```
drop table donateBlood;

drop table patient;

drop table blood;

drop table donor;

drop table bloodBank;

---Set line size and page size

set pagesize 300
set linesize 300

--Table Creation

CREATE TABLE bloodBank (
  bb_id NUMBER(20),
  bb_name VARCHAR2(30),
  bb_city VARCHAR2(30),
  bb_contactNo NUMBER(11),
  bb_area VARCHAR2(30),
  PRIMARY KEY (bb_id)
);


CREATE TABLE donor (
  d_id NUMBER(20),
  d_name VARCHAR2(30),
  d_gender VARCHAR2(6),
  d_age NUMBER(3),
  d_bloodGrp VARCHAR2(10),
  d_city VARCHAR2(30),
  last_date_of_donation DATE,
  d_contactNo NUMBER(11),
  bb_id NUMBER(20),
  PRIMARY KEY (d_id),
```

```
  FOREIGN KEY (bb_id) REFERENCES bloodBank (bb_id)
);

CREATE TABLE blood (
  b_id NUMBER(20),
  bb_id NUMBER(20),
  d_id NUMBER(20),
  b_bloodGrp VARCHAR2(10),
  date_donated DATE,
  PRIMARY KEY (b_id),
  FOREIGN KEY (bb_id) REFERENCES bloodBank (bb_id),
  FOREIGN KEY (d_id) REFERENCES donor (d_id)
);

 CREATE TABLE patient (
  p_id NUMBER(20),
  p_name VARCHAR2(30),
  p_gender VARCHAR2(6),
  p_age NUMBER(3),
  p_bloodGrp VARCHAR2(10),
  p_city VARCHAR2(30),
  p_contactNo NUMBER(11),
  PRIMARY KEY (p_id)
);

 CREATE TABLE donateBlood (
  b_id NUMBER(20),
  d_id NUMBER(20),
  p_id NUMBER(20),
```

```
  bb_id NUMBER(20),
  dateOfDonation DATE,
  PRIMARY KEY (b_id, d_id, p_id, bb_id),
  FOREIGN KEY (b_id) REFERENCES blood (b_id),
  FOREIGN KEY (d_id) REFERENCES donor (d_id),
  FOREIGN KEY (p_id) REFERENCES patient (p_id),
  FOREIGN KEY (bb_id) REFERENCES bloodBank (bb_id)
);

--DDL
ALTER TABLE bloodBank ADD bb_area VARCHAR2(30);
ALTER TABLE bloodBank MODIFY bb_area VARCHAR2(30);
ALTER TABLE bloodBank RENAME COLUMN bb_area TO area;

ALTER TABLE blood DROP COLUMN d_id;
ALTER TABLE blood DROP COLUMN bb_id;

--DML
INSERT INTO bloodBank (bb_id, bb_name, bb_city, bb_contactNo, area)VALUES (1001, 'ABC Blood Bank', 'New York', 1234567890, 'Central');
INSERT INTO bloodBank (bb_id, bb_name, bb_city, bb_contactNo, area)VALUES (1002, 'XYZ Blood Bank', 'Los Angeles', 9876543210, 'West');
INSERT INTO bloodBank (bb_id, bb_name, bb_city, bb_contactNo, area)VALUES (1003, 'PQR Blood Bank', 'Chicago', 5556667777, 'North');
INSERT INTO bloodBank (bb_id, bb_name, bb_city, bb_contactNo, area)VALUES (1004, 'MNO Blood Bank', 'San Francisco', 8889990000, 'West');
INSERT INTO bloodBank (bb_id, bb_name, bb_city, bb_contactNo, area)VALUES (1005, 'DEF Blood Bank', 'New York', 1112223333, 'Central');
```

```
INSERT INTO donor (d_id, d_name, d_gender, d_age, d_bloodGrp, d_city, last_date_of_donation, d_contactNo, bb_id)VALUES (1, 'John Smith', 'Male', 25, 'O+', 'New York', TO_DATE('2023-05-15', 'YYYY-MM-DD'), 1234567890, 1001);

INSERT INTO donor (d_id, d_name, d_gender, d_age, d_bloodGrp, d_city, last_date_of_donation, d_contactNo, bb_id)VALUES (2, 'Emily Johnson', 'Female', 30, 'A-', 'Los Angeles', TO_DATE('2023-05-20', 'YYYY-MM-DD'), 9876543210, 1002);

INSERT INTO donor (d_id, d_name, d_gender, d_age, d_bloodGrp, d_city, last_date_of_donation, d_contactNo, bb_id)VALUES (3, 'Emma Johnson', 'Female', 29, 'B+', 'Los Angeles', TO_DATE('2023-05-25', 'YYYY-MM-DD'), 8888888888, 1001);

INSERT INTO donor (d_id, d_name, d_gender, d_age, d_bloodGrp, d_city, last_date_of_donation, d_contactNo, bb_id)VALUES (4, 'William Wilson', 'Male', 32, 'AB+', 'Chicago', TO_DATE('2023-05-28', 'YYYY-MM-DD'), 7777777777, 1002);

INSERT INTO donor (d_id, d_name, d_gender, d_age, d_bloodGrp, d_city, last_date_of_donation, d_contactNo, bb_id)VALUES (5, 'Olivia Davis', 'Female', 27, 'O-', 'San Francisco', TO_DATE('2023-05-30', 'YYYY-MM-DD'), 9999999999, 1003);


INSERT INTO blood (b_id, b_bloodGrp, date_donated)VALUES (101, 'O+', TO_DATE('2023-05-15', 'YYYY-MM-DD'));

INSERT INTO blood (b_id, b_bloodGrp, date_donated)VALUES (102, 'A-', TO_DATE('2023-05-20', 'YYYY-MM-DD'));

INSERT INTO blood (b_id, b_bloodGrp, date_donated)VALUES (103, 'B+', TO_DATE('2023-05-25', 'YYYY-MM-DD'));

INSERT INTO blood (b_id, b_bloodGrp, date_donated)VALUES (104, 'AB+', TO_DATE('2023-05-28', 'YYYY-MM-DD'));

INSERT INTO blood (b_id, b_bloodGrp, date_donated)VALUES (105, 'O-', TO_DATE('2023-05-30', 'YYYY-MM-DD'));


INSERT INTO patient (p_id, p_name, p_gender, p_age, p_bloodGrp, p_city, p_contactNo)VALUES (10001, 'Sarah Davis', 'Female', 35, 'B+', 'San Francisco', 5551112222);

INSERT INTO patient (p_id, p_name, p_gender, p_age, p_bloodGrp, p_city, p_contactNo)VALUES (10002, 'Michael Brown', 'Male', 40, 'AB-', 'Chicago', 9998887777);

INSERT INTO patient (p_id, p_name, p_gender, p_age, p_bloodGrp, p_city, p_contactNo)VALUES (10003, 'David Smith', 'Male', 50, 'O-', 'New York', 7777777777);

INSERT INTO patient (p_id, p_name, p_gender, p_age, p_bloodGrp, p_city, p_contactNo)VALUES (10004, 'Sophia Taylor', 'Female', 22, 'A+', 'Los Angeles', 8888888888);
```

```
INSERT INTO patient (p_id, p_name, p_gender, p_age, p_bloodGrp, p_city, p_contactNo)VALUES (10005, 'James Anderson', 'Male', 45, 'B-', 'Chicago', 9999999999);


INSERT INTO donateBlood (b_id, d_id, p_id, bb_id, dateOfDonation)VALUES (101, 1, 10001, 1001, TO_DATE('2023-05-15', 'YYYY-MM-DD'));

INSERT INTO donateBlood (b_id, d_id, p_id, bb_id, dateOfDonation)VALUES (102, 2, 10002, 1002, TO_DATE('2023-05-20', 'YYYY-MM-DD'));

INSERT INTO donateBlood (b_id, d_id, p_id, bb_id, dateOfDonation)VALUES (103, 3, 10003, 1003, TO_DATE('2023-05-25', 'YYYY-MM-DD'));

INSERT INTO donateBlood (b_id, d_id, p_id, bb_id, dateOfDonation)VALUES (104, 4, 10004, 1004, TO_DATE('2023-05-28', 'YYYY-MM-DD'));

INSERT INTO donateBlood (b_id, d_id, p_id, bb_id, dateOfDonation)VALUES (105, 5, 10005, 1005, TO_DATE('2023-05-30', 'YYYY-MM-DD'));


--select command

select * from bloodBank;

select * from donor;

select * from blood;

select * from patient;

select * from donateBlood;


--update, delete

update patient set p_name='Sara Devis' where p_id=10001;

INSERT INTO donor (d_id, d_name, d_gender, d_age, d_bloodGrp, d_city, last_date_of_donation, d_contactNo, bb_id)VALUES (6, 'Sara Parker', 'Female', 37, 'O+', 'San Atlanta', TO_DATE('2023-06-30', 'YYYY-MM-DD'), 1999999999, 1003);

delete from  donor where d_id=6;


--string operation

select d_gender from donor where d_gender like 'F%';


--aggregate
```

```
select count(*) from donor;

select count(p_name) as number_of_patients from patient;

select count(distinct b_bloodGrp) as number_of_blood_groups from blood;


--group by

select d_gender,avg(d_age) from donor group by d_gender having avg(d_age)>=10;


--nested

select d_name from donor where d_id=(select d_id from donateBlood where p_id=(select p_id from
donateBlood where bb_id=(select bb_id from bloodBank where bb_name='ABC Blood Bank')));


--membership

SELECT * FROM donor WHERE d_age >= 20 AND d_id IN (SELECT d_id FROM donateBlood WHERE
d_bloodGrp LIKE '%A%');


--PL/SQL


--variable declaration and print

set serveroutput on

declare

bb_id bloodBank.bb_id%type;

bb_name bloodBank.bb_NAME%type;

begin

select bb_id, bb_name into bb_id, bb_name from bloodBank where bb_id=1002;

dbms_output.put_line('BloodBank_id: '||bb_id|| ' BloodBank_name: '||bb_name);

end;

/

--insert and set default value

SET SERVEROUTPUT ON;

DECLARE
```

```
  bb_id bloodBank.bb_id%TYPE := 1006;
  bb_name bloodBank.bb_name%TYPE := 'GHI Blood Bank';
  bb_city bloodBank.bb_city%TYPE := 'Seattle';
  bb_contactNo bloodBank.bb_contactNo%TYPE := 5555555555;
  area bloodBank.area%TYPE := 'Northwest';
BEGIN
  INSERT INTO bloodBank (bb_id, bb_name, bb_city, bb_contactNo, area)
  VALUES (bb_id, bb_name, bb_city, bb_contactNo, area);
  DBMS_OUTPUT.PUT_LINE('Row inserted successfully.');
  commit;
END;
/
--rowtype
SET SERVEROUTPUT ON;
DECLARE
  bloodBank_row bloodBank%ROWTYPE;
BEGIN
  SELECT bb_id, bb_name, bb_city, bb_contactNo, area
  INTO bloodBank_row.bb_id, bloodBank_row.bb_name, bloodBank_row.bb_city,
bloodBank_row.bb_contactNo, bloodBank_row.area
  FROM bloodBank
  WHERE bb_id = 1005;

  DBMS_OUTPUT.PUT_LINE('bb_id: ' || bloodBank_row.bb_id);
  DBMS_OUTPUT.PUT_LINE('bb_name: ' || bloodBank_row.bb_name);
  DBMS_OUTPUT.PUT_LINE('bb_city: ' || bloodBank_row.bb_city);
  DBMS_OUTPUT.PUT_LINE('bb_contactNo: ' || bloodBank_row.bb_contactNo);
  DBMS_OUTPUT.PUT_LINE('area: ' || bloodBank_row.area);
end;
```

```
/
--cursor
SET SERVEROUTPUT ON;
DECLARE
  CURSOR bloodBank_cursor IS
    SELECT bb_id, bb_name, bb_city, bb_contactNo, area
    FROM bloodBank;
  bloodBank_row bloodBank_cursor%ROWTYPE;
  total_rows INTEGER := 0;
BEGIN
  OPEN bloodBank_cursor;
  LOOP
    FETCH bloodBank_cursor INTO bloodBank_row;
    EXIT WHEN bloodBank_cursor%NOTFOUND;

    DBMS_OUTPUT.PUT_LINE('bb_id: ' || bloodBank_row.bb_id || ', bb_name: ' || bloodBank_row.bb_name || ', bb_city: ' || bloodBank_row.bb_city || ', bb_contactNo: ' || bloodBank_row.bb_contactNo || ', area: ' || bloodBank_row.area);
    total_rows := total_rows + 1;
    DBMS_OUTPUT.PUT_LINE('');
  END LOOP;

  CLOSE bloodBank_cursor;
  DBMS_OUTPUT.PUT_LINE('Total rows: ' || total_rows);
END;
/
--if-else
SET SERVEROUTPUT ON
DECLARE
   counter NUMBER := 1;
```

```
  donor_blood_group donor.d_bloodGrp%TYPE;
BEGIN
  counter := 1;
  FOR x IN 1..5
  LOOP
    SELECT d_bloodGrp INTO donor_blood_group FROM donor WHERE d_id = x;
    IF donor_blood_group IN ('A+', 'B+', 'AB+', 'O+') THEN
      DBMS_OUTPUT.PUT_LINE(donor_blood_group || ' is positive');
    ELSIF donor_blood_group IN ('A-', 'B-', 'AB-', 'O-') THEN
      DBMS_OUTPUT.PUT_LINE(donor_blood_group || ' is negative');
    END IF;
  END LOOP;
END;
/
--procedure
CREATE OR REPLACE PROCEDURE GetBloodBankName(
  bb_id IN bloodBank.bb_id%TYPE
)
AS
  BloodBank_name bloodBank.bb_name%TYPE;
BEGIN
  SELECT bb_name INTO BloodBank_name FROM bloodBank WHERE bb_id = GetBloodBankName.bb_id;
  DBMS_OUTPUT.PUT_LINE('Blood Bank Name: ' || BloodBank_name);
END;
/
--callProcedure
SET SERVEROUTPUT ON;
DECLARE
  BloodBank_id bloodBank.bb_id%TYPE := 1001;
```

```
BEGIN
  GetBloodBankName(BloodBank_id);
END;
/
--function
CREATE OR REPLACE FUNCTION GetTotalDonorsByBloodType(
  blood_type IN donor.d_bloodGrp%TYPE
) RETURN NUMBER
AS
  total_donors NUMBER;
BEGIN
  SELECT COUNT(*) INTO total_donors FROM donor WHERE d_bloodGrp = blood_type;
  RETURN total_donors;
END;
/
--callFunction
DECLARE
  blood_type donor.d_bloodGrp%TYPE := 'B+';
  donors_count NUMBER;
BEGIN
  donors_count := GetTotalDonorsByBloodType(blood_type);
  DBMS_OUTPUT.PUT_LINE('Total Donors with Blood Type ' || blood_type || ': ' || donors_count);
END;
/
--drop
drop procedure GetBloodBankName;
drop function GetTotalDonorsByBloodType;
```