drop database pharmacy;

create database pharmacy;

use pharmacy;

create table Drugs

(

drugID int primary key not null ,

dName varchar(25) not null,

dprice decimal(10.2) not null,

dtype varchar(25),

dtreat varchar (100)

);

create table Employee

(

empID int primary key auto\_increment,

fname varchar(10),

lname varchar(10),

salary int

);

create table Bill

(

blID int primary key not null,

empID int,

bldate varchar(20) ,

constraint emp\_fk foreign key(empID) references Employee( empID)

);

create table brg\_bill\_drug

(

blID int ,

drugID int,

constraint bl\_fk foreign key(blID) references Bill( blID),

constraint drug\_fk foreign key(drugID) references Drugs( drugID)

);

start transaction;

insert into Drugs(drugID,dName,dprice,dtype,dtreat)values

(1,'ASPIRIN',6.60,'Suppositories','Pain relief'),

(2,'BETACOR',18.00,'Liquid','SupraventricularTachycardia'),

(3,'CARDIOTON',13.20,'Injections','Hypertension'),

(4,'DERMASOL',5.50,'Drops','skin conditions'),

(5,'EPANUTIN',14.40,'Capsule','Epilepsy'),

(6,'FLUCAND',60.00,'Liquid','skin conditions'),

(7,'GLOBACIN',30.00,'Capsule','Infections of the urinary tract'),

(8,'HYDIKAL',6.00,'Tablets','Hypokalemia in congestive heart failure'),

(9,'INDOMIN',1.90,'Injections','Antipyretic'),

(10,'JULMENTIN',57.50,'Drops','Urethritis'),

(11,'KINSORAM',31.50,'Capsule','Total anesthesia'),

(12,'LIPITOR',160.00,'Suppositories','lower blood levels of cholesterol'),

(13,'MALARAN',40.00,'Liquid','MALARIA'),

(14,'NEURONTIN',64.00,'Capsules','cut heroin'),

(15,'OPRAZOLE',26.25,'Drops','Gastritis'),

(16,'POWERCAPS',19.50,'Injections','To treat influenza'),

(17,'QUADEL',38.00,'Capsule','To treat mood and mental disorders'),

(18,'Rifampicin',9.00,'Suppositories','Antibacterial'),

(19,'SUPRAX',43.25,'Liquid','Resistance to microbial infection'),

(20,'TEKLIVA',48.00,'Drops','For reduction of intraocular pressure');

commit;

start transaction;

insert into Employee(fname,lname,salary)values ('Omar','Fathi',5000),('Omar','Gamal',4000),('Hossam','Hassan',3000),('Amr','Ragab',2500);

commit;

start transaction;

insert into Bill (blID,empID,bldate)values(1,1,'2020-12-30'),(2,3,'2020-12-31'),(3,4,'2020-12-30'),(4,1,'2020-12-20');

commit;

start transaction;

insert into brg\_bill\_drug (blID,drugID)values(1 ,1 ),(1,2),(1,3),(2,4),(2,2),(3,10),(3,15),(4,20);

commit;

start transaction;

delete from Employee where empID=2;

rollback;

/\* PROCEDURE \*/

DELIMITER $$

CREATE PROCEDURE total\_amount\_of\_bill(IN billID INT )

BEGIN

select sum(Drugs.dprice)

from Drugs, brg\_bill\_drug

where brg\_bill\_drug.drugID=Drugs.drugID and brg\_bill\_drug.blID=billID ;

END $$

DELIMITER ;

call total\_amount\_of\_bill(1);

DELIMITER $$

CREATE PROCEDURE drags\_info(IN dID INT )

BEGIN

select dName,dtreat ,dprice

from Drugs

where drugID=dID ;

END $$

DELIMITER ;

call drags\_info(1);

DELIMITER $$

CREATE PROCEDURE emp\_of\_bill(IN bID INT )

BEGIN

select Employee.fname,Employee.lname

from Employee ,Bill

where Bill.blID=bID and Employee.empID=Bill.empID;

END $$

DELIMITER ;

call emp\_of\_bill(1);

DELIMITER $$

CREATE PROCEDURE drugs\_Treat(IN d int )

BEGIN

select dName,dtreat

from Drugs

where drugID = d;

END $$

DELIMITER ;

call drugs\_Treat(1);

/\* PROCEDURE drugs\_Price \*/

DELIMITER $$

CREATE PROCEDURE drugs\_Price(IN d int )

BEGIN

select Drugs.dprice

from Drugs

where drugID = d;

END $$

DELIMITER ;

call drugs\_Price(1);

/\* functions \*/

DELIMITER $$

CREATE FUNCTION total\_benfits()

RETURNS int

BEGIN

DECLARE total int;

select sum(Drugs.dprice)into total

from Drugs,brg\_bill\_drug

where Drugs.drugID=brg\_bill\_drug.drugID;

RETURN (total);

END$$

DELIMITER ;

DELIMITER $$

CREATE FUNCTION total\_Number\_OfDrugs()

RETURNS int

BEGIN

DECLARE cnt int;

select COUNT(Drugs.drugID)into cnt

from Drugs;

return(cnt);

END$$

DELIMITER ;

select total\_Number\_OfDrugs() ;

DELIMITER $$

CREATE FUNCTION salaryofemp(ID int)

RETURNS int

BEGIN

DECLARE sal int ;

SELECT salary

INTO sal

FROM Employee

WHERE ID = empID;

return sal;

END$$

DELIMITER ;

DELIMITER $$

CREATE FUNCTION typeOfDrug(ID int)

RETURNS varchar(25)

BEGIN

DECLARE ans varchar(25);

SELECT dtype

INTO ans

FROM Drugs

WHERE ID = drugid;

return ans;

END$$

DELIMITER ;

select typeOfDrug(1);

DELIMITER $$

CREATE FUNCTION EmployeeLevel(

sal DECIMAL(10,2)

)

RETURNS VARCHAR(20)

BEGIN

DECLARE empLevel VARCHAR(20);

IF sal > 50000 THEN

SET empLevel = 'PLATINUM';

ELSEIF (sal >= 50000 AND sal <= 10000) THEN

SET empLevel = 'GOLD';

ELSEIF sal < 10000 THEN

SET empLevel = 'SILVER';

END IF;

RETURN (empLevel );

END$$

DELIMITER ;

SELECT

fname,lname ,

EmployeeLevel(salary)

FROM

Employee;

CREATE TRIGGER update\_emp\_name

BEFORE INSERT ON Employee

FOR EACH ROW

SET NEW.fname = UPPER(NEW.fname);

CREATE TABLE SalaryBudgets(

total int NOT NULL

);

insert into SalaryBudgets (total)values(0);

CREATE TRIGGER after\_salaries\_delete

AFTER DELETE

ON Employee FOR EACH ROW

UPDATE SalaryBudgets

SET total = total - OLD.salary;

CREATE TRIGGER after\_salaries\_insert

AFTER insert

ON Employee FOR EACH ROW

UPDATE SalaryBudgets

SET total = total + new.salary;

CREATE TRIGGER update\_emp\_last\_name

BEFORE INSERT ON Employee

FOR EACH ROW

SET NEW.lname = lower(NEW.lname);

CREATE TRIGGER update\_Drug\_name

BEFORE INSERT ON Drugs

FOR EACH ROW

SET NEW.dName = UPPER(NEW.dName);