

1. implicit cursor program using % found and %not found attribute.

Begin

```
select eid, ename from emp1;  
if sql% found then  
  dbms_output.put_line('Record found');  
end if;  
end;  
/
```

2. Implicit cursor program using % found and %not found attribute. rowcount%.

declare

n = 800;

begin

```
delete from emp1 where eid = 'e007';  
if sql% found then  
  dbms_output.put_line('Record found');  
  n := sql% Rowcount;  
  dbms_output.put_line('Record deleted');  
elsif sql% not found then  
  dbms_output.put_line('Records not deleted');  
end if;  
end;  
/
```

Date \_\_\_\_ / \_\_\_\_ / \_\_\_\_

3. using implicit cursor update salary of employee by Rs. 500 who are staying in Mumbai print appropriate message about updation of no. of rows.

```

declare
    n number;
Begin
    update emp1 set sal=sal+500 where addr="Mumbai";
    if SQL%found then
        dbms_output.put_line('Record update');
        n:=SQL%rowcount;
    elsif SQL%not found then
        dbms_output.put_line('Record not update');
    endif;
end;
/
    
```

4. using Implicit cursor program using %found and %notfound

```

Declare
    n=&n0;
    delete from emp1 where eid='e001';
    if SQL%found then
        dbms_output.put_line('Record found');
    elsif SQL%not found then
        dbms_output.put_line('Record not found');
    end if;
end;
/
    
```



## 5. explicit cursor demo.

Declare

empid emp1.eid%type;

addr emp1.addr%type;

cursor c1 is select eid, addr, from emp1 order  
by sal;

Begin

open c1;

loop

fetch c1 into empid, addr;

exit when c1%not found;

dbms\_output.put\_line (empid || ' ' || addr);

end loop;

close c1;

end;

/

## 6. explicit cursor with use of its attribute

Declare

cursor s1 is select name from stud;

snm stud.name%type;

Begin

open s1;

IF s1%is open then

:loop

fetch s1 into snm;

: IF s1%not found then

exit;

end if;

dbms\_output.put\_line (snm);

end loop;

```

dbms_output.put_line('Total records: ' || s1 || ' rowscount');
else
dbms_output.put_line('cannot open cursor');
end if;
close s1;
end;
/

```

## 7. updating records with explicit cursor.

Declare

```

m isbn book.isbn % type;
m title book.title % type;
m author book.author % type;
m publisher book.publisher % type;
m price book.price % type;

```

cursor o1 is select title, price from book;

Begin

Open B1;

loop

fetch B1 into m-title, m-price;

update book set price=100 where title='C programming';

exit when B1 % NOT FOUND;

end loop;

close B1;

end;

/



8. Explicit cursor using for loop demo(1)

```
declare
    cursor c1 is select eid, addr from emp1
                order by sal;
begin
    for i in c1
    loop
        dbms_output.put_line ('employee id is' || i.eid ||
                               ' ' || 'Address is' || i.addr);
    end loop;
end;
/
```

9. explicit cursor using for loop demo(2)

```
declare
    cursor s1 is select sno, name from stud
                where course = 'BCA';
begin
    for i in s1
    loop
        dbms_output.put_line (i.sno || ' ' || i.name);
    end loop;
end;
/
```

10. explicit cursor example with user defined value.

declare

m\_isbn book.isbn%type;

m\_title book.title%type;

m\_author book.author%type;

m\_publisher book.publisher%type;

c\_publisher book.publisher%type;

m\_price book.price%type;

cursor c5 is select isbn, title, author, publisher,  
price from book;

begin

c\_publisher := '&publisher';

open c5;

loop

fetch c5 into m\_isbn, m\_title, m\_author, m\_publisher,  
m\_price;

exit when c5%NOTFOUND;

if m\_publisher = c\_publisher then

dbms\_output.put\_line(m\_isbn || m\_title || m\_author  
|| m\_publisher || m\_price);

end if;

end loop;

close c5;

end;

/



11. insertion of record from one table to another table using explicit cursor.

declare

m\_addr emp1.addr%type;

m\_eid emp1.eid%type;

m\_ename emp1.ename%type;

m\_sal emp1.sal%type;

cursor C3 is select eid, ename, addr, sal  
from emp1 order by sal desc;

begin

Open C3;

for lp in 1..5

loop

fetch C3 into m\_eid, m\_ename, m\_addr,  
m\_sal;

insert into top5\_sal values(m\_eid, m\_ename,  
m\_addr, m\_sal);

exit when C3%NOTFOUND;

end loop;

if sql%found then

dbms\_output.put\_line('Record Inserted');

else

dbms\_output.put\_line('not inserted');

end if;

close C3;

end;

/

1. No-data-found exception and Invalid-number exception.

declare

SALARY NUMBER;

begin

SELECT sal INTO SALARY FROM EMP WHERE  
EID = 100;

dbms\_output.put\_line('The salary is: ' ||  
SALARY);

EXCEPTION

WHEN NO\_DATA\_FOUND THEN

dbms\_output.put\_line('No records are found  
for empno 100');

When invalid-number then

dbms\_output.put\_line('invalid value');

END;

/



## 2. No-Data-Found exception.

```
declare
```

```
    id emp1.eid %type;
```

```
    name emp1.ename %type;
```

```
begin
```

```
    SELECT ename INTO name FROM EMP1 WHERE
                                         EID = &id;
```

```
    dbms_output.put_line('The Name is: ' || name);
```

```
EXCEPTION
```

```
    WHEN NO-DATA-FOUND THEN
```

```
        DBMS_OUTPUT.PUT_LINE('No records are found');
```

```
END;
```

```
/
```

## 3. ---to many records

```
declare
```

```
    SALARY emp1.sal %type;
```

```
    dno emp1.deptno %type;
```

```
begin
```

```
    SELECT SAL INTO SALARY FROM EMP1 WHERE DEPTNO
                                         = &dno;
```

```
    DBMS_OUTPUT.PUT_LINE('salary is: ' || SALARY);
```

```
EXCEPTION
```

```
    WHEN TOO-MANY-ROWS THEN
```

```
        DBMS_OUTPUT.PUT_LINE('More than one rows are
                                selected for department');
```

```
    WHEN NO-DATA-FOUND THEN
```

```
        DBMS_OUTPUT.PUT_LINE('No records are found');
```

```
END;
```

```
/
```

4. --- Invalid number exception

declare

NO NUMBER(5);

begin

SELECT TO\_NUMBER('0001A') INTO NO FROM DUAL;

DBMS\_OUTPUT.PUT\_LINE(NO);

EXCEPTION

WHEN INVALID\_NUMBER THEN

DBMS\_OUTPUT.PUT\_LINE('Wrong Data type/Data  
conversion');

END;

/

5. --- Write a program that explains the use of  
ZERO-DIVIDE exception.

declare

N NUMBER(3);

begin

SELECT 100/0 INTO N FROM DUAL;

DBMS\_OUTPUT.PUT\_LINE(N);

EXCEPTION

WHEN ZERO-DIVIDE THEN

DBMS\_OUTPUT.PUT\_LINE('You can not divide the numbers  
by 0');

END;

/