Part 1.

Use the tables created in the previous lab session for the following questions.

a. Write an Update trigger for the Products table to insert a record to the Items_to_Order table if the Quantity available of a given product falls below the re-order level.

b. Write an Insert trigger to Sales_Order _Detail to update the Product table according to the products and the quantities ordered in a given sale. (Eg. If sales order detail table gets a new record for 25 items of product P0001, the Qty_Available in the product table should be updated accordingly.

Part 2.

Consider the following tables.

Employee

eid	ename	age	salary	did	supervId
e001	Saman	23	70,000.00	d001	null
e002	Kamal	31	34,000.00	d001	e001
e003	Nipun	22	56,000.00	d003	e001
e004	Kasun	23	54,000.00	d002	e003
e005	Heshan	31	60,000.00	d002	e001
e006	Aruni	25	47,000.00	d004	e003
e007	Sachini	21	32,000.00	d002	e004

Department

did	dname	budget	mgrId
d001	HR	250,000.00	e002
d002	Sales	340,000.00	e001
d003	Accounts	560,000.00	e001
d004	IT	590,000.00	e003

```
--SQL Script
create table department
(
   did varchar(5),
   dname varchar(20),
   budget money,
   mgrid varchar(5),
   constraint department_pk primary key(did)
)
create table employee
(
   eid varchar(5),
   ename varchar(20),
   age int,
   salary money,
   did varchar(5),
   supervId varchar(5),
   constraint employee_pk primary key(eid),
   constraint employee_fk1 foreign key (did) REFERENCES department(did),
   constraint employee_fk2 foreign key (supervId) REFERENCES employee(eid)
)
alter table department add constraint department_fk foreign key (mgrid) REFERENCES employee(eid)

insert into department values ('d001','HR',250000,null)
insert into department values ('d002','Sales',340000,null)
insert into department values ('d003','Accounts',560000,null)
insert into department values ('d004','IT',590000,null)
insert into employee values('e001','Saman',23,70000,'d001',null)
insert into employee values('e002','Kamal',31,34000,'d001','e001')
insert into employee values('e003','Nipun',22,56000,'d003','e001')
insert into employee values('e004','Kasun',23,54000,'d002','e003')
insert into employee values('e005','Heshan',31,60000,'d002','e001')
insert into employee values('e006','Aruni',25,47000,'d004','e003')
insert into employee values('e007','Sachini',21,32000,'d002','e004')
update department set mgrid='e002' where did='d001'
update department set mgrid='e001' where did='d002'
update department set mgrid='e001' where did='d003'
update department set mgrid='e003' where did='d004'
```

a. Write an insert/update trigger to check the employee's salary not higher than his/her supervisor's salary.
b. Write a view called "DeptMgr_Details" to retrieve all department details with the manager's details.
   Note: you must retrieve the following columns.
   did, dname, budget, mgrid, mgrname, age, salary, supervId
c. Write an "instead of" trigger to the view "DeptMgr_Details" (created in question no 'b') for insert records to the base tables. Note: only insert new tuples to the base tables, if non-existing records for base tables and an existing value for 'supervId'.
d. Write a read-only view called "Low_budgeted_departments" which retrieve all details of the departments of budget less than 500,000/-