

OUTPUT:

```
SQL> CREATE TABLE COURSE12(CCODE VARCHAR2(5), NAME VARCHAR2(2));
```

table created.

```
SQL> INSERT INTO COURSE12 VALUES(12 , 'A');
```

1 row created.

```
SQL> DECLARE
```

```
 2 NEWCCODE VARCHAR2(5) := NULL;  
 3 BEGIN  
 4 UPDATE COURSE12 SET CCODE = NEWCCODE WHERE CCODE = 'C';  
 5 EXCEPTION  
 6 WHEN DUP_VAL_ON_INDEX THEN  
 7 DBMS_OUTPUT.PUT_LINE('DUPLICATE COURSE CODE');  
 8 WHEN OTHERS THEN DBMS_OUTPUT.PUT_LINE(SQLERRM);  
 9 END;  
10 /
```

PL/SQL procedure successfully completed.

```
SQL> CREATE TABLE EMP55(SAL NUMBER(6), NAME VARCHAR2(20), EMPNO NUMBER(20));
```

table created.

```
SQL> INSERT INTO EMP55 VALUES(1000, 'NAVEEN' ,3);
```

1 row created.

```
SQL> DECLARE
```

```
 2 L_SAL EMP55.SAL%TYPE;  
 3 BEGIN  
 4 DBMS_OUTPUT.PUT_LINE('WELCOME');  
 5 SELECT SAL INTO L_SAL FROM EMP55 WHERE EMPNO = &EMPNO;  
 6 DBMS_OUTPUT.PUT_LINE(L_SAL);  
 7 DBMS_OUTPUT.PUT_LINE('THANKYOU');  
 8 EXCEPTION  
 9 WHEN NO_DATA_FOUND THEN  
10 DBMS_OUTPUT.PUT_LINE('INVALID EMPNO');  
11 END;  
12 /
```

Enter value for empno : 4

old 5: SELECT SAL INTO L_SAL FROM EMP55 WHERE EMPNO = &EMPNO;

new 5: SELECT SAL INTO L_SAL FROM EMP55 WHERE EMPNO = 4;

WELCOME
INVALID EMPNO
PL/SQL procedure successfully completed.

SQL> INSERT INTO EMP55 VALUES(2000, 'RISHAB' ,3);

1 row created.

SQL> DECLARE
2 L_SAL EMP55.SAL%TYPE;
3 BEGIN
4 DBMS_OUTPUT.PUT_LINE('WELCOME');
5 SELECT SAL INTO L_SAL FROM EMP55 WHERE EMPNO = 3;
6 DBMS_OUTPUT.PUT_LINE(L_SAL);
7 DBMS_OUTPUT.PUT_LINE('THANKYOU');
8 EXCEPTION
9 WHEN TOO_MANY_ROWS THEN
10 DBMS_OUTPUT.PUT_LINE('MORE THAN ONE ROW');
11 END;
12 /

WELCOME
MORE THAN ONE ROW
PL/SQL procedure successfully completed.

SQL> DECLARE
2 A NUMBER;
3 BEGIN
4 A := 5/0;
5 EXCEPTION
6 WHEN ZERO_DIVIDE THEN
7 DBMS_OUTPUT.PUT_LINE('DO NOT DIVIDE BY 0');
8 END;
9 /

DO NOT DIVIDE BY 0
PL/SQL procedure successfully completed.

SQL> DECLARE
2 OUT_OF_STOCK EXCEPTION;
3 QTY NUMBER;
3 BEGIN
4 QTY := 5;
5 IF QTY<10 THEN

```
6 RAISE OUT_OF_STOCK;
7 END IF;
8 EXCEPTION
9 WHEN OUT_OF_STOCK THEN
10 DBMS_OUTPUT.PUT_LINE('OUT OF STOCK');
11 END;
12 /
```

OUT OF STOCK

PL/SQL procedure successfully completed.

```
SQL> INSERT INTO EMP55 VALUES(3000 , 'ABI' , 5);
```

1 row created.

```
SQL> DECLARE
2 L_SAL EMP55.SAL%TYPE;
3 BEGIN
4 DBMS_OUTPUT.PUT_LINE('WELCOME');
5 SELECT SAL INTO L_SAL FROM EMP55 WHERE EMPNO = 5;
6 IF L_SAL > 2000 THEN
7 RAISE_APPLICATION_ERROR(-20150 , 'SAL IS TOO HIGH');
8 END IF;
9 DBMS_OUTPUT.PUT_LINE('THE SAL IS ...' || L_SAL);
10 END;
11 /
```

WELCOME

DECLARE

*

ERROR AT LINE 1:

ORA-20150 : SAL IS TOO HIGH

ORA-06512 : AT LINE 7