Group 8

1. Arsya Hendratno Putri
2. Fegi Anora
3. Jab Yoan Efendi
4. Zikri Muhammad Madani

**SQL Nonquery Cases**

1. Normalize this table from 1NF to 3NF!  
   Student\_Grade\_Report (StudentNo, StudentName, Major, CourseNo, CourseName, InstructorNo, InstructorName, InstructorLocation, Grade)  
   (***HINT****: in 3NF, there will be 4 tables*)

**Answer:** A table can be said to be in the third normal form should it be in its second normal form and have no transitive partial dependency. So, in order to obtain its third normal form, the table ‘Student\_Grade\_Report’ first must be normalized to the second form (2NF). In this case, the table will be split into two tables.

1. **Students**

The first table consists of three attributes that are listed as follows:

* StudentNo (primary key);
* StudentName;
* and Major.

1. **Grade\_Report**

The second table consists of the remaining attributes that are listed as follows:

* StudentNo (foreign key);
* CourseNo (primary key);
* CourseName;
* InstructorNo;
* InstructorName;
* Instructor Location;
* and Grade.

The first table contains the information in regard to the student while the second one is responsible for saving the information concerning each of the students’ grades in different courses. By separating StudentNo and CourseNo, the composite primary keys have been made. Therefore, it enables a student to take multiple courses all at the same time and, in reverse, allows a course to be taken by multiple students. This organization thus eliminates partial dependencies and makes the now separated table achieve its second form. In order to normalize it into the third form, transitive partial dependencies have to be removed and, for that reason, the table will be split in four.

1. **Students**

The first table holds information regarding students and consists of three attributes that are listed as follows:

* StudentNo (primary key);
* StudentName;
* and Major.

1. **Courses**

The second table holds information regarding courses and consists of two attributes that are listed as follows:

* CourseNo (primary key)
* and CourseName.

1. **Instructors**

The third table holds information regarding instructors and consists of three attributes that are listed as follows:

* InstructorNo (primary key);
* InstructorName;
* and InstructorLocation.

1. **Grades**

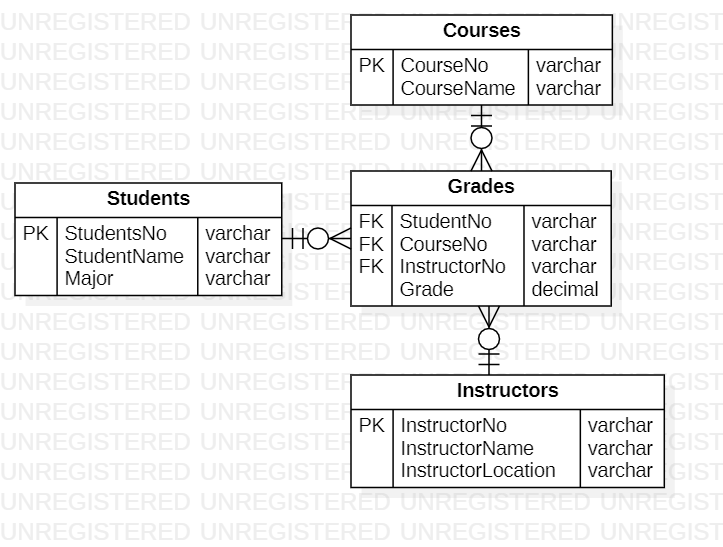
The fourth table holds information regarding grades and serves as a means to link the abovementioned three tables. Th;e attributes contained within it are listed as follows:

* StudentNo;
* CourseNo;
* InstructorNo;
* and Grade.

By splitting the tables, which have been in their second normal form in four, the redundancy has been eliminated. Therefore, each table only serves one particular purpose since it does not have any partial and transitive dependencies.

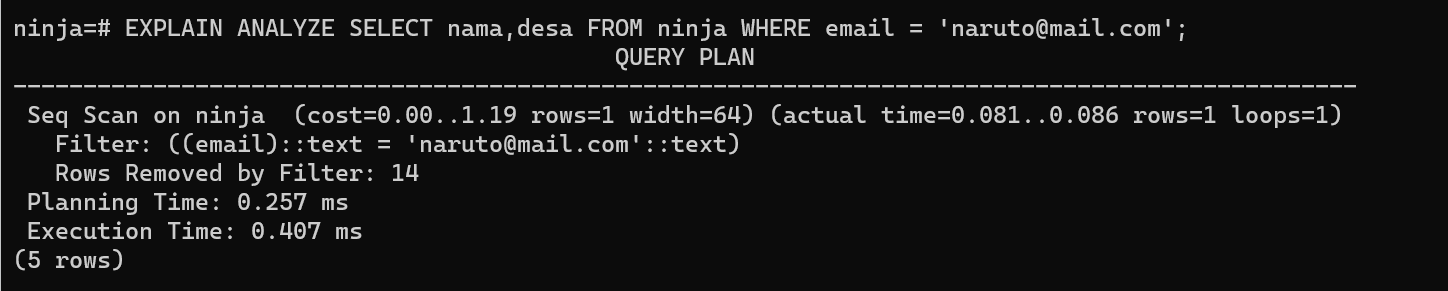
1. Create ERD based on the table you normalized!

**Answer:** The ERD is depicted in the picture below.

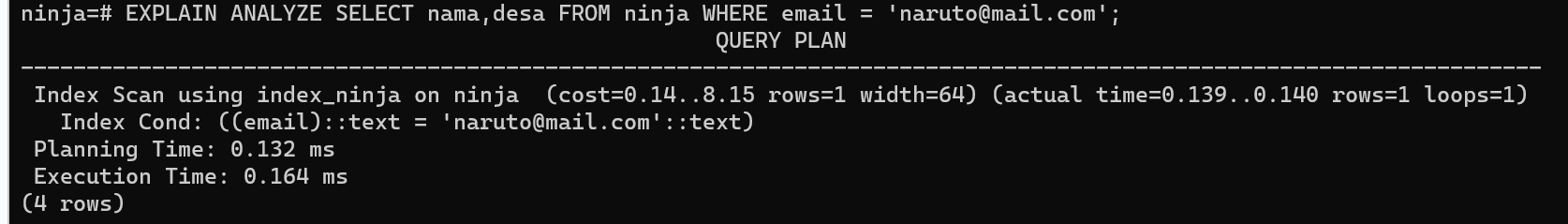


1. Use the ninja dataset, write a query that returns nama and desa, use email as a filter. Create a proper index to satisfy the query, provide the explain result before and after index creation. (do set enable\_seqscan = off first).

**Answer:** The explanation is written in the pictures attached below.

****

The picture above is before indexing while the picture below is **after indexing( enable\_seqscan = off)**.

****

From the pictures above, it is noticeable that the planning time and the execution time after the indexing is faster than before the indexing.