**Experiment-10**

**Aim:**

To Perform PL/SQL Commands.

**Types of PL/SQL:**

Syntax, Comments, Variable Attributes, Conditionals: IF-THEN-ELSE, Case, Loops – For, While

**Input:**

Syntax

DECLARE

message varchar2(20):= 'Hello World From ';

BEGIN

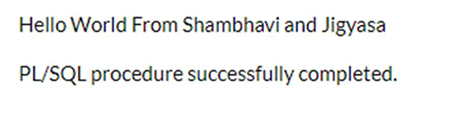
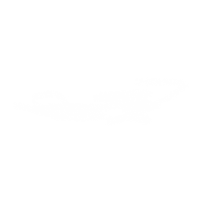
 dbms\_output.put\_line(message);

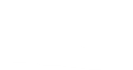
END;

/



**O/P:**





Comments

DECLARE

-- variable declaration

message varchar2(20):= 'Hello, World!';

BEGIN

/\*

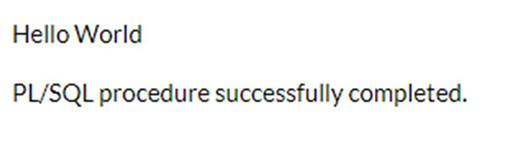
\* PL/SQL executable statement(s)

\*/

dbms\_output.put\_line(message);

END; /

**O/P:**

****

Example

DECLARE

a integer := 30;

b integer := 40;

c integer;

f real;

BEGIN

c := a + b;

dbms\_output.put\_line('Value of c: ' || c);

f := 100.0/3.0;

dbms\_output.put\_line('Value of f: ' || f);

END;

**O/P:**

**A number on a white background

Description automatically generated**

Variable Attributes

*% TYPE*

DECLARE

SALARY EMP.SAL % TYPE;

ECODE EMP.empno % TYPE;

BEGIN

Ecode :=&Ecode;

Select SAL into SALARY from EMP where EMPNO = ECODE;

dbms\_output.put\_line('Salary of ' || ECODE || 'is = || salary');

END;

**O/P:**

**A black text on a white background

Description automatically generated**

*%ROWTYPE*

*DECLARE*

*EMPLOYEE EMP. % ROW TYPE;*

*BEGIN*

*EMPLOYEE.EMPNO := 2092;*

*5 EMPLOYEE.ENAME := 'Sanju';*

*Insert into EMP where (EMPNO, ENAME) Values (employee.empno, employee.ename);*

*dbms\_output.put\_line('Row Inserted');*

*END;*

**O/P:**

**A black text on a white background

Description automatically generated**

Conditionals

1) IF -THEN-ELSE

DECLARE

a number(3) := 500;

BEGIN

-- check the boolean condition using if statement

IF( a < 20 ) THEN

-- if condition is true then print the following

dbms\_output.put\_line('a is less than 20 ' );

ELSE

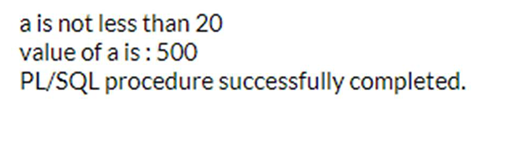
dbms\_output.put\_line('a is not less than 20 ' );

END IF;

dbms\_output.put\_line('value of a is : ' || a);

END;

**O/P:**

****

2) CASE

DECLARE

grade char(1) := 'A';

BEGIN

CASE grade

when 'A' then dbms\_output.put\_line('Excellent');

when 'B' then dbms\_output.put\_line('Very good');

when 'C' then dbms\_output.put\_line('Good');

when 'D' then dbms\_output.put\_line('Average');

when 'F' then dbms\_output.put\_line('Passed with Grace');

else dbms\_output.put\_line('Failed');

END CASE;

END;

**O/P:**

**A close up of a text

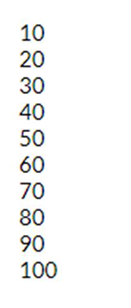
Description automatically generated**

Loop

1) FOR

DECLARE

VAR1 NUMBER;



BEGIN

VAR1:=10;

FOR VAR2 IN 1..10

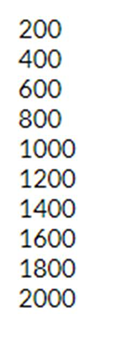
LOOP

DBMS\_OUTPUT.PUT\_LINE (VAR1\*VAR2);

END LOOP;

END;

2) WHILE



DECLARE

VAR1 NUMBER;

VAR2 NUMBER;

BEGIN

VAR1:=200;

VAR2:=1;

WHILE (VAR2<=10)

LOOP

DBMS\_OUTPUT.PUT\_LINE (VAR1\*VAR2);

VAR2:=VAR2+1;

END LOOP;

END;

# Experiment-9

**Aim:**

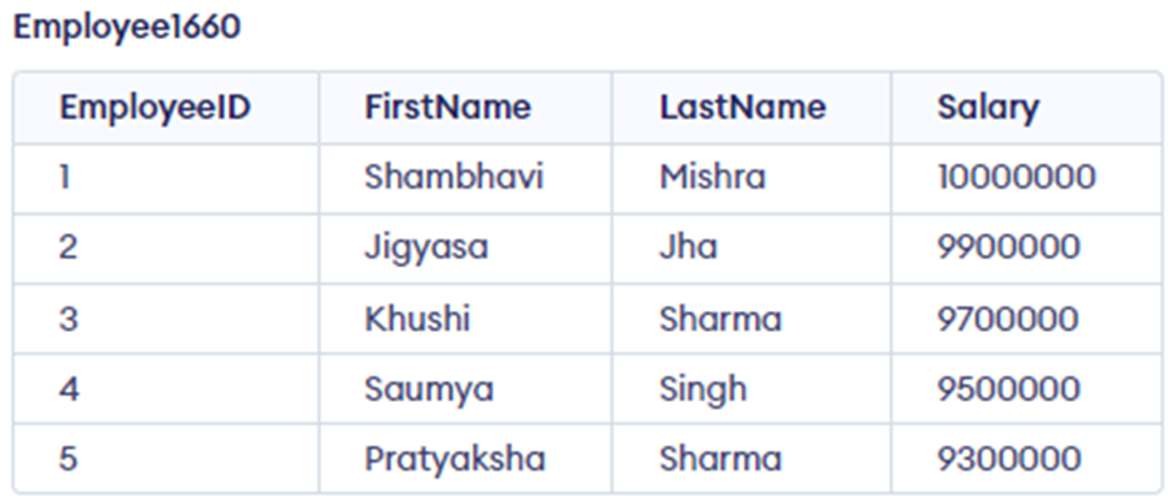
To Practice View Command.

# Input: (Create, Update and Delete)

* CREATE TABLE Employee1660( EmployeeID int NOT NULL PRIMARY KEY, FirstName varchar(255) NOT NULL, LastName varchar(255),

Salary int

);

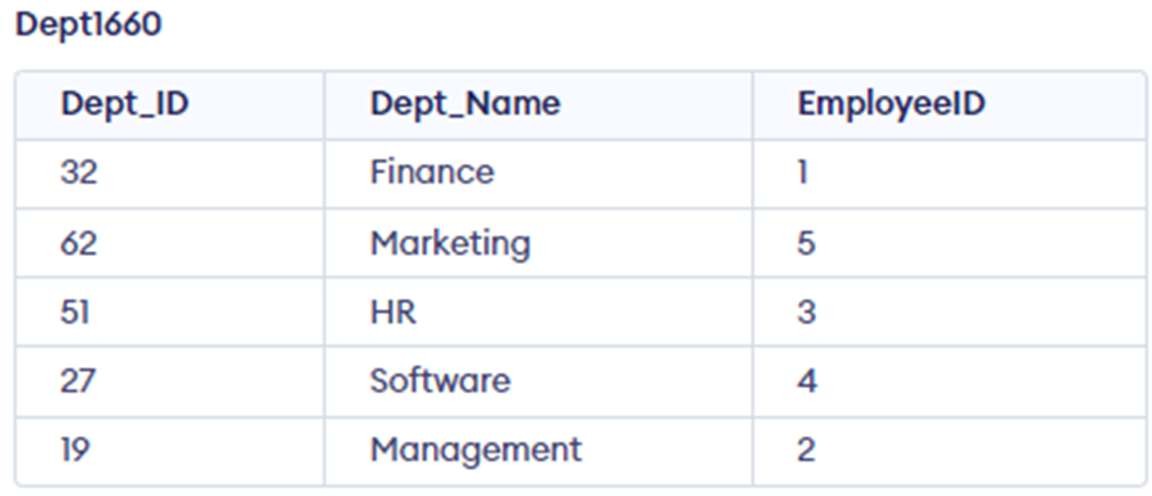


* CREATE TABLE Dept1660(

Dept\_ID int NOT NULL PRIMARY KEY,

Dept\_Name varchar(255) NOT NULL, EID int

);



# Creating a View

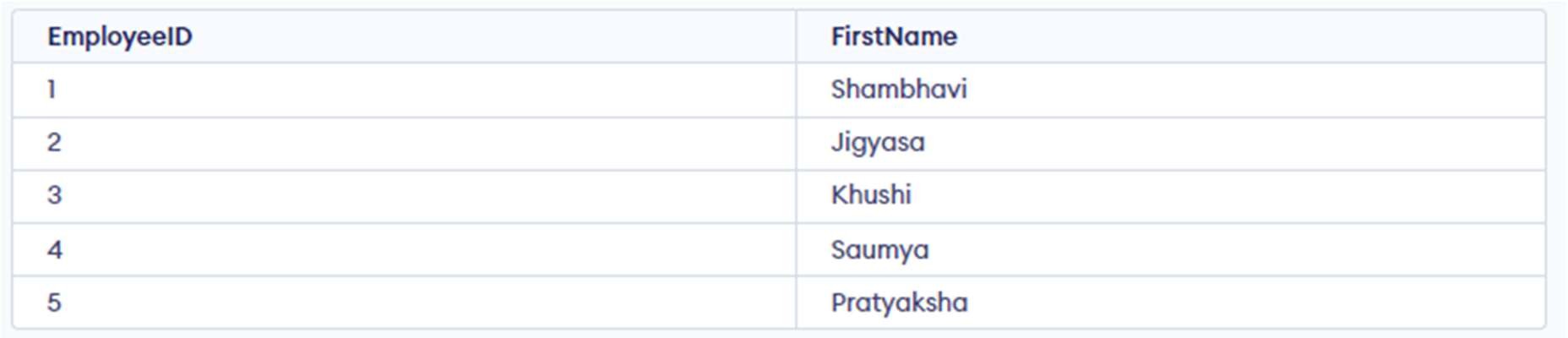
* **From a Single Table**

CREATE VIEW details AS SELECT EmployeeID, FirstName FROM Employee1660

WHERE Salary >= '9100000';

SELECT \*

FROM details;



# From Multiple Tables

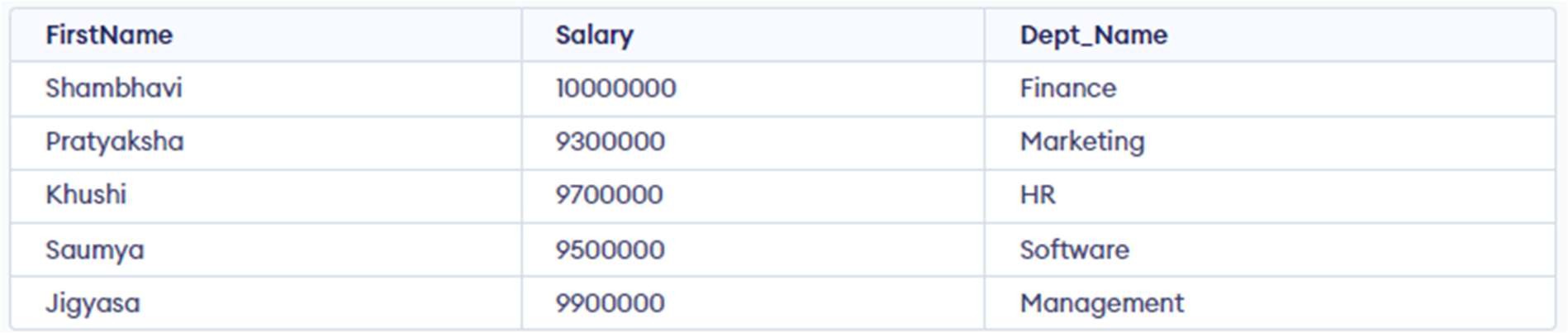
CREATE VIEW details2 AS

SELECT Employee1660.FirstName, Employee1660.Salary, Dept1660.Dept\_Name FROM Employee1660, Dept1660

WHERE Employee1660.EmployeeID = Dept1660.EmployeeID;

SELECT \*

FROM details2;

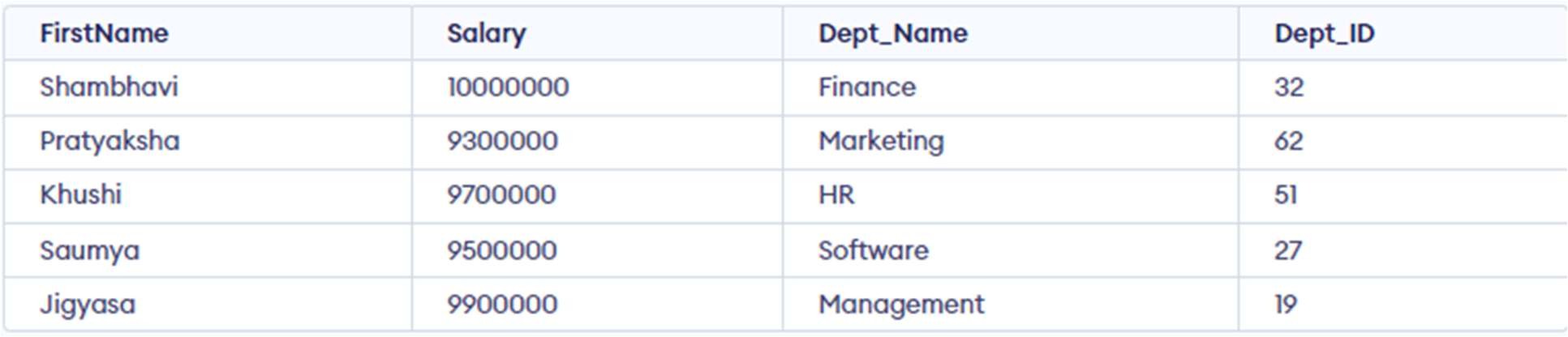


# Updating a View

CREATE OR REPLACE VIEW details AS

SELECT Employee1660.FirstName, Employee1660.Salary, Dept1660.Dept\_Name, Dept1660.Dept\_ID FROM Employee1660, Dept1660

WHERE Employee1660.EmployeeID = Dept1660.EmployeeID;



# Deleting a View

DROP VIEW details;

