) from finance.financialdata;
8
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

avg(income)
42253.11916993164

```
-- the total expenditure on healthcare for individuals aged 30 and above
```

```
select sum(Healthcare) as sum_healthcare from financialdata
```

```
where age >= '30';
```

```
10 •  select sum(Healthcare) as sum_healthcare from financialdata
11    where age >= '30';
12
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

sum_healthcare
3829397.9843444824

```
-- Average utility expenditure for individuals below 30
```

```
select AVG(Uutilities) as avg_utilities
```

from financialdata

where Age < 30;

```
13      -- Average utility expenditure for individuals below 30
14 •  select AVG(Uutilities) as avg_utilities
15    from financialdata
16   where Age < 30;
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

avg_utilities
2592.091034208025

-- average amount spent on groceries by individuals in Tier 1 cities

SELECT avg(Groceries) AS avg\_groceries FROM financialdata WHERE city\_tier = 'Tier\_1';

```
18      -- average amount spent on groceries by individuals in Tier 1 cities
19 •  SELECT avg(Groceries) AS avg_groceries FROM financialdata WHERE city_tier = 'Tier_1';
20
21      -- the number of individuals with insurance expenses greater than 2,000
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

avg_groceries
5354.276519915059

-- the number of individuals with insurance expenses greater than 2,000

select count(\*) from financialdata where insurance > 2000;

```
20
21      -- the number of individuals with insurance expenses greater than 2,000
22 •  select count(*) from financialdata where insurance > 2000;
23
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

avg_groceries
5354.276519915059

-- the average loan repayment amount for individuals with dependents

```
select avg(loan_repayment) as average_loan_repayment from financialdata where dependents > 0;
```

```
24      -- the average loan repayment amount for individuals with dependents
25 •  select avg(loan_repayment) as average_loan_repayment from financialdata where dependents > 0;
26
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
avg_groceries				

▶ 5354.276519915059

-- the number of individuals with transport expenses less than 1,000

```
select count(*) from financialdata where transport < 1000;
```

```
27      -- the number of individuals with transport expenses less than 1,000
28 •  select count(*) from financialdata where transport < 1000;
29
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
count(*)				

▶ 587

-- the number of individuals with desired savings greater than 10,000

```
select count(*) from financialdata where desired_savings > 10000;
```

```
31      -- the number of individuals with desired savings greater than 10,000
32 •  select count(*) from financialdata where desired_savings > 10000;
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
count(*)				

▶ 334

-- the total disposable income for individuals aged 45 and above

```
select sum(disposable_income) as total_disposable_income from financialdata where age >= 45;
```

```
34      -- the total disposable income for individuals aged 45 and above
35 •   select sum(disposable_income) as total_disposable_income from financialdata where age >= 45;
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
	total_disposable_income			

▶ 13729937.027837753

-- the number of individuals with groceries expenses greater than 5,000

```
select count(*) from financialdata where Groceries < 5000;
```

```
37      -- the number of individuals with groceries expenses greater than 5,000
38 •   select count(*) from financialdata where Groceries < 5000;
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
	count(*)			

▶ 1886

-- the total amount spent on eating out by individuals in Tier 2 cities

```
select sum(eating_out) as total_eating_out from financialdata where city_tier = 'Tier_2';
```

```
40      -- the total amount spent on eating out by individuals in Tier 2 cities
41 •   select sum(eating_out) as total_eating_out from financialdata where city_tier = 'Tier_2';
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
	total_eating_out			

▶ 2286722.4198532104

-- Count the number of individuals with disposable incomes less than 5000

```
select count(*) from financialdata where disposable_income < 5000;
```

```
-- Count the number of individuals with disposable incomes less than 5000
select count(*) from financialdata where disposable_income < 5000;
```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
count(*)			

1022

## Conditional Filtering:

– top 5 income in tier 2 cities

```
select income, occupation from finance.financialdata
```

```
where City_Tier = 'Tier_1'
```

```
order by income desc
```

```
limit 5;
```

```
6
7 • select income, occupation from finance.financialdata
8   where City_Tier = 'Tier_1'
9   order by income desc
10  limit 5;
11
```

income	occupation
369221	Retired
361092	Student
326818	Professional
272851	Retired
264366	Retired

-- top 5 income in tier 2 cities

```
select income, occupation from finance.financialdata
```

```
where City_Tier = 'Tier_2'
```

```
order by income desc
```

```
limit 5;
```

```
12      -- top 5 income in tier 2 cities
13 •  select income, occupation from finance.financialdata
14  where City_Tier = 'Tier_2'
15  order by income desc
16  limit 5;
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:	Fetch rows:
	income	occupation			
▶	248806	Self_Employed			
	235070	Self_Employed			
	220434	Self_Employed			
	213885	Retired			
	211241	Student			

-- tier 3

```
select income, occupation from finance.financialdata
```

```
where City_Tier = 'Tier_3'
```

```
order by income desc
```

```
limit 5;
```

```
18      -- tier 3
19 •  select income, occupation from finance.financialdata
20  where City_Tier = 'Tier_3'
21  order by income desc
22  limit 5;
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:	Fetch rows:
	income	occupation			
▶	248806	Self_Employed			
	235070	Self_Employed			
	220434	Self_Employed			
	213885	Retired			
	211241	Student			

-- individual with income and occupation in they younger age

```
select min(age) as min_age, income, Occupation from financialdata
```

```
group by income, Occupation
```

```
order by min_age
```

```
limit 10;  
24      -- individual with income and occupation in they younger age  
25 •  select min(age) as min_age, income, Occupation from financialdata  
26      group by income, Occupation  
27      order by min_age  
28      limit 10;
```

Result Grid			
	min_age	income	Occupation
▶	18	10202.7	Professional
	18	11641.7	Professional
	18	85362.9	Self_Employed
	18	9964.83	Student
	18	31159.4	Retired
	18	15431.9	Self_Employed
	18	72256.2	Self_Employed
	18	11357.1	Student
	18	11144.5	Retired
	18	24652.7	Self_Employed

-- Professionals with transport expenses

```
SELECT COUNT(*) as Prof_Transport  
FROM financialdata  
WHERE Occupation = 'Professional' AND Transport > 0;
```

```
30      -- Professionals with transport expenses  
31 •  SELECT Execute the statement under the keyboard cursor  
32      FROM financialdata  
33      WHERE Occupation = 'Professional' AND Transport > 0;  
34
```

Result Grid		
	Prof_Transport	
▶	730	

```
-- the total expenditure on healthcare across all city tiers
```

```
select sum(healthcare) as sum_healthcare, city_tier  
from finance.financialdata  
group by City_Tier;
```

```
36      -- the total expenditure on healthcare across all city tiers  
37 •  select sum(healthcare) as sum_healthcare, city_tier  
38    from finance.financialdata  
39    group by City_Tier;  
40
```

	sum_healthcare	city_tier
▶	1487644.9891662598	Tier_1
	2627089.3344573975	Tier_2
	947270.6214294434	Tier_3

```
-- Individuals in Tier 2 with desired savings percentage > 15%
```

```
select count(*) as count_Tier2  
FROM financialdata  
  
WHERE City_Tier = 'Tier_2' AND Desired_Savings_Percentage > 15;
```

```
41      -- Individuals in Tier 2 with desired savings percentage > 15%  
42 •  select count(*) as count_Tier2  
43    FROM financialdata  
44    WHERE City_Tier = 'Tier_2' AND Desired_Savings_Percentage > 15;  
45
```

	count_Tier2
▶	108

```
-- Individuals with grocery expenses greater than 5,000
```

```
select COUNT(*) as high_groceries  
from financialdata  
where Groceries > 5000;
```

```
46 -- Individuals with grocery expenses greater than 5,000  
47 • select COUNT(*) as high_groceries  
48 from financialdata  
49 where Groceries > 5000;  
50
```

high_groceries
1113

```
-- average income of individuals in Tier 1 cities  
select avg(income) as avg_income  
from financialdata  
where city_tier = 'tier_1';
```

```
51 -- average income of individuals in Tier 1 cities  
52 • select avg(income) as avg_income  
53 from financialdata  
54 where city_tier = 'tier_1';  
55
```

avg_income
42744.17380214665

```
-- the average income of individuals who spend more on entertainment than on education  
select avg(income), age as avg_income from financialdata  
where Entertainment > Education  
group by age  
order by age  
limit 10;
```

```

56    -- the average income of individuals who spend more on entertainment than on education
57 •  select avg(income), age as avg_income from financialdata
58  where Entertainment > Education
59  group by age
60  order by age
61  limit 10;

```

Result Grid | Filter Rows: \_\_\_\_\_ | Export: \_\_\_\_\_ | Wrap Cell Content: \_\_\_\_\_ | Fetch rows: \_\_\_\_\_

	avg(income)	avg_income
▶	86246.10286458333	18
	28737.164713541668	19
	59973.134765625	20
	38807.281700721156	21
	47212.604771205355	22
	58719.99253627232	23
	41690.684326171875	24
	34834.277506510414	25
	39022.700032552086	26
	53704.794830322266	27

-- the average rent for individuals living in Tier 1 cities and having 2 or more dependents

SELECT avg(Rent) as Avg\_rent

FROM financialdata

WHERE City\_Tier = 'Tier\_1' AND dependents >= 2;

```

64    -- the average rent for individuals living in Tier 1 cities and having 2 or more dependents
65 •  SELECT avg(Rent) as Avg_rent
66  FROM financialdata
67  WHERE City_Tier = 'Tier_1' AND dependents >= 2;
68

```

Result Grid | Filter Rows: \_\_\_\_\_ | Export: \_\_\_\_\_ | Wrap Cell Content: \_\_\_\_\_ | Fetch rows: \_\_\_\_\_

	Avg_rent
▶	13316.886426538453

-- the individuals who spend more on entertainment than on healthcare

SELECT \* FROM financial\_data WHERE Entertainment > Healthcare;

```

70    -- the individuals who spend more on entertainment than on healthcare
71 •  SELECT * FROM financialdata WHERE Entertainment > Healthcare;
72
73    -- How many individuals have a potential savings on groceries greater than 1000 and a potential savings on entertainment less than 500

```

Result Grid | Filter Rows: \_\_\_\_\_ | Export: \_\_\_\_\_ | Wrap Cell Content:  | Fetch rows:

	Income	Age	Dependents	Occupation	City_Tier	Rent	Loan_Repayment	Insurance	Groceries	Transport	Eating_Out	Entertainment	Utilities	Healthcare	Educa
▶	6492.02	22	3	Retired	Tier_3	973.802	0	184.243	955.499	494.824	309.24	229.826	471.87	222.648	590.04
	7547.85	39	4	Self_Employed	Tier_2	1509.57	0	330.189	1032.73	568.236	266.438	277.466	510.595	235.712	521.97
	38574.6	21	4	Professional	Tier_2	7714.93	0	1838.34	5126.82	2614.48	1267.55	1802.74	2081.39	1196.04	3475.9
	9692.52	46	0	Professional	Tier_2	1938.5	0	217.585	1306.54	562.101	396.06	427.908	723.641	305.653	0
	19408.1	52	3	Retired	Tier_1	5822.43	2832.16	516.117	2486.69	1346.56	956.591	930.294	1041.33	592.632	1604.7
	32783.8	37	3	Professional	Tier_2	6556.77	0	946.625	4815.61	1788.11	1519.47	1452.26	1557.94	1175.05	2225.1

-- How many individuals have a potential savings on groceries greater than 1000 and a potential savings on entertainment less than 500

```
SELECT COUNT(*) AS Count_High_Groceries_Low_Entertainment
```

```
FROM financialdata
```

```
WHERE Potential_Savings_Groceries > 1000 AND Potential_Savings_Entertainment < 500;
```

```

73    -- How many individuals have a potential savings on groceries greater than 1000 and a potential savings on entertainment less than 500
74 •  SELECT COUNT(*) AS Count_High_Groceries_Low_Entertainment
75    FROM financialdata
76    WHERE Potential_Savings_Groceries > 1000 AND Potential_Savings_Entertainment < 500;
77

```

Result Grid | Filter Rows: \_\_\_\_\_ | Export: \_\_\_\_\_ | Wrap Cell Content:

Count_High_Groceries_Low_Entertainment
601

-- Top 3 salary that earned at the age of 18

```
select max(age) as young, income from financialdata
```

```
group by income
```

```
order by young desc
```

```
limit 3;
```

```

78      -- Top 3 salary that earned at the age of 18
79 •   select young, income from financialdata
80     group by income
81     order by young desc
82     limit 3;
83

```

Result Grid | Filter Rows:  Export: Wrap Cell Content: Fetch

	young	income
▶	64	47975.1
	64	163845
	64	24275.6

-- Average income between the age of 30 and 40

Select avg(Income) as avg\_income

from financialdata

where Age between 30 and 40;

```

84      -- Average income between the age of 30 and 40
85 •   Select avg(Income) as avg_income
86     from financialdata
87     where Age between 30 and 40;
88

```

Result Grid | Filter Rows:  Export: Wrap Cell Content:

	avg_income
▶	43713.784930245536

-- Rental rate greater than 10000 in tier 3 cities

SELECT \* FROM financialdata

WHERE Rent > 10000 AND City\_Tier = 'Tier\_3';

```

89 -- Rental rate greater than 10000 in tier 3 cities
90 • SELECT * FROM financialdata
91 WHERE Rent > 10000 AND City_Tier = 'Tier_3';

```

Result Grid | Filter Rows: Export: Wrap Cell Content: [X]

	Income	Age	Dependents	Occupation	City_Tier	Rent	Loan_Repayment	Insurance	Groceries	Transport	Eating_Out	Entertainment	Utilities	Healthcare	Education
▶	101456	21	0	Self_Employed	Tier_3	15218.3	6809.44	4889.42	14690.1	7106.13	5040.25	2858.19	6128.55	4571.12	0
	132029	56	2	Self_Employed	Tier_3	19804.3	0	4383.84	15998.7	7954.56	5733.45	4929.44	6807.52	5860.84	8512.7
	92151.4	60	3	Retired	Tier_3	13822.7	0	3927.21	13165.2	7110.06	3629.76	2986.46	7156.3	3907.24	8200.1
	85362.9	18	3	Self_Employed	Tier_3	12804.4	12629	2580.53	10867.2	4970.89	2204.79	3397.41	4128.24	3926.66	4693.0
	136815	37	1	Retired	Tier_3	20522.3	0	6277.03	15328.2	9653.34	6821.5	4110.41	6937.69	4469.73	10396
	74414.4	45	1	Self_Employed	Tier_3	11162.2	0	2182.59	10345.1	5082.47	3668.14	1575.2	4412.88	3196.78	6131.1

## Reference:

### Dataset link :

<https://www.kaggle.com/datasets/shriyashjagtap/indian-personal-finance-and-spending-habits>









