Name: Leah Sigana

Assignment: Essential Data Retrieval & Filtering (Focus on Expense Tracker Data)

1.1 Retrieving All Expenses:

SQL

```
SELECT *
FROM Expenses;
```

This query uses the `SELECT *` clause to retrieve all columns (`*`) from the "Expenses" table. The `FROM` clause specifies the table from which data is retrieved.

1.2 Specific Columns:

SQL

```
SELECT date, category, amount
FROM Expenses;
```

This query selects only the "date," "category," and "amount" columns from the "Expenses" table.

1.3 Filtering by Date Range:

SQL

```
SELECT *
FROM Expenses
WHERE date >= '2024-01-01' AND date < '2024-02-16';
```

This query retrieves all expenses where the "date" column falls within a specific range. Here, it selects expenses from January 1, 2024 (inclusive) to February 15, 2024 (exclusive). The date format used is YYYY-MM-DD.

2.1 Filtering by Category:

SQL

```
SELECT *
FROM Expenses
WHERE category = 'Entertainment';
```

This query selects all expenses where the "category" column value exactly matches "'Entertainment'"

2.2 Filtering with Comparison Operators:

SQL

```
SELECT *
FROM Expenses
WHERE amount > 50;
```

This query selects all expenses where the "amount" column value is greater than 50.

2.3 Combining Filters (AND):

SQL

```
SELECT *
FROM Expenses
WHERE amount > 75 AND category = 'Food';
```

This query combines two filters using the `AND` operator. It retrieves expenses where the "amount" is greater than 75 AND the "category" is 'Food'.

2.4 Combining Filters (OR):

SQL

```
SELECT *
FROM Expenses
WHERE category = 'Transportation' OR category = 'Groceries';
```

This query uses the `OR` operator to filter expenses belonging to either 'Transportation' OR 'Groceries' category.

2.5 Filtering with NOT:

SQL

```
SELECT *
FROM Expenses
WHERE category <> 'Rent';
```

This query uses the `NOT` operator (often represented by `<>` or `!=`). It selects all expenses where the "category" is NOT equal to 'Rent'.

3.1 Sorting by Amount:

SQL

```
SELECT *
FROM Expenses
ORDER BY amount DESC;
```

This query sorts all expenses based on the "amount" column in descending order (DESC) from highest to lowest spending.

3.2 Sorting by Date and Category:

SQL

```
SELECT *
FROM Expenses
ORDER BY date DESC, category ASC;
```