

CASE STUDY ARICAL ON SAIT.



Written By Yash Panjwani

Professor Dr. Junaid Qazi

Contents

1. Introduction.....	3
1.1 Purpose of Database Design.....	3
1.2 Overview of SAIT.	3
2. Mission.....	3
2.1 Mission Statement.	3
2.2 Objective.....	3
3. Database Design Overview.....	4
4. Tables and attributes.....	4
5. Relationships Between Tables.	5
6. Entity Relationship Diagram (ERD).....	6
6.1 Overview of ERD.....	6
7. Conclusion.....	6
8. Appendix.....	7

1. Introduction.

1.1 Purpose of the Database Design

This case study looks at how SAIT College handles student enrollment and how important it is for a good student experience. To make the process easier for both students and staff, SAIT created a database system. This system helps with student registration, course management, and keeping track of student records. In this case study, we'll explain how the system works, its main parts, how everything is connected, and the benefits it brings to the enrollment process.

1.2 Overview of SAIT

SAIT College is a well-known for offering practical programs in areas like technology, business, and trades. It helps students gain real-world skills and gives hands-on experience to succeed in their careers. With up-to-date facilities and strong industry connections, SAIT prepares students for the job market.

2. Mission.

2.1 Mission Statement.

SAIT College aims to help students succeed by providing practical education that aligns with industry needs. We focus on keeping all processes, including enrollment, simple and easy to navigate, allowing students to concentrate on their studies without facing unnecessary complications.

2.2 Objectives

- **Skills for the Future** - SAIT College teaches skills that prepare students for future jobs and a changing job market.
- **Learning for Life** - The College encourages students to keep learning and growing even after they graduate.
- **Global Perspective** - SAIT prepares students to understand and succeed in a global market by teaching them about international trends and diverse cultures, helping them compete effectively on a worldwide stage.
- **Industry-Driven** - The programs at SAIT are shaped by industry needs, ensuring students learn what employers want.
- **Commitment to Excellence** - SAIT College provides high-quality education and constantly improving its programs.

3. Database Design Overview

The SAIT enrollment process is organized using several important database tables, each representing a different part of the system. These tables hold key information about students, courses, departments, faculty members, classrooms, enrollments, and grades. This setup helps manage and access information easily, making the enrollment process smoother for students and staff. The following is a breakdown of the main tables that support the enrollment system.

4. Tables and Attributes.

1. Student Table

- The Student Table holds personal and academic details for each student.
- Fields: Student_ID, Firstname, Lastname, DOB (Date of Birth), Address, Gender, Phone, Email, Department_ID

2. Course Table

- The Course Table lists all available courses, along with their related department and faculty.
- Fields: Course_ID, Course_Name, Department_ID, Faculty_ID.

3. Enrollment Table

- The Enrollment Table records which students are enrolled in which courses and tracks the date when students signed up.
- Fields: Enrollment_ID, Student_ID, Course_ID, Enrollment_Date.

4. Department Table

- The Department Table contains information about the different academic departments at SAIT, each offering various courses.
- Fields: Department_ID, Department_Name.

5. Faculty Table

- The Faculty Table stores information about faculty members, including their names and contact details, as well as their department affiliation.
- Fields: Faculty_ID, Firstname, Lastname, Phone, Email, Department_ID.

6. Classroom Table

- The Classroom Table keeps track of where courses are held, detailing room numbers and building names.
- Fields: Classroom_ID, Classroom_No, Building_Name.

7. Grades Table

- The Grades Table records the grades that students receive in their courses, linking each grade to the respective student and course.
- Fields: Grades_ID, Student_ID, Course_ID, Grade.

8. Process Table

- The Process Table monitors the steps in the enrollment process, making sure that all necessary actions are tracked for each student.
- Fields: Process_ID, Steps, Date, Status.

5. Relation between Tables

1. **The Student to Department relationship is (one-to-many).**

Each student belongs to one department, but a department can have many students. The **Department_ID** in the Student table links to the Department table.

2. **The Department to Course relationship is also (one-to-many).**

Each department offers many courses, while each course belongs to just one department. The **Department_ID** in the Course table connects to the Department table.

3. **In the Student to Enrollment relationship, which is (one-to-many)?**

A student can enroll in many courses, and each course can have many students. The **Student_ID** in the Enrollment table links to the Student table, and the **Course_ID** connects to the Course table.

4. **The Course to Faculty relationship (one-to-many).**

A faculty member can teach several courses, but each course is taught by just one faculty member. The **Faculty_ID** in the Course table links to the Faculty table.

5. **The Student to Grades relationship is (one-to-many).**

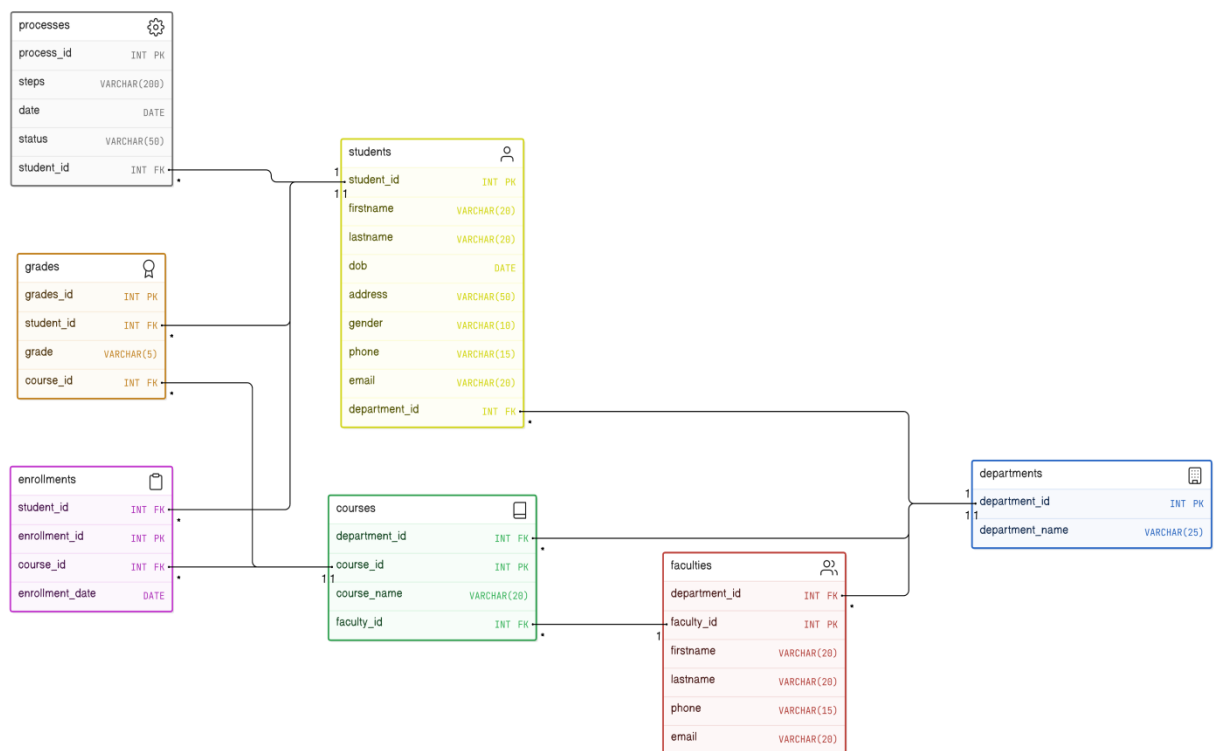
A student can receive grades for many courses. The **Student_ID** in the Grades table links to the Student table, while the **Course_ID** connects to the Course table.

6. ERD (Entity Relationship Diagram)

6.1 Overview of ERD.

The Entity Relationship Diagram (ERD) outlines how the enrollment database at SAIT is set up. It shows how students, courses, departments, and faculty are linked together. This organized system makes enrollment easier, reduces mistakes, and keeps information safe.

6.2 ER-Diagram.



7. Conclusion

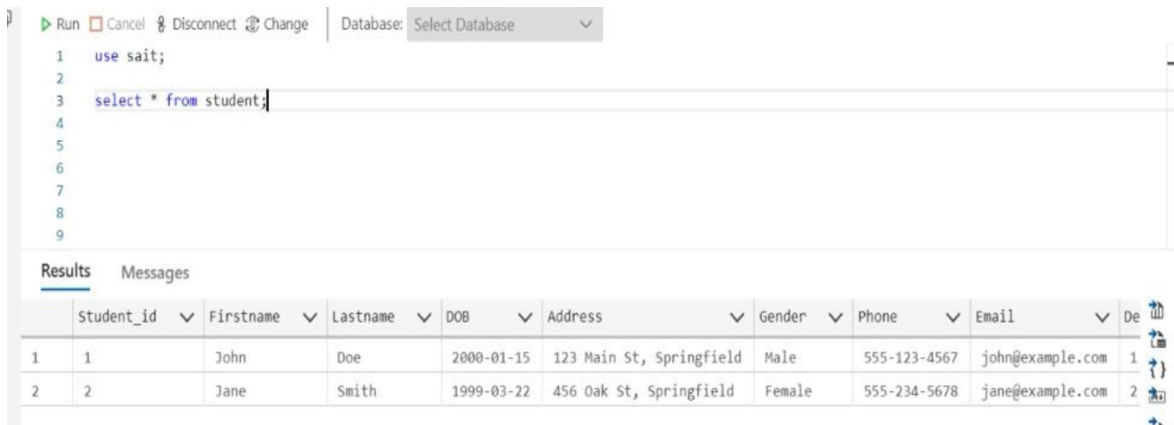
The enrollment process at SAIT is an important part of the student experience. With a well-designed database system, SAIT allows students to sign up for courses easily while keeping their academic records correct. This system not only makes the enrollment process more efficient but also ensures that information is clear, secure, and able to grow as the college expands.

8. Appendix

8.1 Table with tested database and query.

1. Student.

Field Name	Data Type
Student_id	INT PK
Firstname	VARCHAR (20)
Lastname	VARCHAR (20)
DOB(Date of Birth)	DATE
Address	VARCHAR (50)
Gender	VARCHAR (10)
Phone	VARCHAR (15)
Email	VARCHAR (20)
Department_id	INT FK

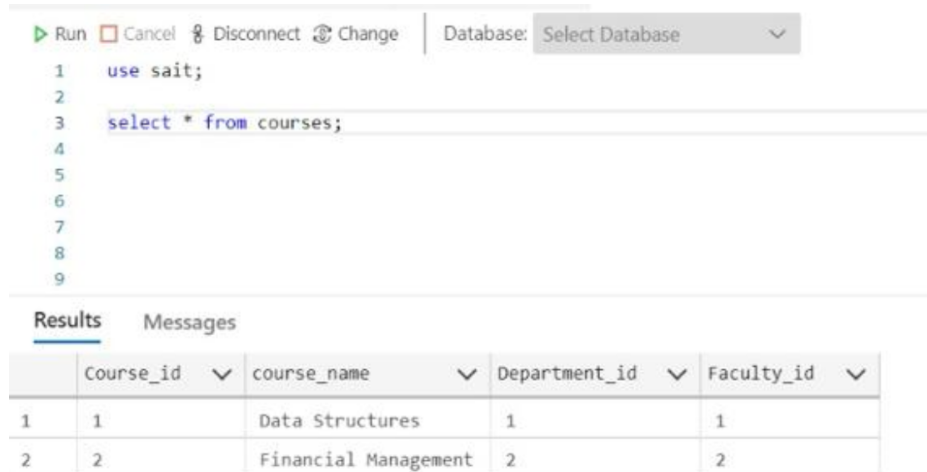


The screenshot shows a database query interface. At the top, there are buttons for 'Run', 'Cancel', 'Disconnect', and 'Change', along with a 'Database:' dropdown menu set to 'Select Database'. Below this is a text area containing the SQL query: `use sait;` followed by `select * from student;`. The query is numbered 1 through 9. Below the query editor, there are two tabs: 'Results' and 'Messages'. The 'Results' tab is active, displaying a table with 11 columns: Student_id, Firstname, Lastname, DOB, Address, Gender, Phone, Email, and De. The table contains two rows of data. Row 1: Student_id 1, Firstname John, Lastname Doe, DOB 2000-01-15, Address 123 Main St, Springfield, Gender Male, Phone 555-123-4567, Email john@example.com, De 1. Row 2: Student_id 2, Firstname Jane, Lastname Smith, DOB 1999-03-22, Address 456 Oak St, Springfield, Gender Female, Phone 555-234-5678, Email jane@example.com, De 2.

	Student_id	Firstname	Lastname	DOB	Address	Gender	Phone	Email	De
1	1	John	Doe	2000-01-15	123 Main St, Springfield	Male	555-123-4567	john@example.com	1
2	2	Jane	Smith	1999-03-22	456 Oak St, Springfield	Female	555-234-5678	jane@example.com	2

2. Course Table

Field Name	Data Type
Course_id	INT PK
Course_name	VARCHAR (20)
Department_id	INT FK
Faculty_id	INT FK



Run Cancel Disconnect Change Database: Select Database

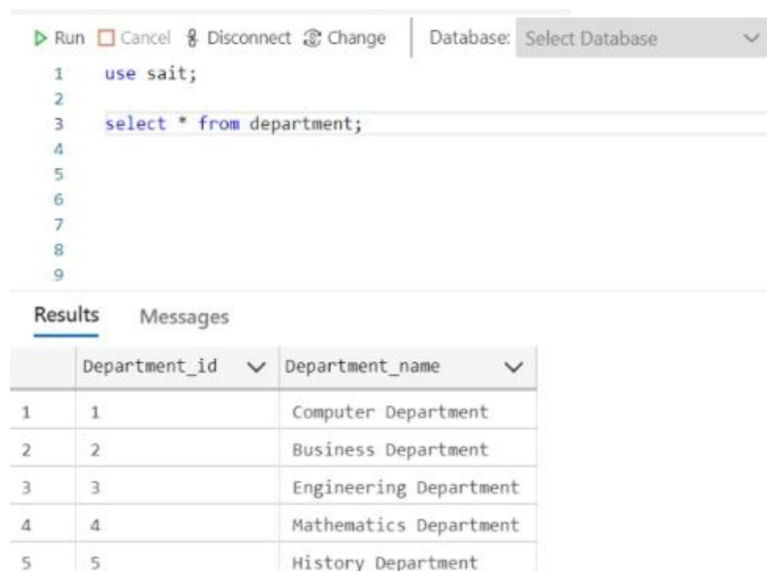
```
1 use sait;  
2  
3 select * from courses;  
4  
5  
6  
7  
8  
9
```

Results Messages

	Course_id	course_name	Department_id	Faculty_id
1	1	Data Structures	1	1
2	2	Financial Management	2	2

3. Department Table

Field Name	Data Type
Department_id	INT PK
Department_name	VARCHAR (25)



Run Cancel Disconnect Change Database: Select Database

```
1 use sait;  
2  
3 select * from department;  
4  
5  
6  
7  
8  
9
```

Results Messages

	Department_id	Department_name
1	1	Computer Department
2	2	Business Department
3	3	Engineering Department
4	4	Mathematics Department
5	5	History Department

4. Classroom Table.

Field Name	Data Type
Classroom_id	INT PK
Classroom_no	VARCHAR(20)
Building_name	VARCHAR(20)

Run Cancel Disconnect Change Database: Select Database

```

1 use sait;
2
3 select * from classroom;
4
5
6
7
8
9

```

Results Messages

	Classroom_id	Classroom_no	Building_name
1	1	C-101	Science Building
2	2	B-201	Business Building

5. Faculty Table

Field Name	Data Type
Faculty_id	INT PK
First name	VARCHAR (20)
Lastname	VARCHAR (20)
Phone	VARCHAR(15)
Email	VARCHAR (20)
Department_id	INT FK

Run Cancel Disconnect Change Database: Select Database

```

1 use sait;
2
3 select * from faculty;
4
5
6
7
8
9

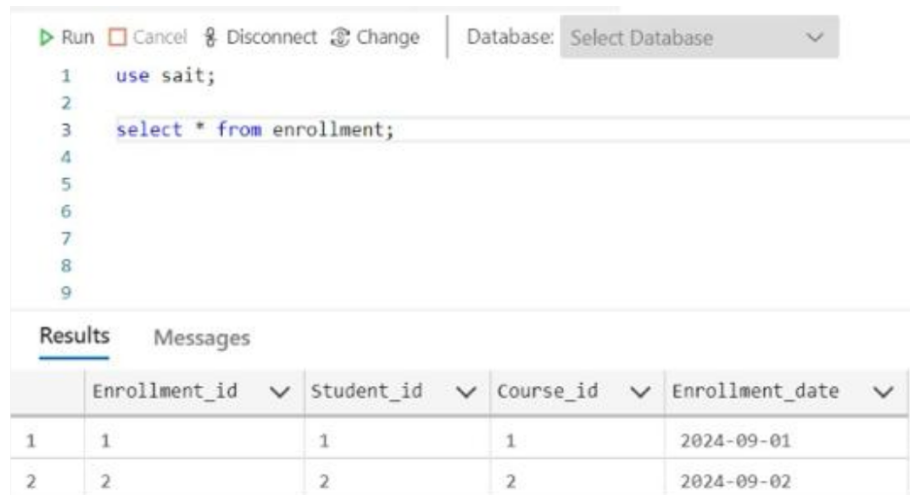
```

Results Messages

	Faculty_id	Firstname	Lastname	Phone	Email	Department_id
1	1	Michael	Jordan	555-987-6543	michael@example.com	1
2	2	Sarah	Connor	555-876-5432	sarah@example.com	3

6. Enrollment Table.

Field Name	Data Type
Enrollment_id	INT PK
Student_id	INT FK
Course_id	INT FK
Enrollment_date	DATE

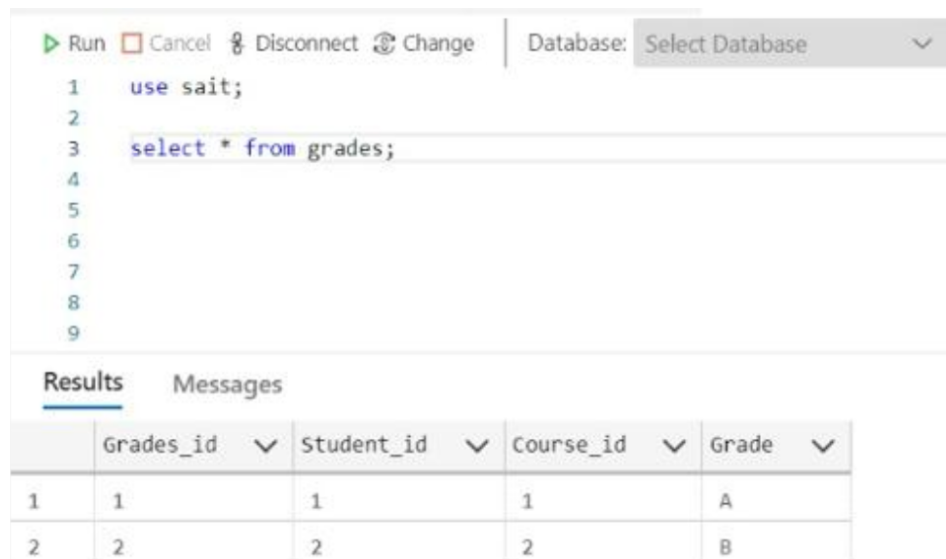


The screenshot shows a SQL query execution window. The query is `use sait;` followed by `select * from enrollment;`. The results are displayed in a table with columns: Enrollment_id, Student_id, Course_id, and Enrollment_date. The data shows two rows: (1, 1, 1, 2024-09-01) and (2, 2, 2, 2024-09-02).

	Enrollment_id	Student_id	Course_id	Enrollment_date
1	1	1	1	2024-09-01
2	2	2	2	2024-09-02

7. Grades Table

Field Name	Data Type
Grades_id	INT PK
Student_id	INT FK
Course_id	INT FK
Grade	VARCHAR (5)



The screenshot shows a SQL query execution window. The query is `use sait;` followed by `select * from grades;`. The results are displayed in a table with columns: Grades_id, Student_id, Course_id, and Grade. The data shows two rows: (1, 1, 1, A) and (2, 2, 2, B).

	Grades_id	Student_id	Course_id	Grade
1	1	1	1	A
2	2	2	2	B

8. Process Table.

Field Name	Data Type
Process_id	INT PK
Status (Completed, Pending)	VARCHAR (50)
Date	DATE
Steps	VARCHAR (200)

- Join Query.

The Enrollment table is the main one used in this query. It connects to the student table through Student_id to get student information, and to the Courses table using Course_id to retrieve course details. Additionally, the Grades table is joined using both Student_id and Course_id to get each student's grade for their courses. Finally, the ORDER BY clause sorts the results by student and course.

The screenshot shows a SQL IDE interface with a query editor and a results pane. The query editor contains the following SQL code:

```
1 SELECT
2     Student.Student_id,
3     Student.Firstname AS Student_Firstname,
4     Student.Lastname AS Student_Lastname,
5     Student.Email AS Student_Email,
6     Courses.Course_id,
7     Courses.Course_name,
8     Grades.Grade,
9     Enrollment.Enrollment_date
10 FROM
11     Enrollment
12 INNER JOIN Student ON Enrollment.Student_id = Student.Student_id
13 INNER JOIN Courses ON Enrollment.Course_id = Courses.Course_id
14 INNER JOIN Grades ON Student.Student_id = Grades.Student_id AND Courses.Course_id = Grades.Course_id
15 ORDER BY Student.Student_id, Courses.Course_id;
16
```

The results pane shows the following data:

Results	Messages						
Student_id	Student_Firstname	Student_Lastname	Student_Email	Course_id	Course_name	Grade	Er
1	John	Doe	john@example.com	1	Data Structures	A	2
2	Jane	Smith	jane@example.com	2	Financial Management	B	2

- View Query.

This view is created using an inner join query, which helps make data retrieval easier, saves time and effort, and ensures that the same logic and relationships between tables are maintained every time the view is applied.

