Part1

Task 1:

Create the above database schema using CREATE TABLE statements, including primary key constraints, and the constraint that salary is integer in the range [5000,20000]. You can assume CHAR (20) type for all other attributes.

```sql
CREATE TABLE Employee (
    eid CHAR(20) PRIMARY KEY,
    name CHAR(20),
    salary INT CHECK (salary >= 5000 AND salary <= 20000),
    dept CHAR(20)
);

CREATE TABLE Sales (
    dept CHAR(20),
    item CHAR(20),
    PRIMARY KEY (dept, item)
);

CREATE TABLE Types (
    item CHAR(20),
    color CHAR(20),
    PRIMARY KEY (item, color)
);
```

Task2:

Insert the above records into the tables using INSERT statements.

```sql
INSERT INTO Employee (eid, name, salary, dept) VALUES
('111', 'Jane', 8000, 'Household'),
('222', 'Anderson', 8000, 'Toy'),
('333', 'Morgan', 10000, 'Cosmetics'),
('444', 'Lewis', 12000, 'Stationery'),
('555', 'Nelson', 6000, 'Toy'),
('666', 'Hoffman', 16000, 'Cosmetics');

INSERT INTO Sales (dept, item) VALUES
('Stationery', 'pen'),
('Cosmetics', 'lipstick'),
('Toy', 'puzzle'),
('Stationery', 'ink'),
('Household', 'disk'),
('Sports', 'skates'),
('Toy', 'lipstick');

INSERT INTO Types (item, color) VALUES
```

```
('pen', 'red'),
('lipstick', 'red'),
('pen', 'black'),
('puzzle', 'black'),
('ink', 'red'),
('ink', 'blue');
```

Task3:

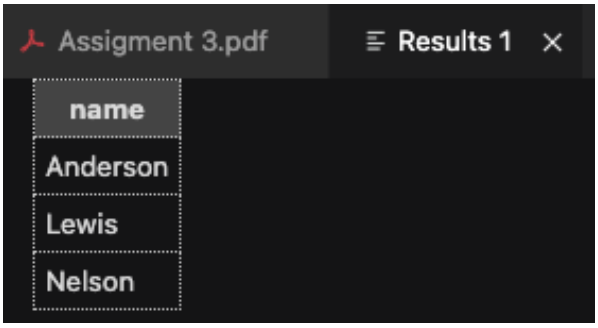
1. Compute the maximum salary for each department that sells at least two distinct items.

```
-- 1.3.1
SELECT dept, MAX(salary)
FROM Employee
WHERE dept IN (
        SELECT dept
        FROM Sales
        GROUP BY dept
        HAVING COUNT(DISTINCT item) >= 2
    )
GROUP BY dept;
```

dept	MAX(salary)
Toy	8000
Stationery	12000

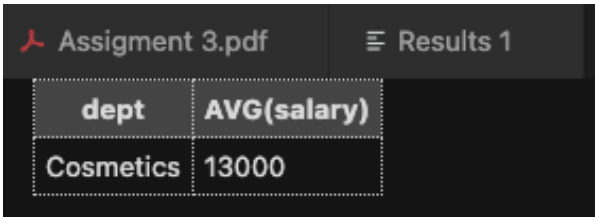
2. Compute the names of the employees who work in a department that sells some item in black color

```sql
-- 1.3.2
SELECT DISTINCT E.name
FROM Employee AS E
JOIN Sales AS S ON E.dept = S.dept
JOIN Types AS T ON S.item = T.item
WHERE T.color = 'Black';
```



3. For each department that has a larger average salary than that of "Stationery" department, find its average salary.

```sql
-- 1.3.3
SELECT dept, AVG(salary)
FROM Employee
GROUP BY dept
HAVING AVG(salary) > (
        SELECT AVG(salary)
        FROM Employee
        WHERE dept = 'Stationery'
    );
```



4. Find the number of the departments that have a smaller average salary than that of "Stationery" department.

```sql
-- 1.3.4
SELECT COUNT(DISTINCT dept)
FROM Employee
WHERE dept IN (SELECT dept
                FROM Employee
                GROUP BY dept
                HAVING AVG(salary) < (
                        SELECT AVG(salary)
                        FROM Employee
                        WHERE dept = 'Stationery'
                    ));
```

5. Which department pays every of its employees at least 7000?

```sql
-- 1.3.5
SELECT dept
FROM Employee
GROUP BY dept
HAVING MIN(salary) >= 7000;
```

6. Which departments sell all items sold by “Cosmetics” department

```sql
-- 1.3.6
SELECT dept
FROM Sales
WHERE item IN (
    SELECT item
    FROM Sales
    WHERE dept = 'Cosmetics'
)
GROUP BY dept
HAVING COUNT(DISTINCT item) = (
    SELECT COUNT(DISTINCT item)
    FROM Sales
    WHERE dept = 'Cosmetics'
);
```