FINANCE MANAGEMENT – CASE STUDY- SQL

By KARTHIKA KALIMUTHU, Excel Engineerinng College

```
create database Finance
use Finance
create table Users (
        user_id int primary key,
        username varchar (50),
        password varchar(30),
        email varchar (100)
    );

create table Expenses (
        expense_id int primary key,
        user_id int,
        amount decimal(10,2),
        category_id int,
        date date,
        description varchar (50),
        foreign key (user_id) references Users (user_id),
        foreign key (category_id) references ExpenseCategories (category_id)
    );

create table ExpenseCategories (
        category_id int primary key,
        category_name varchar (50)
    );
```

```
insert into Users values
(1, 'karthika', 'dj97478bs9', 'karthika@example.com'),
(2, 'meena', 'kdj953jvu9', 'meena@example.com'),
(3, 'jayasree', 'senck0984g', 'jayasree@example.com'),
(4, 'asha', 'e98mv65i45', 'asha@example.com'),
(5, 'amutha', 'wer876iu34', 'amutha@email.com')

select * from Users

insert into Expenses values
(101, 1, 250.00, 1, '2025-06-01', 'lunch at restaurant'),
(102, 2, 1200.00, 2, '2025-06-03', 'cab to airport'),
(103, 3, 7000.00, 3, '2025-06-05', 'monthly rent'),
(104, 4, 450.00, 4, '2025-06-08', 'movie with friends'),
(105, 5, 1000.00, 5, '2025-06-10', 'electricity bill'),
(106, 1, 150.00, 1, '2025-06-11', 'snacks'),
(107, 2, 2000.00, 1, '2025-06-12', 'dinner with family')

select * from Expenses

insert into ExpenseCategories values
(1, 'food'),
(2, 'travel'),
(3, 'rent'),
(4, 'entertainment'),
(5, 'utilities')

select * from ExpenseCategories
```

----------**Queries**----------

1. List all users with their email addresses.

```
select username, email from Users
```

2. Display all expenses greater than 500.

```
select amount from Expenses
where amount > 500
```

3. Expenses happened between any specific periods.

```
select amount, date from Expenses
where date between '2025-06-03' and '2025-06-10'
```

4. name started with a.

```
select username from Users
where username like 'a%'
```

5. Expenses based on electricity bill.

```
select description, amount from Expenses
where description = 'electricity bill'
```

6. Expenses in descending order.

```
select amount as expenses from Expenses
order by expenses desc
```

7. Highest expenses in specific period.

```
select amount, date from Expenses
where date between '2025-06-03' and '2025-06-10' order by amount desc
```

8. total amount spend by each user.

```
select user_id, sum (amount) as tot_amt from Expenses
group by user_id
```

9. Average expenses amounts per category.

```
select category_id, avg (amount) as avg_amt from Expenses
group by category_id
```

10. Top 3 highest expenses.

```
select top (3) amount as highest_expenses from Expenses
order by amount desc
```

11. categories where average of expenses more than 1000.

```
select category_id, avg (amount) as avg_expenses from Expenses
group by category_id
having avg (amount) > 500
```

12. users who have spent more than 2000 in total.

```
select user_id, sum (amount) total_exp from Expenses
group by user_id
having sum (amount) >2000
```

13. List the expenses with username, category name and amount.

```
select u.username, e.amount, ec.category_name from Users as u
inner join Expenses as e
on u.user_id = e.user_id
inner join ExpenseCategories as ec
on e.user_id = ec.category_id
```

14. List all the users with their expenses.

```
select u.user_id, u.username, e.expense_id, e.amount, e.date from Users as u
left join Expenses as e
on u.user_id = e.user_id
```

15. list all expenses descriptions with user email and category name.

```
select ec.category_name, e.expense_id, e.amount from ExpenseCategories as ec
right join Expenses as e
on ec.category_id = e.category_id
```

16. Each user with their total expenses more than 2000.

```
select u.username, sum (e.amount) as  tot_amt from Users as u
inner join Expenses as e
on u.user_id = e.user_id
group by u.username
having sum (e.amount) > 2000
```

17.List all expenses made in most recent date.

```
select * from Expenses
where date = (select max (date) from Expenses)
```

18. List all expenses greater than the average expense amount.

```
select amount as expenses from Expenses
where amount > (select avg (amount) avg_amt from Expenses)
```

19. description for the expense with the highest amount.

```
select description, amount from Expenses
where amount = (select max (amount) as Highest_amt from Expenses)
```

20. Find the usernames of users who made the smallest expense.

```
select u.username,e.amount from Users as u

inner join Expenses as e

on u.user_id = e.user_id

where e.amount = (select min (amount) from Expenses)
```

21. for each category what is the min and max expenses.

```
select ec.category_name, max (e.amount) as max_exp, min (e.amount) as min_exp from ExpenseCategories as ec

join Expenses as e

on ec.category_id = e.category_id

group by ec.category_name
```