6f.Aim: To study nested queries

6f.Theory:

SUBQUERIES:

0. Nesting of queries one within the other is termed as a subquery.
1. A statement containing subquery is called a parent statement.
2. Subqueries are used to retrieve data from tables that depend on the values in the table itself.
3. Subquery will be evaluated first followed by the main query.
4. Subqueries can also return more than one value by including operators like any, all, in or not in between the comparison operator and the subquery.
5. If the subquery is selected from the same table as the main query, then the main query must define an alias for the table name, and the subquery must use the alias to refer to the column's value in the main query.

Subquery can be used by the following commands:

1. To insert records in a target table.
   Insert into <table1> (select column1, column2 from <table2> where condition);

2. To create tables and insert records in the table created.
   Create table <tablename> (select column1, column2 from <table2> where condition);

3. To update records in a target table.
   Update <table1> set <columnname> = <value> where <columnname> = (select <columnname> from <table2> where <condition>);

4. To provide values for conditions in where, having , in and so on used with select, update, and delete statements.
   Select * from <table1> where <columnname> = (select <columnname> from <table2> where <condition>):

6f.Exercise

NESTED QUERIES : IN/NOT IN

1. **Find the names of customer who has ordered book with Isbn 0-07-123151.**

```
select c.cname
from customer c
where c.cid in (select o.cid
                from order o
                where o.isbn='0-07-123151');
```

2. **Find the names of the customer who have ordered book with title DBS.**

```
select c.cname
from customer c
where c.cid in (select o.cid
                from order o
                where o.isbn in (select b.isbn
                                 from books b
                                 where b.title='DBS'));
```

3. **Find the names of the customer who have not ordered book with title DBS.**

```
select c.cname
from customer c
where c.cid not in (select o.cid
                from order o
                where o.isbn in (select b.isbn
                                 from books b
                                 where b.title='DBS'));
```

CORRELATED NESTED QUERIES: EXISTS/NOT EXISTS

4. **Find the names of customers who have ordered book with Isbn 0-07-123151.**

```
select c.cname
from customer c
where exists ( select *
               from order o, customer c
               where o.isbn='0-07-123151' and o.cid=c.cid);
```

SET COMPARISON OPERATIONS: >ANY, >ALL

5. **Find those books whose Qty_in stock is better than some book with title DBS.**

```
select b1.title
from books b1
where b1.qty_in_stock > any (select b2.qty_in_stock
                             from book b2
                             where b2.title='DBS');
```

6. **Find those books with highest Qty_in_stock.**

```
select b1.title
from books b1
where qty_in_stock >= all (select b2.title
                           from books b2);
```