

## HOMEWORK 2 (Overview)

### Functional dependencies and normalization

In this homework you will improve the database designed in the previous homework. You will use a method called normalization to reduce data redundancy and improve data integrity.

#### Important requirement:

- A maximum of 8 pages including the P+ assignment.
- A Normalisation Guide is available on Canvas as a required tool for this homework.

### REQUIRED TASKS (P)

1. Implement modifications to your solution if you:
  - a. received feedback to make modifications/changes to your schemas in homework 1
  - b. simply want to make adjustments of your own accord

For each modification, clearly state what has been modified and explain the reason behind the change.
2. The following table is an example of a poorly normalized relation.
  - a. In short words, explain how updating a row may lead to an update anomaly
  - b. In short words, explain how deleting the course 'Linear Algebra' may lead to a deletion anomaly
  - c. In short words, explain how adding a student who does not yet take a course may lead to an insertion anomaly
  - d. In short words, explain if this table suffers from redundancy. How can you tell if it does/doesn't?
  - e. If you want to, you may also try to create a schema that represents the table. This question will **NOT** be graded but serves as great exercise (Hint: StudentID is not a candidate key).

StudentID	StudentName	age	StudentDepartment	Course
29	Bob	22	Computer Science	Database Systems
29	Bob	22	Computer Science	Logical Programming
29	Bob	22	Computer Science	Intro to Python
89	Cliff	24	Maths	Linear Algebra
30	Anna	19	Maths	Calculus
30	Anna	19	Maths	Linear Algebra
19	Mike	29	Arts	Color Theory

**NOTE:** 1) Homework **MUST** be handed in via Canvas **only** and in the correct assignment folder, in one file representing the entire homework. 2) Your names, group number must be included in the document. 3) Take into consideration this **exercise is accumulative** and builds upon your solutions. 4) **All homeworks are based solely on the course case study.**

### Important notes for tasks 3-9:

- If any changes were made in task 1, use the updated schemas.
  - Make sure that your solution is in line with the case study's Detailed Requirements.
  - Any assumptions not included in the Detailed Requirements must be clearly stated.
  - Adding new schemas is permitted but you will need to make sure and explain how those meet the requirements of the previous normal forms too.
3. Include your schemas from HW 1.
  4. For each of your own schemas, make a list of all candidate keys in that schema.
  5. 1NF
    - a. What conditions must a table fulfill in order to be in 1NF?
    - b. In the following schema, explain why it is NOT in 1NF.
      - i. *Store(StoreID: int, StoreName: String, Owner:String, Workers:String[]).*
    - c. For each of your own schemas, explain why each is or is not in 1NF. If any schema is not in 1NF, change it to meet the requirements and explain what changes you have made.
  6. 2NF
    - a. What conditions must a table fulfill in order to be in 2NF?
    - b. In the following schema, explain why it is NOT in 2NF.
      - i. *Order(OrderNumber: int, ItemID: int, UnitPrice: int, Quantity:int)*
    - c. For each of your own schemas, explain why each is or is not in 2NF. If any schema is not in 2NF, change it to meet the requirements and explain what changes you have made.
  7. 3NF
    - a. What conditions must a table fulfill in order to be in 3NF?
    - b. In the following schema, explain why it is NOT in 3NF.
      - i. *Book(id: int, amountInStock: int, Title: String, ISBN: int)*
    - c. For each of your own schemas, explain why each is or is not in 3NF. If any schema is not in 3NF, change it to meet the requirements and explain what changes you have made.
  8. BCNF
    - a. What conditions must a table fulfill in order to be in BCNF?
    - b. In the following schema, explain why it is NOT in BCNF. Assume an athlete only runs one distance but may participate in multiple rounds where a round refers to a level of a tournament such as 'quarter-final', 'semi-final' etc.
      - i. *RaceResult(Distance: int, Round: String, Position: int, athlete: string, time: int)*
    - c. For each of your own schemas, explain why each is or is not in BCNF. If any schema is not in BCNF, change it to meet the requirements and explain what changes you have made.
  9. Make a list of your normalized schemas. Include any new schemas that you have made.

**NOTE: 1)** Homework **MUST** be handed in via Canvas **only** and in the correct assignment folder, in one file representing the entire homework. **2)** Your names, group number must be included in the document. **3)** Take into consideration this **exercise is accumulative** and builds upon your solutions. **4)** All homeworks are based solely on the course case study.

### Grading Criteria:

- Is the feedback for homework 1, if such was given, implemented?
- Are all changes in task 1, if such were made, implemented correctly and explained?
- Does the solution to task 2.a-d describe why each anomaly occurs?
- Does the solution to task 3-9 use the updated schemas from task 1, if such changes were made?
- Is the solution to task 3-9 in line with the case study's Detailed Requirements?
- Are all assumptions in task 3-9 not included in the Detailed Requirements clearly stated?
- Are all schemas in the final list in task 9 in BCNF?
- Do the schemas in task 3-9 include all the Database Data Requirements?
- Are all problems requiring normalization in task 5-8 explained?
- Are the normalized schemas correct in terms of format, fitting domains, valid primary keys, and non-nullable attributes never forced to be null?
- Is a list with all normalized schemas included.

## HOMEWORK 2 P+

### Important requirement:

To be able to pass the P+ assignment, the P assignment has to pass.

### TASKS (P+):

1. Observe the relation represented by schema *Teacher*(Teacher\_ID:integer, School\_ID:integer, Classes:string, School\_Address:string, School\_Zip\_Code:integer, School\_Country:string).
  - a) List all prime attributes, non prime attributes and candidate keys of *Teacher*.
  - b) For each of the 4 normal forms, list how *Teacher* violates it.

### Important notes:

- Assume Teacher\_ID is unique in any given school but not across different schools, that there is only 1 school per School\_Address but not zip, and that there is only 1 school per School\_ID.
- Assume no attribute in *Teacher* is nullable.
- Find at least one unique problem per normal form, e.g. it is not acceptable to find a violation for 1NF and just repeat it for the others.
- A problem must not break prior normal forms. E.g. a violation of 3NF may not be a violation in 2NF or 1NF.
- Beware of the Important Notes on page 1 in the Normalisation Guide.

**NOTE:** 1) Homework **MUST** be handed in via Canvas **only** and in the correct assignment folder, in one file representing the entire homework. 2) Your names, group number must be included in the document. 3) Take into consideration this **exercise is accumulative** and builds upon your solutions. 4) **All homeworks are based solely on the course case study.**

- Assume that each country has a unique zip code format with no overlap.
- Do not change the schema after finding a violation.

**Example:** In [Normal Form], [attribute 1] can't functionally depend on [attribute 2] because ...

**Grading Criteria:**

- Were *all* the prime attributes, non-prime attributes *and* candidate keys identified?
- Was there at least one unique problem per normal form correctly identified?
- Do the identified problems in no way contradict the Important Notes on page 1 in the Normalization Guide?

**NOTE:** 1) Homework **MUST** be handed in via Canvas **only** and in the correct assignment folder, in one file representing the entire homework. 2) Your names, group number must be included in the document. 3) Take into consideration this **exercise is accumulative** and builds upon your solutions. 4) **All homeworks are based solely on the course case study.**