

PLSQL→Procedural language sql

Why we are using PLSQL

1. To increase the security by hiding table names and queries inside the procedure
2. If any operation needs to execute multiple queries at one go. Then instead of sending queries one by one from front end application, we may wrap these queries, inside the procedure, and call procedure only once, it will reduce network traffic, and increases the speed of execution.

Using PLSQL

1. We may write procedures, functions and triggers
2. If you want to return single value then use function, functions can be called in select and where clause.
3. If you want to execute operation automatically without user's knowledge, then you write triggers.
4. Otherwise write procedures.

### Procedures

1. To a procedure we need to pass some parameters,  
These parameters are of 3 types
  - a. In
  - b. Out
  - c. In out

In	out	In out
These parameters are used to pass data to the procedure	These parameters are used to get data as output from the procedure	These parameters are used for passing data as input also, the values can be modified inside the procedure and send as output to frontend application
These parameters are read only parameters	These are write only parameters	These are read and write parameters
If you don't mention the type of parameters then by default the type is in.	You need to explicitly specify out word to define out parameters	you need to explicitly specify in out word to define in out parameters

1. Write a procedure to insert a record in dept table

Create procedure insert\_dept(dno int,dnm varchar(20),dloc varchar(20))

As

Begin

Insert into dept values(dno,dnm,dloc);

End//

2. Write a procedure to find number of rows in emp table

Delimiter //

Create procedure gentcnt(out cnt int)

As

Begin

    Select count(\*) into cnt

From emp;

End//

Delimiter ;

Call gentcnt(@c);

Select @cnt;

3. Write a procedure to add a number of employees in department 10, into totalcnt

Create procedure gentcnt(in out totalcnt int,out cnt int)

As

Begin

    Select count(\*) into cnt

Set totalcnt=totalcnt+cnt;

From emp;

End//

4. Find number of employees of the given department

delimiter //

create procedure getcnt(in dno int,out cnt int)

begin

    select count(\*) into cnt

    from emp

    where deptno=dno;

end//

delimiter ;

call getcnt(10,@c);

```
select @c;
```

```
delimiter //
```

```
create procedure insert_dept(dno int,dnm varchar(20),dl varchar(20))
```

```
begin
```

```
    insert into dept values(dno,dnm,dl);
```

```
    select *
```

```
    from dept;
```

```
end//
```

```
delimiter ;
```

```
call insert_dept(50,'purchase','Pune');
```

5. Using inout parameter, add value of number of employees in department 10 into tot;

```
create procedure gettotal(inout tot int)
```

```
begin
```

```
declare cnt int;
```

```
select count(*) into cnt
```

```
from emp
```

```
where deptno=10;
```

```
set tot=tot+cnt;
```

```
select tot;
```

```
end//
```

```
set @num=20;
```

```
call gettotal(@num)
```

```
select @num;
```