Warmup #1

Consider the following set of functional dependencies, $F$, for the schema $R(A, B, C, D, E)$

- $AB \rightarrow C$
- $C \rightarrow D$
- $D \rightarrow EA$
- $E \rightarrow B$

1. Find the candidate keys for the schema $R$.
2. Compute the closure, $F^+$, for the set $F$.

**Hint:** you can use * to refer to any subset of attributes in $R$, and $\alpha$ to indicate the full set $\{A, B, C, D, E\}$. So $A* \rightarrow \alpha$ means that $A$, along with any and all subsets of $\{B, C, D, E\}$, functionally determines all attributes.

Warmup #2

Using the functional dependency set from the first warmup, compute the canonical cover $F_c$.

Warmup #3

Consider the following schema:

- student($ID$, name, departmentName, totalCredits)
- professor($ID$, name, departmentName, salary)
- classroom($building$, roomNumber, capacity)
- course($courseID$, semester, year, building, roomNumber, professorID)
- takes($studentID$, courseID, grade)

Note that:

- the $building$, $roomNumber$ attributes of $course$ relation are references to the $classroom$ relation
- $professorID$ of $course$ is a reference to $professor$
- $studentID$ of $takes$ is a reference to $student$
- $courseID$ of $takes$ is a reference to $course$

1. Considering the given database schema, write SQL queries to construct this database, where the attributes name, departmentName, building are strings of various lengths (maximum length of 50), semester is only one character, and all remaining attributes are integers.
2. Write a SQL query to find the name of every professor in the CS department.

3. If the salary attribute of professor represents monthly salary, write a SQL query to find every possible pairing of student ID and professor ID, where professor IDs included in the result consist of the IDs of the professors whose yearly salary is greater than $10,000 and student IDs included in the result consist of IDs of students who have completed fewer than 10 credits.

**Warmup #4**

Write a SQL query for each of the following, using the schema from warmup 3:

1. If the salary attribute of professor represents monthly salary, find the names and yearly salaries of professors who hold classes in rooms that do not exist in classroom.

2. Find the name, total credits, and GPA of every student whose GPA is greater than 3.5. GPA can be calculated by summing a student’s grades and dividing by total credits.

3. Write a SQL query for finding the ID and name of the professor(s) who earns the highest salary, without using aggregate functions.

**Problem 5 (To Be Graded)**

Using the following schema, write SQL queries to answer the following questions.

\[
\begin{align*}
\text{advisor} & : (s\_ID, i\_ID) \\
\text{classroom} & : (building, room\_number, capacity) \\
\text{course} & : (course\_id, title, dept\_name, credits) \\
\text{department} & : (dept\_name, building, budget) \\
\text{instructor} & : (ID, name, dept\_name, salary) \\
\text{section} & : (course\_id, sec\_id, semester, year, building, room\_number, time\_slot\_id) \\
\text{student} & : (ID, name, dept\_name, tot\_cred) \\
\text{takes} & : (ID, course\_id, sec\_id, semester, year, grade) \\
\text{teaches} & : (ID, course\_id, sec\_id, semester, year)
\end{align*}
\]

1. Find the students who have had classes in the highest number of distinct buildings.

2. Find the instructor who has given the highest number of A’s.

3. Display the names of all students along with their advisor’s name. Note that not every student has an advisor.

4. The university needs to determine whether a student is eligible for graduation. Here are the following requirements for graduation:
   - the student has taken at least 130 credits
   - the student has taken courses in at least five different departments
   - the student has taken at least seven courses in a single department. This requirement is not restricted to the department associated with their student ID.

Find all students who are eligible to graduate.
Problem 6 (To Be Graded)

For each set of relations and functional dependencies, answer the following questions:

1. What are all the nontrivial functional dependencies that follow for the given functional dependencies? You should restrict yourself to FD’s with single attributes on the right side.

2. What are the candidate keys of the relation? List all possible candidate keys.

3. Give a canonical cover for the given relation and functional dependencies.

\[
R(A, B, C, D)
\]
\[
AB \rightarrow C
\]
\[
C \rightarrow D
\]
\[
D \rightarrow A
\]

\[
S(A, B, C, D)
\]
\[
A \rightarrow B
\]
\[
B \rightarrow C
\]
\[
B \rightarrow D
\]

\[
T(A, B, C, D)
\]
\[
AB \rightarrow C
\]
\[
BC \rightarrow D
\]
\[
CD \rightarrow A
\]
\[
AD \rightarrow B
\]
\[
ADC \rightarrow B
\]