

Part1 Using MySQL

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.

```
1 CREATE TABLE Employee (  
2     eid CHAR(20) PRIMARY KEY,  
3     name CHAR(20),  
4     salary INT CHECK (salary >= 5000 AND salary <= 20000),  
5     dept CHAR(20)  
6 );  
7  
8 CREATE TABLE Sales (  
9     dept CHAR(20),  
10    item CHAR(20),  
11    PRIMARY KEY (dept, item)  
12 );  
13  
14 CREATE TABLE Types (  
15     item CHAR(20),  
16     color CHAR(20),  
17     PRIMARY KEY (item, color)  
18 );
```

Task2:

Insert the above records into the tables using INSERT statements.

```
1 INSERT INTO Employee (eid, name, salary, dept) VALUES  
2 ('111', 'Jane', 8000, 'Household'),  
3 ('222', 'Anderson', 8000, 'Toy'),  
4 ('333', 'Morgan', 10000, 'Cosmetics'),  
5 ('444', 'Lewis', 12000, 'Stationery'),  
6 ('555', 'Nelson', 6000, 'Toy'),  
7 ('666', 'Hoffman', 16000, 'Cosmetics');  
8  
9 INSERT INTO Sales (dept, item) VALUES  
10 ('Stationery', 'pen'),  
11 ('Cosmetics', 'lipstick'),  
12 ('Toy', 'puzzle'),  
13 ('Stationery', 'ink'),  
14 ('Household', 'disk'),  
15 ('Sports', 'skates'),  
16 ('Toy', 'lipstick');  
17  
18 INSERT INTO Types (item, color) VALUES
```

```
19 ('pen', 'red'),
20 ('lipstick', 'red'),
21 ('pen', 'black'),
22 ('puzzle', 'black'),
23 ('ink', 'red'),
24 ('ink', 'blue');
```

Task3:

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

```
1  -- 1.3.1
2  SELECT dept, MAX(salary)
3  FROM Employee
4  WHERE dept IN (
5      SELECT dept
6      FROM Sales
7      GROUP BY dept
8      HAVING COUNT(DISTINCT item) >= 2
9  )
10 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

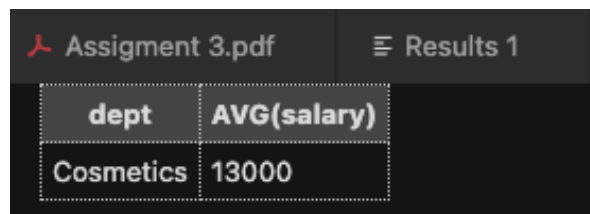
```
1  -- 1.3.2
2  SELECT DISTINCT E.name
3  FROM Employee AS E
4  WHERE E.dept IN (
5      SELECT S.dept
6      FROM Sales AS S
7      WHERE S.item IN (
8          SELECT T.item
9          FROM Types AS T
10         WHERE T.color = 'Black'
11     )
12 );
```



name	
Anderson	
Lewis	
Nelson	

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

```
1  -- 1.3.3
2  SELECT dept, AVG(salary)
3  FROM Employee
4  GROUP BY dept
5  HAVING AVG(salary) > (
6      SELECT AVG(salary)
7      FROM Employee
8      WHERE dept = 'Stationery'
9  );
```



dept	AVG(salary)
Cosmetics	13000

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

```
1  -- 1.3.4
2  SELECT COUNT(DISTINCT dept)
3  FROM Employee
4  WHERE dept IN (SELECT dept
5                  FROM Employee
6                  GROUP BY dept
7                  HAVING AVG(salary) < (
8                      SELECT AVG(salary)
9                      FROM Employee
10                     WHERE dept = 'Stationery'
11                     ));
```

Assignment 3.pdf	Results 1
COUNT(DISTINCT dept)	
2	

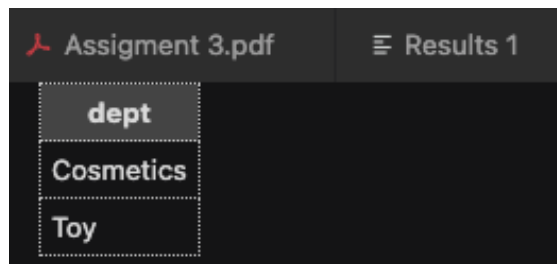
5. Which department pays every of its employees at least 7000?

```
1  -- 1.3.5
2  SELECT dept
3  FROM Employee
4  GROUP BY dept
5  HAVING MIN(salary) >= 7000;
```

Assignment 3.pdf	Results 1
dept	
Household	
Cosmetics	
Stationery	

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

```
1  -- 1.3.6
2  SELECT dept
3  FROM Sales
4  WHERE item IN (
5      SELECT item
6      FROM Sales
7      WHERE dept = 'Cosmetics'
8  )
9  GROUP BY dept
10 HAVING COUNT(DISTINCT item) = (
11     SELECT COUNT(DISTINCT item)
12     FROM Sales
13     WHERE dept = 'Cosmetics'
14 );
```



dept	Results 1
Cosmetics	
Toy	

Part2

Question 1

S	A	C	T	D
David	7924	CMPT412	Yasutaka Furukawa	CS
David	7924	CMPT354	Ke Wang	CS
Tom	8850	CMPT354	Ke Wang	CS

Data redundancy:

- Data is redundancy i.e. `tuple(David, 7924)` and `tuple(CMPT354, Ke Wang, CS)`.
- “Ke Wang” and “CS” can be figured out by using FD’s `C→T` and `C→D`.
- “7924” can be figured out by using FD’s `S→A`

Update anomaly: if `David` is transferred to `8850`, will we remember to change each of his tuples

Deletion anomaly: if no one takes `CMPT354`, we will lose track of the fact that `tuple(CMPT354, Ke Wang, CS)`

Insertion anomaly: Impossible to insert a student without knowing his/her course schedule.

Question 2

2.1 Find all keys of R with respect to F.

key = {S,C} It is the only key.

2.2 Test if R in BCNF with respect to F, why?

NO

C, S, T aren't super key so they violate BCNF.

2.3 Produce a BCNF decomposition through a series of binary decomposition. For each binary decomposition, tell the FD used for the decomposition and show the FDs holding on the decomposed tables.

- Pick BCNF violation: $S \rightarrow A$
 - FD used for the decomposition: $S \rightarrow A$:
 - $R_1 = S^+ = \{S, A\}$ holds $S \rightarrow A$
 - $R_2 = (R - S^+) \cup S = \{S, C, T, D\}$ holds $C \rightarrow T$, $T \rightarrow D$
- BCNF violation R2: $C \rightarrow T$
 - FD used for the decomposition: $C \rightarrow T$, $T \rightarrow D$
 - $R_3 = C^+ = \{C, T, D\}$ holds $C \rightarrow T$, $T \rightarrow D$
 - $R_4 = (R_2 - C^+) \cup C = \{S, C\}$
- BCNF violation R3: $T \rightarrow D$
 - FD used for the decomposition: $T \rightarrow D$, $C \rightarrow T$
 - $R_5 = T^+ = \{T, D\}$ holds $T \rightarrow D$
 - $R_6 = (R_3 - T^+) \cup T = \{C, T\}$ holds $C \rightarrow T$
- Final result:
 - There are 4 relationships (R_1, R_2, \dots is renamed):
 - $R_1 = \{T, D\}$ holds $T \rightarrow D$, T is key
 - $R_2 = \{S, A\}$ holds $S \rightarrow A$, S is key
 - $R_3 = \{C, T\}$ holds $C \rightarrow T$, C is key
 - $R_4 = \{S, C\}$, {S, C} is key

2.4 Explain why the decomposed tables produced in 3 is a better representation than the original single table R.

There isn't data redundancy, update anomaly. Insertion anomaly and deletion anomaly.

2.5 Is the final decomposition in 3 dependency-preserving, why

$R1 = \{T, D\}$ holds $F1 = T \rightarrow D$, T is key

$R2 = \{S, A\}$ holds $F2 = S \rightarrow A$, S is key

$R3 = \{C, T\}$ holds $F3 = C \rightarrow T$, C is key

$R4 = \{S, C\}$, {S, C} is key

$F1 \cup F2 \cup F3 = F$. In other words $F1, F2, F3$ implies F . So it is dependency-preserving

2.6 Is the original schema R in 3NF with respect to F, why

No

$F = T \rightarrow D, S \rightarrow A, C \rightarrow T$.

Key = {S, C}

D, A, T is not prime (each of them isn't a member of any key).

S, T, C is not superkey.

2.7 If the answer to 6 is no, produce a 3NF decomposition that is lossless and dependency-preserving.

$F = T \rightarrow D, S \rightarrow A, C \rightarrow T$.

(3 steps: Split right sides into single attributes, Remove redundant attributes from left sides of FDs, Remove redundant FDs)

minimal cover after 3 steps is still $T \rightarrow D, S \rightarrow A, C \rightarrow T$.

$R1 = \{T, D\}$ holds $T \rightarrow D$

$R2 = \{S, A\}$ holds $S \rightarrow A$

$R3 = \{C, T\}$ holds $C \rightarrow T$

no key is contained in any R_i :

$R4 = \{S, C\}$

Final result 3NF Decomposition {TD, SA, CT, SC}

2.8 Is the decomposition produced in 7 in BCNF?

Yes

$$T^+ \rightarrow R1$$

T is superkey of R1 = {T, D} holds T→D, D isn't contained in T, it is in BCNF

$$S^+ \rightarrow R2$$

S is superkey of R2 = {S, A} holds S→A, A isn't contained in S, it is in BCNF

$$C^+ \rightarrow R3$$

C is superkey of R3 = {C, T} holds C→T, T isn't contained in C, it is in BCNF

For R4 there is no FD so it is in BCNF