

# WEEK 2 DATABASE ASSIGNMENT

## PART 1

### 1.1 Retrieving all expenses.

SELECT \* FROM Expenses;

Output:

The screenshot shows the MySQL Workbench interface. The SQL Editor contains the query `SELECT * FROM Expenses;`. The Results window displays the following data:

expense_id	amount	date	category
1	21.00	2023-09-17	Entertainment
2	74.00	2023-11-07	Transportation
3	33.00	2023-03-21	Other
4	77.00	2021-11-22	Groceries
5	80.00	2022-06-16	Entertainment
6	42.00	2021-01-24	Transportation
7	83.00	2021-07-30	Other

The bottom panel shows the Action Output for the query, indicating that 20 rows were returned.

### 1.2 Specific columns - SELECT date, category, amount FROM Expenses;

The screenshot shows the MySQL Workbench interface. The SQL Editor contains the query `SELECT date, category, amount FROM Expenses;`. The Results window displays the following data:

date	category	amount
2023-09-17	Entertainment	21.00
2023-11-07	Transportation	74.00
2023-03-21	Other	33.00
2021-11-22	Groceries	77.00
2022-06-16	Entertainment	80.00
2021-01-24	Transportation	42.00
2021-07-30	Other	83.00

The bottom panel shows the Action Output for the query, indicating that 20 rows were returned.

### 1.3 Filtering by date range

SELECT date, category, amount

FROM Expenses

WHERE date BETWEEN '2021-01-01' AND '2024-12-15';

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 SELECT date, category, amount
2 FROM Expenses
3 WHERE date BETWEEN '2021-01-01' AND '2024-12-15';
4
5
6
```

The 'Result Grid' displays the following data:

#	date	category	amount
1	2023-09-17	Entertainment	21.00
2	2023-11-07	Transportation	74.00
3	2023-03-21	Other	33.00
4	2021-11-22	Groceries	77.00
5	2022-06-16	Entertainment	80.00
6	2021-01-24	Transportation	42.00
7	2021-07-30	Other	83.00

The 'Action Output' pane shows the following log:

#	Time	Action	Message	Duration / Fetch
25	09:31:15	-- Call the procedure to insert sample data CALL InsertSampleData()	1 row(s) affected	2.672 sec
26	09:31:18	DROP PROCEDURE IF EXISTS InsertSampleData	0 row(s) affected	0.281 sec
27	09:31:18	DROP FUNCTION IF EXISTS GetRandomDate	0 row(s) affected	0.110 sec
28	09:31:55	SELECT * FROM expense_tracker.expenses LIMIT 0, 1000	20 row(s) returned	0.000 sec / 0.000 sec
29	08:30:47	SELECT * FROM Expenses LIMIT 0, 1000	20 row(s) returned	0.016 sec / 0.000 sec
30	08:36:21	SELECT date, category, amount FROM Expenses LIMIT 0, 1000	20 row(s) returned	0.000 sec / 0.000 sec
31	08:41:53	SELECT date, category, amount FROM Expenses WHERE date BETWEEN 2021-01-01 A...	19 row(s) returned	0.015 sec / 0.000 sec

## PART 2

### 2.1 Filtering by category

SELECT category

FROM Expenses

WHERE category = 'Entertainment';

MySQL Workbench

Local instance MySQL80 (practice) x Local instance MySQL80 (exp... x

File Edit View Query Database Server Tools Scripting Help

Navigator: Filter objects

SCHEMAS

- bursary
- employees
- expense\_tracker
  - Tables
    - expenses
  - Views
  - Stored Procedures
  - Functions
- mydb
- practice
- sys

SQL File 3\* expenses SQL File 5\*

```
1 SELECT category
2 FROM Expenses
3 WHERE category = 'Entertainment';
4
5
6
```

Result Grid Filter Rows: Export: Wrap Cell Contents:

category
Entertainment
Entertainment
Entertainment
Entertainment

Expenses 4 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
26	09:31:18	DROP PROCEDURE IF EXISTS InsertSampleData	0 row(s) affected	0.281 sec
27	09:31:18	DROP FUNCTION IF EXISTS GetRandomDate	0 row(s) affected	0.110 sec
28	09:31:55	SELECT * FROM expense_tracker.expenses LIMIT 0, 1000	20 row(s) returned	0.000 sec / 0.000 sec
29	08:30:47	SELECT * FROM Expenses LIMIT 0, 1000	20 row(s) returned	0.016 sec / 0.000 sec
30	08:36:21	SELECT date, category, amount FROM Expenses LIMIT 0, 1000	20 row(s) returned	0.000 sec / 0.000 sec
31	08:41:53	SELECT date, category, amount FROM Expenses WHERE date BETWEEN '2021-01-01' A...	19 row(s) returned	0.015 sec / 0.000 sec
32	08:46:13	SELECT category FROM Expenses WHERE category = 'Entertainment' LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec

Table: expenses

Columns:

- expense\_id int AI PK
- amount decimal(10,2)
- date date
- category varchar(50)

Object Info Session

Type here to search

14°C Sunny 8:46 AM 05-Jul-24

## 2.2 Filtering with comparison operators

SELECT date, category, amount

FROM Expenses

WHERE amount > 50;

MySQL Workbench

Local instance MySQL80 (practice) x Local instance MySQL80 (exp... x

File Edit View Query Database Server Tools Scripting Help

Navigator: Filter objects

SCHEMAS

- bursary
- employees
- expense\_tracker
  - Tables
    - expenses
  - Views
  - Stored Procedures
  - Functions
- mydb
- practice
- sys

SQL File 3\* expenses SQL File 5\*

```
1 SELECT date, category, amount
2 FROM Expenses
3 WHERE amount > 50;
4
5
6
```

Result Grid Filter Rows: Export: Wrap Cell Contents:

date	category	amount
2023-11-07	Transportation	74.00
2021-11-22	Groceries	77.00
2022-06-16	Entertainment	80.00
2021-07-30	Other	83.00
2022-03-14	Entertainment	88.00
2022-08-27	Transportation	86.00
2022-05-01	Groceries	82.00

Expenses 5 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
27	09:31:18	DROP FUNCTION IF EXISTS GetRandomDate	0 row(s) affected	0.110 sec
28	09:31:55	SELECT * FROM expense_tracker.expenses LIMIT 0, 1000	20 row(s) returned	0.000 sec / 0.000 sec
29	08:30:47	SELECT * FROM Expenses LIMIT 0, 1000	20 row(s) returned	0.016 sec / 0.000 sec
30	08:36:21	SELECT date, category, amount FROM Expenses LIMIT 0, 1000	20 row(s) returned	0.000 sec / 0.000 sec
31	08:41:53	SELECT date, category, amount FROM Expenses WHERE date BETWEEN '2021-01-01' A...	19 row(s) returned	0.015 sec / 0.000 sec
32	08:46:13	SELECT category FROM Expenses WHERE category = 'Entertainment' LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
33	08:49:16	SELECT date, category, amount FROM Expenses WHERE amount > 50 LIMIT 0, 1000	12 row(s) returned	0.015 sec / 0.000 sec

Table: expenses

Columns:

- expense\_id int AI PK
- amount decimal(10,2)
- date date
- category varchar(50)

Object Info Session

Type here to search

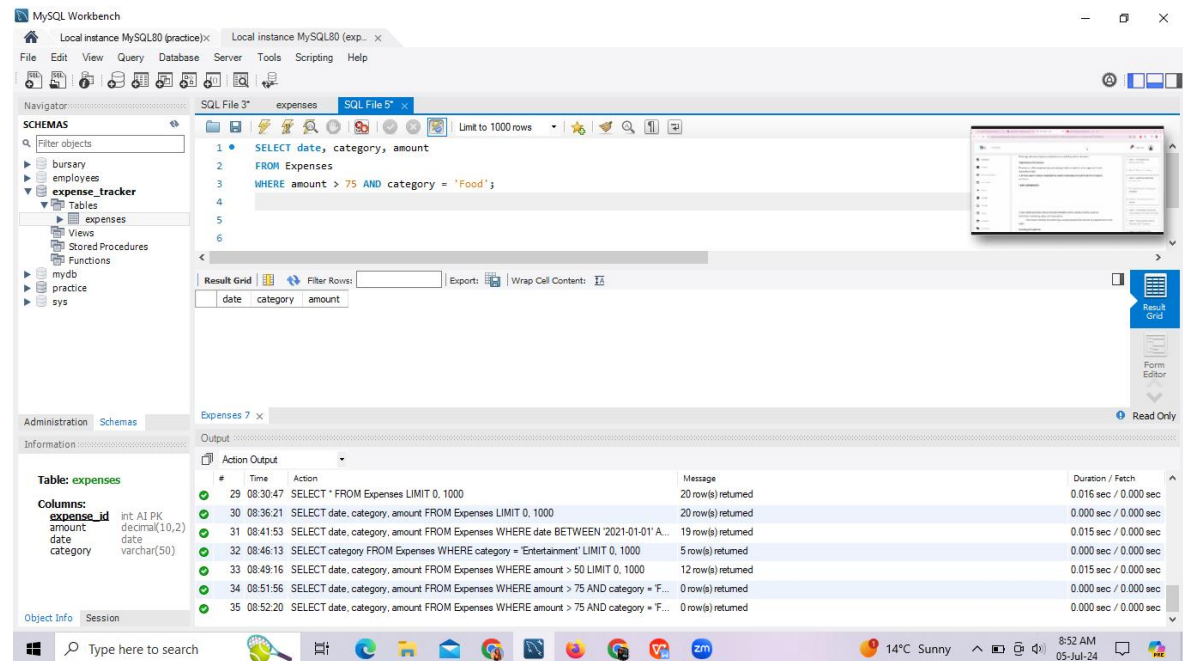
14°C Sunny 8:49 AM 05-Jul-24

## 2.3 Combining filters (AND)

SELECT date, category, amount

FROM Expenses

WHERE amount > 75 AND category = 'Food';



## 2.4 Combining filters (OR)

SELECT date, category, amount

FROM Expenses

WHERE category = 'Transportation' OR category = 'Groceries';



The screenshot shows the MySQL Workbench interface. The SQL Editor contains the following query:

```
1 SELECT date, category, amount
2 FROM Expenses
3 WHERE category = 'Transportation' OR category = 'Groceries';
4
5
6
```

The Result Grid displays the following data:

date	category	amount
2023-11-07	Transportation	74.00
2021-11-22	Groceries	77.00
2021-01-24	Transportation	42.00
2023-11-04	Groceries	22.00
2022-08-27	Transportation	86.00
2022-05-01	Groceries	82.00
2020-07-19	Transportation	83.00

The Output tab shows the execution log with the following messages:

#	Time	Action	Message	Duration / Fetch
30	08:36:21	SELECT date, category, amount FROM Expenses LIMIT 0, 1000	20 row(s) returned	0.000 sec / 0.000 sec
31	08:41:53	SELECT date, category, amount FROM Expenses WHERE date BETWEEN 2021-01-01' A...	19 row(s) returned	0.015 sec / 0.000 sec
32	08:46:13	SELECT category FROM Expenses WHERE category = 'Entertainment' LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
33	08:49:16	SELECT date, category, amount FROM Expenses WHERE amount > 50 LIMIT 0, 1000	12 row(s) returned	0.015 sec / 0.000 sec
34	08:51:56	SELECT date, category, amount FROM Expenses WHERE amount > 75 AND category = F...	0 row(s) returned	0.000 sec / 0.000 sec
35	08:52:20	SELECT date, category, amount FROM Expenses WHERE amount > 75 AND category = F...	0 row(s) returned	0.000 sec / 0.000 sec
36	08:55:34	SELECT date, category, amount FROM Expenses WHERE category = 'Transportation' OR c...	10 row(s) returned	0.000 sec / 0.000 sec

## 2.5 Filtering with NOT

SELECT date, category, amount

FROM Expenses

WHERE NOT category = 'Rent';

The screenshot shows the MySQL Workbench interface. The SQL Editor contains the following query:

```
1 SELECT date, category, amount
2 FROM Expenses
3 WHERE NOT category = 'Rent';
4
5
6
7
```

The Result Grid displays the following data:

date	category	amount
2023-09-17	Entertainment	21.00
2023-11-07	Transportation	74.00
2023-03-21	Other	33.00
2021-11-22	Groceries	77.00
2022-06-16	Entertainment	80.00
2021-01-24	Transportation	42.00
2021-07-30	Other	83.00
2023-11-04	Groceries	22.00

The Output tab shows the execution log with the following messages:

#	Time	Action	Message	Duration / Fetch
31	08:41:53	SELECT date, category, amount FROM Expenses WHERE date BETWEEN 2021-01-01' A...	19 row(s) returned	0.015 sec / 0.000 sec
32	08:46:13	SELECT category FROM Expenses WHERE category = 'Entertainment' LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
33	08:49:16	SELECT date, category, amount FROM Expenses WHERE amount > 50 LIMIT 0, 1000	12 row(s) returned	0.015 sec / 0.000 sec
34	08:51:56	SELECT date, category, amount FROM Expenses WHERE amount > 75 AND category = F...	0 row(s) returned	0.000 sec / 0.000 sec
35	08:52:20	SELECT date, category, amount FROM Expenses WHERE amount > 75 AND category = F...	0 row(s) returned	0.000 sec / 0.000 sec
36	08:55:34	SELECT date, category, amount FROM Expenses WHERE category = 'Transportation' OR c...	10 row(s) returned	0.000 sec / 0.000 sec

## PART 3

### 3.1 Sorting by amount

SELECT \*

FROM Expenses

ORDER BY amount DESC;

The screenshot shows the MySQL Workbench interface. The SQL Editor contains the following query:

```
1 SELECT *
2 FROM Expenses
3 ORDER BY amount DESC;
4
5
6
7
```

The Results window displays the following data:

expense_id	amount	date	category
15	95.00	2021-07-01	Other
9	88.00	2022-03-14	Entertainment
18	88.00	2023-09-06	Transportation
10	86.00	2022-08-27	Transportation
7	83.00	2021-07-30	Other
14	83.00	2020-07-19	Transportation
12	82.00	2022-05-01	Groceries
5	80.00	2022-06-16	Entertainment

The Output window shows the execution log:

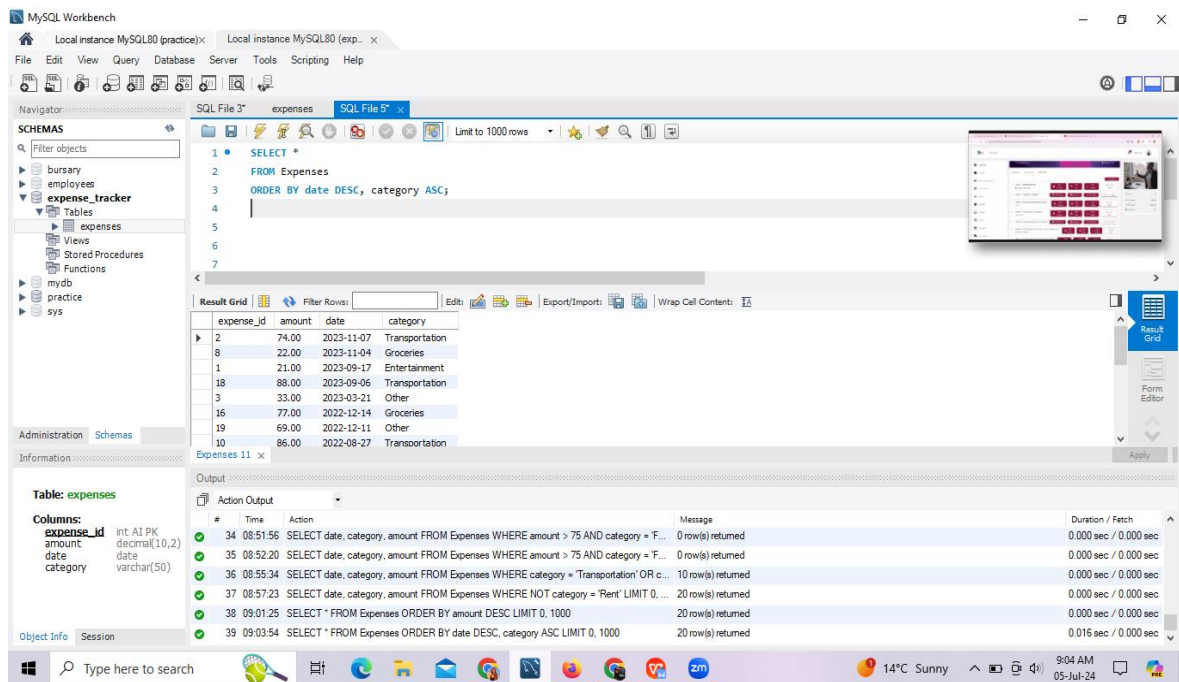
#	Time	Action	Message	Duration / Fetch
33	08:49:16	SELECT date, category, amount FROM Expenses WHERE amount > 50 LIMIT 0, 1000	12 row(s) returned	0.015 sec / 0.000 sec
34	08:51:56	SELECT date, category, amount FROM Expenses WHERE amount > 75 AND category = 'F...	0 row(s) returned	0.000 sec / 0.000 sec
35	08:52:20	SELECT date, category, amount FROM Expenses WHERE amount > 75 AND category = 'F...	0 row(s) returned	0.000 sec / 0.000 sec
36	08:55:34	SELECT date, category, amount FROM Expenses WHERE category = 'Transportation' OR c...	10 row(s) returned	0.000 sec / 0.000 sec
37	08:57:23	SELECT date, category, amount FROM Expenses WHERE NOT category = 'Rent' LIMIT 0, ...	20 row(s) returned	0.000 sec / 0.000 sec
38	09:01:25	SELECT * FROM Expenses ORDER BY amount DESC LIMIT 0, 1000	20 row(s) returned	0.000 sec / 0.000 sec

### 3.2 Sorting by date and category

SELECT \*

FROM Expenses

ORDER BY date DESC, category ASC;



## PART 4

### 4.1 Creating income table

CREATE TABLE Income (

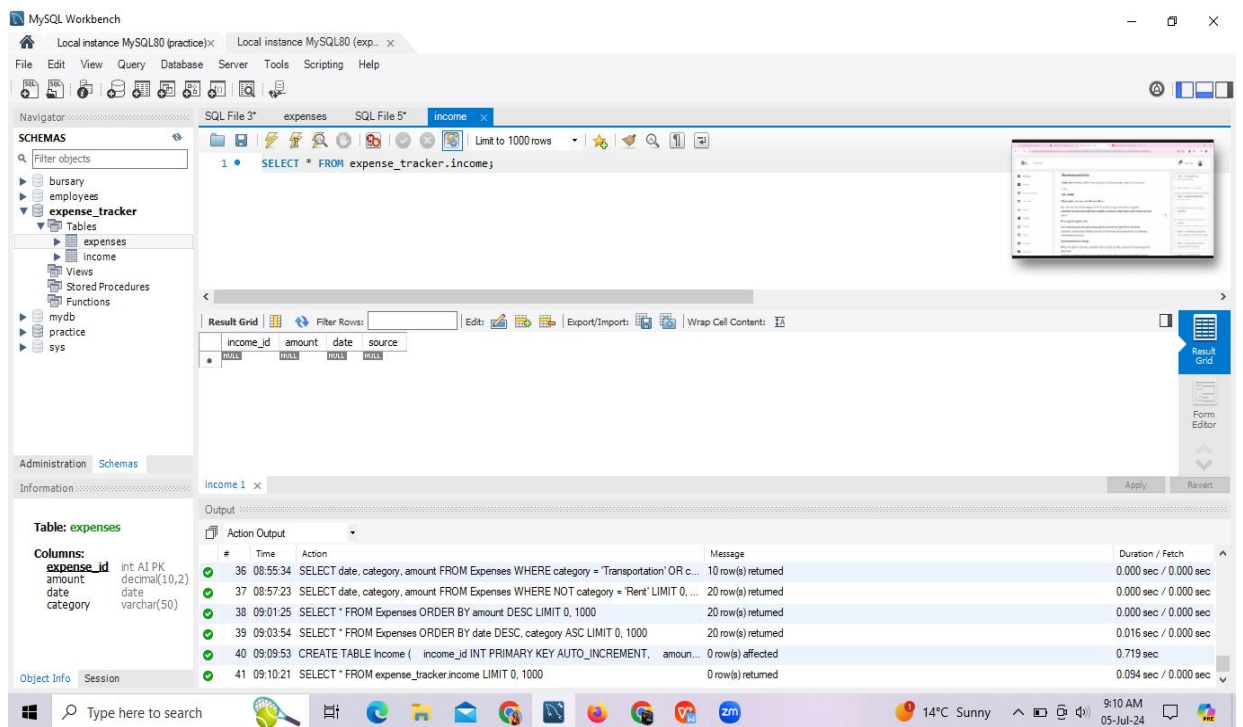
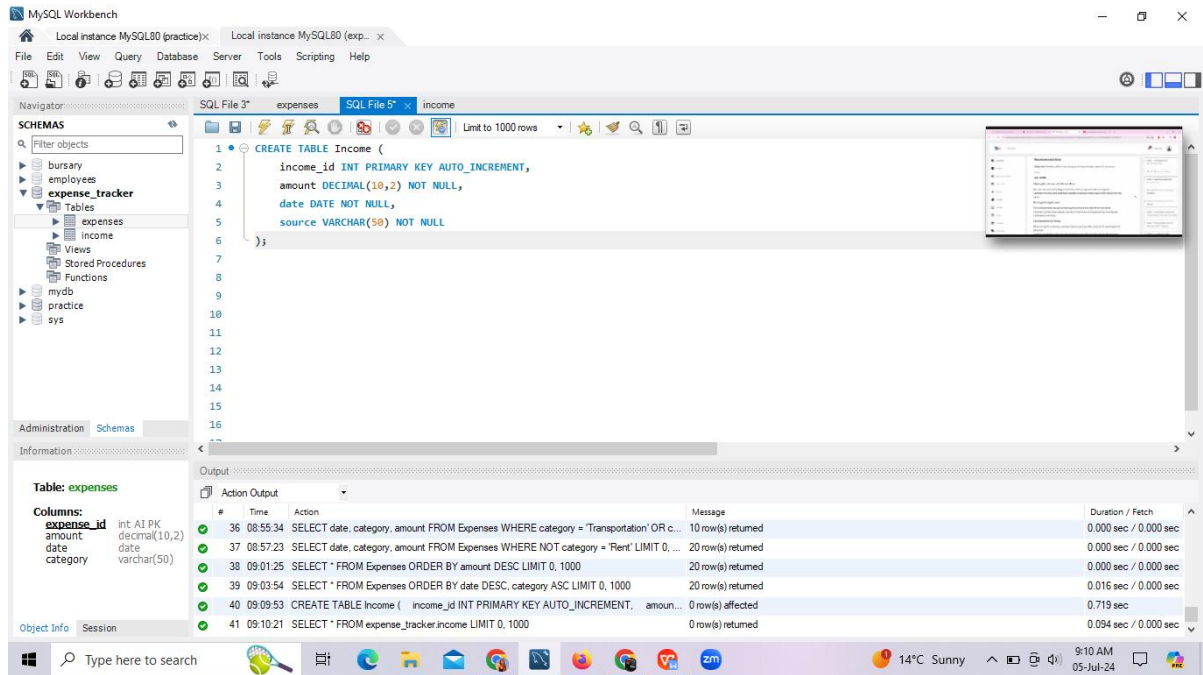
income\_id INT PRIMARY KEY AUTO\_INCREMENT,

amount DECIMAL(10,2) NOT NULL,

date DATE NOT NULL,

source VARCHAR(50) NOT NULL

);

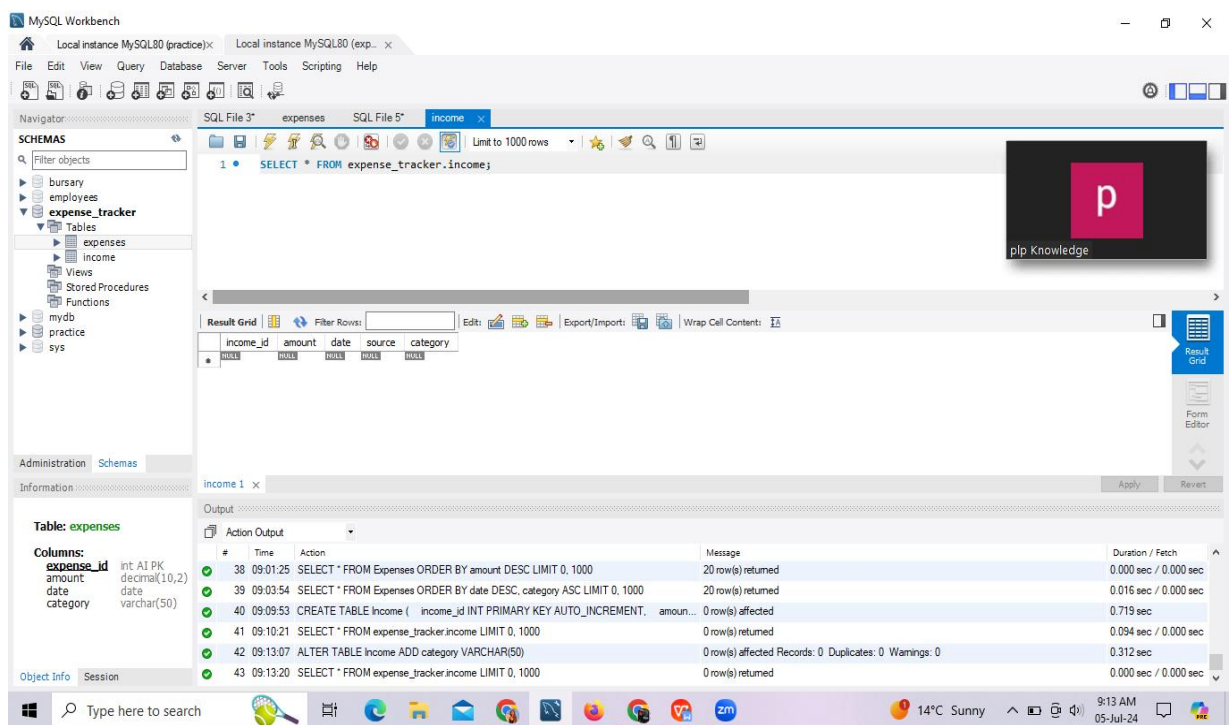
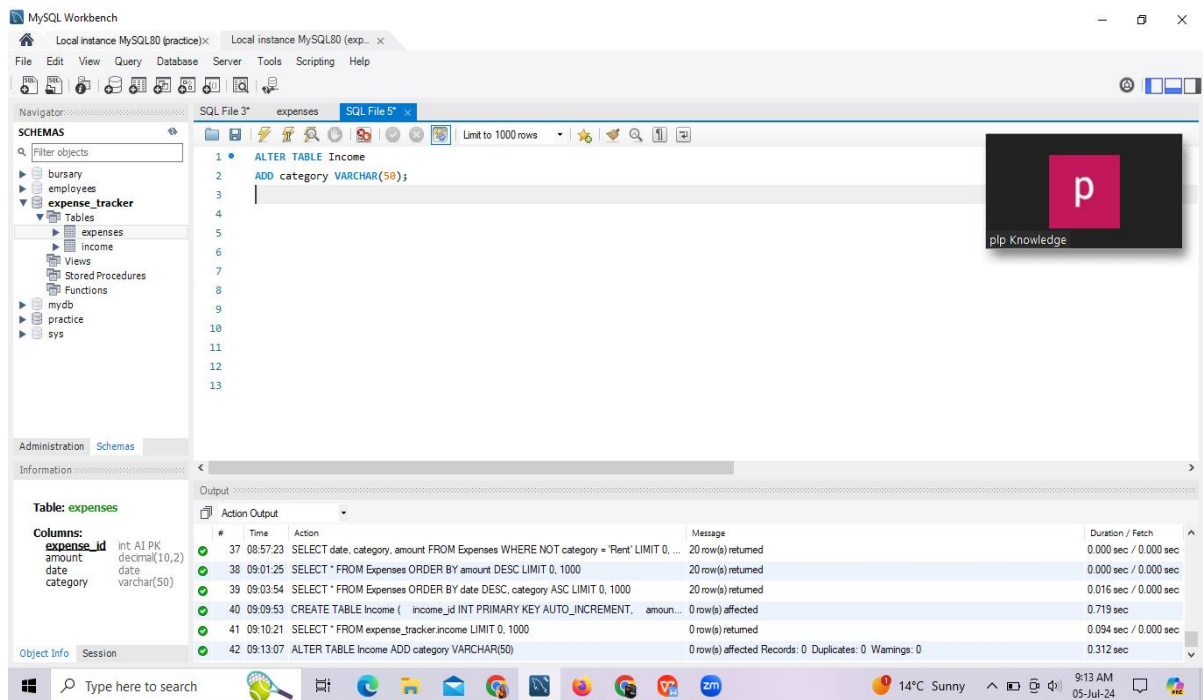


## 4.2 Adding category

ALTER TABLE Income

ADD category VARCHAR(50);

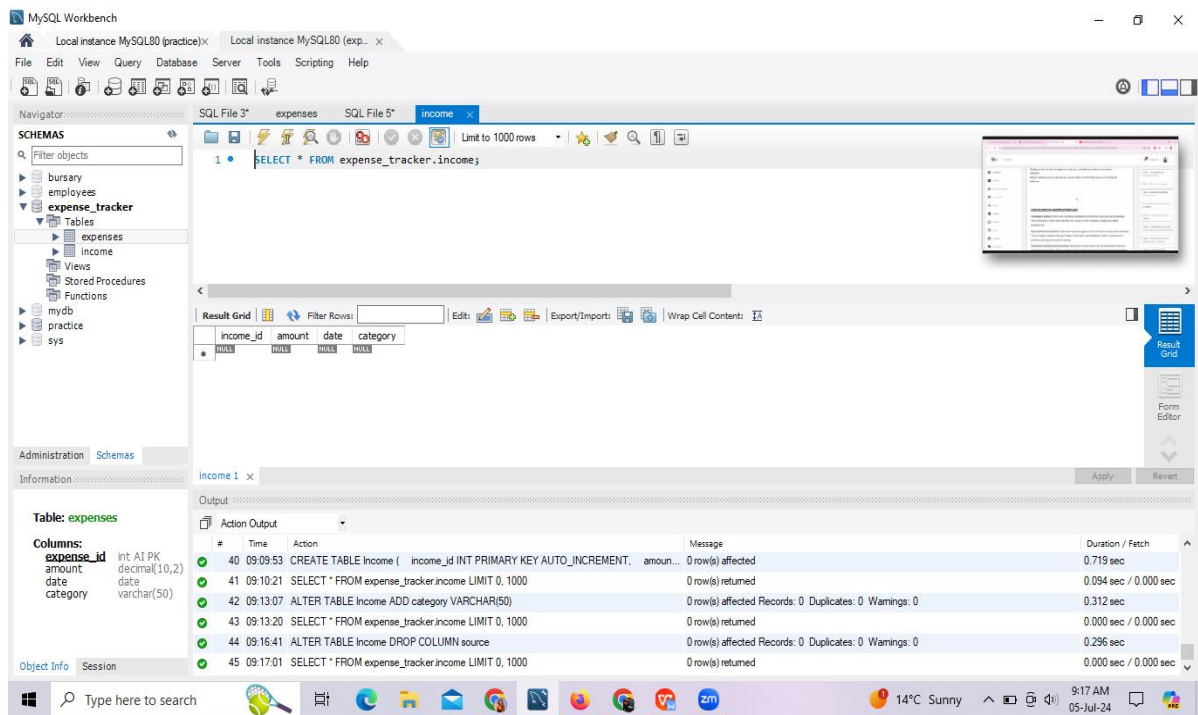
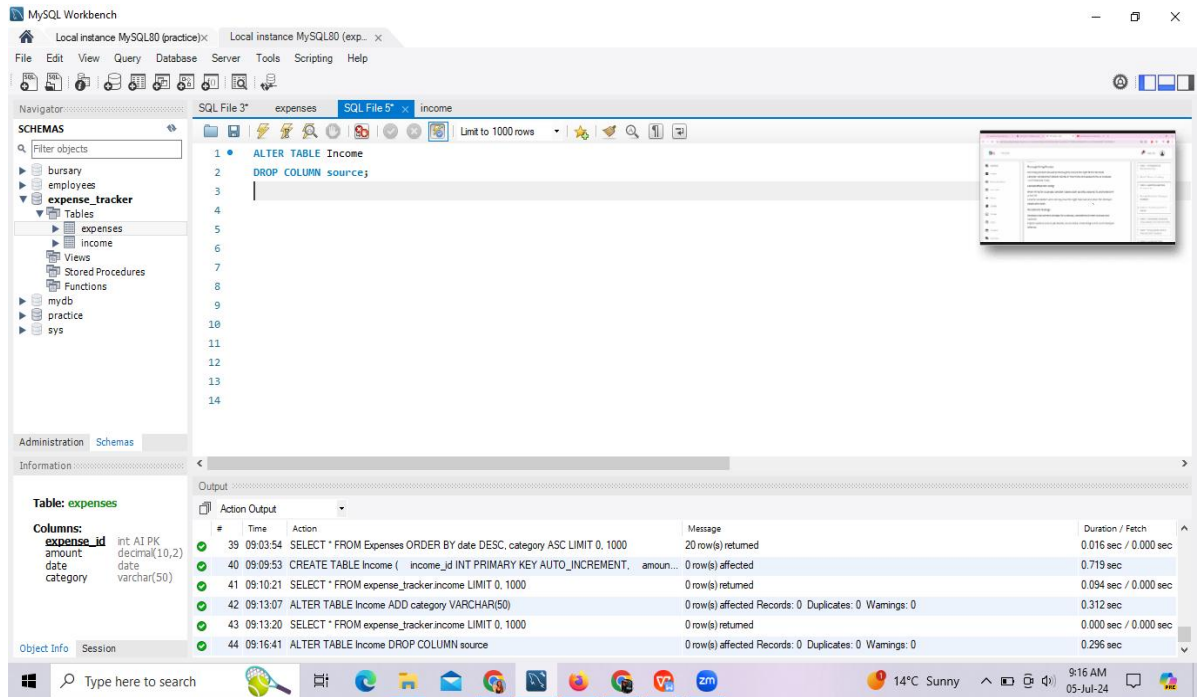




### 4.3 Dropping source

ALTER TABLE Income

DROP COLUMN source;



Dropping the income table

MySQL Workbench

Local instance MySQL80 (practice)x Local instance MySQL80 (exp... x

File Edit View Query Database Server Tools Scripting Help

SQL File 3\* expenses SQL File 5\* income SQL File 6\* x

Limit to 1000 rows

1 DROP TABLE Income;

2

SCHEMAS

Filter objects

- bursary
- employees
- expense\_tracker
  - Tables
    - expenses
    - Views
    - Stored Procedures
    - Functions
  - mydb
  - practice
  - sys

Administration Schemas

Information

Table: expenses

Columns:

- expense\_id int AI PK
- amount decimal(10,2)
- date date
- category varchar(50)

Object Info Session

Output

#	Time	Action	Message	Duration / Fetch
41	09:10:21	SELECT * FROM expense_tracker.income LIMIT 0, 1000	0 row(s) returned	0.094 sec / 0.000 sec
42	09:13:07	ALTER TABLE Income ADD category VARCHAR(50)	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.312 sec
43	09:13:20	SELECT * FROM expense_tracker.income LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec
44	09:16:41	ALTER TABLE Income DROP COLUMN source	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.296 sec
45	09:17:01	SELECT * FROM expense_tracker.income LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec
46	09:18:55	DROP TABLE Income	0 row(s) affected	0.532 sec

14°C Sunny 9:19 AM 05-Jul-24