Stored Procedures and User-Defined Functions - Task 9

RDBMS used : MySql

Stored procedure

A stored procedure is a group of SQL statements that are **pre-complied**, and can be reused anytime. Stored procedures can be used to perform different database operations such as inserting, updating, or deleting data.

SYNTAX:

```
DELIMITER //

CREATE PROCEDURE procedure_name([IN|OUT|INOUT] parameter_name parameter_datatype)
BEGIN
        -- SQL statements to be executed
END //
```

- **Procedure without any parameters**

```
delimiter //
create procedure GetProdInfo()
begin
        select * from products where price>2000;
end //

call GetProdInfo();
```

prod_id	prod_name	category	price	color	size	gender
1001	sneakers	casual	2500	dark brown	6	1
1002	loafers	formal	3500	deep black	7	1
1003	boots	semi-formal	3000	black	5	0

- **Procedure with IN parameter**
  - used to receive input value from the calling program.

```
delimiter //
create procedure GetProdWithSize(IN reqsize tinyint)
begin
        select prod_id, prod_name, category, price from products
```

```
   where size = reqsize;
end //
```

prod_id	prod_name	category	price
1003	boots	semi-formal	3000
1004	flip-flops	casual	500

- **Getting all orders within a specific date range**

```
delimiter //
create procedure GetTotalOrders(IN fromdate date, IN todate date)
begin
        select * from orders
   where order_date between fromdate and todate;
end //

call GetTotalOrders("2020-05-01","2021-08-31");
```

order_id	order_date	prod_id	quantity	order_status	region_id
2	2020-07-17	1001	4	packaged	AP
4	2021-02-22	1003	1	shipped	KA

- **Getting the total order quantity within a certain date range**

```
delimiter //
create procedure GetTotalQty(IN fromdate date, IN todate date)
begin
        select sum(quantity) as TOTAL_QTY from orders
   where order_date between fromdate and todate;
end //

call GetTotalQty("2020-05-01","2021-08-31");
```

TOTAL_QTY
5

➢ **Procedure with OUT parameter**

- Used to send the output values to the calling program.
- It is necessary to specify the **OUT** keyword to an output parameter when creating the procedure.
- At the time of calling, a variable, prefixed with '@' is used to hold the returned value.
- Then, a SELECT statement can be used on the variable to display the output of the procedure.

```
delimiter //
create procedure GetProdDet(IN prodid int, OUT prodprice decimal)
begin
        select price INTO prodprice from products
   where prod_id = prodid;
end //

call GetProdDet(1002, @p);
select @p as price;
```

	price
▶	3500

User-Defined Functions

A stored function is a set of SQL statements that perform some operation and return a single value. The **CREATE FUNCTION** statement is used for creating a stored function and user-defined functions.

Syntax:

```
DELIMITER //
CREATE FUNCTION function_name(func_parameter1, func_parameter2, ..)
RETURNS datatype [characteristics]
BEGIN
        MySQL statements
        RETURN expression;
END //
```

- **Function that calculates discount based on input parameters**

```
delimiter //
create function CalcDisc(amt decimal)
returns decimal deterministic
begin
        declare discamt decimal;
   set discamt = amt - amt*0.10;
   return discamt;
end //

select prod_id, prod_name, price, CalcDisc(price) as After_discount
from products;
```

prod_id	prod_name	price	After_discount
1001	sneakers	2500	2250
1002	loafers	3500	3150
1003	boots	3000	2700
1004	flip-flops	500	450
1005	boots	500	450
1006	flip-flops	300	270