



AMERICAN INTERNATIONAL UNIVERSITY–BANGLADESH (AIUB)

**Dept. of Computer Science
Faculty of Science and Technology**

CSC2210: OBJECT ORIENTED PROGRAMMING 2

Spring 2024-2025

Section: [W]

Group No: 00

Project Report On

Project Name [Learning Management System]

Supervised By

Syeda Ashrafa Islam

Submitted By:

Name	ID
1. Hasibur Rahman Rahim	22-47044-1
2. Ahnaf Hasan	23-53376-3
3. MD Fatin Hasnat Patwary	23-53353-3
4. Tarikul Islam Joy	23-53237-3

CO2: Display and verify the mean of a real-life Project using the concepts of C# Graphical User Interface based environment with database integration to depict a desktop-based application.

Assessment Criteria	Not Attended/ Incorrect (0)	Inadequate (1-2)	Average (3)	Good (4)	Excellent (5)
Evaluation Criteria	Evaluation Definition				Total =
Requirement fulfillment	Properly demonstrate a real-life scenario-based project with proper functional requirement identification for the Object-Oriented Programming project development activities.				
Validation	Ensuring the ability of students' proper demonstration on validation forms in their system in terms of dealing with the data.				
Verification	Identifying if the students can verify the system data along with proper functional requirements in terms of data flow.				

Table of Contents:	Page no.
1. Chapter :01 (Introduction)-	03
2. Chapter :02(User Story)-	03
3. Chapter :03 (ER Diagram)-	05
4. Chapter :03 (SQL Queries)-	06
5. Chapter:04 (Screenshots) -	09

Introduction

The Learning Management System (LMS) is a comprehensive digital platform designed to streamline the educational experience for administrators, teachers, and students within academic institutions. The LMS project aims to centralize and simplify the management of courses, assignments, grades, feedback, and communication through a secure, role-based web application. It empowers administrators to manage users and content, enables teachers to facilitate learning and assess student performance, and supports students in accessing materials, submitting work, and tracking progress.

The system enforces strict access controls and user-based dashboards to ensure that each user interacts only with the functionalities relevant to their role. Admins have exclusive access to manage users and academic structures such as departments and course assignments. Teachers have full control over their course materials, assignment creation, grading, and student communication. Students benefit from intuitive access to learning resources, assignment uploads, feedback submission, and grade viewing.

This LMS enhances transparency, efficiency, and academic performance tracking through integrated views and reports, a calendar-based deadline system, and two-way communication tools—all supported by a structured relational database with clearly defined entities and relationships.

User Story

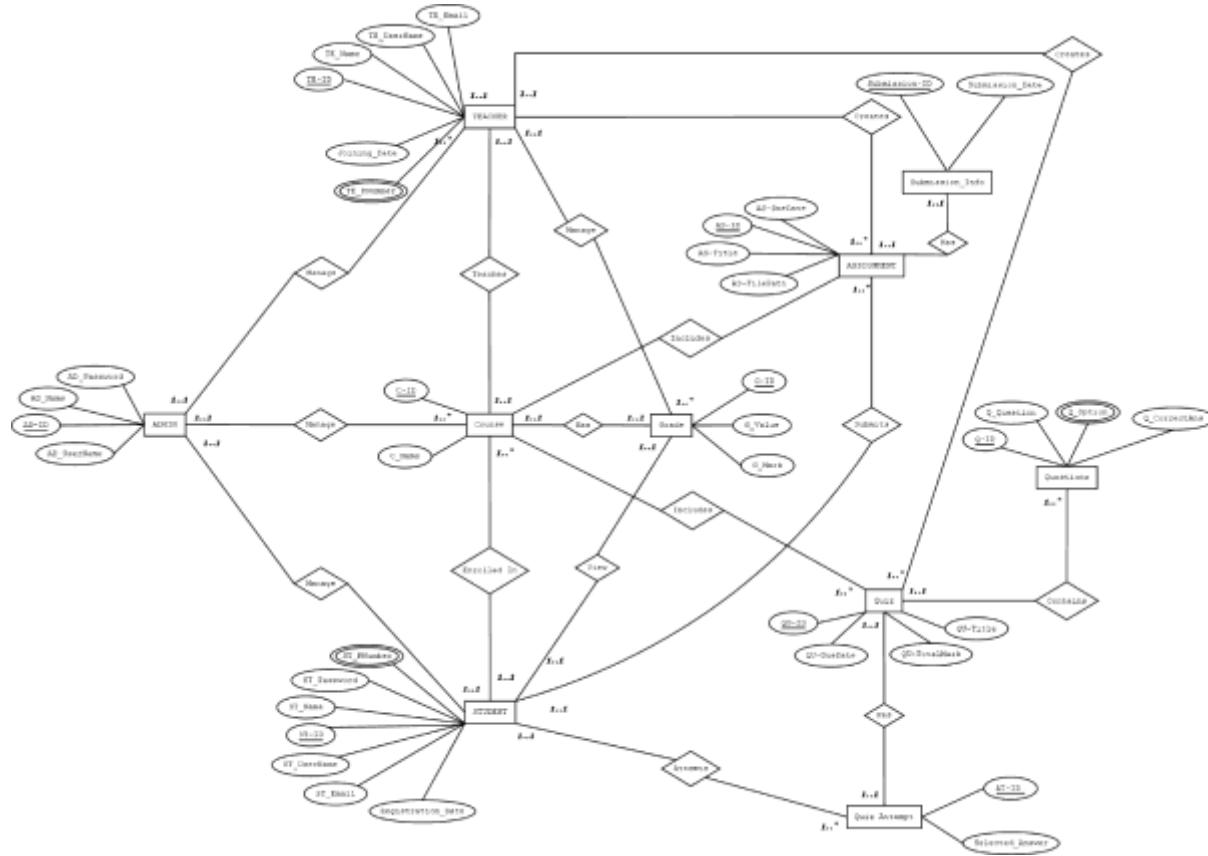
As an **Admin**, I want to create and manage user accounts for teachers and students, assign them to relevant courses, and generate academic reports. I should not access individual student grades or quizzes, keeping the academic evaluation solely in the hands of the teachers.

As a **Teacher**, I want to upload course materials, create and assign homework and quizzes, set deadlines, grade submissions, and communicate with students through a built-in messaging system. I want to manage multiple courses and receive anonymous feedback through the system, which is reviewed and optionally shared by the Admin.

As a **Student**, I want to access course content, download study materials, upload assignments before the deadline, attempt quizzes, and see my grades. I also want to be able to communicate with my teachers, receive notifications, and submit anonymous feedback on the courses and teaching methods.

Each role interacts with a personalized dashboard and operates within their domain of responsibility, making the LMS an effective, role-sensitive educational tool.

ER Diagram



NORMALIZATION

1. Mange: Admin & Teacher

UNF	AD_ID (Primary Key), AD_Name, AD_Username, AD_Password, TE_ID (Primary Key), TE_Name, TE_Username, TE_Password, TE_Email, Joining_Date, TE_PNUMBER
1NF	AD_ID (Primary Key), AD_Name, AD_Username, AD_Password, TE_ID (Primary Key), TE_Name, TE_Username, TE_Password, TE_Email, Joining_Date, TE_PNUMBER
2NF	<ol style="list-style-type: none"> AD_ID (Primary Key), AD_Name, AD_Username, AD_Password, TE_ID (Primary Key), TE_Name, TE_Username, TE_Password, TE_Email, Joining_Date TE_ID (Foreign Key), TE_PNUMBER AD_ID (Foreign Key), TE_ID (Foreign Key),
3NF	<ol style="list-style-type: none"> AD_ID (Primary Key), AD_Name, AD_Username, AD_Password, TE_ID (Primary Key), TE_Name, TE_Username, TE_Password, TE_Email, Joining_Date TE_ID (Foreign Key), TE_PNUMBER AD_ID (Foreign Key), TE_ID (Foreign Key),

2. Mange: Admin & Student

UNF	AD_ID (Primary Key), AD_Name, AD_Username, AD_Password, ST_ID (Primary Key), ST_Name, ST_Username, ST_Password, ST_Email, Registration_Date, ST_PNUMBER
1NF	AD_ID (Primary Key), AD_Name, AD_Username, AD_Password, ST_ID (Primary Key), ST_Name, ST_Username, ST_Password, ST_Email, Registration_Date, ST_PNUMBER
2NF	<ol style="list-style-type: none"> 1. AD_ID (Primary Key), AD_Name, AD_Username, AD_Password 2. ST_ID (Primary Key), ST_Name, ST_Username, ST_Password, ST_Email, Registration_Date, 3. ST_ID (Primary Key), ST_PNUMBER 4. AD_ID (Foreign Key), ST_ID (Foreign Key),
3NF	<ol style="list-style-type: none"> 1. AD_ID (Primary Key), AD_Name, AD_Username, AD_Password 2. ST_ID (Primary Key), ST_Name, ST_Username, ST_Password, ST_Email, Registration_Date, 3. ST_ID (Primary Key), ST_PNUMBER 4. AD_ID (Foreign Key), ST_ID (Foreign Key)

3. Mange: Admin & Course

UNF	AD_ID (Primary Key), AD_Name, AD_Username, AD_Password, C_ID (Primary Key), C_Name
1NF	AD_ID (Primary Key), AD_Name, AD_Username, AD_Password, C_ID (Primary Key), C_Name
2NF	<ol style="list-style-type: none"> 1. AD_ID (Primary Key), AD_Name, AD_Username, AD_Password. 2. C_ID (Primary Key), C_Name. 3. AD_ID (Foreign Key), C_ID (Foreign Key),
3NF	<ol style="list-style-type: none"> 1. AD_ID (Primary Key), AD_Name, AD_Username, AD_Password. 2. C_ID (Primary Key), C_Name. 3. AD_ID (Foreign Key), C_ID (Foreign Key),

4. Enrolled in: Student & Course

UNF	ST_ID (Primary Key), ST_Name, ST_Username, ST_Password, ST_Email, Registration_Date, ST_PNUMBER, C_ID (Primary Key), C_Name
1NF	ST_ID (Primary Key), ST_Name, ST_Username, ST_Password, ST_Email, Registration_Date, ST_PNUMBER, C_ID (Primary Key), C_Name
2NF	<ol style="list-style-type: none"> 1. ST_ID (Primary Key), ST_Name, ST_Username, ST_Password, ST_Email, Registration_Date, 2. C_ID (Primary Key), C_Name. 3. ST_ID (Foreign Key), ST_PNUMBER 4. ST_ID (Foreign Key), C_ID (Foreign Key)
3NF	<ol style="list-style-type: none"> 1. ST_ID (Primary Key), ST_Name, ST_Username, ST_Password, ST_Email, Registration_Date, 2. C_ID (Primary Key), C_Name. 3. ST_ID (Foreign Key), ST_PNUMBER 4. ST_ID (Foreign Key), C_ID (Foreign Key)

5. Teaches: Teacher & Course

UNF	TE_ID (Primary Key), TE_Name, TE_Username, TE_Password, TE_Email, Joining_Date, TE_PNUMBER, C_ID (Primary Key), C_Name
1NF	TE_ID (Primary Key), TE_Name, TE_Username, TE_Password, TE_Email, Joining_Date, TE_PNUMBER, C_ID (Primary Key), C_Name
2NF	<ol style="list-style-type: none"> 1. TE_ID (Primary Key), TE_Name, TE_Username, TE_Password, TE_Email, Joining_Date 2. C_ID (Primary Key), C_Name. 3. TE_ID (Foreign Key), TE_PNUMBER 4. TE_ID (Foreign Key), C_ID (Foreign Key)
3NF	<ol style="list-style-type: none"> 1. TE_ID (Primary Key), TE_Name, TE_Username, TE_Password, TE_Email, Joining_Date 2. C_ID (Primary Key), C_Name. 3. TE_ID (Foreign Key), TE_PNUMBER 4. TE_ID (Foreign Key), C_ID (Foreign Key)

6. Has: Grade & Course

UNF	G_ID, G_Value, G_Mark, C_ID (Primary Key), C_Name
1NF	G_ID, G_Value, G_Mark, C_ID (Primary Key), C_Name
2NF	<ol style="list-style-type: none"> 1. G_ID (Primary Key), G_Value, G_Mark 2. C_ID (Primary Key), C_Name. 3. G_ID (Foreign Key), C_ID (Foreign Key)
3NF	<ol style="list-style-type: none"> 1. G_ID(Primary Key), G_Value, G_Mark 2. C_ID (Primary Key), C_Name. 3. G_ID (Foreign Key), C_ID (Foreign Key)

7. Manage: Grade & Teacher

UNF	G_ID (Primary Key), G_Value, G_Mark, TE_ID (Primary Key), TE_Name, TE_Username, TE_Password, TE_Email, Joining_Date, TE_PNUMBER,
1NF	G_ID (Primary Key), G_Value, G_Mark, TE_ID (Primary Key), TE_Name, TE_Username, TE_Password, TE_Email, Joining_Date, TE_PNUMBER,
2NF	<ol style="list-style-type: none"> 1. G_ID (Primary Key), G_Value, G_Mark 2. TE_ID (Primary Key), TE_Name, TE_Username, TE_Password, TE_Email, Joining_Date 3. TE_ID (Foreign Key), TE_PNUMBER 4. G_ID (Foreign Key), TE_ID (Foreign Key)
3NF	<ol style="list-style-type: none"> 1. G_ID (Primary Key), G_Value, G_Mark 2. TE_ID (Primary Key), TE_Name, TE_Username, TE_Password, TE_Email, Joining_Date 3. TE_ID (Foreign Key), TE_PNUMBER 4. G_ID (Foreign Key), TE_ID (Foreign Key)

8. View: Student & Grade

UNF	ST_ID (Primary Key), ST_Name, ST_Username, ST_Password, ST_Email, Registration_Date, , ST_PNUMBER, G_ID (Primary Key), G_Value, G_Mark
1NF	ST_ID (Primary Key), ST_Name, ST_Username, ST_Password, ST_Email, Registration_Date, , ST_PNUMBER, G_ID (Primary Key), G_Value, G_Mark
2NF	<ol style="list-style-type: none"> 1. ST_ID (Primary Key), ST_Name, ST_Username, ST_Password, ST_Email, Registration_Date, 2. G_ID (Primary Key), G_Value, G_Mark 3. ST_ID (Foreign Key), TE_PNUMBER 4. G_ID (Foreign Key), ST_ID (Foreign Key)
3NF	<ol style="list-style-type: none"> 1. ST_ID (Primary Key), ST_Name, ST_Username, ST_Password, ST_Email, Registration_Date, 2. G_ID (Primary Key), G_Value, G_Mark 3. ST_ID (Foreign Key), TE_PNUMBER 4. G_ID (Foreign Key), ST_ID (Foreign Key)

9. include: Assignment & Course

UNF	AS_ID (Primary Key), AS_Title, AS_Description, AS_DueDate, AS_FilePath, C_ID (Primary Key), C_Name
1NF	AS_ID (Primary Key), AS_Title, AS_Description, AS_DueDate, AS_FilePath, C_ID (Primary Key), C_Name
2NF	<ol style="list-style-type: none"> 1. AS_ID (Primary Key), AS_Title, AS_DueDate, AS_FilePath. 2. C_ID (Primary Key), C_Name 3. AS_ID (Foreign Key), C_ID (Foreign Key)
3NF	<ol style="list-style-type: none"> 1. AS_ID (Primary Key), AS_Title, AS_DueDate, AS_FilePath. 2. C_ID (Primary Key), C_Name 3. AS_ID (Foreign Key), C_ID (Foreign Key)

10. has: Assignment & Submission_Info

UNF	AS_ID (Primary Key), AS_Title, AS_Description, AS_DueDate, AS_FilePath, Submission_ID (Primary Key), Submission_Date.
1NF	AS_ID (Primary Key), AS_Title, AS_Description, AS_DueDate, AS_FilePath, Submission_ID (Primary Key), Submission_Date.
2NF	<ol style="list-style-type: none"> 1. AS_ID (Primary Key), AS_Title, AS_DueDate, AS_FilePath. 2. Submission_ID (Primary Key), Submission_Date. 3. AS_ID (Foreign Key), Submission_ID (Foreign Key)
3NF	<ol style="list-style-type: none"> 1. AS_ID (Primary Key), AS_Title, AS_DueDate, AS_FilePath. 2. Submission_ID (Primary Key), Submission_Date.

	3. AS_ID (Foreign Key), Submission_ID (Foreign Key)
--	---

11. Manage: Assignment & Teacher

UNF	AS_ID , AS_Title, AS_Description, AS_DueDate, AS_FilePath, TE_ID , TE_Name, TE_Username, TE_Password, TE_Email, Joining_Date, TE_PNUMBER,
1NF	AS_ID , AS_Title, AS_Description, AS_DueDate, AS_FilePath, TE_ID , TE_Name, TE_Username, TE_Password, TE_Email, Joining_Date, TE_PNUMBER,
2NF	<ol style="list-style-type: none"> 1. AS_ID (Primary Key), AS_Title, AS_DueDate, AS_FilePath. 2. TE_ID (Primary Key), TE_Name, TE_Username, TE_Password, TE_Email, Joining_Date 3. TE_ID (Foreign Key), TE_PNUMBER 4. AS_ID (Foreign Key), TE_ID (Foreign Key)
3NF	<ol style="list-style-type: none"> 1. AS_ID (Primary Key), AS_Title, AS_DueDate, AS_FilePath. 2. TE_ID (Primary Key), TE_Name, TE_Username, TE_Password, TE_Email, Joining_Date 3. TE_ID (Foreign Key), TE_PNUMBER 4. AS_ID (Foreign Key), TE_ID (Foreign Key)

12. Submits: Assignment & Student

UNF	AS_ID ,AS_Title, AS_Description, AS_DueDate, AS_FilePath, ST_ID, ST_Name, ST_Username, ST_Password, ST_Email, Registration_Date, ST_PNUMBER
1NF	AS_ID ,AS_Title, AS_Description, AS_DueDate, AS_FilePath, ST_ID, ST_Name, ST_Username, ST_Password, ST_Email, Registration_Date, ST_PNUMBER
2NF	<ol style="list-style-type: none"> 1. AS_ID (Primary Key), AS_Title, AS_DueDate, AS_FilePath. 2. ST_ID (Primary Key), ST_Name, ST_Username, ST_Password, ST_Email, Registration_Date. 3. ST_ID (Foreign Key), ST_PNUMBER 4. AS_ID (Foreign Key), TE_ID (Foreign Key)
3NF	<ol style="list-style-type: none"> 1. AS_ID (Primary Key), AS_Title, AS_DueDate, AS_FilePath. 2. ST_ID (Primary Key), ST_Name, ST_Username, ST_Password, ST_Email, Registration_Date 3. ST_ID (Foreign Key), ST_PNUMBER 4. AS_ID (Foreign Key), ST_ID (Foreign Key)

13. Includes: Course & Quiz

UNF	C_ID , C_Name QU_ID, QU_DueDate, QU_TotalMark, QU_Title.
1NF	C_ID (Primary Key), C_Name ,QU_ID, QU_DueDate, QU_TotalMark, QU_Title.

2NF	<ol style="list-style-type: none"> 1. C_ID (Primary Key), C_Name 2. QU_ID (Primary Key), QU_DueDate, QU_TotalMark, QU_Title. 3. C_ID (Foreign Key), QU_ID (Foreign Key).
3NF	<ol style="list-style-type: none"> 1. QU_ID (Primary Key), QU_DueDate, QU_TotalMark, QU_Title. 2. TE_ID (Primary Key), TE_Name, TE_Username, TE_Password, TE_Email, Joining_Date. 3. TE_ID (Foreign Key), TE_PNUMBER. 4. QU_ID (Foreign Key), TE_ID (Foreign Key).

14. Creates : Quiz & Teacher

UNF	QU_ID, QU_DueDate, QU_TotalMark, QU_Title, TE_ID (Primary Key), TE_Name, TE_Username, TE_Password, TE_Email, Joining_Date, TE_PNUMBER
1NF	QU_ID, QU_DueDate, QU_TotalMark, QU_Title, TE_ID (Primary Key), TE_Name, TE_Username, TE_Password, TE_Email, Joining_Date, TE_PNUMBER
2NF	<ol style="list-style-type: none"> 4. QU_ID (Primary Key), QU_DueDate, QU_TotalMark, QU_Title. 5. TE_ID (Primary Key), TE_Name, TE_Username, TE_Password, TE_Email, Joining_Date. 6. TE_ID (Foreign Key), TE_PNUMBER. 7. QU_ID (Foreign Key), TE_ID (Foreign Key).
3NF	<ol style="list-style-type: none"> 5. QU_ID (Primary Key), QU_DueDate, QU_TotalMark, QU_Title. 6. TE_ID (Primary Key), TE_Name, TE_Username, TE_Password, TE_Email, Joining_Date. 7. TE_ID (Foreign Key), TE_PNUMBER. 8. QU_ID (Foreign Key), TE_ID (Foreign Key).

15. Attempt: Quiz attempt & Student

UNF	AT-ID , Selected_Answer, ST_ID , ST_Name, ST_Username, ST_Password, ST_Email, Registration_Date
1NF	AT-ID (Primary Key), Selected_Answer, ST_ID (Primary Key), ST_Name, ST_Username, ST_Password, ST_Email, Registration_Date
2NF	<ol style="list-style-type: none"> 1. AT-ID (Primary Key), Selected_Answer 2. ST_ID (Primary Key), ST_Name, ST_Username, ST_Password, ST_Email, Registration_Date 3. ST_ID (Foreign Key), ST_PNUMBER. 4. AT_ID (Foreign Key), ST_ID (Foreign Key).
3NF	<ol style="list-style-type: none"> 1. AT-ID (Primary Key), Selected_Answer 2. ST_ID (Primary Key), ST_Name, ST_Username, ST_Password, ST_Email, Registration_Date 3. ST_ID (Foreign Key), ST_PNUMBER. 4. AT_ID (Foreign Key), ST_ID (Foreign Key).

16. Attempt: Quiz attempt & Quiz

UNF	AT-ID , Selected_Answer, QU_ID, QU_DueDate, QU_TotalMark, QU_Title
1NF	AT-ID (Primary Key), Selected_Answer, QU_ID (Primary Key), QU_DueDate, QU_TotalMark, QU_Title
2NF	<ol style="list-style-type: none"> 1. AT-ID (Primary Key), Selected_Answer 2. QU_ID (Primary Key), QU_DueDate, QU_TotalMark, QU_Title 3. AT_ID (Foreign Key), QU_ID (Foreign Key).
3NF	<ol style="list-style-type: none"> 1. AT-ID (Primary Key), Selected_Answer 2. QU_ID (Primary Key), QU_DueDate, QU_TotalMark, QU_Title 3. AT_ID (Foreign Key), QU_ID (Foreign Key).

17. Has: Quiz & Questions

UNF	QU_ID (Primary Key), QU_DueDate, QU_TotalMark, QU_Title, Q_ID (Primary Key), Q_Question, Q_CorrectAns, Q_Option.
1NF	QU_ID (Primary Key), QU_DueDate, QU_TotalMark, QU_Title, Q_ID (Primary Key), Q_Question, Q_Option.
2NF	<ol style="list-style-type: none"> 1. QU_ID (Primary Key), QU_DueDate, QU_TotalMark, QU_Title. 2. Q_ID (Primary Key), Q_Question, Q_CorrectAns. 3. Q_ID (Foreign Key), Q_Option. 4. QU_ID (Foreign Key), Q_ID (Foreign Key).
3NF	<ol style="list-style-type: none"> 1. QU_ID (Primary Key), QU_DueDate, QU_TotalMark, QU_Title. 2. Q_ID (Primary Key), Q_Question, Q_CorrectAns. 3. Q_ID (Foreign Key), Q_Option. 4. QU_ID (Foreign Key), Q_ID (Foreign Key).

SQL Queries

CREATE:

```
CREATE TABLE LMS_Teachers (
    TE_ID INT PRIMARY KEY IDENTITY(1,1),
    TE_Name NVARCHAR(100) NOT NULL,
    TE_Username NVARCHAR(100) NOT NULL UNIQUE,
    TE_Password NVARCHAR(100) NOT NULL,
    TE_Email NVARCHAR(100) NOT NULL UNIQUE,
    Joining_Date DATE NOT NULL
);
```

```
CREATE TABLE LMS_TeacherPhone (
    TE_ID INT, FOREIGN KEY (TE_ID) REFERENCES LMS_Teachers(TE_ID) ON
    DELETE CASCADE,
    PhoneNumber NVARCHAR(20),
```

```

PRIMARY KEY (TE_ID, PhoneNumber)
);

CREATE TABLE LMS_Students (
    ST_ID INT PRIMARY KEY IDENTITY(1,1),
    ST_Name NVARCHAR(100) NOT NULL,
    ST_Username NVARCHAR(100) NOT NULL UNIQUE,
    ST_Password NVARCHAR(100) NOT NULL,
    ST_Email NVARCHAR(100) NOT NULL UNIQUE,
    Registration_Date DATE NOT NULL
);

CREATE TABLE LMS_StudentPhone (
    ST_ID INT, FOREIGN KEY (ST_ID) REFERENCES LMS_Students(ST_ID) ON
    DELETE CASCADE,
    PhoneNumber NVARCHAR(20),
    PRIMARY KEY (ST_ID, PhoneNumber),
);
;

CREATE TABLE LMS_Courses (
    C_ID INT PRIMARY KEY IDENTITY(1000,1),
    C_Name NVARCHAR(100) NOT NULL
);

SELECT
    IDENT_CURRENT('LMS_Courses') AS Current_Identity,
    IDENT_INCR('LMS_Courses') AS Identity_Increment,
    IDENT_CURRENT('LMS_Courses') + IDENT_INCR('LMS_Courses') AS Next_Identity;

CREATE TABLE LMS_Assignment (
    AS_ID INT PRIMARY KEY IDENTITY(1,1),
    AS_Name VARCHAR(255),
    AS_Description TEXT,
    AS_FilePath VARBINARY(MAX),
    AS_DueDate DATE,
    AS_TotalMarks DECIMAL(5,2)
);
drop table LMS_Assignment

CREATE TABLE LMS_Submission_Info (
    Submission_ID INT PRIMARY KEY IDENTITY(1,1),
    Submission_Timestamp DATETIME,
    Submission_FilePath VARBINARY(MAX),
    Submission_Grade DECIMAL(5,2)
);
drop table LMS_Submission_Info

```

```

CREATE TABLE LMS_Quiz (
    QU_ID INT PRIMARY KEY IDENTITY(1,1),
    QU_Name VARCHAR(255),
    QU_DueDate DATE,
    QU_TotalMarks DECIMAL(5,2)
);
drop table LMS_Quiz

CREATE TABLE LMS_Quiz_Attempt (
    AT_ID INT PRIMARY KEY IDENTITY(1,1),
    AT_SelectedAns VARCHAR(255),
);
drop table LMS_Quiz_Attempt

CREATE TABLE LMS_Questions (
    Q_ID INT PRIMARY KEY IDENTITY(1,1),
    Q_Question TEXT,
    Q_CorrectAnswer VARCHAR(255),
);
drop table LMS_Questions

CREATE TABLE LMS_Feedback (
    FB_ID INT PRIMARY KEY,
    FB_Content TEXT,
);
drop table LMS_Feedback

CREATE TABLE LMS_Grade (
    G_ID INT PRIMARY KEY IDENTITY(1,1),
    G_Value VARCHAR(25),
    G_Marks DECIMAL(5,2),
);
drop table LMS_Grade

CREATE TABLE LMS_Question_Option (
    Q_ID INT,FOREIGN KEY (Q_ID) REFERENCES LMS_Questions(Q_ID)ON DELETE CASCADE,
    Q_Option VARCHAR(255),
    PRIMARY KEY (Q_ID, Q_Option),
);
drop table LMS_Question_Option

CREATE TABLE Student_Course_EnrolledIn (
    ST_ID INT,FOREIGN KEY (ST_ID) REFERENCES LMS_Students(ST_ID) ON DELETE CASCADE,
    C_ID INT,FOREIGN KEY (C_ID) REFERENCES LMS_Courses(C_ID) ON DELETE CASCADE,
    PRIMARY KEY (ST_ID, C_ID),
);
drop table Student_Course_EnrolledIn

CREATE TABLE Teacher_Teaches_Course (

```

```
    TE_ID INT,FOREIGN KEY (TE_ID) REFERENCES LMS_Teachers(TE_ID)ON
    DELETE CASCADE,
    C_ID INT,FOREIGN KEY (C_ID) REFERENCES LMS_Courses(C_ID)ON DELETE
    CASCADE,
    PRIMARY KEY (TE_ID, C_ID),
);
```

```
CREATE TABLE Course_Grade (
    G_ID INT, FOREIGN KEY (G_ID) REFERENCES LMS_Grade(G_ID)ON DELETE
    CASCADE,
    C_ID INT, FOREIGN KEY (C_ID) REFERENCES LMS_Courses(C_ID)ON DELETE
    CASCADE,
    PRIMARY KEY (G_ID, C_ID),
);
drop table Course_Grade
```

```
CREATE TABLE Grade_Manage (
    G_ID INT,FOREIGN KEY (G_ID) REFERENCES LMS_Grade(G_ID)ON DELETE
    CASCADE,
    TE_ID INT,FOREIGN KEY (TE_ID) REFERENCES LMS_Teachers(TE_ID)ON
    DELETE CASCADE,
    PRIMARY KEY (G_ID, TE_ID),
);
drop table Grade_Manage
```

```
CREATE TABLE StudentGrades (
    ST_ID INT,FOREIGN KEY (ST_ID) REFERENCES LMS_Students(ST_ID) ON
    DELETE CASCADE,
    G_ID INT,FOREIGN KEY (G_ID) REFERENCES LMS_Grade(G_ID) ON DELETE
    CASCADE,
    PRIMARY KEY (ST_ID, G_ID)
);
drop table StudentGrades
```

```
CREATE TABLE Course_Assignment (
    AS_ID INT, FOREIGN KEY (AS_ID) REFERENCES LMS_Assignment(AS_ID) ON
    DELETE CASCADE,
    C_ID INT, FOREIGN KEY (C_ID) REFERENCES LMS_Courses(C_ID)ON DELETE
    CASCADE,
    PRIMARY KEY (AS_ID, C_ID),
);
drop table Course_Assignment
```

```
CREATE TABLE Assignment_Submission(
    AS_ID INT, FOREIGN KEY (AS_ID) REFERENCES LMS_Assignment(AS_ID)ON
    DELETE CASCADE,
```

```
Submission_ID INT,FOREIGN KEY (Submission_ID) REFERENCES  
LMS_Submission_Info(Submission_ID)ON DELETE CASCADE,  
    PRIMARY KEY (AS_ID, Submission_ID)  
);  
drop table Assignment_Submission
```

```
CREATE TABLE Assignment_Manage (  
    AS_ID INT,FOREIGN KEY (AS_ID) REFERENCES LMS_Assignment(AS_ID)ON  
DELETE CASCADE,  
    TE_ID INT,FOREIGN KEY (TE_ID) REFERENCES LMS_Teachers(TE_ID)ON  
DELETE CASCADE,  
    PRIMARY KEY (AS_ID, TE_ID),  
);
```

```
drop table Assignment_Manage
```

```
CREATE TABLE Assignment_Submits (  
    AS_ID INT,FOREIGN KEY (AS_ID) REFERENCES LMS_Assignment(AS_ID)ON  
DELETE CASCADE,  
    ST_ID INT,FOREIGN KEY (ST_ID) REFERENCES LMS_Students(ST_ID)ON  
DELETE CASCADE,  
    PRIMARY KEY (AS_ID, ST_ID),  
);
```

```
drop table Assignment_Submits
```

```
CREATE TABLE Student_AttemptQuiz (  
    AT_ID INT,FOREIGN KEY (ST_ID) REFERENCES LMS_Students(ST_ID)ON  
DELETE CASCADE,  
    ST_ID INT,FOREIGN KEY (AT_ID) REFERENCES LMS_Quiz_Attempt(AT_ID)ON  
DELETE CASCADE,  
    PRIMARY KEY (AT_ID, ST_ID)  
);  
drop table Student_AttemptQuiz
```

```
CREATE TABLE Course_Quiz (  
    C_ID INT,FOREIGN KEY (C_ID) REFERENCES LMS_Courses(C_ID)ON DELETE  
CASCADE,  
    QU_ID INT,FOREIGN KEY (QU_ID) REFERENCES LMS_Quiz(QU_ID)ON DELETE  
CASCADE,  
    PRIMARY KEY (C_ID, QU_ID)  
);
```

```
drop table Course_Quiz
```

```
CREATE TABLE QuizAttempt_Quiz (
```

```
    AT_ID INT,FOREIGN KEY (AT_ID) REFERENCES LMS_Quiz_Attempt(AT_ID)ON  
    DELETE CASCADE,  
    QU_ID INT,FOREIGN KEY (QU_ID) REFERENCES LMS_Quiz(QU_ID)ON DELETE  
    CASCADE,  
    PRIMARY KEY (AT_ID, QU_ID)  
);  
drop table QuizAttempt_Quiz
```

```
CREATE TABLE Quiz_Question (  
    QU_ID INT,FOREIGN KEY (QU_ID) REFERENCES LMS_Quiz(QU_ID)ON DELETE  
    CASCADE,  
    Q_ID INT,FOREIGN KEY (Q_ID) REFERENCES LMS_Questions(Q_ID)ON DELETE  
    CASCADE,  
    PRIMARY KEY (QU_ID, Q_ID)  
);  
drop table Quiz_Question
```

```
CREATE TABLE Teacher_Feedback (  
    TE_ID INT,FOREIGN KEY (TE_ID) REFERENCES LMS_Teachers(TE_ID)ON  
    DELETE CASCADE,  
    FB_ID INT,FOREIGN KEY (FB_ID) REFERENCES LMS_Feedback(FB_ID)ON  
    DELETE CASCADE,  
    PRIMARY KEY (TE_ID, FB_ID)  
);
```

```
CREATE TABLE Student_Feedback_GivesFeedback (  
    ST_ID INT,FOREIGN KEY (ST_ID) REFERENCES LMS_Students(ST_ID)ON  
    DELETE CASCADE,  
    FB_ID INT,FOREIGN KEY (FB_ID) REFERENCES LMS_Feedback(FB_ID)ON  
    DELETE CASCADE,  
    PRIMARY KEY (ST_ID, FB_ID)  
);
```

--view tables

```
CREATE VIEW View_Course_Enrollments AS  
SELECT s.ST_Name, s.ST_Email, c.C_Name  
FROM LMS_Students s, LMS_Courses c, LMS_Enrollments e  
WHERE s.ST_ID = e.ST_ID AND c.C_ID = e.C_ID;
```

```
CREATE VIEW View_Teacher_Assignments AS  
SELECT t.TE_Name, a.AS_Title, c.C_Name, a.AS_DueDate  
FROM LMS_Teachers t, LMS_Assignments a, LMS_Courses c  
WHERE t.TE_ID = a.TE_ID AND a.C_ID = c.C_ID;
```

```
CREATE VIEW View_Quiz_Results AS
```

```
SELECT s.ST_Name, q.QU_Title, qa.Selected_Answer  
FROM LMS_Students s, LMS_Qizzes q, LMS_QuizAttempts qa  
WHERE s.ST_ID = qa.ST_ID AND q.QU_ID = qa.QU_ID;
```

```
CREATE VIEW View_Student_Grades AS  
SELECT s.ST_Name, g.G_Value, g.G_Mark, c.C_Name  
FROM LMS_Students s, LMS_StudentGrades sg, LMS_Grades g, LMS_Courses c  
WHERE s.ST_ID = sg.ST_ID AND g.G_ID = sg.G_ID AND g.C_ID = c.C_ID;
```

```
CREATE VIEW View_Upcoming_Assignments AS  
SELECT s.ST_Name, a.AS_Title, a.AS_DueDate, c.C_Name  
FROM LMS_Students s, LMS_Enrollments e, LMS_Assignments a, LMS_Courses c  
WHERE s.ST_ID = e.ST_ID AND e.C_ID = a.C_ID AND c.C_ID = a.C_ID  
AND a.AS_DueDate >= GETDATE();
```

```
CREATE VIEW View_Message_Log AS  
SELECT m.M_Time, m.M_SenderRole, m.M_SenderID, m.M_ReciverRole,  
m.M_ReciverID, m.M_Content  
FROM LMS_Messages m;
```

```
CREATE VIEW View_Feedback_Summary AS  
SELECT  
    f.FB_ID,  
    f.FB_Content,  
    s.ST_Name AS GivenByStudent,  
    t.TE_Name AS AboutTeacher  
FROM  
    LMS_Feedbacks f,  
    LMS_StudentFeedback sf,  
    LMS_Students s,  
    LMS_TeacherFeedback tf,  
    LMS_Teachers t  
WHERE  
    f.FB_ID = sf.FB_ID AND  
    sf.ST_ID = s.ST_ID AND  
    f.FB_ID = tf.FB_ID AND  
    tf.TE_ID = t.TE_ID;
```

```
CREATE VIEW LMS_TeacherFullDetails AS  
SELECT  
    T.TE_ID,  
    T.TE_Name,  
    T.TE_Username,  
    T.TE_Password,  
    T.TE_Email,
```

```

T.Joining_Date,
TP.TE_PNUMBER
FROM
LMS_Teachers T,
LMS_TeacherPhone TP
WHERE
T.TE_ID = TP.TE_ID;

CREATE VIEW LMS_StudentFullDetails AS
SELECT
S.ST_ID,
S.ST_Name,
S.ST_Username,
S.ST_Email,
S.ST_Password,
S.Registration_Date,
SP.ST_PNUMBER
FROM
LMS_Students S,
LMS_StudentPhone SP
WHERE
S.ST_ID = SP.ST_ID;

```

```

CREATE VIEW QuestionDetails AS
SELECT
c.C_ID,
c.C_Name,
q.Q_ID,
cq.QU_ID,
q.Q_Question,
o.Q_Option,
q.Q_CorrectAnswer,
CASE
WHEN o.Q_Option = q.Q_CorrectAnswer THEN 'Yes'
ELSE 'No'
END AS IsCorrect

FROM
LMS_Questions q,
LMS_Question_Option o,
Quiz_Question qq,
LMS_Quiz cq,
Course_QuiZ ca,
LMS_Courses c
WHERE
q.Q_ID = o.Q_ID AND
q.Q_ID = qq.Q_ID AND
qq.QU_ID = cq.QU_ID AND
cq.QU_ID = ca.QU_ID AND

```

```
ca.C_ID = c.C_ID
```

```
drop view QuestionDetails
```

```
select * from QuestionDetails  
CREATE VIEW CourseAssignList AS  
SELECT  
    c.C_ID,  
    t.TE_ID,  
    s.ST_ID,  
    t.TE_Name,  
    c.C_Name,  
    s.ST_Name  
FROM  
    LMS_Courses c,  
    LMS_Teachers t,  
    LMS_Students s,  
    Teacher_Teaches_Course CT,  
    Student_Course_EnrolledIn CS
```

```
WHERE
```

```
    c.C_ID = CT.C_ID AND  
    c.C_ID = CS.C_ID AND  
    t.TE_ID = CT.TE_ID AND  
    s.ST_ID = CS.ST_ID
```

```
drop view CourseAssignList
```

```
select * from LMS_Courses
```

```
    select * from
```

```
DELETE ROW FROM L
```

```
CREATE VIEW AssignmentStudentTeacherView AS
```

```
SELECT  
    a.AS_ID,  
    a.AS_Name AS AS_FileName, -- Assuming AS_Name is the assignment  
    filename  
    si.Submission_Timestamp AS AS_UploadDate, -- Upload timestamp  
    ca.C_ID,  
    t.TE_ID,  
    s.ST_Username  
FROM  
    LMS_Assignment a,  
    LMS_Teachers t,  
    LMS_Students s,  
    Assignment_Manage am,  
    Assignment_Submits ast,  
    Course_Assignment ca,  
    Assignment_Submission asi,  
    LMS_Submission_Info si
```

```

WHERE
    a.AS_ID = am.AS_ID AND
    a.AS_ID = ast.AS_ID AND
    a.AS_ID = ca.AS_ID AND
    t.TE_ID = am.TE_ID AND
    s.ST_ID = ast.ST_ID AND
    asi.AS_ID = a.AS_ID AND
    si.Submission_ID = asi.Submission_ID;

SELECT * FROM LMS_Assignment
;

CREATE VIEW QuizQuestionsWithOptions AS
SELECT
    q.Q_ID,
    q.Q_Question,
    q.Q_CorrectAnswer,
    o.Q_Option
FROM
    LMS_Questions q,
    LMS_Question_Option o
WHERE
    q.Q_ID = o.Q_ID
ORDER BY
    q.Q_ID;

CREATE TABLE Teacher_Quiz (
    TE_ID INT,
    QU_ID INT,
    PRIMARY KEY (TE_ID, QU_ID),
    FOREIGN KEY (TE_ID) REFERENCES LMS_Teachers(TE_ID) ON DELETE CASCADE,
    FOREIGN KEY (QU_ID) REFERENCES LMS_Quiz(QU_ID) ON DELETE CASCADE
);
CREATE VIEW View_TeacherQuiz AS
SELECT
    t.TE_ID,
    t.TE_Name,
    q.QU_ID,
    q.QU_Name
FROM
    LMS_Teachers t,
    LMS_Quiz q,
    Teacher_Quiz tq
WHERE
    t.TE_ID = tq.TE_ID AND
    q.QU_ID = tq.QU_ID;

```

```
CREATE TABLE Student_Quiz (
    ST_ID INT,
    QU_ID INT,
    PRIMARY KEY (ST_ID, QU_ID),
    FOREIGN KEY (ST_ID) REFERENCES LMS_Students(ST_ID) ON DELETE CASCADE,
    FOREIGN KEY (QU_ID) REFERENCES LMS_Quiz(QU_ID) ON DELETE CASCADE
);
```

INSERT:

```
INSERT INTO LMS_Admins (AD_Name, AD_Username, AD_Password)
VALUES
('Md Fatin Hasnat Patwary', 'HBoss01', '23-53353-3'),
('TARIKUL ISLAM JOY', 'joy07', '23-53237-3'),
('Ahnaf Hasan', 'ahnaf4647', '23-53376-3'),
('HASIBUR RAHMAN RAHIM', 'hasibur22', '22-47044-1');
```

```
Select * from LMS_Admins;
```

```
INSERT INTO LMS_Teachers (TE_Name, TE_Username, TE_Password, TE_Email,
Joining_Date)
VALUES
('Rahul Islam', 'rahul_islam', 'teacher01', 'rahul_islam@gmail.com', '2022-02-07'),
('Farhana Yasmin', 'farhana_yasmin', 'teacher02', 'farhana_yasmin@gmail.com', '2023-09-09'),
('Hasib Rahman', 'hasib_rahman', 'teacher03', 'hasib_rahman@gmail.com', '2023-03-06'),
('Sajid Hossain', 'sajid_hossain', 'teacher04', 'sajid_hossain@gmail.com', '2024-08-24'),
('Tanvir Alam', 'tanvir_alam', 'teacher05', 'tanvir_alam@gmail.com', '2024-09-26');
```

```
INSERT INTO LMS_TeacherPhone (TE_ID, PhoneNumber)
VALUES
(1, '01832364516'),
(1, '01656101575'),
(2, '01668545203'),
(2, '01952306737'),
(3, '01884926839'),
(3, '01513649671'),
(4, '01951814396'),
(5, '01421476609');
```

```
Select * from LMS_TeacherFullDetails;
```

```
UPDATE LMS_Teachers
SET TE_Password = 'tanvir05'
WHERE TE_Username = 'tanvir_alam';
```

```
INSERT INTO LMS_Students (ST_Name, ST_Username, ST_Password, ST_Email, Registration_Date)
VALUES
('Hasib Rahman', 'hasib_rahman0', 'hasib001', 'hasib_rahman0@gmail.com', '2023-02-25'),
('Rahul Islam', 'rahul_islam1', 'rahul11', 'rahul_islam1@gmail.com', '2023-03-20'),
('Sabrina Akter', 'sabrina_akter2', 'sabrina33', 'sabrina_akter2@gmail.com', '2023-04-02');
```

```
INSERT INTO Course_Quiz (C_ID, QU_ID)
VALUES (
    (SELECT C_ID FROM LMS_Courses WHERE C_Name = 'Web Development'),
    (SELECT QU_ID FROM LMS_Quiz WHERE QU_Name = 'Web Development Basics')
);
```

```
INSERT INTO LMS_Questions (Q_Question, Q_CorrectAnswer)
VALUES ('Which HTML tag is used to define an internal style sheet?', '<style>');
DECLARE @QID INT = SCOPE_IDENTITY();
INSERT INTO LMS_Question_Option (Q_ID, Q_Option) VALUES
(@QID, '<script>'),
(@QID, '<style>'),
(@QID, '<css>'),
(@QID, '<link>');
INSERT INTO Quiz_Question (QU_ID, Q_ID)
VALUES ((SELECT QU_ID FROM LMS_Quiz WHERE QU_Name = 'Web Development Basics'), @QID);
```

```
INSERT INTO LMS_Questions (Q_Question, Q_CorrectAnswer)
VALUES
('What is 2 + 2?', '4'),
('Capital of France?', 'Paris'),
('Sun rises from?', 'East'),
('5 * 3 = ?', '15'),
('Which is a fruit?', 'Apple'),
('Water freezes at?', '0'),
('HTML stands for?', 'HyperText Markup Language'),
('CPU means?', 'Central Processing Unit'),
('Which one is a programming language?', 'Python'),
('Opposite of hot?', 'Cold');
```

```
INSERT INTO Quiz_Question (QU_ID, Q_ID)
VALUES
(1, 1), (1, 2), (1, 3), (1, 4), (1, 5),
(1, 6), (1, 7), (1, 8), (1, 9), (1, 10);
INSERT INTO Grades (G_ID, G_Value, G_Mark) VALUES
(501, 'A+', 95.00),
(502, 'A', 90.00),
(503, 'B+', 85.00),
(504, 'B', 80.00),
```

(505, 'C+', 75.00),
(506, 'C', 70.00),
(507, 'D', 60.00),
(508, 'F', 40.00);

Screenshots



The screenshot shows the Learnex website homepage. At the top, there is a navigation bar with links for Home, About, Courses, and Contact, along with a search icon and a user profile icon. Below the navigation bar is a large, colorful illustration featuring a man sitting on a stack of books and a woman standing next to a large computer monitor. The monitor displays a video player interface with a play button and some text. A large yellow lightbulb is positioned above the monitor, symbolizing ideas or learning. The background of the illustration includes clouds and a blue sky. To the right of the illustration, there is a promotional text block: "Unlock your potential, upgrade your career, and achieve your goals — anytime, anywhere. At Learnex, expert-led learning meets convenience to help you grow and succeed." Below this text is a prominent orange "GET STARTED" button.

Featured Courses

At Learnex, expert-led learning meets convenience to help you grow and succeed.

Web Development



8 Hours
Beginner

Graphic Design



10 Hours
Intermediate

Digital Marketing



6 Hours
Beginner

OOP2



6 Hours
Beginner



Featured Instructor

Syeda Ashrafa Islam



Instructor for OOP2 (C# Programming)
Known for her clarity, engaging content,
and practical examples.

What We Offer !!

1. Web Development -8 Hours | Beginner Level

2. Graphic Design-10 Hours | Intermediate.

3. Digital Marketing -6 Hours | Beginner Level

4. OOP2 (C#)-6 Hours | Beginner Level

Our Vision

To create a world where high-quality education is within everyone's reach
empowering individuals to transform their future through learning.

Join Us Today !!

Unlock your potential, Upgrade your skills. Start your
journey with Learnex.

All Available Courses

Web Development



8 Hours
Beginner

[Enroll Now](#)

Data Structure



10 Hours
Intermediate

[Enroll Now](#)

JavaScript



6 Hours
Beginner

[Enroll Now](#)

OOP2



6 Hours
Beginner

[Enroll Now](#)

Python Programming



8 Hours
Beginner

[Enroll Now](#)

UI Design



10 Hours
Intermediate

[Enroll Now](#)

Algorithms



6 Hours
Beginner

[Enroll Now](#)

DataBase Management



6 Hours
Beginner

[Enroll Now](#)



The image shows the Learnex contact us page. At the top, there is a navigation bar with links for "Home", "About", "Courses", "Contact" (which is highlighted in blue), and two small icons. Below the navigation bar is a section titled "***Contact Us***". This section contains four contact options: "Call Us" (with a phone icon and the number +88 02 541 4046-9), "E-mail Us" (with an envelope icon and the email address contact.learnex@gmail.com), "Address" (with a location pin icon and the address 405/1, Kurilok, Khilkhet, Dhaka), and "WhatsApp" (with a WhatsApp icon and the number +8801944250046).



Sign Up



Teacher

Name

User Name

Password

Email address

Phone Number

Sign Up[Already have an account?](#)[Log in](#)

Password Reset

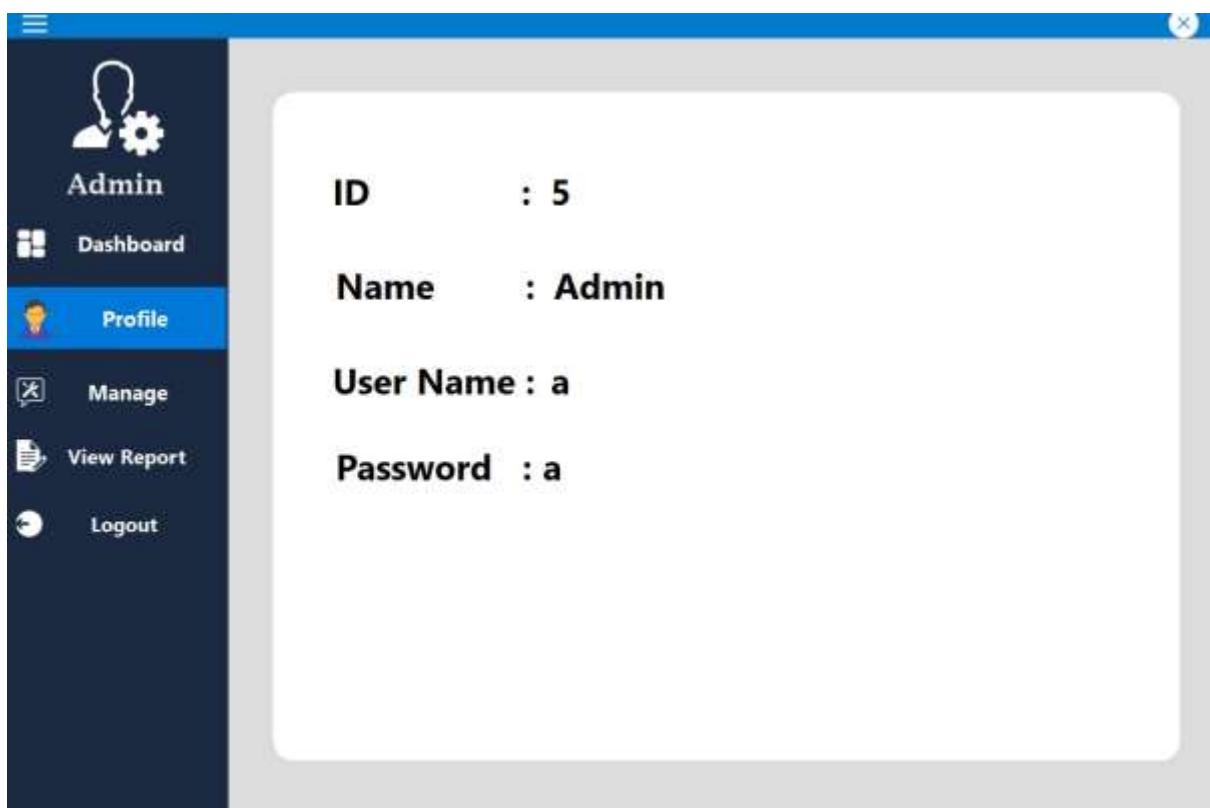
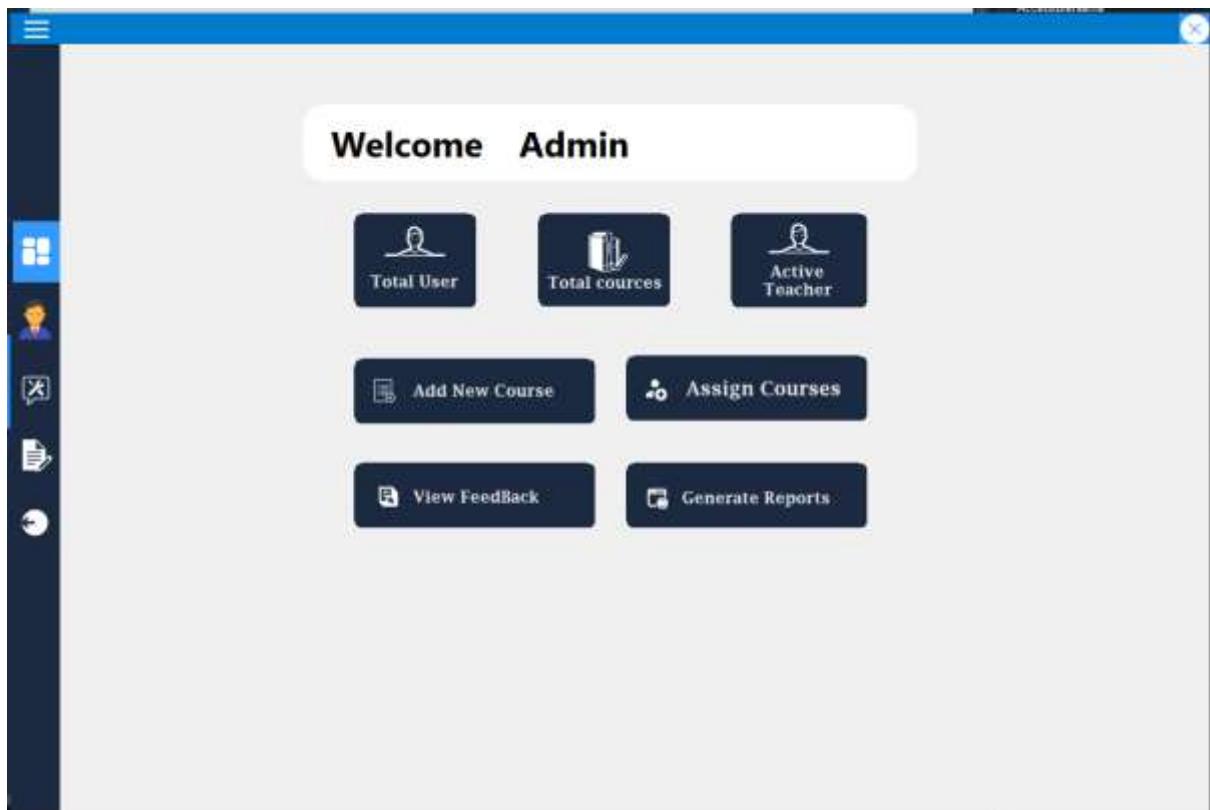
Teacher

User Name

Email address

New Password

Reset



LEARNEX

HOME QUIZES NOTES ASSIGNMENTS STUDENT

1 Courses 3 quizzes 2 Assignments 1 Upcoming Events

My Schedule

June 2025

1	2	3	4	5	6	
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30				

< Previous Next >

Add reminder title: 11:01:20 AM

Add Reminder

Update Delete Reminder

Quizzes

OOP2

- cop principles
- 09/15/2025
- 30 min
- 3 questions

Results

OOP2: B+ Math 6: A Device: B+

LEARNEX

HOME QUIZES NOTES ASSIGNMENTS S

 Browse

Upload selected document

AS_ID	AS_RenName	AS_UploadDate	C_ID	TE_ID	ST_Username
1	Accounting.pdf	6/25/2025 5:24 AM	1007	8	s
2	PDA-Spring-2025.pdf	6/25/2025 5:24 AM	1007	8	e
*					

LEARNEX

HOME QUizzes NOTES ASSIGNMENTS S

Courses:

Quiz Name:

Start the Quiz

LEARNEX



Teacher

Dashboard

Profile

Courses

Quizzes

Assignments

Logout

ID : 8

Name : Teacher

User Name : t

Email : teacher@aiub.edu

Password : t

Phone Number: +8812345678910,

Joining Date : 13-06-2025

LEARNEX



Teacher

Dashboard

Profile

Courses

Quiz

Assignments

Logout

LEARNEX

Quiz Name: **Due Date:**

Total Mark:

Courses:

Question	Option1	Option2
Option3	Option4	Answer

Save **Edit** **Reset** **Delete**

Question Bank

C_Name	Q_Question	Q_Option	Q_CorrectAnswer	isCorrect
Web Development	What does HTML stand for?	Hyper Text Markup Language	Hyper Text Markup Language	Yes
Web Development	Which tag is used to create a hy...			Yes
Web Development	Which HTML tag is used to inser...			Yes
Web Development	What is the correct file extension for a Java applet?	.html	.html	Yes
Web Development	Which tag is used to create a li...			Yes
Web Development	What tag is used to define a tabl...	<tbl>	<tbl>	Yes
Web Development	How can you make a numbered lis...			Yes
Web Development	What is the purpose of the t...	Line Break	Line Break	Yes
Web Development	Which attribute is used to provid...	alt	alt	Yes
Data Structure	Which data structure uses LIFO pr...	Stack	Stack	Yes
Data Structure	Which data structure uses LIFO pr...	Stack	Stack	Yes
Data Structure	Which data structure is used for ...	Queue	Queue	Yes
OOPII	Which keyword is used to create ...	new	new	Yes

Courses and Student List :

C_Name	ST_Name
Database Management	STUDENT

The screenshot shows the LEARNEX teacher dashboard. On the left sidebar, under the 'Teacher' section, the 'Assignments' option is selected, indicated by a blue background. Other options include Dashboard, Profile, Courses, Quizzes, and Logout. The main content area displays a table of uploaded assignments:

AS_ID	AS_FileName	AS_UploadDate	C_ID	TE_ID	ST_Username
1	Accounting.pdf	6/25/2025 5:24 AM	1007	8	s
2	POA-Spring-2025.pdf	6/25/2025 5:24 AM	1007	8	s

A blue button at the bottom right of the table area says 'Download selected document'.

The screenshot shows the LEARNEX teacher dashboard. The main content area displays the following course statistics:

- Courses:** 1 Course
- Quizzes:** 0 Quizzes
- Assignments:** 2 Assignments
- Students:** 1 Student


Admin

-  [Dashboard](#)
-  [Profile](#)
-  [Manage](#)
-  [Manage Faculty](#)
-  [Manage Student](#)
-  [View Report](#)
-  [Logout](#)

Name :

UserName :

Password :

Email :

Number :

Course :

Registration Date

Insert **Update** **Delete** **Search**

ST_ID	ST_Name	ST_Username	ST_Email	ST_Password	Registration_Date	ST_PNUMBER
1	Heeb Rahman	heeb_rahman0	heeb_rahman0@gmail.com	heeb001	3/25/2023	01600112233
2	Rahul Islam	rahul_islam1	rahul_islam1@gmail.com	rahul11	3/20/2023	01800334455
3	Rahul Islam	rahul_islam1	rahul_islam1@gmail.com	rahul11	3/20/2023	01900445566
3	Sabina Alter	sabina_alter2	sabina_alter2@gmail.com	sabina33	4/2/2023	01319972204
3	Sabina Alter	sabina_alter2	sabina_alter2@gmail.com	sabina33	4/2/2023	01500006677
7	STUDENT	-	student@abu.edu	-	6/13/2025	01844250076


Admin

-  [Dashboard](#)
-  [Profile](#)
-  [Manage](#)
-  [Manage Faculty](#)
-  [Manage Student](#)
-  [View Report](#)
-  [Logout](#)

Name :

UserName :

Password :

Email :

Number :

Joining Date

Insert **Update** **Delete** **Search**

TE_ID	TE_Name	TE_Username	TE_Password	TE_Email	Joining_Date	TE_PNUMBER
1	Rafiq Hasan	rafiq.hasan	teacher01	rafiq@gmail.com	2/7/2022	01600000076
1	Rafiq Hasan	rafiq.hasan	teacher01	rafiq@gmail.com	2/7/2022	01032364518
2	Fathana Yassmin	fathana_yassmin	teacher02	fathana_yassmin@gmail.com	3/3/2023	01669545203
2	Fathana Yassmin	fathana_yassmin	teacher02	fathana_yassmin@gmail.com	3/3/2023	01952305737
3	Jay Rahman	jay_rahman	teacher03	jay@gmail.com	3/6/2023	01513649671
3	Jay Rahman	jay_rahman	teacher03	jay@gmail.com	3/6/2023	01884326339
4	Sajid Hossain	sajid_hossain	teacher04	sajid_hossain@gmail.com	5/24/2024	01961914396
5	Tarvir Rani	Tarvir5	tarvir555	tarvir@gmail.com	6/11/2024	01753062900
7	ankit11	ankit	1234	ankit@gmail.com	5/8/2025	01888809771
8	Teacher	-	-	teacher@abu.edu	6/13/2025	12345678912