**INTRODUCTION TO DATABASE**

**SECTION: C**

**PROJECT TITLE: UNIVERSITY MANAGEMENT SYSTEM DATABASE**

**PRESENTED TO: REZWAN AHMED**

**GROUP MEMBER**

|  |  |
| --- | --- |
| NAME | ID |
| MUNTASIR RAHMAN | 20-43763-2 |
| LIMA AKTER | 20-43806-2 |
| SHEK IFTIER ISLAM | 20-43757-2 |

**INTRODUCTION**

A great university management system guarantees improved academic delivery, working efficiency and better student achievements. It provides several advantages for candidates, staff, and the corporation. Because the training panorama is experiencing a constant alternate, it has become essential for universities to apply an innovative college management system to overcome gift and future troubles. There are a lot of benefit from the management system.

Key economic advantages of a university management system:

**1. Time, money and resource saving:** A great university management system gets the student, teacher and staffs to save a lot of amounts of time and resources saving. And it’s a lot more efficient.

**2. E-learning:** The management itself is a great way for learning community. Cause it’ll contain all the data and information of the university. Moreover, there can be notes, results, notice, books and instruction uploading through the managing system.

**3. Improved database:** A great university management contains a great efficient of database system to manage all the data of the system.

**4. Backup:** The data can be backup through clouds or other way. Which makes it more reliable.

**5. Self-service:** This type of management system lets people to have self-service. Such as: enrolment, course choosing, course selection or registration, a portal community for teachers and staff to manage their schedule and develop the system furthermore.

**6. Improved communication:** The management system is a great platform to create a great eternal communication community through the student-teacher portal hub.

**OVERVIEW OF THE MANAGEMENT SYSTEM**

The department table has the information about the department and the faculty members working for the department the table has a unique id D\_ID

Every department has many students. To store student information the studet\_info table das same attributes the primary key of the table is S\_ID and other attributes are S\_NAME, D\_OF\_B, CITY, AGE, C\_ID, DEPARTMENT, CGPA.

One student have many courses and one course can be taken by many student so that we have a course table that has attribute C\_ID, C\_NAME, CREADIT, ROOM\_NO here C\_ID is the primary key

There is a classroom table to store the information about course schedule and the course instructor it has ROOM\_NO, SCHEDULE\_, F\_ID where ROOM\_NO is the primary key

Every department has many faculty member but one faculty member have only one department to work for so we have a table to store information about the faculty members that has a primary key F\_ID and other attributes F\_NAME, F\_EMAIL, F\_COURSE, SAL

**ER DIAGRAM**

Diagram

Description automatically generated

**NORMALIZATION**

***Table:*** DEPARTMENT (D\_ID, D\_NAME, FACULTY\_MEMBER)

1NF: No multivalued attributes.

2NF:

|  |  |
| --- | --- |
| D\_ID | D\_NAME |

|  |  |
| --- | --- |
| D\_ID | F\_ID |

3NF: There is no transitive dependency

|  |  |
| --- | --- |
| D\_ID | D\_NAME |

|  |  |
| --- | --- |
| D\_ID | F\_ID |

Finally,

|  |  |
| --- | --- |
| D\_ID | D\_NAME |

|  |  |
| --- | --- |
| D\_ID | F\_ID |

***Table:*** STUDENT (S\_ID, S\_NAME, D\_OF\_B, CITY, AGE, C\_ID, DEPARTMENT, CGPA)

1NF: No multivalued attributes.

2NF:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| S\_ID | S\_NAME | D\_OF\_B | CITY | AGE | D\_ID |

|  |  |  |
| --- | --- | --- |
| S\_ID | C\_ID | CGPA |

3NF: There is no transitive dependency

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| S\_ID | S\_NAME | D\_OF\_B | CITY | AGE | D\_ID |

|  |  |  |
| --- | --- | --- |
| S\_ID | C\_ID | CGPA |

Finaly,

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| S\_ID | S\_NAME | D\_OF\_B | CITY | AGE | D\_ID |

|  |  |  |
| --- | --- | --- |
| S\_ID | C\_ID | CGPA |

***Table:*** FACULTY(F\_ID, F\_NAME, F\_EMAIL, F\_COURSE, SAL)

1NF: No multivalued attributes.

2NF:

|  |  |  |  |
| --- | --- | --- | --- |
| F\_ID | F\_NAME | F\_EMAIL | F\_SAL |

|  |  |
| --- | --- |
| F\_ID | C\_ID |

3NF: There is no transitive dependency

|  |  |  |  |
| --- | --- | --- | --- |
| F\_ID | F\_NAME | F\_EMAIL | F\_SAL |

|  |  |
| --- | --- |
| F\_ID | C\_ID |

Finaly,

|  |  |  |  |
| --- | --- | --- | --- |
| F\_ID | F\_NAME | F\_EMAIL | F\_SAL |

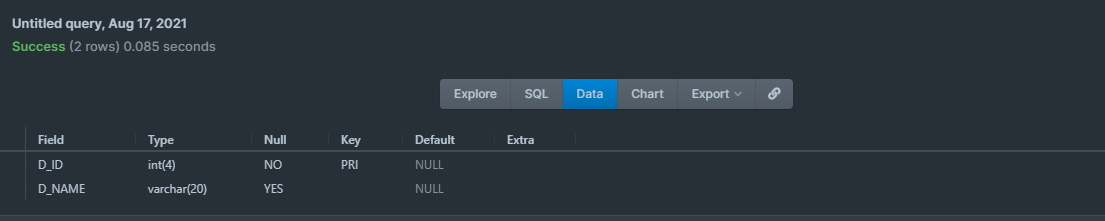
|  |  |
| --- | --- |
| F\_ID | C\_ID |

**FINAL TABLE LIST:**

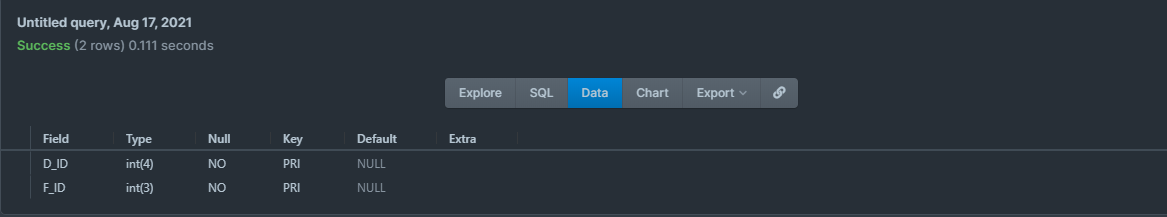
|  |  |
| --- | --- |
| Table name | atribute |
| department | D\_ID, D\_NAME |
| faculty\_members | D\_ID, F\_ID |
| student\_info | S\_ID, S\_NAME, D\_OF\_B, CITY, AGE, D\_ID |
| student\_courses | S\_ID, C\_ID, CGPA |
| course\_info | C\_ID, C\_NAME, CREADIT, ROOM\_NO |
| class\_info | ROOM\_NO, SCHEDULE\_, F\_ID |
| faculty\_info | F\_ID, F\_NAME, F\_EMAIL, F\_SAL |
| faculty\_courses | F\_ID, C\_ID |

**TABLE CREATION**

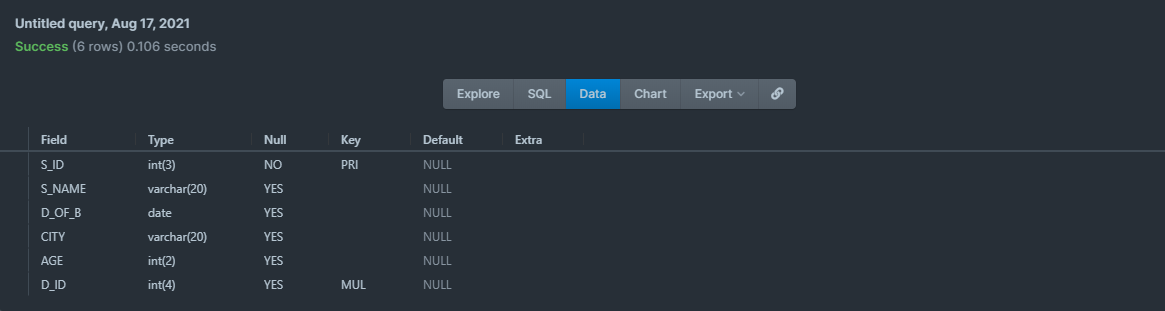
**1. department table:**



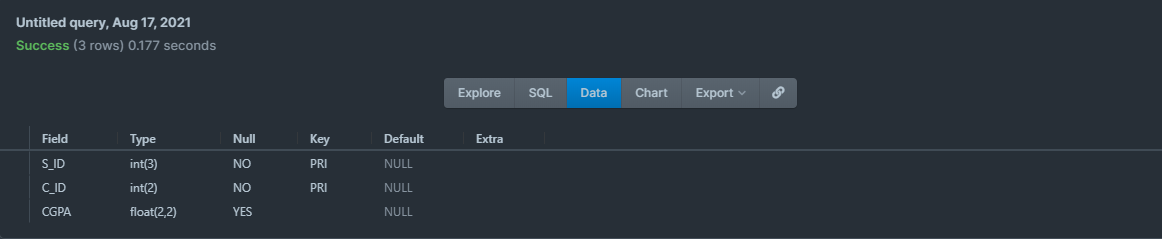
**2. faculty\_members table:**



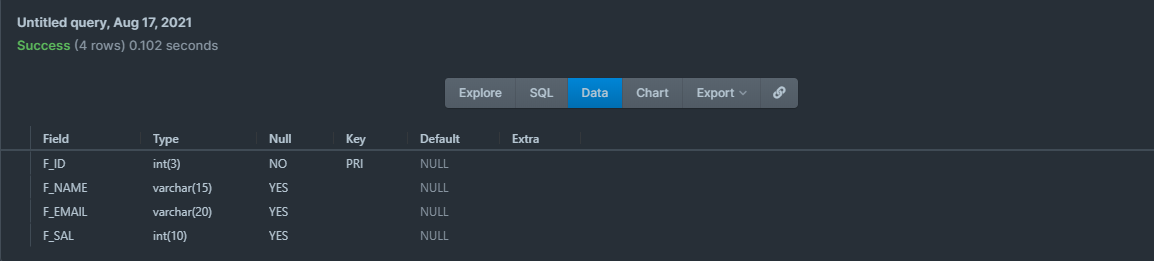
**3. student\_info table:**



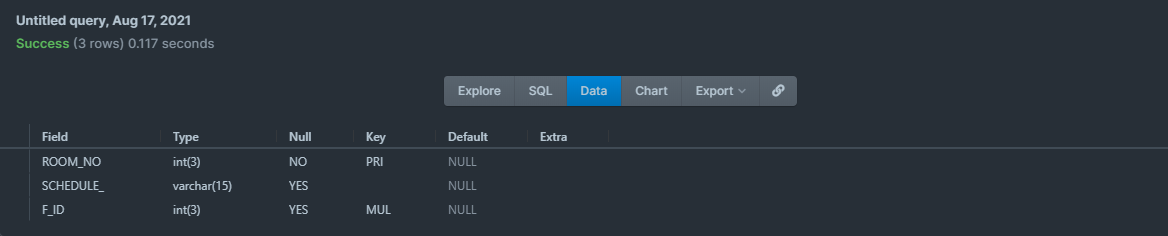
**4. student\_courses table:**



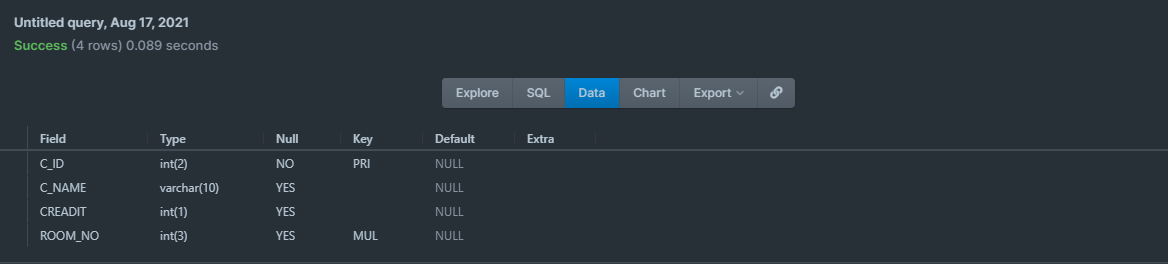
**5. faculty\_info table:**



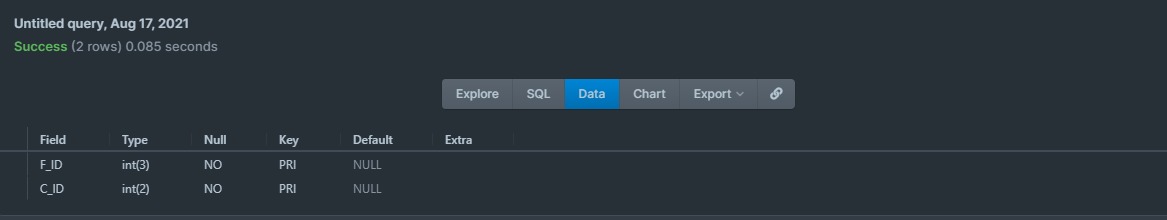
**6. class\_info table:**



**7. course\_info table:**

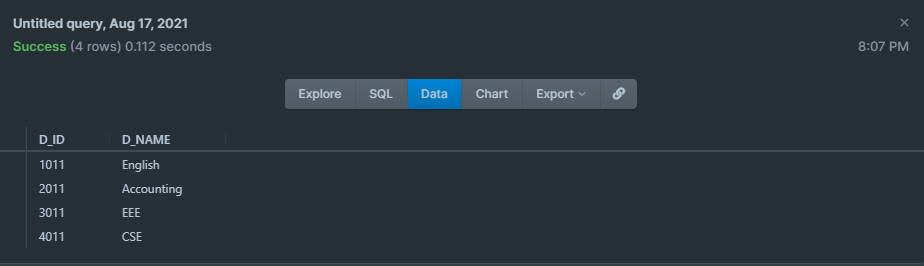


**8. faculty\_courses table:**

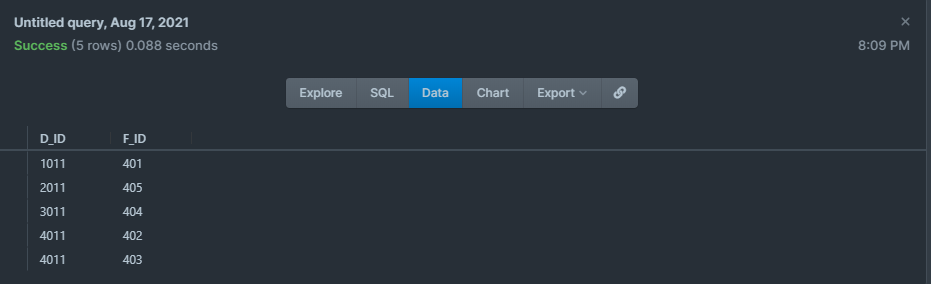


**DATA INSERTION**

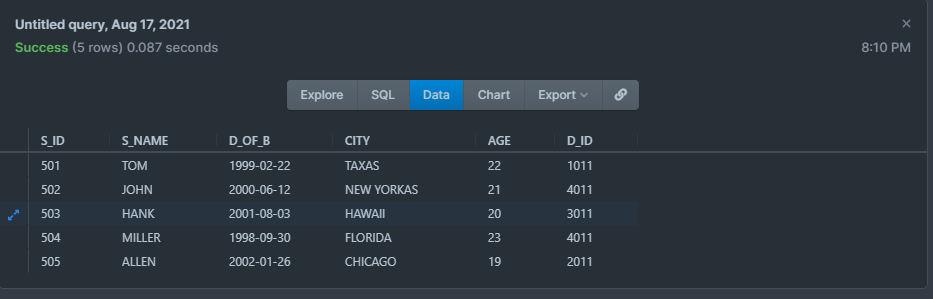
**1.department :**



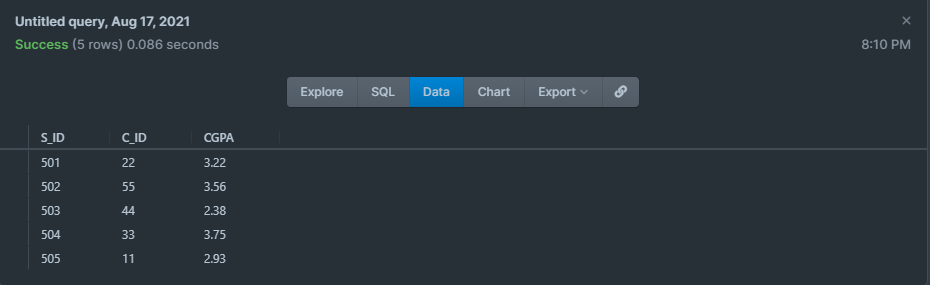
**2.faculty\_members :**



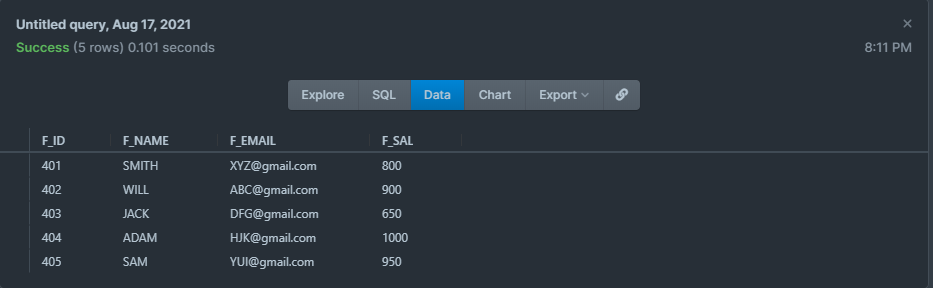
**3.student\_info :**



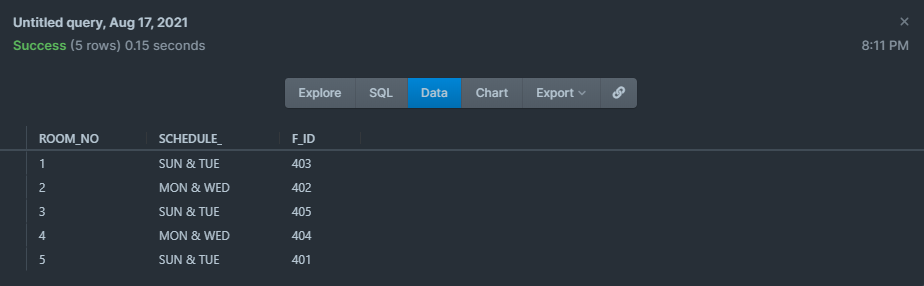
**4.student\_courses :**



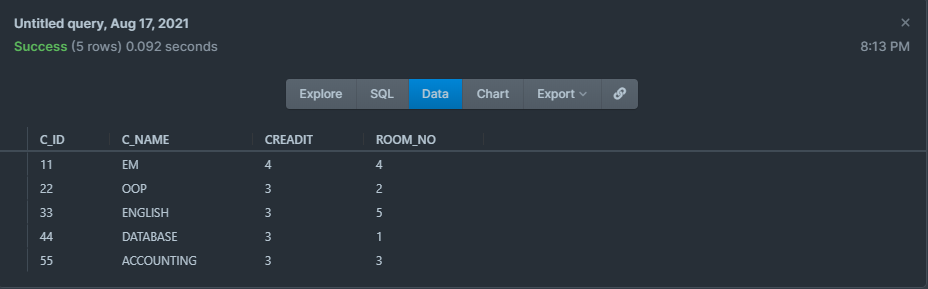
**5.faculty\_info :**



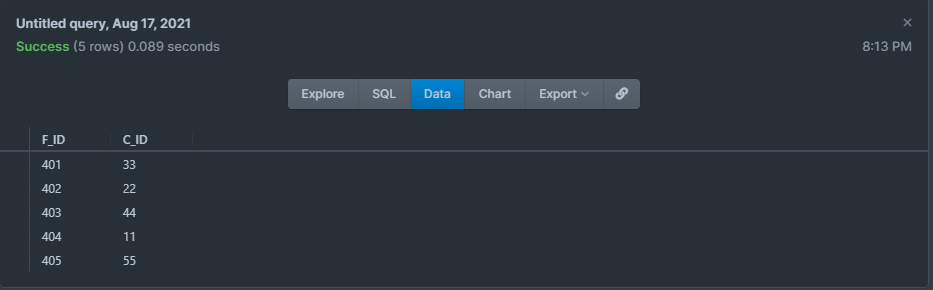
**6.class\_info :**



**7.course\_info :**



**8.faculty\_courses :**



**QUERY**

**-- 1) Show all the student details of the university**

= SELECT \*FROM student\_info

**-- 2) Show all the student from CSE department**

= SELECT student\_info.\* FROM student\_info,department

WHERE student\_info.D\_ID = department.D\_ID AND D\_NAME = 'CSE';

**-- 3) Show all the faculty members of CSE department**

= SELECT \*FROM faculty\_info

WHERE F\_ID IN (SELECT F\_ID FROM faculty\_members WHERE D\_ID = (SELECT D\_ID FROM department WHERE D\_NAME = 'CSE'))

**--4) Show all the courses available in the university that have not more than 3 credits**

= SELECT \*FROM course\_info WHERE CREADIT<4

**--5) Show all the courses available in the university**

= SELECT c\_name FROM course\_info

**--6) Which faculty take class in the room number 002 show his/her details**

= SELECT \*FROM faculty\_info

WHERE F\_ID = (SELECT F\_ID FROM class\_info WHERE ROOM\_NO = 002)

**--7) How many students are in the university show department wise**

= SELECT COUNT(\*),D\_NAME FROM student\_info,department

WHERE student\_info.D\_ID = department.D\_ID GROUP BY D\_NAME

**--8) Show the highest paid faculty member**

= SELECT \*FROM faculty\_info WHERE F\_SAL = (SELECT MAX(F\_SAL) FROM faculty\_info)

**--9) Show the class schedule of faculty id 403**

= SELECT SCHEDULE\_ FROM faculty\_info,class\_info

WHERE class\_info.F\_ID = faculty\_info.F\_ID AND faculty\_info.F\_ID = 403

**--10) Create a view and query the view**

= CREATE VIEW faculty\_contact AS

SELECT F\_NAME,F\_EMAIL

FROM faculty\_info;

SELECT \* FROM faculty\_contact

**PART B:**

From this database project we learned about sql language, ER-Diagrams, Tables, inserting data into tables and Normalization. We also faced some problems while doing normalizations and specially creating the table student\_courses in that table there is an attribute that store students cgpa of a particular subject, for that we have to create a Colom so that it can hold double type variable but by mistake we created the column with wrong data type [float (size, d)] as a result we had to face some problem while inserting data in to that table after realizing the mistake we had to use altar table command to modify the data type. In the end we have achieved skills on creating ER-Diagrams and Normalizing. From this particular project we have learned new things and in the near future, we can develop more advanced management systems.