

1. Total & Average Daily Energy Consumption by Zone:

The screenshot shows the MySQL Workbench interface. In the top navigation bar, the database is set to 'Local instance MySQL80' and the schema is 'smartcity_db'. The main area displays a query in 'Query 1' tab:

```
3
4 • select * from smartcityenergy;
5
6 • SELECT
7     Zone,
8     SUM(EnergyConsumed_kWh) AS Total_Consumption,
9     AVG(EnergyConsumed_kWh) AS Avg_Daily_Consumption
10    FROM SmartCityEnergy
11   GROUP BY Zone;
```

The results grid shows the following data:

Zone	Total_Consumption	Avg_Daily_Consumption
North	40617.240000000005	534.4373684210527
Central	43640.56000000001	589.7372972972975
South	41683.210000000014	578.9334722222225
West	56704.00000000001	550.5242718446602
East	46227.51	616.3668

The 'Output' pane at the bottom shows the execution log:

#	Time	Action	Message	Duration / Fetch
10	16:53:41	select * from smartcity_db LIMIT 0, 1000	Error Code: 1146. Table 'smartcity_db.smartcity_db' doesn't exist	0.031 sec
11	16:54:50	select * from smartcityenergy LIMIT 0, 1000	400 row(s) returned	0.016 sec / 0.000 sec
12	16:57:17	SELECT Zone, SUM(EnergyConsumed_kWh) AS Total_Consumption, AV...	5 row(s) returned	0.031 sec / 0.000 sec

2. Top 5 Highest Energy Consuming Consumers:-

The screenshot shows the MySQL Workbench interface. In the top navigation bar, the database is set to 'Local instance MySQL80' and the schema is 'smartcity_db'. The main area displays a query in 'Query 1' tab:

```
15 • SELECT
16     MeterID,
17     ConsumerType,
18     SUM(EnergyConsumed_kWh) AS Total_Usage
19    FROM SmartCityEnergy
20   GROUP BY MeterID, ConsumerType
21   ORDER BY Total_Usage DESC
22   LIMIT 5;
```

The results grid shows the following data:

MeterID	ConsumerType	Total_Usage
MTR00399	Residential	999.22
MTR00139	Industrial	998.56
MTR00366	Residential	995.35
MTR00268	Residential	995.16
MTR00299	Industrial	993.18

The 'Output' pane at the bottom shows the execution log:

#	Time	Action	Message	Duration / Fetch
11	16:54:50	select * from smartcityenergy LIMIT 0, 1000	400 row(s) returned	0.016 sec / 0.000 sec
12	16:57:17	SELECT Zone, SUM(EnergyConsumed_kWh) AS Total_Consumption, AV...	5 row(s) returned	0.031 sec / 0.000 sec
13	17:02:41	SELECT MeterID, ConsumerType, SUM(EnergyConsumed_kWh) AS Total...	5 row(s) returned	0.016 sec / 0.000 sec

3. Monthly Trend of Consumption :

The screenshot shows the MySQL Workbench interface with a query editor window titled "Query 1". The query retrieves monthly consumption data from the "SmartCityEnergy" table:

```
20 GROUP BY MeterID, ConsumerType
21 ORDER Execute the selected portion of the script or everything, if there is no selection
22 LIMIT 5;
23
24 -- Monthly Trend of Consumption
25 SELECT
26     YEAR(Date) AS Year,
27     MONTH(Date) AS Month,
28     SUM(EnergyConsumed_kWh) AS Monthly_Consumption
29 FROM SmartCityEnergy
```

The results are displayed in a grid:

Year	Month	Monthly_Consumption
2025	1	19800.289999999997
2025	2	14675.250000000004
2025	3	17865.65
2025	4	17390.540000000005
2025	5	19781.09

Below the results, the "Output" pane shows the execution log:

#	Time	Action	Message	Duration / Fetch
12	16:57:17	SELECT	Zone, SUM(EnergyConsumed_kWh) AS Total_Consumption, AVG(EnergyConsumed_kWh * TariffRate) AS Avg_Cost	0.031 sec / 0.000 sec
13	17:02:41	SELECT	MeterID, ConsumerType, SUM(EnergyConsumed_kWh) AS Total_Consumption, AVG(EnergyConsumed_kWh * TariffRate) AS Avg_Cost	0.016 sec / 0.000 sec
14	17:04:17	SELECT	YEAR(Date) AS Year, MONTH(Date) AS Month, SUM(EnergyConsumed_kWh) AS Monthly_Consumption	0.000 sec / 0.000 sec

4. Average Cost per Zone :-

The screenshot shows the MySQL Workbench interface with a query editor window titled "Query 1". The query retrieves average consumption costs by zone from the "SmartCityEnergy" table:

```
28 SUM(EnergyConsumed_kWh) AS Monthly_Consumption
29 Execute the selected portion of the script or everything, if there is no selection
30 GROUP BY YEAR(Date), MONTH(Date)
31 ORDER BY Year, Month;
32
33
34 -- Average Cost per Zone
35 SELECT
36     Zone,
37     AVG(EnergyConsumed_kWh * TariffRate) AS Avg_Cost
```

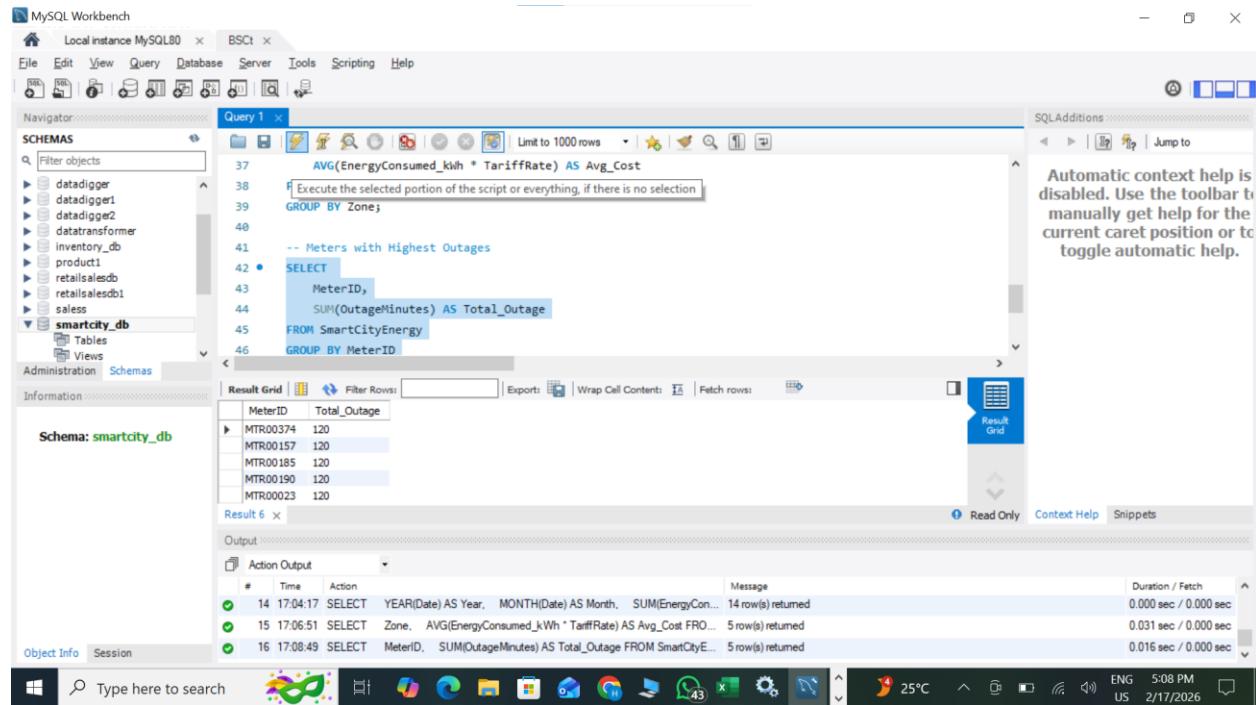
The results are displayed in a grid:

Zone	Avg_Cost
North	3253.6178921052624
Central	3481.614428378378
South	3359.6199347222228
West	3409.388183495144
East	3642.8594706666663

Below the results, the "Output" pane shows the execution log:

#	Time	Action	Message	Duration / Fetch
13	17:02:41	SELECT	MeterID, ConsumerType, SUM(EnergyConsumed_kWh) AS Total_Consumption, AVG(EnergyConsumed_kWh * TariffRate) AS Avg_Cost	0.016 sec / 0.000 sec
14	17:04:17	SELECT	YEAR(Date) AS Year, MONTH(Date) AS Month, SUM(EnergyConsumed_kWh) AS Monthly_Consumption	0.000 sec / 0.000 sec
15	17:06:51	SELECT	Zone, AVG(EnergyConsumed_kWh * TariffRate) AS Avg_Cost FROM SmartCityEnergy	0.031 sec / 0.000 sec

5. Meters with Highest Outages :



The screenshot shows the MySQL Workbench interface with a query editor window titled "Query 1". The code in the editor is:

```

37     AVG(EnergyConsumed_kWh * TariffRate) AS Avg_Cost
38
39     GROUP BY Zone;
40
41 -- Meters with Highest Outages
42
43     SELECT
44         MeterID,
45         SUM(OutageMinutes) AS Total_Outage
46
47     FROM SmartCityEnergy
48
49     GROUP BY MeterID

```

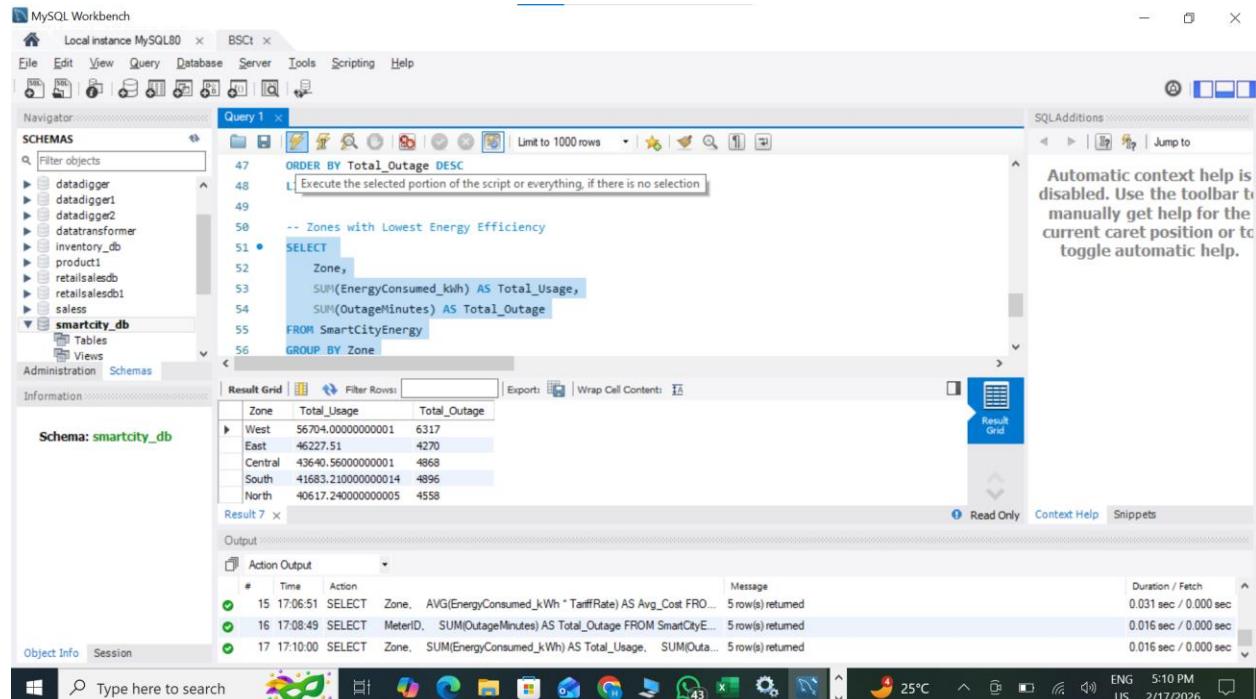
The "Result Grid" pane shows the following data:

MeterID	Total_Outage
MTR00374	120
MTR00157	120
MTR00185	120
MTR00190	120
MTR00023	120

The "Output" pane shows the execution log:

#	Time	Action	Message	Duration / Fetch
14	17:04:17	SELECT	YEAR(Date) AS Year, MONTH(Date) AS Month, SUM(EnergyCon... 14 row(s) returned	0.000 sec / 0.000 sec
15	17:06:51	SELECT	Zone, AVG(EnergyConsumed_kWh * TariffRate) AS Avg_Cost FRO... 5 row(s) returned	0.031 sec / 0.000 sec
16	17:08:49	SELECT	MeterID, SUM(OutageMinutes) AS Total_Outage FROM SmartCityE... 5 row(s) returned	0.016 sec / 0.000 sec

6. Zones with Lowest Energy Efficiency :-



The screenshot shows the MySQL Workbench interface with a query editor window titled "Query 1". The code in the editor is:

```

47     ORDER BY Total_Outage DESC
48
49
50 -- Zones with Lowest Energy Efficiency
51
52     SELECT
53         Zone,
54         SUM(EnergyConsumed_kWh) AS Total_Usage,
55         SUM(OutageMinutes) AS Total_Outage
56
57     FROM SmartCityEnergy
58
59     GROUP BY Zone

```

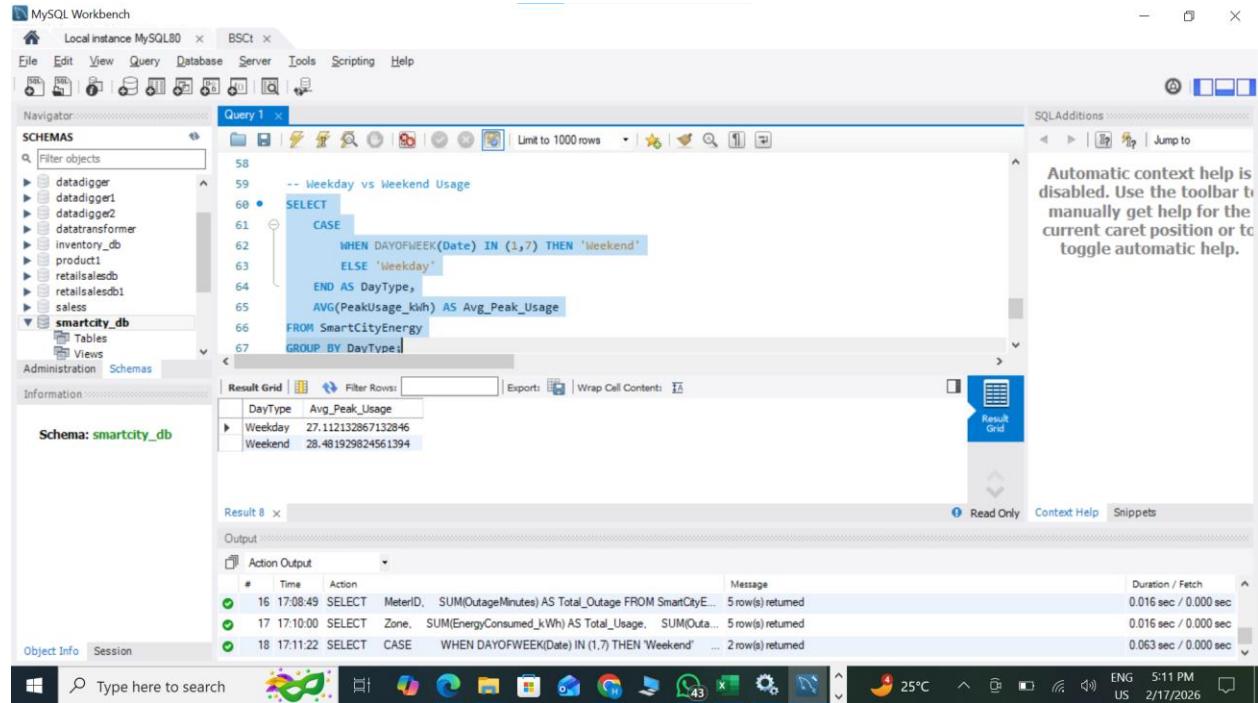
The "Result Grid" pane shows the following data:

Zone	Total_Usage	Total_Outage
West	56704.0000000001	6317
East	46227.51	4270
Central	43640.5600000001	4868
South	41683.21000000014	4896
North	40617.24000000005	4558

The "Output" pane shows the execution log:

#	Time	Action	Message	Duration / Fetch
15	17:06:51	SELECT	Zone, AVG(EnergyConsumed_kWh * TariffRate) AS Avg_Cost FRO... 5 row(s) returned	0.031 sec / 0.000 sec
16	17:08:49	SELECT	MeterID, SUM(OutageMinutes) AS Total_Outage FROM SmartCityE... 5 row(s) returned	0.016 sec / 0.000 sec
17	17:10:00	SELECT	Zone, SUM(EnergyConsumed_kWh) AS Total_Usage, SUM(Outa... 5 row(s) returned	0.016 sec / 0.000 sec

7. Weekday vs Weekend Usage :-



The screenshot shows the MySQL Workbench interface with a query editor window titled "Query 1". The SQL code is as follows:

```
58 -- weekday vs weekend Usage
59
60 • SELECT
61     CASE
62         WHEN DAYOFWEEK(Date) IN (1,7) THEN 'Weekend'
63         ELSE 'Weekday'
64     END AS DayType,
65     AVG(PeakUsage_kWh) AS Avg_Peak_Usage
66 FROM SmartCityEnergy
67 GROUP BY DayType;
```

The results grid shows two rows:

DayType	Avg_Peak_Usage
Weekday	27.112132867132846
Weekend	28.481929824561394

The "Output" tab displays the execution log:

#	Time	Action	Message	Duration / Fetch
16	17:08:49	SELECT MeterID, SUM(OutageMinutes) AS Total_Outage FROM SmartCityE...	5 row(s) returned	0.016 sec / 0.000 sec
17	17:10:00	SELECT Zone, SUM(EnergyConsumed_kWh) AS Total_Usage, SUM(Outa...	5 row(s) returned	0.016 sec / 0.000 sec
18	17:11:22	SELECT CASE WHEN DAYOFWEEK(Date) IN (1,7) THEN 'Weekend' ...	2 row(s) returned	0.063 sec / 0.000 sec