

Chandigarh College of Engineering & Technology (Degree Wing)



Department of Computer Science and Engineering

Database Systems (Practical)

CS 352

Practical -3

DOP: - 19/07/2024

DOS:-26/07/2024

Submitted By:

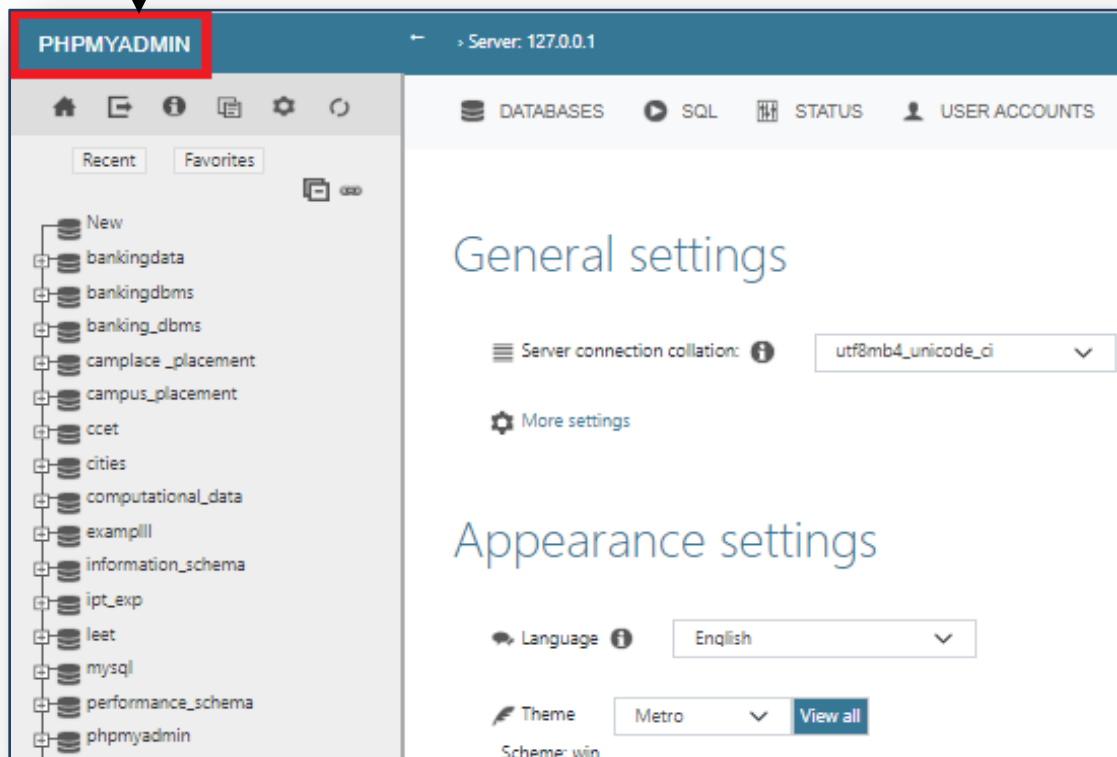
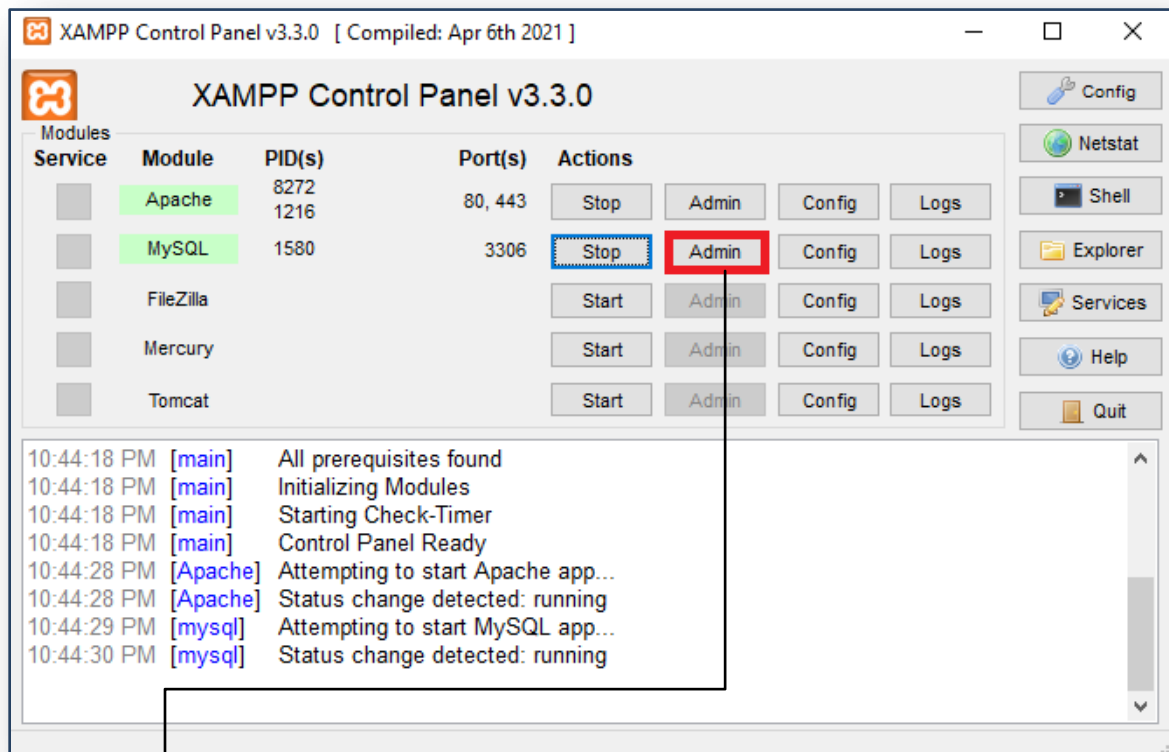
Abhay Pratap Singh
(Roll No: - CO23306)

Submitted To:

Dr. Dheerendra Singh
CSE Department

STEP 1:- Starting XAMPP

- Open the “XAMPP” Control Panel and start the Apache and MySQL services, by clicking on start and then admin.
- The “phpMyAdmin” will open in your default web browser.



STEP 2:- Creating new database

- In phpMyAdmin, click on the "**Databases**" tab.
- Enter the name for your new database in the "**Database name**" field.
- Choose the desired collation (e.g., **utf8_general_ci**) from the drop-down menu if necessary.
- Click on "**Create**", to create the database.

The first screenshot shows the 'Create database' form in phpMyAdmin. The 'Database name' field contains 'Customer_Data' and the 'Collation' dropdown is set to 'utf8mb4_general_ci'. The 'Create' button is highlighted. Annotations include an arrow pointing to the 'Database name' field labeled 'Database name' and an arrow pointing to the 'Collation' dropdown labeled 'Choose desired collation'.

The second screenshot shows the 'Databases' tab after the database has been created. The 'customer_data' database is listed at the bottom of the table, with its collation set to 'utf8mb4_general_ci'. An annotation points to this row with the label 'Database created'.

Database	Collation	Action
<input type="checkbox"/> bankingdata	Latin1_swedish_ci	<input type="checkbox"/> Check privileges
<input type="checkbox"/> bankingdbms	Latin1_swedish_ci	<input type="checkbox"/> Check privileges
<input type="checkbox"/> banking_dbms	Latin1_swedish_ci	<input type="checkbox"/> Check privileges
<input type="checkbox"/> camplace_placement	Latin1_swedish_ci	<input type="checkbox"/> Check privileges
<input type="checkbox"/> campus_placement	Latin1_swedish_ci	<input type="checkbox"/> Check privileges
<input type="checkbox"/> ccet	Latin1_swedish_ci	<input type="checkbox"/> Check privileges
<input type="checkbox"/> cities	Latin1_swedish_ci	<input type="checkbox"/> Check privileges
<input type="checkbox"/> computational_data	Latin1_swedish_ci	<input type="checkbox"/> Check privileges
<input type="checkbox"/> customer_data	utf8mb4_general_ci	<input type="checkbox"/> Check privilege

STEP 3:- Creating Tables

- For creating a table, enter its name in “**Table name**” section and define the “**Number of columns**”.
- You will be redirected to the window shown, now fill out “**Name**”, “**Type**” and “**Length/Value**”.
- Now click on “**Save**” to create the table.

STRUCTURE SQL SEARCH QUERY EXPORT

No tables found in database.

Create new table

Table name: Customer_transactions

Number of columns: 3

Create

Number of Columns

Table Name

STRUCTURE SQL SEARCH QUERY EXPORT IMPORT

Name	Type	Length/Values	Default
ID	INT	50	None
NAME	VARCHAR	50	None
TOTAL_SPENT	DECIMAL	50	None

Attributes Name

Attributes Type

Length of each Attribute

PARTITION definition:

Partition by: (Expression or column list)

Partitions:

Preview SQL Save

Click on save to create table

STEP 4:- Adding Data

- Now a table will be displayed. Then again click on “Go”.
- The second image will be shown. Now add the information, and click on “Go”.

Navigation: BROWSE | **STRUCTURE** | SQL | SEARCH | INSERT | EXPORT | IMPORT | PRIVILEGES

Sub-tabs: **TABLE STRUCTURE** | RELATION VIEW

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/> 1	ID	int(50)			No	None			Change Drop More
<input type="checkbox"/> 2	NAME	varchar(50)	utf8mb4_general_ci		No	None			Change Drop More
<input type="checkbox"/> 3	TOTAL_SPENT	decimal(50,0)			No	None			Change Drop More

With selected: Browse Change Drop Primary Unique Index Spatial Fulltext

Print Propose table structure Track table Move columns Normalize

Add 1 column(s) after TOTAL_SPENT **Go**

Navigation: BROWSE | STRUCTURE | **INSERT** | EXPORT | IMPORT | PRIVILEGES | OPERATIONS

Column	Type	Function	Null	Value
ID	int(50)			1
NAME	varchar(50)			John Smith
TOTAL_SPENT	decimal(50,0)			150.00

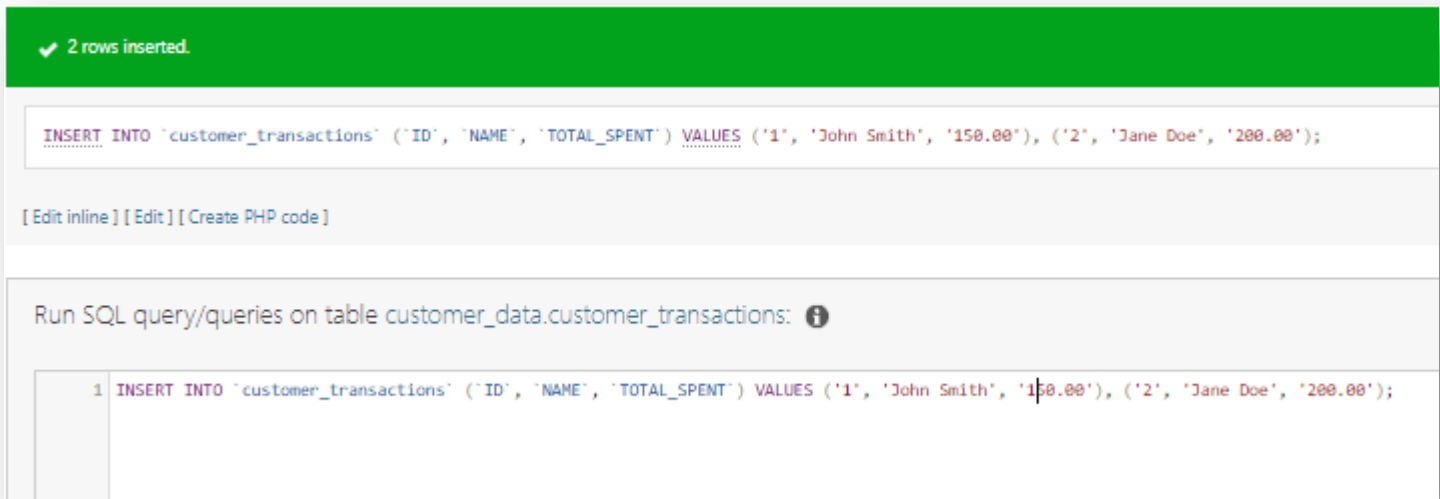
☐ Ignore

Column	Type	Function	Null	Value
ID	int(50)			2
NAME	varchar(50)			Jane Doe
TOTAL_SPENT	decimal(50,0)			200.00

Insert as new row and then Go back to previous page

Console Preview SQL Reset **Go**

- Once you click on “Go” you will get the following message.



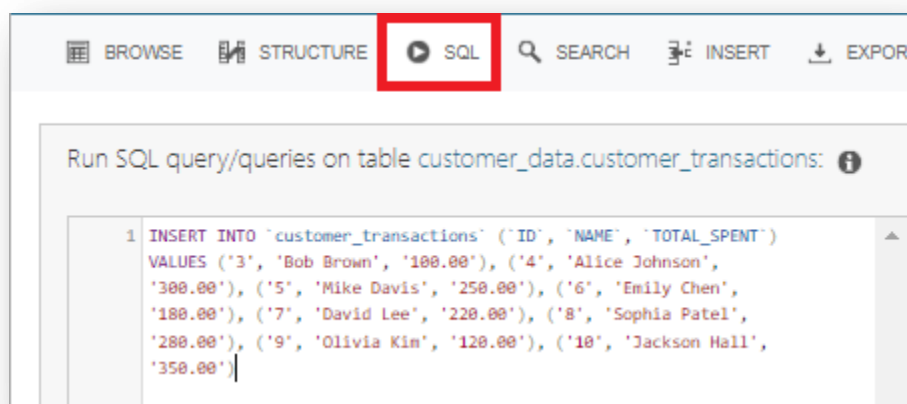
- Now click on “Browse” to see your table, you will see the two rows inserted by you.

Show all | Number of rows: 25 ▾

Extra options

ID	NAME	TOTAL_SPENT
1	John Smith	150
2	Jane Doe	200

- Now add more rows by going to “SQL” and run this query to insert more rows to your table.



- Click on “Go” and the following message will be displayed.

Run SQL query/queries on table `customer_data.customer_transactions`:

```
1 INSERT INTO `customer_transactions` (`ID`, `NAME`, `TOTAL_SPENT`)
VALUES ('3', 'Bob Brown', '100.00'), ('4', 'Alice Johnson',
'300.00'), ('5', 'Mike Davis', '250.00'), ('6', 'Emily Chen',
'180.00'), ('7', 'David Lee', '220.00'), ('8', 'Sophia Patel',
'280.00'), ('9', 'Olivia Kim', '120.00'), ('10', 'Jackson Hall',
'350.00');
```

SELECT * SELECT INSERT UPDATE DELETE Clear Format

Get auto-saved query

☐ Bind parameters

Bookmark this SQL query:

Delimiter: ☐ Show this query here again ☐ Retain query box ☐ Rollback when finished

☒ Enable foreign key checks **Go**

Show query box

✓ 8 rows inserted. (Query took 0.0894 seconds.)

```
INSERT INTO `customer_transactions` (`ID`, `NAME`, `TOTAL_SPENT`) VALUES ('
('8', 'Sophia Patel', '280.00'), ('9', 'Olivia Kim', '120.00'), ('10', 'Jac
```

[Edit inline] [Edit] [Create PHP code]

- Now click on “**Browse**” to see your full table, as shown below.

✓ Showing rows 0 - 9 (10 total, Query took 0.0003 seconds.)

```
SELECT * FROM `customer_transactions`
```

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

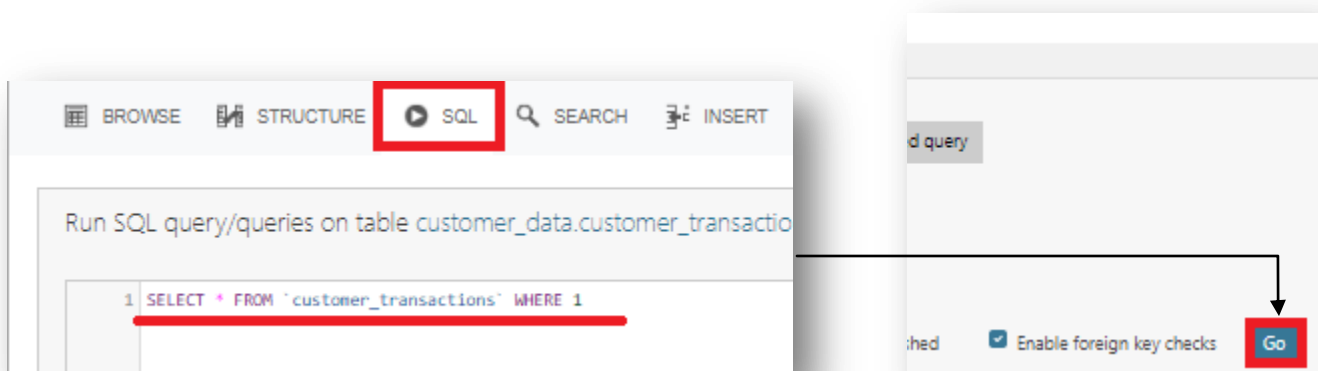
☐ Show all | Number of rows: 25 ▾

Extra options

ID	NAME	TOTAL_SPENT
1	John Smith	150
2	Jane Doe	200
3	Bob Brown	100
4	Alice Johnson	300
5	Mike Davis	250
6	Emily Chen	180
7	David Lee	220
8	Sophia Patel	280
9	Olivia Kim	120
10	Jackson Hall	350

STEP 4:- Retrieval of Rows using “SELECT” statement

- Now go to “SQL” and run the following query – “**SELECT * FROM `customer_transactions` WHERE 1**”, by clicking on “Go”.



- Once you run the query the following table will be displayed.

✓ Showing rows 0 - 9 (10 total, Query took 0.0003 seconds.)

```
SELECT * FROM `customer_transactions` WHERE 1;
```

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

☐ Show all | Number of rows: 25 ▼

Extra options

ID	NAME	TOTAL_SPENT
1	John Smith	150
2	Jane Doe	200
3	Bob Brown	100
4	Alice Johnson	300
5	Mike Davis	250
6	Emily Chen	180
7	David Lee	220
8	Sophia Patel	280
9	Olivia Kim	120
10	Jackson Hall	350

Step 5:- Conditional Retrieval of Rows

- Now again go to “SQL” and run the following query – “**SELECT * FROM customer_transactions WHERE TOTAL_SPENT BETWEEN 150 AND 250;**” and click on “Go”.

The screenshot shows the SQL query editor interface. The 'SQL' tab is selected and highlighted with a red box. The query entered is: `1 SELECT * FROM customer_transactions`
`2 WHERE TOTAL_SPENT BETWEEN 150 AND 250;` The query is underlined in red. To the right, the 'Go' button is also highlighted with a red box. An arrow points from the 'Go' button in the right panel to the 'Go' button in the left panel.

- As you click on “Go”, then all the rows with value of total spent between 150 to 250 will be retrieved, as shown in image below.

The screenshot shows a database query interface with a green status bar indicating 'Showing rows 0 - 4 (5 total, Query took 0.0003 seconds.)'. The SQL query entered is: `SELECT * FROM customer_transactions WHERE TOTAL_SPENT BETWEEN 150 AND 250;`. Below the query, there are options for 'Show all' and 'Number of rows: 25'. The results are displayed in a table with columns ID, NAME, and TOTAL_SPENT.

ID	NAME	TOTAL_SPENT
1	John Smith	150
2	Jane Doe	200
5	Mike Davis	250
6	Emily Chen	180
7	David Lee	220

At the bottom, there are options for 'Query results operations' including Print, Copy to clipboard, Export, Display chart, and Create view.

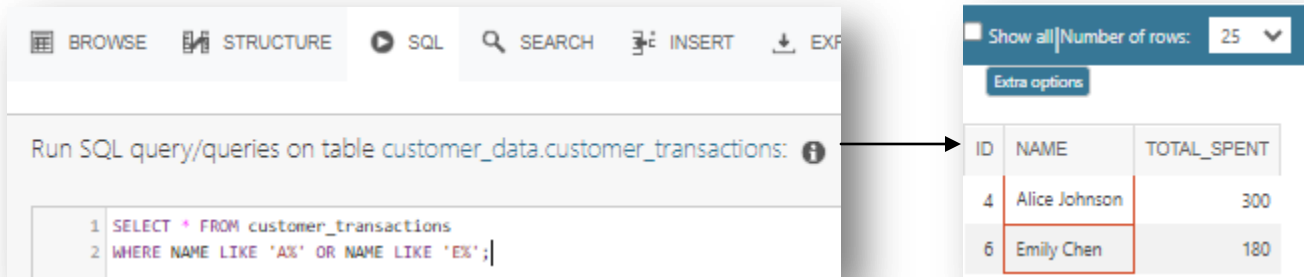
- Similarly some conditional operations that are done on database are shown below.

- Where **TOTAL_SPENT** is less than 150 and greater than 300.

The screenshot shows a database query interface with a status bar indicating 'Showing rows 0 - 2 (3 total, Query took 0.0003 seconds.)'. The SQL query entered is: `1 SELECT * FROM customer_transactions`
`2 WHERE TOTAL_SPENT < 150 OR TOTAL_SPENT > 300;`. Below the query, there are options for 'Show all' and 'Number of rows: 25'. The results are displayed in a table with columns ID, NAME, and TOTAL_SPENT.

ID	NAME	TOTAL_SPENT
3	Bob Brown	100
9	Olivia Kim	120
10	Jackson Hall	350

- Getting customer names starting from “A” or “E”.



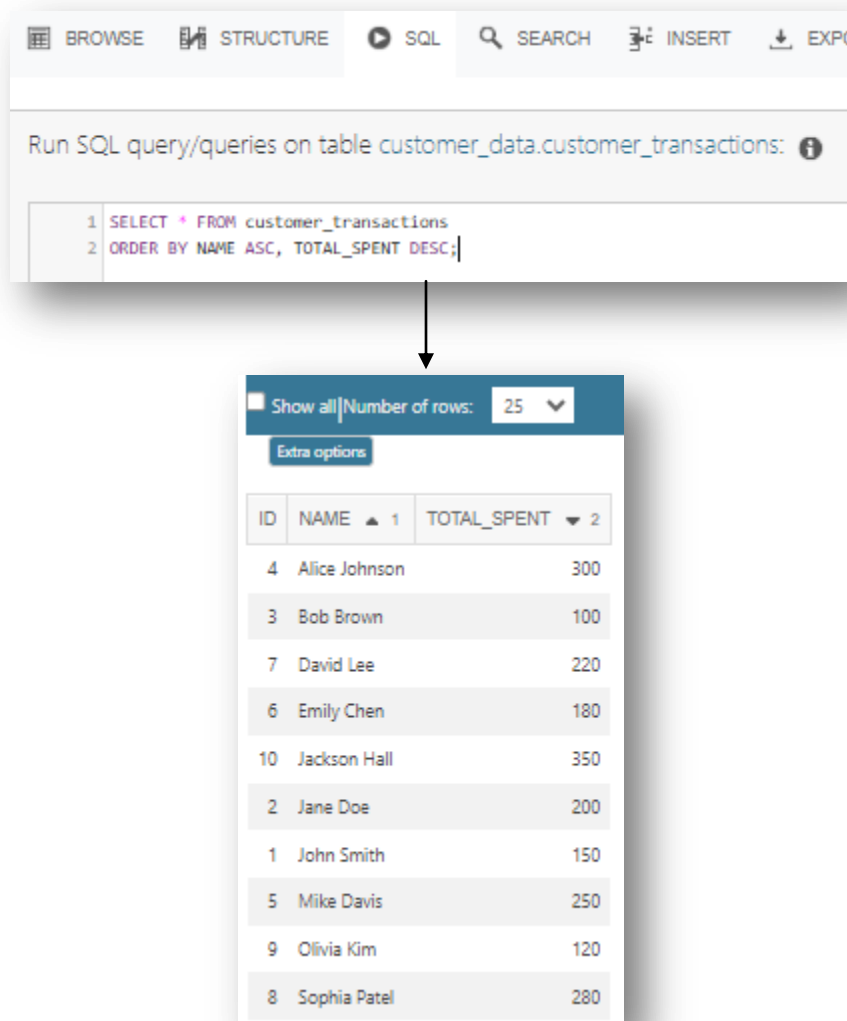
The screenshot shows a SQL query interface with a toolbar containing icons for BROWSE, STRUCTURE, SQL, SEARCH, INSERT, and EXPORT. Below the toolbar, a text prompt says "Run SQL query/queries on table customer_data.customer_transactions:". The SQL editor contains the following query:

```
1 SELECT * FROM customer_transactions
2 WHERE NAME LIKE 'A%' OR NAME LIKE 'E%';
```

An arrow points from the SQL editor to a results table. The results table has a header with "Show all" and "Number of rows: 25". Below the header is a button labeled "Extra options". The table has three columns: ID, NAME, and TOTAL_SPENT. The data rows are:

ID	NAME	TOTAL_SPENT
4	Alice Johnson	300
6	Emily Chen	180

- Ordering the data by “NAME” in ascending and “TOTAL_SPENT” in descending.



The screenshot shows a SQL query interface with a toolbar containing icons for BROWSE, STRUCTURE, SQL, SEARCH, INSERT, and EXPORT. Below the toolbar, a text prompt says "Run SQL query/queries on table customer_data.customer_transactions:". The SQL editor contains the following query:

```
1 SELECT * FROM customer_transactions
2 ORDER BY NAME ASC, TOTAL_SPENT DESC;
```

An arrow points from the SQL editor to a results table. The results table has a header with "Show all" and "Number of rows: 25". Below the header is a button labeled "Extra options". The table has three columns: ID, NAME, and TOTAL_SPENT. The data rows are:

ID	NAME	TOTAL_SPENT
4	Alice Johnson	300
3	Bob Brown	100
7	David Lee	220
6	Emily Chen	180
10	Jackson Hall	350
2	Jane Doe	200
1	John Smith	150
5	Mike Davis	250
9	Olivia Kim	120
8	Sophia Patel	280