QUESTION 1:

1.

```
        CREATE TABLE CUSTOMER(
        CUST_NUM INT,
        CUST_LNAME VARCHAR(255),
        CUST_FNAME VARCHAR(255),
        CUST_BALANCE FLOAT);

        CREATE TABLE CUSTOMER_2(
        CUST_NUM INT,
        CUST_LNAME VARCHAR(255),
        CUST_FNAME VARCHAR(255));

        CREATE TABLE INVOICE(
        INV_NUM INT,
        CUST_NUM INT,
        INV_DATE DATE,
        INV_AMOUNT FLOAT);
```

2.

```
INSERT INTO CUSTOMER(CUST_NUM, CUST_LNAME, CUST_FNAME, CUST_BALANCE)
VALUES
(1000, 'Smith', 'Jeanne', 1050.11),
(1001, 'Ortega', 'Juan', 840.92);

INSERT INTO CUSTOMER_2(CUST_NUM, CUST_LNAME, CUST_FNAME)
VALUES
(2000, 'McPherson', 'Anne'),
(2001, 'Ortega', 'Juan'),
(2002, 'Kowalski', 'Jan'),
(2003, 'Chen', 'George');

INSERT INTO INVOICE(INV_NUM, CUST_NUM, INV_DATE, INV_AMOUNT)
VALUES
(8000, 1000, '2016-03-23', 235.89),
(8001, 1001, '2016-03-23', 312.82),
(8002, 1001, '2016-03-30', 528.10),
(8003, 1000, '2016-04-12', 194.78),
(8004, 1000, '2016-04-16', 619.44);
```

3.

```
SELECT CUSTOMER.CUST_LNAME, CUSTOMER.CUST_FNAME FROM CUSTOMER
UNION ALL
SELECT CUSTOMER_2.CUST_LNAME, CUSTOMER_2.CUST_FNAME FROM
CUSTOMER_2;
```

4.

```
SELECT CUSTOMER.CUST_LNAME, CUSTOMER.CUST_FNAME FROM CUSTOMER
UNION
SELECT CUSTOMER_2.CUST_LNAME, CUSTOMER_2.CUST_FNAME FROM
CUSTOMER_2;
```

5.

```
SELECT CUST_LNAME, CUST_FNAME
FROM(
SELECT CUST_LNAME, CUST_FNAME FROM CUSTOMER
UNION ALL
SELECT CUST_LNAME, CUST_FNAME FROM CUSTOMER_2) AS theTable
GROUP BY CUST_LNAME, CUST_FNAME HAVING COUNT(*) > 1;
```

6.

```
SELECT CUST_LNAME, CUST_FNAME FROM CUSTOMER_2
WHERE(CUST_LNAME, CUST_FNAME)
NOT IN (SELECT CUST_LNAME, CUST_FNAME FROM CUSTOMER);
```

7.

```
SELECT INVOICE.INV_NUM, INVOICE.CUST_NUM, CUSTOMER.CUST_FNAME,
CUSTOMER.CUST_LNAME, INVOICE.INV_DATE, INVOICE.INV_AMOUNT
FROM INVOICE
INNER JOIN CUSTOMER ON INVOICE.CUST_NUM = CUSTOMER.CUST_NUM
WHERE CUSTOMER.CUST_BALANCE >= 1000;
```

8.

```
SELECT INV_NUM, INV_AMOUNT, (SELECT AVG(INV_AMOUNT) FROM INVOICE),
(SELECT AVG(INV_AMOUNT) FROM INVOICE) - INV_AMOUNT FROM INVOICE;
```

9.

We do not have Oracle but rather mySQL.
Can only auto increment one column

```
CREATE TABLE new_INVOICE(
INV_NUM INT auto_increment primary key,
CUST_NUM INT ,
INV_DATE DATE,
INV_AMOUNT FLOAT,
foreign key(CUST_NUM) references CUSTOMER(CUST_NUM))
auto_increment = 5000;

INSERT INTO new_INVOICE( CUST_NUM, INV_DATE, INV_AMOUNT)
VALUES
(1000, '2016-03-23', 235.89),
(1001, '2016-03-23', 312.82),
(1001, '2016-03-30', 528.10),
(1000, '2016-04-12', 194.78),
(1000, '2016-04-16', 619.44);

SELECT * FROM new_INVOICE;
```

10.

```
ALTER TABLE CUSTOMER ADD CUST_DOB DATE;
ALTER TABLE CUSTOMER ADD CUST_AGE DOUBLE;

UPDATE CUSTOMER set CUST_DOB ='1979-03-15' WHERE CUST_NUM = 1000;
UPDATE CUSTOMER set CUST_DOB ='1988-12-22' WHERE CUST_NUM = 1001;

UPDATE CUSTOMER set CUST_AGE = TIMESTAMPDIFF(YEAR,'1979-03-15',CURDATE())
WHERE CUST_NUM = 1000;
UPDATE CUSTOMER set CUST_AGE = TIMESTAMPDIFF(YEAR,'1988-12-22',CURDATE())
WHERE CUST_NUM = 1001;
```

QUESTION 2

1.

```
SELECT EMPLOYEE.Fname, EMPLOYEE.Lname
FROM PROJECT
INNER JOIN WORKS_ON ON PROJECT.Pnumber = WORKS_ON.Pno
INNER JOIN EMPLOYEE ON WORKS_ON.Essn = EMPLOYEE.Ssn
WHERE EMPLOYEE.Dno = 5 AND WORKS_ON.Hours > 10 AND PROJECT.Pname =
'ProductX';
```

2.

```
SELECT EMPLOYEE.Fname, EMPLOYEE.Lname
FROM EMPLOYEE
INNER JOIN DEPENDENT ON EMPLOYEE.Ssn = DEPENDENT.Essn
WHERE EMPLOYEE.Fname = DEPENDENT.Dependent_name;
```

3.

```
SELECT Fname, Lname FROM EMPLOYEE WHERE Super_ssn = (SELECT Ssn FROM
EMPLOYEE WHERE Fname = 'Franklin' and Lname = 'Wong');
```

4.

```
SELECT Fname, Lname FROM EMPLOYEE WHERE Lname LIKE 'W%';
```

5.

```
SELECT Fname, Lname FROM EMPLOYEE WHERE Salary > 40000;
```

6.

```
SELECT EMPLOYEE.Fname, EMPLOYEE.Lname
FROM EMPLOYEE
INNER JOIN DEPARTMENT ON EMPLOYEE.Dno = DEPARTMENT.Dnumber
WHERE Dname = 'Research';
```

7. Have emailed Harshita about this question. She told me to explain the reason why this question is not possible. The reason why is because in the tables the hire dates of all employees are not available. I only see Manager hire dates. Hence, this problem is not solvable.

8.

```
SELECT DEPARTMENT.Dname, COUNT(*)
FROM DEPARTMENT
INNER JOIN EMPLOYEE ON DEPARTMENT.Dnumber = EMPLOYEE.Dno
GROUP BY DEPARTMENT.Dnumber HAVING COUNT(*) < 4;
```

9.

```
SELECT Fname, Lname FROM EMPLOYEE WHERE Super_ssn is NULL;
```

10.

```
SELECT DEPARTMENT.Dnumber, COUNT(DEPARTMENT.Dnumber)
FROM DEPARTMENT
INNER JOIN EMPLOYEE ON DEPARTMENT.Dnumber = EMPLOYEE.Dno
GROUP BY DEPARTMENT.Dnumber;
```

QUESTION 3

1.

```
SELECT CUST_FNAME, CUST_LNAME FROM Customer WHERE CUST_STATE = 'NY';
```

2.

```
SELECT COUNT(CUST_FNAME), CUST_STATE FROM Customer GROUP BY CUST_STATE
ORDER BY COUNT(CUST_FNAME) DESC;
```

3.

```
SELECT Brand.BRAND_NAME, Brand_name_num.NUMPRODUCTS_BRAND AS
Number_of_Products
FROM Brand
INNER JOIN Brand_name_num ON Brand.BRAND_NAME =
Brand_name_num.BRAND_NAME
ORDER BY Brand.BRAND_NAME;
```

4.

```
SELECT EMP_NUM, EMP_LNAME, EMP_EMAIL, EMP_TITLE, DEPT_NAME FROM
Employee
WHERE EMP_TITLE LIKE '%ASSOCIATE' ORDER BY DEPT_NAME, EMP_TITLE;
```

5.

```
SELECT NUMPRODUCTS_Prod_detailed_num, PROD_TYPE, PROD_BASE FROM
Prod_detailed_num WHERE PROD_BASE = 'Water';
```

6.

```
SELECT PROD_BASE,PROD_TYPE, SUM(NUMPRODUCTS_Prod_detailed_num) FROM
Prod_detailed_num
GROUP BY PROD_TYPE, PROD_BASE;
```

7.

```
SELECT BRAND_ID, SUM(TOTALINVENTORY) FROM Brand_id_inv GROUP BY BRAND_ID
ORDER BY BRAND_ID DESC;
```

8.

```
SELECT CUST_FNAME, CUST_LNAME FROM Customer WHERE CUST_STATE = 'FL';
```

9.

```
SELECT BRAND_ID, BRAND_NAME, AVGPRICE FROM Brand ORDER BY BRAND_NAME;
```

10.

```
SELECT DEPT_NUM , MOSTRECENT FROM Dept_recent ORDER BY DEPT_NUM;
```