

```
SQL> CREATE TABLE Employees (  
2 EmployeeID INT PRIMARY KEY,  
3 FirstName VARCHAR2(50),  
4 LastName VARCHAR2(50),  
5 DepartmentID INT,  
6 Salary NUMBER(10,2),  
7 JoinDate DATE  
8 );
```

Table created.

```
SQL> CREATE TABLE Departments (  
2 DepartmentID INT PRIMARY KEY,  
3 DepartmentName VARCHAR2(100)  
4 );
```

Table created.

```
SQL> INSERT INTO Departments (DepartmentID, DepartmentName) VALUES  
2 (1, 'HR');
```

1 row created.

```
SQL> INSERT INTO Departments (DepartmentID, DepartmentName) VALUES  
2 (2, 'Finance');
```

1 row created.

```
SQL> INSERT INTO Departments (DepartmentID, DepartmentName) VALUES  
2 (3, 'IT');
```

1 row created.

```
SQL> INSERT INTO Departments (DepartmentID, DepartmentName) VALUES  
2 (4, 'Marketing');
```

1 row created.

```
SQL>  
SQL> INSERT INTO Employees (EmployeeID, FirstName, LastName, DepartmentID, Salary, JoinDate)  
VALUES  
2 (1, 'John', 'Doe', 1, 5000.00, TO_DATE('2020-01-15', 'YYYY-MM-DD'));
```

1 row created.

```
SQL> SELECT table_name FROM user_tables;
```

TABLE\_NAME

-----

STUDENT

PERSON

STUDENTS

STUDENT3

REST

CLASS

CLASSROOM

PRODUCTS  
EMPLOYEES  
DEPARTMENTS

10 rows selected.

```
SQL> CREATE OR REPLACE PROCEDURE sp_GetEmployeesByDepartment (  
  2   p_DeptID IN NUMBER  
  3 )  
  4 IS  
  5 BEGIN  
  6   FOR empRec IN (  
  7     SELECT  
  8       EmployeeID,  
  9       FirstName,  
10       LastName,  
11       DepartmentID,  
12       Salary,  
13       JoinDate  
14   FROM Employees  
15   WHERE DepartmentID = p_DeptID  
16 )  
17 LOOP  
18   DBMS_OUTPUT.PUT_LINE(  
19     'EmployeeID: ' || empRec.EmployeeID ||  
20     ', Name: ' || empRec.FirstName || ' ' || empRec.LastName ||  
21     ', DepartmentID: ' || empRec.DepartmentID ||  
22     ', Salary: ' || empRec.Salary ||  
23     ', JoinDate: ' || TO_CHAR(empRec.JoinDate, 'YYYY-MM-DD')  
24   );  
25 END LOOP;  
26 END;  
27 /
```

Procedure created.

```
SQL> SET SERVEROUTPUT ON;  
SQL> EXEC sp_GetEmployeesByDepartment(2);
```

PL/SQL procedure successfully completed.

```
SQL> CREATE OR REPLACE PROCEDURE sp_InsertEmployee (  
  2   p_FirstName  IN VARCHAR2,  
  3   p_LastName   IN VARCHAR2,  
  4   p_DepartmentID IN NUMBER,  
  5   p_Salary     IN NUMBER,  
  6   p_JoinDate   IN DATE  
  7 )  
  8 IS  
  9 BEGIN  
10   INSERT INTO Employees (EmployeeID, FirstName, LastName, DepartmentID, Salary, JoinDate)  
11   VALUES (  
12     (SELECT NVL(MAX(EmployeeID), 0) + 1 FROM Employees), -- Auto-increment workaround  
13     p_FirstName,  
14     p_LastName,  
15     p_DepartmentID,  
16     p_Salary,
```

```

17     p_JoinDate
18 );
19
20 DBMS_OUTPUT.PUT_LINE('Employee inserted successfully. ');
21 END;
22 /

```

Procedure created.

```

SQL> SET SERVEROUTPUT ON;
SQL>
SQL> EXEC sp_InsertEmployee(
BEGIN sp_InsertEmployee(; END;

```

\*

ERROR at line 1:

ORA-06550: line 1, column 25:

PLS-00103: Encountered the symbol ";" when expecting one of the following:

( ) - + case mod new not null <an identifier>

<a double-quoted delimited-identifier> <a bind variable>

table continue avg count current exists max min prior sql

stddev sum variance execute multiset the both leading

trailing forall merge year month day hour minute second

timezone\_hour timezone\_minute timezone\_region timezone\_abbrev

time timestamp interval date

<a string literal with character set specification>

```

SQL> p_FirstName => 'Alice',
SP2-0734: unknown command beginning "p_FirstNam..." - rest of line ignored.
SQL> p_LastName => 'Williams',
SP2-0734: unknown command beginning "p_LastName..." - rest of line ignored.
SQL> p_DepartmentID => 3,
SP2-0734: unknown command beginning "p_Departme..." - rest of line ignored.
SQL> p_Salary => 6200.00,
SP2-0734: unknown command beginning "p_Salary =..." - rest of line ignored.
SP2-0044: For a list of known commands enter HELP
and to leave enter EXIT.
SQL> p_JoinDate => TO_DATE('2022-06-01', 'YYYY-MM-DD')
SP2-0734: unknown command beginning "p_JoinDate..." - rest of line ignored.
SQL> );
SP2-0042: unknown command ")" - rest of line ignored.
SQL> SET SERVEROUTPUT ON;
SQL>
SQL> BEGIN
2   sp_InsertEmployee(
3     p_FirstName  => 'Alice',
4     p_LastName   => 'Williams',
5     p_DepartmentID => 3,
6     p_Salary     => 6200.00,
7     p_JoinDate   => TO_DATE('2022-06-01', 'YYYY-MM-DD')
8   );
9 END;
10 /

```

Employee inserted successfully.

PL/SQL procedure successfully completed.

```
SQL> EXEC sp_InsertEmployee('Alice', 'Williams', 3, 6200.00, TO_DATE('2022-06-01', 'YYYY-MM-DD'));
```

Employee inserted successfully.

PL/SQL procedure successfully completed.

```
SQL>
```