# HW 2 Nawras Rawas Qalaji

1.)

```
CREATE TABLE Test_Info
        ( FAA_no        INTEGER,
        reg_no          INTEGER,
        SSN             INTEGER
        date            DATE,
        score           REAL,
        hours           INTEGER,
        PRIMARY KEY     (FAA_no, reg_no, SSN)
        FOREIGN KEY     (FAA_no) REFERENCES Test,
        FOREIGN KEY     (SSN) REFERENCES Technician,
        FOREIGN KEY     (reg_no) REFERENCES Plane
        )
CREATE TABLE Type
        (reg_no         INTEGER NOT NULL,
        model_no        INTEGER,
        PRIMARY KEY     (reg_no),
        FOREIGN KEY     (model_no) REFERENCES Model,
        FOREIGN KEY     (reg_no) REFERENCES Plane
        )
CREATE TABLE Expert
        (SSN            INTEGER,
        model_no        INTEGER,
        PRIMARY KEY     (model_no, SSN),
        FOREIGN KEY     (model_no) REFERENCES Model,
        FOREIGN KEY     (SSN) REFERENCES Technician
        )
CREATE TABLE Employees
        (union_mem_no INTEGER,
        SSN             INTEGER,
        PRIMARY KEY     (SSN)
        )
```

```
CREATE TABLE Technician
        (SSN            INTEGER,
        salary          INTEGER,
        name            CHAR(30),
        phone_num       INTEGER,
        address         CHAR(30),
        PRIMARY KEY (SSN),
        FOREIGN KEY (SSN) REFERENCES Employees
        )
CREATE TABLE Traffic_control
        (SSN            INTEGER,
        exam_date       DATE,
        PRIMARY KEY (SSN),
        FOREIGN KEY (SSN) REFERENCES Employees
        )
CREATE TABLE Model
        (model_no       INTEGER,
        capacity        REAL,
        weight          REAL,
        PRIMARY KEY (model_no)
        )
CREATE TABLE Plane
        (reg_no         INTEGER NOT NULL,
        PRIMARY KEY (reg_no),
        )
CREATE TABLE Test
        (FAA_no         INTEGER,
        name            CHAR(30),
        score           INTEGER,
        PRIMARY KEY (FAA_no)
        )
```

Everything can be captured in the translation

2.a) Prints all the part names that are in catalog C

2.b)

```
SELECT S.sname
FROM Suppliers S
WHERE NOT EXISTS ((SELECT P.pid
                   FROM Parts P
                   WHERE NOT EXISTS (SELECT C.cid
                                     FROM Catalog C
                                     WHERE C.cid = P.pid AND C.cid = S.sid))
```

2.c) Prints the names of suppliers who supply all red parts

2.d)

```
SELECT P.pname
FROM Suppliers S Parts P, Catalog C
WHERE S.sname = “Acme Widget Suppliers” AND C.pid = P.pid AND C.sid = S.sid
```

2.e) Every distinct supplier id who supplies a part that is higher than the average cost of that part

2.f) For each unique part it will find the name of the supplier who is selling it for the highest price and what part id it is

2.g)

```
SELECT DISTINCT C.sid
FROM Catalog C
WHERE NOT EXISTS (  SELECT *
                    FROM Parts P
                    WHERE P.pid = C.pid AND P.color <> “red”)
```

2.h) Returns supplier id who supply both Red parts and Green parts

2.i)

```
SELECT DISTINCT C.sid
FROM Catalog C, Parts P
WHERE C.pid = P.pid AND P.color = 'Red'
UNION
SELECT DISTINCT C1.sid
FROM Catalog C1, Parts P1
WHERE C1.pid = P1.pid AND P1.color = 'Green'
```

2.j)

Prints the supplier name and the number of green parts she sells for everyone that only supplies green parts

2.k)

```
SELECT P.pname, MAX(C.cost) as maxCost
FROM Parts P, Catalog C, Supplier S
GROUP BY S.sname, S.sid
HAVING ANY (P.color = "green") AND ANY (P.color = "red")
```

3.a)

```
CREATE TABLE Emp
        (eid             INTEGER,
        ename            CHAR(15),
        age              INTEGER,
        salary           REAL,
        PRIMARY KEY (eid),
        CHECK (salary >= 20000)
        )
```

3.b)

```
CREATE ASSERTION managerAlsoEmployee
CHECK ((Select COUNT (*)
          FROM Dept D
          WHERE D.managerid NOT IN (SELECT * FROM Emp)) = 0)
```

3.c)

```
CREATE TABLE Works
        (eid             INTEGER,
        did              INTEGER,
        pct_time         INTEGER,
        PRIMARY KEY (eid, did),
```

```
        CHECK (pct_time < 100)
        )
```

3.d)

```
CREATE ASSERTION mSalaryHigherE
CHECK ((SELECT E.eid
        FROM Emp E
        WHERE MAX(E.salary))
        =
        (SELECT D.managerid
        FROM Dept D)
        )
```

3.e)

```
CREATE TRIGGER IncrManagerSalary
        AFTER INSERT ON Emp
        FOR EACH STATEMENT
        UPDATE Emp
                Set E.salary = E1.salary
                FROM Emp E, Emp E1, Dept D
                WHERE E.salary > E1.salary AND E1.eid = D.managerid
```