

ASSIGNMENT 9

Aim : Write and execute PL/SQL stored procedure and function to perform a suitable task on above DB

Problem Statement : • PL/SQL assignment on tables created

- Write simple PL/SQL programs to perform different operations on table
- Write trigger and execute it on table.

Objective : • To understand PL/SQL

- To understand the concept of procedures / functions

Theory :

Creating a stored procedure :

eg- delimiter //

```
create procedure p2 ()
language sql
deterministic
sql security definer
comment 'A procedure'
begin
    select 'hello world!';
end //
```

- The first part of statement creates procedure • The next clauses define optional characteristics of procedure. Then follows the name and body.
- Stored procedure names are case insensitive • You cannot create procedures with same name
- ALTER PROCEDURE - modify a routine
- Delete a stored procedure
delete procedure if exists p2;

You can create procedures with different parameter types •

IN	OUT	INOUT
delimiter //	delimiter //	delimiter
create procedure	create procedure proc-out	create procedure
proc-IN (IN var1 int)	(OUT var1 varchar (100))	proc_inout (INOUT var1 int)
begin	begin	begin
select var1+2 as result;	set var1 = 'test' ;	set var1 = var1 *2;
end //	end //	end //

Variables: You must declare variables explicitly at the start of begin/end block along with their data types.

Syntax: declare var-name DATA-TYPE DEFAULT def-value;

Flow Control Structures: MYSQL supports IF, CASE, ITERATE, LEAVE LOOP, WHILE and REPEAT constructs for flow control within stored programs:

IF	CASE	WHILE
<ul style="list-style-type: none"> with if statement we can handle tasks which involve conditions 	<ul style="list-style-type: none"> another way to check conditions, excellent way to replace multiple if statements 	<ul style="list-style-type: none"> there are 3 standard loops: WHILE, LOOP, REPEAT
<p>Eg- delimiter //</p> <pre> create procedure proc_if (in par1 int) begin declare v1 int; set v1 = par1 + 1; if v1 = 0 then select v1; end if; if par1 = 0 then select 'value = 0'; else select 'value <> 0'; end if; end //</pre>	<p>Eg- delimiter //</p> <pre> create procedure proc_case (in par1 int) begin declare v1 int; set v1 = par1 + 1; case v1 when 0 then insert into t1 values (par1); when 1 then insert into t1 values (v1); else insert into t1 values (99); end case; end //</pre>	<p>Eg- delimiter //</p> <pre> create procedure proc_while (in par1 int) begin declare v1, v2 int; set v1 = 0; while v1 < par1 do insert into table1 values (par1); select count (*) into v2 from table1; set v1 = v1 + 1; end while; end //</pre>

Conclusion: Understood the concept of PL/SQL block by implementing all types of procedures and functions on DB.