**Project Report**

**Fundamentals of database systems**

**Project: College database management system**

**Project overview:**

We're building a database for a college that includes coursework in relational database design, development, and maintenance, which includes information about students, courses, faculty, location, and events, as well as a better source of data that aids in understanding the information in a more straightforward manner.

It organizes, centralizes, and simplifies the management system in basic terms. With a single database system and department integration, information is flowing effortlessly. It gives decision-makers and administrative employees access to data 24 hours a day, 7 days a week, allowing them to monitor, control, and aid in record management.

**ADO Dot Net overview:**

ADO.NET ensures that data sources such as SQL Server are accessed consistently. ADO.NET can connect to these data sources and retrieve, handle, and update the data they contain in data-sharing consumer applications.

ADO.NET is a Microsoft.NET Framework data access solution that uses a common set of components to communicate between relational and non-relational systems.

ADO.NET is a set of programming components that allow programmers to retrieve data and data services from a database. It's a part of the Microsoft.NET Framework's base class library. It is most typically used by programmers to access and alter data in relational database systems, but it may also retrieve data from non-relational data sources.

ADO.NET is frequently referred to as an evolution of the ActiveX Data Objects (ADO) technology, although it has undergone so many changes that it is now regarded a whole separate product.

Graphical user interface

Description automatically generated with low confidence**Structure of the database (table):**

Entities/Tables of our project:

* Students
* Courses
* Faculty
* Sections
* Administration
* Location
* Office hours
* Bus
* Events
* Examination
* Department

**Functionality of database:**

Basically, the format and look and feel of the database interface is going to be changed in terms of naïve users understanding. This is the functionality we are going to achieve; additionally, adding new attributes we are providing new information that might help to the users.

**Target user / audience description for your database**

Students, Faculty, Administration, Department, Event Coordinators, Bus Coordinators

**ER diagram**

**Diagram

Description automatically generated**

**Database design process :**

**Structure of relevant tables**

A database table consists of

* Rows 🡪 record/tuples
* Columns 🡪 fields/attributes
* Record: Consistent way to combine information about something
* Field: Single item of information, that appears in every record

Integrity constraints:

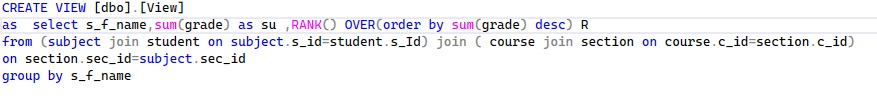
* Primary key
* Foreign key
* Candidate key
* Super Key

A database schema consists of

* Joins
* Triggers
* Views

**Source code / SQL code**

**Views**

****

**Text

Description automatically generated**

**Output**

**Text

Description automatically generated**

**Triggers**

**Graphical user interface, text

Description automatically generated**

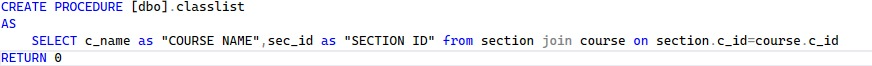
**Graphical user interface, text, application

Description automatically generated**

**Output**

****

**Stored Procedures**

****

**Graphical user interface, text

Description automatically generated with medium confidence**

Output

Text

Description automatically generated

**Functions**

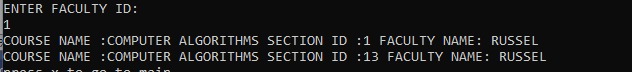
**Text

Description automatically generated with medium confidence**

**Graphical user interface, text, application

Description automatically generated**

**Output**

****

**Index**

****

**SQL table descriptions**

Here we are describing the tables for our database. Every table in the database is described with the values and the keys. We will use SQL command DESCRIBE which tells about specific data of the table in the database.

Syntax: MySQL: describe table\_name

The tables and their description of our project is as follows:

|  |  |
| --- | --- |
| **Describe student** | **CREATE TABLE [dbo].[student] (**  **[s\_Id] INT NOT NULL,**  **[s\_f\_name] VARCHAR (45) NOT NULL,**  **[s\_l\_name] VARCHAR (45) NULL,**  **[s\_address] VARCHAR (100) NOT NULL,**  **[batch] INT NOT NULL,**  **[department\_name] VARCHAR (30) NULL,**  **PRIMARY KEY CLUSTERED ([s\_Id] ASC),**  **FOREIGN KEY ([department\_name]) REFERENCES [dbo].[department] ([dept\_name])**  **);** |
| **Describe department** | **CREATE TABLE [dbo].[department] (**  **[dept\_id] INT NOT NULL,**  **[dept\_name] VARCHAR (30) NOT NULL,**  **[dept\_info] VARCHAR (1000) NOT NULL,**  **PRIMARY KEY CLUSTERED ([dept\_id] ASC),**  **UNIQUE NONCLUSTERED ([dept\_name] ASC),**  **UNIQUE NONCLUSTERED ([dept\_name] ASC)**  **);** |
| **Describe section** | **CREATE TABLE [dbo].[section] (**  **[sec\_id] VARCHAR (5) NOT NULL,**  **[c\_id] INT NOT NULL,**  **[f\_id] INT NOT NULL,**  **[sec\_day] VARCHAR (10) NULL,**  **[sec\_time] DATETIME NULL,**  **PRIMARY KEY CLUSTERED ([sec\_id] ASC),**  **FOREIGN KEY ([c\_id]) REFERENCES [dbo].[course] ([c\_id]),**  **FOREIGN KEY ([f\_id]) REFERENCES [dbo].[faculty] ([f\_id]),**  **CHECK ([sec\_day]='SATURDAY' OR [sec\_day]='FRIDAY' OR [sec\_day]='THURSDAY' OR [sec\_day]='WEDNESDAY' OR [sec\_day]='TUESDAY' OR [sec\_day]='MONDAY')**  **);** |
| **Describe faculty** | **CREATE TABLE [dbo].[faculty] (**  **[f\_id] INT NOT NULL,**  **[f\_f\_name] VARCHAR (50) NOT NULL,**  **[f\_l\_name] VARCHAR (50) NULL,**  **[f\_dept] VARCHAR (30) NULL,**  **PRIMARY KEY CLUSTERED ([f\_id] ASC),**  **FOREIGN KEY ([f\_dept]) REFERENCES [dbo].[department] ([dept\_name])**  **);** |
| **Describe course** | **CREATE TABLE [dbo].[course] (**  **[c\_id] INT NOT NULL,**  **[c\_name] VARCHAR (50) NOT NULL,**  **[c\_faculty] INT NOT NULL,**  **[c\_description] VARCHAR (100) NULL,**  **[dept\_name] VARCHAR (30) NULL,**  **[c\_term] VARCHAR (10) NULL,**  **[c\_year] INT NULL,**  **PRIMARY KEY CLUSTERED ([c\_id] ASC),**  **FOREIGN KEY ([c\_faculty]) REFERENCES [dbo].[faculty] ([f\_id]),**  **FOREIGN KEY ([dept\_name]) REFERENCES [dbo].[department] ([dept\_name]),**  **CHECK ([c\_term] = 'FALL'**  **OR [c\_term] = 'SUMMER'**  **OR [c\_term] = 'SPRING')**  **);** |
| **Describe job\_role** | **CREATE TABLE [dbo].[jobrole] (**  **[j\_id] INT NOT NULL,**  **[j\_catergory] VARCHAR (30) NULL,**  **[j\_sal] FLOAT (53) NULL,**  **[j\_hrs] INT NULL,**  **[j\_wsa] VARCHAR (5) NULL,**  **[sec\_id] VARCHAR (5) NULL,**  **[s\_id] INT NULL,**  **PRIMARY KEY CLUSTERED ([j\_id] ASC),**  **FOREIGN KEY ([sec\_id]) REFERENCES [dbo].[section] ([sec\_id]),**  **FOREIGN KEY ([s\_id]) REFERENCES [dbo].[student] ([s\_Id]),**  **CHECK ([j\_catergory]='PART TIME' OR [j\_catergory]='GRADUATE ASSISTANT' OR [j\_catergory]='GRADER' OR [j\_catergory]='RESEARCH ASSISTANT' OR [j\_catergory]='TEACHING ASSISTANT'),**  **CHECK ([j\_wsa]='FALSE' OR [j\_wsa]='TRUE')**  **);** |
| **Describe office\_hours** | **CREATE TABLE [dbo].[office\_hours] (**  **[oh\_id] INT NOT NULL,**  **[f\_id] INT NULL,**  **[job\_id] INT NULL,**  **[oh\_time] DATETIME NULL,**  **[oh\_loc] INT NULL,**  **PRIMARY KEY CLUSTERED ([oh\_id] ASC),**  **FOREIGN KEY ([f\_id]) REFERENCES [dbo].[faculty] ([f\_id]),**  **FOREIGN KEY ([job\_id]) REFERENCES [dbo].[jobrole] ([j\_id]),**  **FOREIGN KEY ([oh\_loc]) REFERENCES [dbo].[location] ([l\_id])**  **);** |
| **Describe bus** | **CREATE TABLE [dbo].[bus] (**  **[b\_num] INT NOT NULL,**  **[b\_route] VARCHAR (50) NULL,**  **PRIMARY KEY CLUSTERED ([b\_num] ASC)**  **);** |
| **Describe administration** | **CREATE TABLE [dbo].[administration] (**  **[c\_id] INT NULL,**  **[s\_id] INT NULL,**  **FOREIGN KEY ([c\_id]) REFERENCES [dbo].[course] ([c\_id]),**  **FOREIGN KEY ([s\_id]) REFERENCES [dbo].[student] ([s\_Id])**  **);** |
| **Describe subject** | **CREATE TABLE [dbo].[subject] (**  **[sub\_id] INT IDENTITY (1, 1) NOT NULL,**  **[s\_id] INT NOT NULL,**  **[sec\_id] VARCHAR (5) NOT NULL,**  **[grade] INT NULL,**  **PRIMARY KEY CLUSTERED ([sub\_id] ASC),**  **FOREIGN KEY ([s\_id]) REFERENCES [dbo].[student] ([s\_Id]),**  **FOREIGN KEY ([sec\_id]) REFERENCES [dbo].[section] ([sec\_id]),**  **CHECK ([grade]='9' OR [grade]='8' OR [grade]='7' OR [grade]='6' OR [grade]='5' OR [grade]='4' OR [grade]='3' OR [grade]='2' OR [grade]='1')**  **);** |
| **Describe location** | **CREATE TABLE [dbo].[location] (**  **[l\_id] INT NOT NULL,**  **[l\_name] VARCHAR (50) NOT NULL,**  **[sec\_id] VARCHAR (5) NULL,**  **[bus\_number] INT NULL,**  **PRIMARY KEY CLUSTERED ([l\_id] ASC),**  **FOREIGN KEY ([sec\_id]) REFERENCES [dbo].[section] ([sec\_id]),**  **FOREIGN KEY ([bus\_number]) REFERENCES [dbo].[bus] ([b\_num])**  **);** |
| **Describe event** | **CREATE TABLE [dbo].[event] (**  **[e\_id] INT NOT NULL,**  **[e\_catergory] VARCHAR (15) NULL,**  **[e\_description] VARCHAR (100) NULL,**  **[e\_time] DATETIME NULL,**  **[e\_duration] INT NULL,**  **[e\_location] INT NULL,**  **PRIMARY KEY CLUSTERED ([e\_id] ASC),**  **FOREIGN KEY ([e\_location]) REFERENCES [dbo].[location] ([l\_id])**  **);** |

**Screenshot:**

**Graphical user interface, application, Teams

Description automatically generated**

**Graphical user interface, application, Teams

Description automatically generated**

**Screenshots from sample queries and output from the database**

**Views**

**Text

Description automatically generated**

**Triggers**

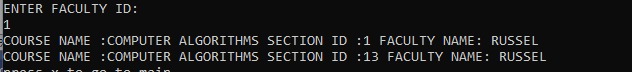
****

**Stored Procedures**

Text

Description automatically generated

**Functions**

****

**Extra things =>**

Relationships between tables:

|  |  |
| --- | --- |
| **Dependent tables** | **Main tables** |
| Sections, Location | Courses |
| Office hours | Faculty |
| Administration, section, office hours | Student |
| Subject | section |
| Bus | location |

**Create Objectives for your project by asking questions:**

Are you going to make an existing system more convenient?

Yes, we believe that we are making a convenient system with better interface. As we are using ADO Dot Net tool referred as ActiveX Data Objects (ADO) technology helped us writing the views for the tables we expected for the interface in naïve terms and we included all the basic modules that a college database must include that is student information, faculty information, course information, bus information, events information and admin who will manage all these roles.

Are you going to be providing new services?

Yes, we are providing new service in terms of additional information for the existing modules. For example,

Instead of searching the course by Course\_ID, we can simply have entire course details available for the term chooses. Then add the course if willing to study that course.