



Information
Technology
Institute

EXAMINATION SYSTEM

2025
2026

Team Members:

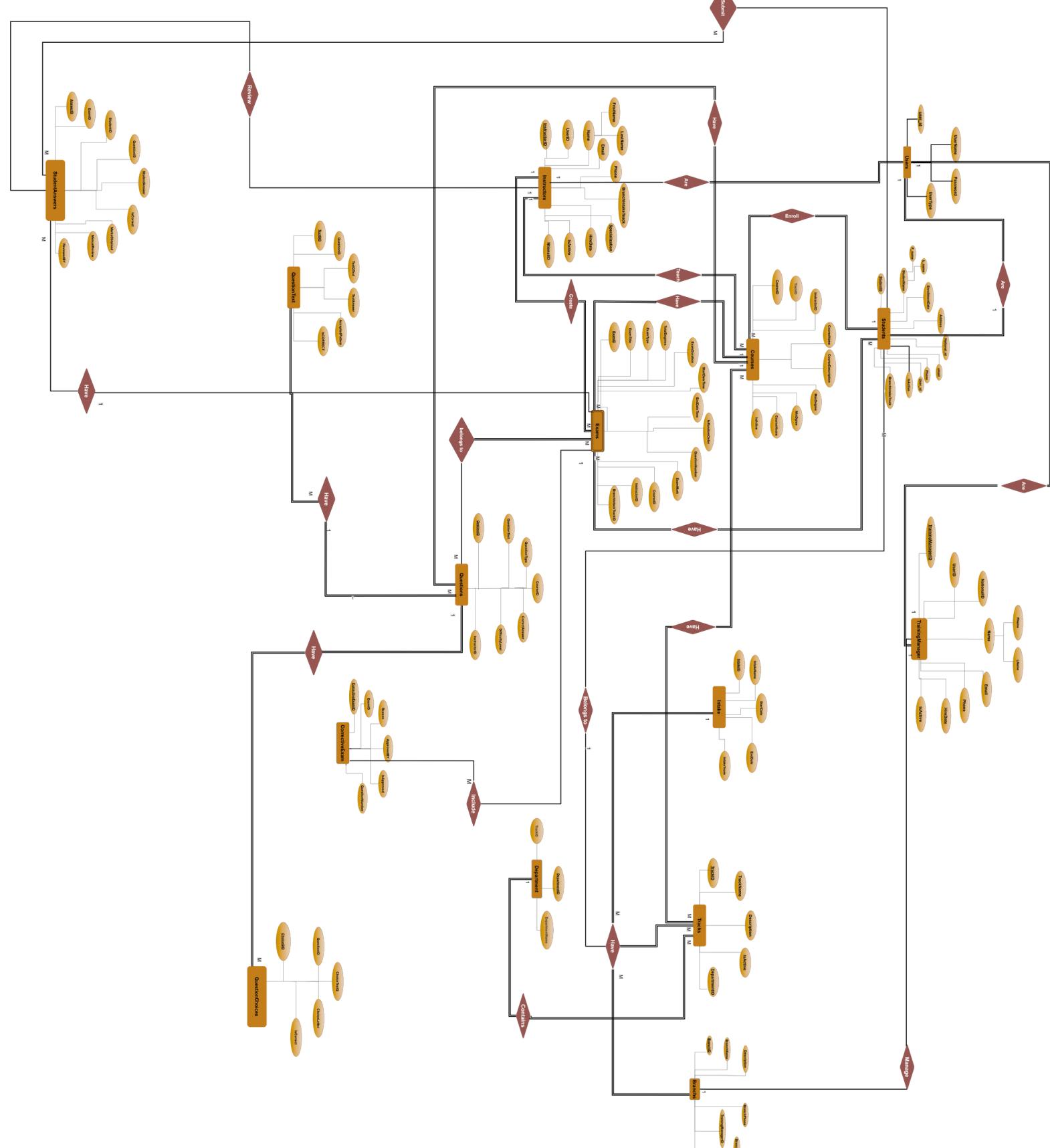
**Asmaa Gamal Abdalaal Sayed
Ahmed Magdi Ali Baraka
Esraa Ramadan Sayed Darder
Bishoy Samir Najeeb Tawdros
Yousef Mohamed Soliman Abo-eleneen
Moaaz Omar Abdelfattah Abdelhamed**

PROJECT OVERVIEW

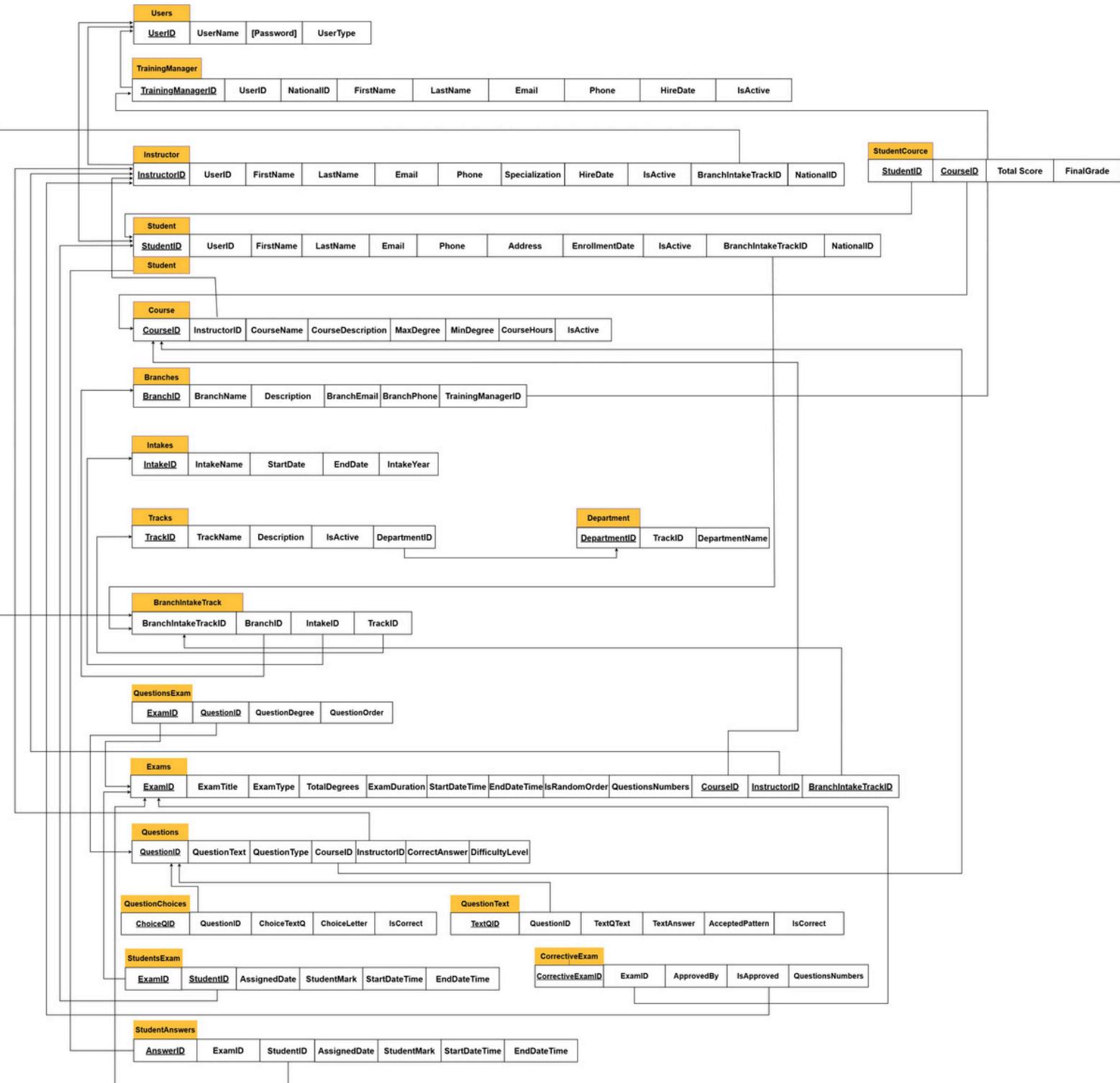
Overview:

- Database-driven system to manage courses, instructors, students, and exams.
- Provides a question pool with multiple-choice, true/false, and text-based questions.
- Allows instructors to generate exams randomly or manually.
- Supports secure user accounts for training managers, instructors, and students with role-based access.
- Handles exam scheduling, result calculation, and answer storage automatically.
- Ensures data integrity, high performance, and daily backups for reliability.
- Stores course details, instructor assignments, and student information for accurate record-keeping.
- Enables manual review and grading of text-based answers with flexible evaluation.
- Offers search and reporting features to track performance and system activity.

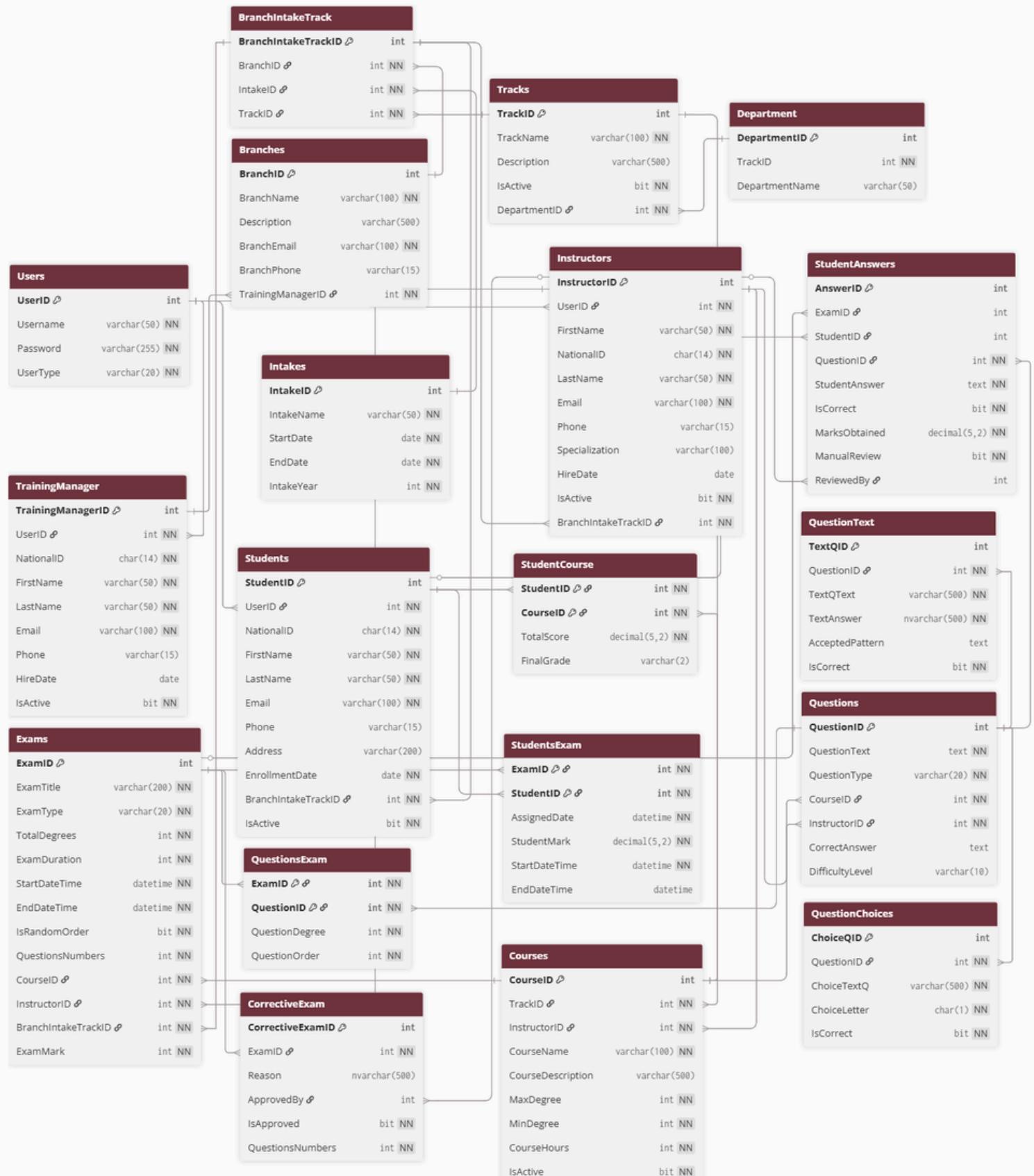
ENTITY-RELATIONSHIP DIAGRAM (ERD)



MAPPING



MAPPING



SQL DDL IMPLEMENTATION

```
CREATE TABLE dbo.Users (
    UserID      INT          IDENTITY(1,1) PRIMARY KEY,
    Username    VARCHAR(50)   NOT NULL UNIQUE,
    [Password]  VARCHAR(255)  NOT NULL,
    UserType    VARCHAR(20)   NOT NULL
);
GO

-- 2) TrainingManager
CREATE TABLE dbo.TrainingManager (
    TrainingManagerID INT          IDENTITY(1,1) PRIMARY KEY,
    UserID            INT          NOT NULL,
    NationalID       CHAR(14)     NOT NULL,
    FirstName        VARCHAR(50)  NOT NULL,
    LastName         VARCHAR(50)  NOT NULL,
    Email            VARCHAR(100) NOT NULL,
    Phone            VARCHAR(15)   NULL,
    HireDate         DATE         NULL,
    IsActive         BIT          NOT NULL DEFAULT(1),
    CONSTRAINT FK_TM_User FOREIGN KEY (UserID) REFERENCES dbo.Users(UserID)
);
GO

-- 3) Branches
CREATE TABLE dbo.Branches (
    BranchID      INT          IDENTITY(1,1) PRIMARY KEY,
    BranchName    VARCHAR(100) NOT NULL,
    [Description] VARCHAR(500)  NULL,
    BranchEmail   VARCHAR(100) NOT NULL,
    BranchPhone   VARCHAR(15)   NULL,
    TrainingManagerID INT        NOT NULL,
    CONSTRAINT FK_Branches_TM FOREIGN KEY (TrainingManagerID) REFERENCES dbo.TrainingManager(TrainingManagerID)
);
GO

-- 4) Intakes
CREATE TABLE dbo.Intakes (
    IntakeID      INT          IDENTITY(1,1) PRIMARY KEY,
    IntakeName    VARCHAR(50)  NOT NULL,
    StartDate     DATE         NOT NULL,
    EndDate       DATE         NOT NULL,
    IntakeYear    INT          NOT NULL
);
GO

-- 5) Department
CREATE TABLE dbo.Department (
    DepartmentID  INT NOT NULL PRIMARY KEY,
    TrackID       INT NOT NULL,
    DepartmentName VARCHAR(50)
);
GO

-- 6) Tracks
CREATE TABLE dbo.Tracks (
    TrackID      INT          IDENTITY(1,1) PRIMARY KEY,
    TrackName    VARCHAR(100) NOT NULL,
    [Description] VARCHAR(500)  NULL,
    IsActive     BIT          NOT NULL DEFAULT(1),
    DepartmentID INT          NOT NULL,
    CONSTRAINT FK_Department_Track FOREIGN KEY (DepartmentID) REFERENCES dbo.Department(DepartmentID)
);
```

SQL DDL IMPLEMENTATION

```
-- 7) BranchIntakeTrack
CREATE TABLE dbo.BranchIntakeTrack (
    BranchIntakeTrackID INT IDENTITY(1,1) PRIMARY KEY,
    BranchID INT NOT NULL,
    IntakeID INT NOT NULL,
    TrackID INT NOT NULL,
    CONSTRAINT FK_BIT_B FOREIGN KEY (BranchID) REFERENCES dbo.Branches(BranchID),
    CONSTRAINT FK_BIT_I FOREIGN KEY (IntakeID) REFERENCES dbo.Intakes(IntakeID),
    CONSTRAINT FK_BIT_T FOREIGN KEY (TrackID) REFERENCES dbo.Tracks(TrackID)
);
GO

-- 8) Students
CREATE TABLE dbo.Students (
    StudentID INT IDENTITY(1,1) PRIMARY KEY,
    UserID INT NOT NULL,
    NationalID CHAR(14) NOT NULL,
    FirstName VARCHAR(50) NOT NULL,
    LastName VARCHAR(50) NOT NULL,
    Email VARCHAR(100) NOT NULL,
    Phone VARCHAR(15) NULL,
    Address VARCHAR(200) NULL,
    EnrollmentDate DATE NOT NULL,
    BranchIntakeTrackID INT NOT NULL,
    IsActive BIT NOT NULL DEFAULT(1),
    CONSTRAINT FK_Students_User FOREIGN KEY (UserID) REFERENCES dbo.Users(UserID),
    CONSTRAINT FK_BranchIntakeTrack FOREIGN KEY (BranchIntakeTrackID) REFERENCES dbo.BranchIntakeTrack(BranchIntakeTrackID)
);
GO

-- 9) Instructors
CREATE TABLE dbo.Instructors (
    InstructorID INT IDENTITY(1,1) PRIMARY KEY,
    UserID INT NOT NULL,
    FirstName VARCHAR(50) NOT NULL,
    NationalID CHAR(14) NOT NULL,
    LastName VARCHAR(50) NOT NULL,
    Email VARCHAR(100) NOT NULL,
    Phone VARCHAR(15) NULL,
    Specialization VARCHAR(100) NULL,
    HireDate DATE NULL,
    IsActive BIT NOT NULL DEFAULT(1),
    BranchIntakeTrackID INT NOT NULL,
    CONSTRAINT FK_Instructors_User FOREIGN KEY (UserID) REFERENCES dbo.Users(UserID),
    CONSTRAINT FK_Instructors_BIT FOREIGN KEY (BranchIntakeTrackID) REFERENCES dbo.BranchIntakeTrack(BranchIntakeTrackID)
);
GO

-- 10) Courses
CREATE TABLE dbo.Courses (
    CourseID INT IDENTITY(1,1) PRIMARY KEY,
    TrackID INT NOT NULL,
    InstructorID INT NOT NULL,
    CourseName VARCHAR(100) NOT NULL,
    CourseDescription VARCHAR(500) NULL,
    MaxDegree INT NOT NULL,
    MinDegree INT NOT NULL,
    CourseHours INT NOT NULL,
    IsActive BIT NOT NULL DEFAULT(1),
    CONSTRAINT FK_Courses_Track FOREIGN KEY (TrackID) REFERENCES dbo.Tracks(TrackID),
    CONSTRAINT FK_Courses_Inst FOREIGN KEY (InstructorID) REFERENCES dbo.Instructors(InstructorID)
);
```

SQL DDL IMPLEMENTATION

```
-- 11) Exams
CREATE TABLE dbo.Exams (
    ExamID      INT          IDENTITY(1,1) PRIMARY KEY,
    ExamTitle   VARCHAR(200)  NOT NULL,
    ExamType    VARCHAR(20)   NOT NULL,
    TotalDegrees INT         NOT NULL,
    ExamDuration INT        NOT NULL,
    StartDateTime DATETIME   NOT NULL,
    EndDateTime DATETIME    NOT NULL,
    IsRandomOrder BIT        NOT NULL DEFAULT(0),
    QuestionsNumbers INT      NOT NULL,
    CourseID     INT         NOT NULL,
    InstructorID INT        NOT NULL,
    BranchIntakeTrackID INT    NOT NULL,
    ExamMark     INT        NOT NULL,
    CONSTRAINT FK_Exams_Course FOREIGN KEY (CourseID) REFERENCES dbo.Courses(CourseID),
    CONSTRAINT FK_Exams_Instructor FOREIGN KEY (InstructorID) REFERENCES dbo.Instructors(InstructorID),
    CONSTRAINT FK_Exams_Context FOREIGN KEY (BranchIntakeTrackID) REFERENCES dbo.BranchIntakeTrack(BranchIntakeTrackID)
);
GO

-- 12) Questions
CREATE TABLE dbo.Questions (
    QuestionID  INT          IDENTITY(1,1) PRIMARY KEY,
    QuestionText TEXT        NOT NULL,
    QuestionType VARCHAR(20)  NOT NULL,
    CourseID    INT         NOT NULL,
    InstructorID INT        NOT NULL,
    CorrectAnswer TEXT       NULL,
    DifficultyLevel VARCHAR(10) NULL,
    CONSTRAINT FK_Questions_Course FOREIGN KEY (CourseID) REFERENCES dbo.Courses(CourseID),
    CONSTRAINT FK_Questions_Instructor FOREIGN KEY (InstructorID) REFERENCES dbo.Instructors(InstructorID)
);
GO

-- 13) QuestionChoices
CREATE TABLE dbo.QuestionChoices (
    ChoiceQID   INT IDENTITY(1,1) PRIMARY KEY,
    QuestionID  INT NOT NULL,
    ChoiceTextQ  VARCHAR(500) NOT NULL,
    ChoiceLetter CHAR(1) NOT NULL,
    IsCorrect    BIT NOT NULL,
    CONSTRAINT FK_MCQ_Question FOREIGN KEY (QuestionID) REFERENCES dbo.Questions(QuestionID)
);
GO

-- 14) QuestionText
CREATE TABLE dbo.QuestionText (
    TextQID     INT IDENTITY(1,1) PRIMARY KEY,
    QuestionID  INT NOT NULL,
    TextQText   VARCHAR(500) NOT NULL,
    TextAnswer   NVARCHAR(500) NOT NULL,
    AcceptedPattern TEXT NULL,
    IsCorrect    BIT NOT NULL,
    CONSTRAINT FK_TextQ_Question FOREIGN KEY (QuestionID) REFERENCES dbo.Questions(QuestionID)
);
GO

-- 15) QuestionsExam
CREATE TABLE dbo.QuestionsExam (
    ExamID      INT NOT NULL,
    QuestionID  INT NOT NULL,
    QuestionDegree INT NOT NULL,
    QuestionOrder INT NOT NULL,
    CONSTRAINT PK_EQ PRIMARY KEY (ExamID, QuestionID),
    CONSTRAINT FK_EQ_Exam FOREIGN KEY (ExamID) REFERENCES dbo.Exams(ExamID),
    CONSTRAINT FK_EQ_Question FOREIGN KEY (QuestionID) REFERENCES dbo.Questions(QuestionID)
);
GO
```

SQL DDL IMPLEMENTATION

```
-- 16) StudentsExam
CREATE TABLE dbo.StudentsExam (
    ExamID      INT      NOT NULL,
    StudentID   INT      NOT NULL,
    AssignedDate DATETIME NOT NULL,
    StudentMark DECIMAL(5,2) NOT NULL,
    StartDateTime DATETIME NOT NULL,
    EndDateTime DATETIME NULL,
    CONSTRAINT PK_ExamStudents PRIMARY KEY (ExamID, StudentID),
    CONSTRAINT FK_ES_Exam FOREIGN KEY (ExamID) REFERENCES dbo.Exams(ExamID),
    CONSTRAINT FK_ES_Student FOREIGN KEY (StudentID) REFERENCES dbo.Students(StudentID)
);
GO

-- 17) CorrectiveExam
CREATE TABLE dbo.CorrectiveExam (
    CorrectiveExamID INT IDENTITY(1,1) PRIMARY KEY,
    ExamID          INT NOT NULL,
    Reason          NVARCHAR(500) NULL,
    ApprovedBy     INT NULL,
    IsApproved     BIT NOT NULL DEFAULT(0),
    QuestionsNumbers INT NOT NULL,
    CONSTRAINT FK_CE_Exam FOREIGN KEY (ExamID) REFERENCES dbo.Exams(ExamID),
    CONSTRAINT FK_CE_Approver FOREIGN KEY (ApprovedBy) REFERENCES dbo.Instructors(InstructorID)
);
GO

-- 18) StudentAnswers
CREATE TABLE dbo.StudentAnswers (
    AnswerID      INT IDENTITY(1,1) PRIMARY KEY,
    ExamID        INT NULL,
    StudentID     INT NULL,
    QuestionID    INT NOT NULL,
    StudentAnswer TEXT NOT NULL,
    IsCorrect     BIT NOT NULL,
    MarksObtained DECIMAL(5,2) NOT NULL,
    ManualReview  BIT NOT NULL DEFAULT(0),
    ReviewedBy    INT NULL,
    CONSTRAINT FK_SA_Question FOREIGN KEY (QuestionID) REFERENCES dbo.Questions(QuestionID),
    CONSTRAINT FK_SA_Reviewer FOREIGN KEY (ReviewedBy) REFERENCES dbo.Instructors(InstructorID),
    CONSTRAINT FK_SA_Exam FOREIGN KEY (ExamID) REFERENCES dbo.Exams(ExamID),
    CONSTRAINT FK_SA_Student FOREIGN KEY (StudentID) REFERENCES dbo.Students(StudentID)
);
GO

-- 19) StudentCourse
CREATE TABLE dbo.StudentCourse (
    StudentID    INT NOT NULL,
    CourseID     INT NOT NULL,
    TotalScore   DECIMAL(5,2) NOT NULL,
    FinalGrade   VARCHAR(2) NULL,
    CONSTRAINT PK_SC PRIMARY KEY (StudentID, CourseID),
    CONSTRAINT FK_SC_Student FOREIGN KEY (StudentID) REFERENCES dbo.Students(StudentID),
    CONSTRAINT FK_SC_Course FOREIGN KEY (CourseID) REFERENCES dbo.Courses(CourseID)
);
GO

-- 20) AuditStudents
CREATE TABLE dbo.AuditStudents (
    AuditID      INT IDENTITY(1,1) PRIMARY KEY,
    StudentID    INT,
    FirstName    VARCHAR(50),
    LastName     VARCHAR(50),
    Email        VARCHAR(100),
    Phone        VARCHAR(15),
    Address      VARCHAR(200),
    EnrollmentDate DATE,
    BranchIntakeTrackID INT NOT NULL,
    CONSTRAINT FK_AuditStudents_BRANCH INT FOREIGN KEY (BranchIntakeTrackID) REFERENCES dbo.BranchIntakeTrack(BranchIntakeTrackID),
    DeletedAt    DATETIME DEFAULT GETDATE()
);
GO
```

SQL SEED DATA

Example of insertion into tables

```
-- Insert Users (Admin, Training Managers, Instructors, Students)
INSERT INTO dbo.Users ([Username], [Password], UserType)
VALUES
    ('admin', 'admin123', 'Admin'),
    ('tm_ahmed', 'tm123', 'TrainingManager'),
    ('tm_mohammed', 'tm456', 'TrainingManager'),
    ('tm_nada', 'tm789', 'TrainingManager'),
    ('inst_rawhia', 'inst123', 'Instructor'),
    ('inst_rahaa', 'inst456', 'Instructor'),
    ('inst_nora', 'inst789', 'Instructor'),
    ('inst_sarah', 'inst101', 'Instructor'),
    ('inst_esraa', 'inst102', 'Instructor'),
    ('inst_moaz', 'inst103', 'Instructor'),
    ('inst_youssef', 'inst104', 'Instructor'),
    ('inst_ahmed', 'inst105', 'Instructor'),
    ('inst_bishoy', 'inst106', 'Instructor'),
    ('inst_asmaa', 'inst107', 'Instructor'),
    ('stud_ahmed', 'stud123', 'Student'),
    ('stud_sara', 'stud456', 'Student'),
    ('stud_youssef', 'stud789', 'Student'),
    ('stud_mona', 'stud101', 'Student'),
    ('stud_rania', 'stud102', 'Student'),
    ('stud_farida', 'stud103', 'Student'),
    ('stud_arwa', 'stud104', 'Student'),
    ('stud_nada', 'stud105', 'Student'),
    ('stud_noaa', 'stud106', 'Student'),
    ('stud_rana', 'stud107', 'Student'),
    ('stud_omar', 'stud108', 'Student'),
    ('stud_hassan', 'stud109', 'Student'),
    ('stud_fatma', 'stud110', 'Student'),
    ('stud_ali', 'stud111', 'Student'),
    ('stud_dina', 'stud112', 'Student');

GO

-- Insert Training Managers
INSERT INTO dbo.TrainingManager ([UserID], [NationalID], [FirstName], [LastName], [Email], [Phone], [HireDate], [IsActive])
VALUES
    (2, '3020202020202', 'Ahmed', 'Magdy', 'ahmed.magdy@company.com', '01023659887', '2024-05-15', 1),
    (3, '30303030300303', 'Mohammed', 'Ali', 'mohammed.ali@company.com', '01063258974', '2024-04-20', 1),
    (4, '30404040400404', 'Nada', 'Alaa', 'nada.alaa@company.com', '01152369741', '2024-02-10', 1);

GO

-- Insert Intakes
INSERT INTO dbo.Intakes ([IntakeName], [StartDate], [EndDate], [IntakeYear])
VALUES
    ('Intake 46', '2025-10-12', '2026-07-30', 2025),
    ('Intake 45', '2024-10-12', '2025-07-30', 2024),
    ('Intake 44', '2023-10-12', '2024-07-30', 2023),
    ('Intake 43', '2022-10-12', '2023-07-30', 2022);

GO

-- Insert Branches
INSERT INTO dbo.Banches ([BranchName], [Description], [BranchEmail], [BranchPhone], [TrainingManagerID])
VALUES
    ('Smart Village Branch', 'Main campus in Smart Village, Bldg B14B on Cairo-Alex Desert Road', 'smartvillage@company.com', '0226594100', 1),
    ('Alexandria Branch', 'Located at Post Office Building, Shohada Square (MISR Station) in Alexandria', 'alexandria@company.com', '0334567899', 2),
    ('New Capital Branch', 'Knowledge City (smart section)', 'newcapital@company.com', '01500123456', 1),
    ('Assuit Branch', 'Assuit University - Info Network Building', 'assuit@company.com', '0882345678', 3),
    ('Minia Branch', 'Minia University - Creativa Building', 'minia@company.com', '0862456789', 2);

GO

-- Insert BranchIntakeTrack
INSERT INTO dbo.BranchIntakeTrack ([BranchID], [IntakeID], [TrackID])
VALUES
    (1, 1, 1), (1, 1, 2), (1, 1, 3), (1, 1, 4),
    (2, 1, 1), (2, 1, 4),
    (3, 2, 2), (3, 2, 3),
    (4, 2, 3), (4, 2, 2),
    (5, 3, 1), (5, 3, 4),
    (1, 2, 1), (2, 2, 2), (3, 3, 3);

GO

-- Insert Instructors
INSERT INTO dbo.Instructors ([UserID], [FirstName], [LastName], [Email], [Phone], [Specialization], [HireDate], [IsActive], [BranchIntakeTrackID], [NationalID])
VALUES
    (5, 'Rania', 'Abdulrahman', 'rania.abdulrahman@company.com', '01033333333', 'Web Development', '2023-04-05', 1, 1, '299004182345671'),
    (6, 'Rahma', 'Mohamed', 'rahma.mohamed@company.com', '01044444444', 'Data Science', '2023-05-20', 1, 2, '300052012345670'),
    (7, 'Nora', 'Magdy', 'nora.magdy@company.com', '01055555555', 'Cybersecurity', '2023-06-15', 1, 3, '30106152345679'),
    (8, 'Sarah', 'Salah', 'sarah.salah@company.com', '01066666666', 'Cloud Computing', '2023-07-01', 1, 4, '30207012345679'),
    (9, 'Esraa', 'Haroon', 'esraa.haroon@company.com', '01077777777', 'Data Science', '2023-08-10', 1, 5, '30308102345670'),
    (10, 'Moaz', 'Omar', 'moaz.omar@company.com', '01088888888', 'Machine Learning', '2023-09-25', 1, 6, '30409252345671'),
    (11, 'Youssef', 'Mahmoud', 'youssef.mahmoud@company.com', '01099999999', 'Full Stack Development', '2023-10-30', 1, 11, '30510302345672'),
    (12, 'Ahmed', 'Magdy', 'ahmed.magdy@company.com', '01100000000', 'Cybersecurity', '2023-01-01', 1, 9, '3060102345673'),
    (13, 'Bishoy', 'Samir', 'bishoy.samir@company.com', '01111111111', 'Software Engineering', '2024-03-01', 1, 12, '30703012345674'),
    (14, 'Asmaa', 'Gamal', 'asmaa.gamal@company.com', '01222222222', 'Data Analytics', '2024-04-01', 1, 10, '308004012345675');

GO

-- Insert Students
INSERT INTO dbo.Students ([UserID], [NationalID], [FirstName], [LastName], [Email], [Phone], [Address], [EnrollmentDate], [BranchIntakeTrackID], [IsActive])
VALUES
    (15, '31515151515151', 'Ahmed', 'Mohamed', 'ahmed.mohamed@student.com', '01000000001', '123 Nasr City, Cairo', '2025-01-01', 1, 1),
    (16, '31616161616161', 'Sara', 'Gamal', 'sara.gamal@student.com', '01000000002', '456 Dokki, Giza', '2025-01-01', 2, 1),
    (17, '31717171717171', 'Youssef', 'Ibrahim', 'youssef.ibrahim@student.com', '01000000003', '789 Sidi Gaber, Alexandria', '2025-01-01', 5, 1),
    (18, '31818181818181', 'Mona', 'Nafee', 'mona.nafee@student.com', '01000000004', '321 Heliopolis, Cairo', '2025-01-01', 3, 1),
    (19, '31919191919191', 'Rania', 'Alaa', 'rania.alaa@student.com', '01000000005', '654 Mansoura Center', '2025-07-01', 9, 1),
    (20, '32020202020202', 'Farida', 'Mohamed', 'farida.mohamed@student.com', '01000000006', '987 Stanley, Alexandria', '2025-07-01', 11, 1),
    (21, '32121212121212', 'Arwa', 'Sami', 'arwa.sami@student.com', '01000000007', '147 Assuit University St.', '2025-07-01', 8, 1),
    (22, '32222222222222', 'Nada', 'Ahmed', 'nada.ahmed@student.com', '01000000008', '258 Ismailia Downtown', '2025-01-01', 4, 1),
    (23, '32323232323232', 'Noha', 'Rashid', 'noha.rashid@student.com', '01000000009', '369 New Administrative Capital', '2026-01-01', 13, 1),
    (24, '32424242424242', 'Rana', 'Ahmed', 'rana.ahmed@student.com', '01000000010', '741 Minia University Area', '2026-01-01', 10, 1),
    (25, '32525252525252', 'Omar', 'Hassan', 'omar.hassan@student.com', '01000000011', '852 Zamalek, Cairo', '2025-01-01', 1, 1),
    (26, '32626262626262', 'Hassan', 'Ali', 'hassan.ali@student.com', '01000000012', '963 Mohandessin, Giza', '2025-01-01', 2, 1),
    (27, '32727272727272', 'Fatma', 'Omar', 'fatma.omar@student.com', '01000000013', '159 Smouha, Alexandria', '2025-07-01', 5, 1),
    (28, '32828282828282', 'Ali', 'Mahmoud', 'ali.mahmoud@student.com', '01000000014', '357 Maadi, Cairo', '2025-01-01', 3, 1),
    (29, '32929292929292', 'Dina', 'Youssef', 'dina.youssef@student.com', '01000000015', '468 6th October City', '2025-07-01', 12, 1);

GO
```

SQL SEED DATA

Example of insertion into tables

```
-- Insert Courses
INSERT INTO dbo.Courses
VALUES
(1, 2, 'HTML & CSS Fundamentals', 'Introduction to web markup and styling languages', 100, 60, 40, 1),
(1, 2, 'JavaScript Programming', 'Client-side scripting and DOM manipulation', 100, 60, 50, 1),
(1, 8, 'React Development', 'Modern Frontend Framework for building user interfaces', 100, 60, 45, 1),
(1, 10, 'Node.js Backend', 'Server-side JavaScript development', 100, 60, 50, 1),
(2, 3, 'Python for Data Science', 'Python programming for data analysis and visualization', 100, 60, 40, 1),
(2, 3, 'Machine Learning Basics', 'Statistical methods for machine learning applications', 100, 60, 70, 1),
(2, 11, 'Data Visualization', 'Creating meaningful charts and dashboards', 100, 60, 40, 1),
(2, 6, 'Statistical Analysis', 'Statistical methods for data interpretation', 100, 60, 55, 1),
(3, 4, 'Network Security', 'Fundamentals of cybersecurity and network protection', 100, 60, 50, 1),
(3, 9, 'Ethical Hacking', 'Penetration testing and vulnerability assessment', 100, 60, 60, 1),
(3, 4, 'Cryptography', 'Encryption methods and security protocols', 100, 60, 45, 1),
(4, 5, 'Machine Learning Project', 'Building a machine learning model', 100, 60, 55, 1),
(4, 5, 'DevOps Fundamentals', 'CI/CD pipelines and automation tools', 100, 60, 50, 1),
(4, 5, 'Container Technologies', 'Docker and Kubernetes for scalable applications', 100, 60, 45, 1),
(1, 2, 'Database Design', 'Relational database concepts and SQL', 100, 60, 45, 1);

-- Insert Exams
INSERT INTO dbo.Exams
VALUES
('HTML & CSS Final Exam', 'Final', 100, 120, '2025-03-15 11:00:00', 1, 10, 3, 2, 1, 100),
('JavaScript Midterm', 'Midterm', 80, 90, '2025-04-10 11:30:00', 1, 8, 4, 2, 1, 80),
('Python Programming Exam', 'Exam', 80, 90, '2025-05-15 10:00:00', 1, 8, 4, 2, 1, 80),
('React Development Assessment', 'Final', 100, 150, '2025-06-20 11:30:00', 1, 12, 11, 4, 3, 100),
('Cloud Computing Practical', 'Practical', 75, 120, '2025-07-15 13:00:00', 1, 9, 6, 14, 5, 4, 75),
('React Development Project', 'Project', 100, 180, '2025-08-10 09:00:00', 1, 9, 6, 14, 5, 4, 75),
('Machine Learning Final', 'Final', 100, 150, '2025-09-25 10:00:00', 1, 15, 8, 7, 2, 100);

-- Insert Questions
INSERT INTO dbo.Questions
VALUES
('QuestionText: QuestionType, CourseID, InstructorID, CorrectAnswer, DifficultyLevel')
VALUES
-- HTML & CSS Fundamentals (CourseID = 3, InstructorID = 2)
('HTML is a programming language.', 'TrueFalse', 3, 2, 'False', 'Easy'),
('Explain the box model in CSS.', 'Text', 3, 2, 'The box model describes rectangular boxes with content, padding, border, and margin areas.', 'Medium'),
('Which HTML tag is used for line breaks?', 'MultipleChoice', 3, 2, '<br>', 'Easy'),
-- JavaScript (CourseID = 4, InstructorID = 2)
('What is the correct way to declare a variable in JavaScript?', 'MultipleChoice', 4, 2, 'let variableName;', 'Easy'),
('JavaScript is case-sensitive.', 'TrueFalse', 4, 2, 'True', 'Easy'),
('Explain the difference between let and var.', 'Text', 4, 2, 'let has block scope while var has function scope. let prevents hoisting issues.', 'Medium'),
('Which method adds an element to the end of an array?', 'MultipleChoice', 4, 2, 'push()', 'Easy'),
('What does DOM stand for?', 'MultipleChoice', 4, 2, 'Document Object Model', 'Easy'),
-- Python (CourseID = 7, InstructorID = 3)
('What is the correct file extension for Python files?', 'MultipleChoice', 7, 3, '.py', 'Easy'),
('Python is an interpreted language.', 'TrueFalse', 7, 3, 'True', 'Easy'),
('Explain list comprehension in Python.', 'Text', 7, 3, 'List comprehension provides a concise way to create lists using a single line of code.', 'Medium'),
('Which keyword is used to define a function in Python?', 'MultipleChoice', 7, 3, 'def', 'Easy'),
('What is the output of print(type([]))?', 'MultipleChoice', 7, 3, '<class \'list\'>', 'Medium'),
-- Network Security (CourseID = 11, InstructorID = 4)
('What does HTTPS stand for?', 'MultipleChoice', 11, 4, 'HyperText Transfer Protocol Secure', 'Easy'),
('Encryption converts readable data into unreadable format.', 'TrueFalse', 11, 4, 'True', 'Easy'),
('Explain two-factor authentication.', 'Text', 11, 4, 'Two-factor authentication adds extra security by requiring two different authentication factors.', 'Medium'),
('Which port is commonly used for HTTPS?', 'MultipleChoice', 11, 4, '443', 'Easy'),
('What is a firewall?', 'Text', 11, 4, 'A firewall is a network security system that monitors and controls network traffic based on security rules.', 'Medium'),
-- Cloud Platforms (CourseID = 14, InstructorID = 5)
('What does SaaS stand for?', 'MultipleChoice', 14, 5, 'Software as a Service', 'Easy'),
('Cloud computing eliminates the need for local storage.', 'TrueFalse', 14, 5, 'False', 'Medium'),
('Name three major cloud service providers.', 'Text', 14, 5, 'Amazon Web Services (AWS), Microsoft Azure, Google Cloud Platform (GCP)', 'Easy'),
('What is auto-scaling in cloud computing?', 'Text', 14, 5, 'Auto-scaling automatically adjusts computing resources based on demand to maintain performance.', 'Medium'),
('Which AWS service is used for object storage?', 'MultipleChoice', 14, 5, 'S3', 'Easy');

-- Insert QuestionChoices (Multiple Choice Options)
INSERT INTO dbo.QuestionChoices (QuestionID, ChoiceText0, ChoiceLetter, IsCorrect)
VALUES
-- Q2: HTML is a programming language.
(2, 'True', 'A', 0),
(2, 'False', 'B', 1)

-- Q4: Which HTML tag is used for line breaks?
(4, '<br>', 'A', 1),
(4, '<break>', 'B', 0),
(4, '<lb>', 'C', 0),
(4, '<newline>', 'D', 0)

-- Q5: What is the correct way to declare a variable in JavaScript?
(5, 'let variableName;', 'A', 1),
(5, 'variable variableName;', 'B', 0),
(5, 'v variableName;', 'C', 0),
(5, 'declare variableName;', 'D', 0)

-- Q6: JavaScript is case-sensitive.
(6, 'True', 'A', 1),
(6, 'False', 'B', 0)

-- Q8: Which method adds an element to the end of an array?
(8, 'push()', 'A', 1),
(8, 'add()', 'B', 0),
(8, 'append()', 'C', 0),
(8, 'insert()', 'D', 0)

-- Q9: What does DOM stand for?
(9, 'Document Object Model', 'A', 1),
(9, 'Data Object Management', 'B', 0),
(9, 'Dynamic Object Method', 'C', 0),
(9, 'Document Oriented Model', 'D', 0)

-- Q10: What is the correct file extension for Python files?
(10, '.py', 'A', 1),
(10, '.python', 'B', 0),
(10, '.pt', 'C', 0),
(10, '.pyt', 'D', 0)

-- Q11: Python is an interpreted language.
(11, 'True', 'A', 1),
(11, 'False', 'B', 0)
```

SQL SEED DATA

Example of insertion into tables

```
-- Insert QuestionText (Text Question Details)
INSERT INTO dbo.QuestionText (QuestionID, TextQText, TextAnswer, AcceptedPattern, IsCorrect)
VALUES
(4, 'Explain the box model in CSS.', 'The CSS box model describes rectangular boxes with content, padding, border, and margin areas.', '%content%padding%border%margin%', 1),
(8, 'Explain the difference between let and var.', 'let has block scope while var has function scope. let prevents hoisting issues.', '%let%var%scope%', 1),
(13, 'Explain list comprehension in Python.', 'List comprehension provides a concise way to create lists using a single line of code.', '%list%comprehension%create%', 1),
(18, 'Explain two-factor authentication.', 'Two-factor authentication adds extra security by requiring two different authentication factors.', '%two%factor%authentication%security%', 1),
(20, 'What is a Firewall?', 'A firewall is a network security system that monitors and controls network traffic based on security rules.', '%firewall%network%security%traffic%', 1),
(23, 'Name three major cloud service providers.', 'Amazon Web Services (AWS), Microsoft Azure, Google Cloud Platform (GCP)', '%AWS%Azure%Google%', 1),
(24, 'What is auto-scaling in cloud computing?', 'Auto-scaling automatically adjusts computing resources based on demand to maintain performance.', '%auto%scaling%resources%demand%', 1);
GO

-- Insert QuestionsExam (Link Questions to Exams)

INSERT INTO dbo.QuestionsExam (ExamID, QuestionID, QuestionDegree, QuestionOrder)
VALUES
HTML & CSS Final Exam (ExamID = 3, 10 questions)
(3, 2, 10, 1), (3, 3, 10, 2), (3, 4, 10, 3), (3, 5, 15, 4), (3, 6, 10, 5),
(3, 7, 10, 6), (3, 8, 15, 7), (3, 9, 10, 8), (3, 10, 10, 9), (3, 11, 10, 10),

-- JavaScript Midterm (ExamID = 4, 8 questions)
(4, 5, 10, 1), (4, 6, 10, 2), (4, 7, 10, 3), (4, 8, 15, 4),
(4, 9, 10, 5), (4, 3, 10, 6), (4, 4, 10, 7), (4, 2, 5, 8),

-- Python Data Science Quiz (ExamID = 5, 5 questions)
(5, 10, 10, 1), (5, 11, 10, 2), (5, 12, 15, 3), (5, 13, 10, 4), (5, 14, 5, 5),

-- Network Security Assessment (ExamID = 6, 8 questions)
(6, 15, 8, 1), (6, 16, 8, 2), (6, 17, 12, 3), (6, 18, 8, 4),
(6, 19, 12, 5), (6, 20, 8, 6), (6, 21, 8, 7), (6, 22, 8, 8),

-- Cloud Computing Practical (ExamID = 7, 6 questions)
(7, 20, 12, 1), (7, 21, 13, 2), (7, 22, 15, 3),
(7, 23, 15, 4), (7, 24, 10, 5), (7, 16, 10, 6),

-- React Development Project (ExamID = 8, 6 questions)
(8, 5, 12, 1), (8, 6, 13, 2), (8, 7, 15, 3),
(8, 8, 12, 4), (8, 9, 12, 5), (8, 4, 12, 6),

-- Machine Learning Final (ExamID = 9, 10 questions)
(9, 10, 7, 1), (9, 11, 7, 2), (9, 12, 10, 3), (9, 13, 7, 4), (9, 14, 7, 5),
(9, 20, 6, 6), (9, 21, 6, 7), (9, 22, 10, 8),
(9, 23, 10, 9), (9, 24, 6, 10);

-- Insert StudentsExam (Assign Students to Exams)
INSERT INTO dbo.StudentsExam (ExamID, StudentID, AssignedDate, StudentMark, StartDateTime, EndDateTime)
VALUES
(3, 1, '2025-03-01 08:00:00', 85.00, '2025-03-15 09:00:00', '2025-03-15 11:00:00'),
(3, 3, '2025-03-01 08:00:00', 92.00, '2025-03-15 09:00:00', '2025-03-15 11:00:00'),
(3, 11, '2025-03-01 08:00:00', 78.00, '2025-03-15 09:00:00', '2025-03-15 11:00:00'),

-- ExamID = 4 (JavaScript Midterm)
(4, 2, '2025-03-02 08:00:00', 88.00, '2025-03-16 09:00:00', '2025-03-16 11:00:00'),
(4, 5, '2025-03-02 08:00:00', 91.00, '2025-03-16 09:00:00', '2025-03-16 11:00:00'),
(4, 7, '2025-03-02 08:00:00', 76.00, '2025-03-16 09:00:00', '2025-03-16 11:00:00'),

-- ExamID = 5 (Python Data Science Quiz)
(5, 4, '2025-03-03 08:00:00', 89.00, '2025-03-17 09:00:00', '2025-03-17 11:00:00'),
(5, 6, '2025-03-03 08:00:00', 93.00, '2025-03-17 09:00:00', '2025-03-17 11:00:00'),
(5, 8, '2025-03-03 08:00:00', 81.00, '2025-03-17 09:00:00', '2025-03-17 11:00:00'),

-- ExamID = 6 (Network Security Assessment)
(6, 9, '2025-03-04 08:00:00', 84.00, '2025-03-18 09:00:00', '2025-03-18 11:00:00'),
(6, 10, '2025-03-04 08:00:00', 79.00, '2025-03-18 09:00:00', '2025-03-18 11:00:00'),
(6, 2, '2025-03-04 08:00:00', 91.00, '2025-03-18 09:00:00', '2025-03-18 11:00:00'),

-- ExamID = 7 (Cloud Computing Practical)
(7, 3, '2025-03-05 08:00:00', 87.00, '2025-03-19 09:00:00', '2025-03-19 11:00:00'),
(7, 6, '2025-03-05 08:00:00', 90.00, '2025-03-19 09:00:00', '2025-03-19 11:00:00'),
(7, 12, '2025-03-05 08:00:00', 74.00, '2025-03-19 09:00:00', '2025-03-19 11:00:00'),

-- ExamID = 8 (React Development Project)
(8, 5, '2025-03-06 08:00:00', 92.00, '2025-03-20 09:00:00', '2025-03-20 11:00:00'),
(8, 7, '2025-03-06 08:00:00', 88.00, '2025-03-20 09:00:00', '2025-03-20 11:00:00'),
(8, 9, '2025-03-06 08:00:00', 80.00, '2025-03-20 09:00:00', '2025-03-20 11:00:00'),

-- ExamID = 9 (Machine Learning Final)
(9, 8, '2025-03-07 08:00:00', 86.00, '2025-03-21 09:00:00', '2025-03-21 11:00:00'),
(9, 10, '2025-03-07 08:00:00', 90.00, '2025-03-21 09:00:00', '2025-03-21 11:00:00'),
(9, 11, '2025-03-07 08:00:00', 77.00, '2025-03-21 09:00:00', '2025-03-21 11:00:00');

-- Insert CorrectiveExam
INSERT INTO dbo.CorrectiveExam (ExamID, Reason, ApprovedBy, IsApproved, QuestionsNumbers)
VALUES
(3, 'Student failed original exam and requested retake due to technical issues', 2, 1, 10),
(4, 'Make-up exam for absent student due to medical emergency', 3, 1, 8),
(6, 'Retake exam approved after appeal for unfair grading', 7, 1, 12),
(7, 'Additional attempt granted due to system malfunction during original exam', 8, 0, 6);
```

SQL SEED DATA

Example of insertion into tables

```
-- Insert StudentAnswers
<--> INSERT INTO dbo.StudentAnswers (StudentID, ExamID, QuestionID, StudentAnswer, IsCorrect, MarksObtained)
VALUES
-- ExamID 3 ? HTML & CSS Final Exam
(1, 3, 2, 'False', 1, 1),
(1, 3, 3, 'The box model includes margin, border, padding, and content', 1, 1),
(1, 3, 4, '<br>', 1, 1),

-- ExamID 4 ? JavaScript Midterm
(2, 4, 5, 'var x = 10;', 1, 1),
(2, 4, 6, 'True', 1, 1),
(2, 4, 7, 'let is block scoped, var is function scoped', 1, 1),
(2, 4, 8, 'push()', 1, 1),
(2, 4, 9, 'Document Object Model', 1, 1),

-- ExamID 5 ? Python Data Science Quiz
(3, 5, 10, '.py', 1, 1),
(3, 5, 11, 'True', 1, 1),
(3, 5, 12, '[x for x in range(5)]', 1, 1),
(3, 5, 13, 'def', 1, 1),
(3, 5, 14, '<class ''list''>', 1, 1),

-- ExamID 6 ? Network Security Assessment
(4, 6, 15, 'Hypertext Transfer Protocol Secure', 1, 1),
(4, 6, 16, 'True', 1, 1),
(4, 6, 17, 'Something you know and something you have', 1, 1),
(4, 6, 18, '443', 1, 1),
(4, 6, 19, 'A system that monitors and controls network traffic', 1, 1),

-- ExamID 7 ? Cloud Computing Practical
(5, 7, 20, 'Software as a Service', 1, 1),
(5, 7, 21, 'True', 1, 1),
(5, 7, 22, 'AWS, Azure, Google Cloud', 1, 1),
(5, 7, 23, 'Automatically adjusts resources based on demand', 1, 1),
(5, 7, 24, 'Amazon S3', 1, 1);

-- Insert StudentCourse (Student Course Enrollment and Results)
<--> INSERT INTO dbo.StudentCourse (StudentID, CourseID, TotalScore, FinalGrade)
VALUES
(1, 15, 62.00, 'C'),
(2, 5, 89.75, 'A'),
(2, 6, 91.50, 'A'),
(2, 7, 78.25, 'B'),
(3, 3, 85.75, 'A'),
(4, 9, 88.25, 'A'),
(4, 11, 82.00, 'B'),
(5, 9, 76.75, 'B'),
(5, 10, 84.50, 'B'),
(6, 3, 95.25, 'A'),
(6, 4, 87.75, 'A'),
(7, 9, 82.50, 'B'),
(7, 10, 79.25, 'B'),
(8, 12, 68.75, 'C'),
(8, 13, 71.50, 'B'),
(9, 15, 55.00, 'F'),
(10, 12, 71.25, 'B'),
(10, 14, 77.50, 'B'),
(12, 5, 42.50, 'F'),
(12, 6, 93.75, 'A'),
(13, 7, 89.25, 'A'),
(13, 8, 86.50, 'A'),
(14, 9, 73.25, 'B'),
(15, 3, 87.00, 'A'),
(15, 4, 91.25, 'A');
```

SECURITY & ROLE-BASED PERMISSION

Roles and Permissions

• Admin

- Full system control.
- Create and manage Training Managers.
- Create and manage Instructors.
- Perform backup/restore and schema changes.

• Training Manager

- Manage training operations within a branch.
- Add and manage Instructors.
- Add Students.
- Create and manage Courses.
- Assign Instructors to Courses.

• Instructor

- Create and manage Exams.
- Enter exam questions.
- Review and grade Student answers.

• Student

- Register in Courses.
- Attempt Exams.
- View Results and Feedback.
- Get Exam Roster

Permission	Admin	Training Manager	Instructor	Student
Full system control	✓	✗	✗	✗
Manage Training Managers	✓	✗	✗	✗
Manage Instructors	✓	✓	✗	✗
Manage Students	✓	✓	✗	✗
Manage Courses	✓	✓	✗	✗
Create Exams	✓	✗	✓	✗
Add Exam Questions	✓	✗	✓	✗
Grade Students	✓	✗	✓	✗
Register in Courses	✗	✗	✗	✓

DATABASE OBJECTS

OBJECT NAME	TYPE	DESCRIPTION
sec.ProvisionApp UserLogin	Stored Procedure	Creates SQL Login + DB User, assigns the correct Role based on Users table.
AddStudentBy Manager	Stored Procedure	Allows Training Manager to add a new Student to the system.
AddInstructorBy Manager	Stored Procedure	Allows Training Manager to add a new Instructor to the system.
sec.sp_Manager AddBranch	Stored Procedure	Allows the Training Manager to add a new branch under their management
sec.sp_Manager DeleteBranch	Stored Procedure	Allows the Training Manager to delete one of their own branches (only if it)
sec.sp_Manager AddIntake	Stored Procedure	Adds a new intake (academic period) with name, start/end dates, and year
sec.sp_Manager AddDepartment	Stored Procedure	Adds a new department under a given track.
sp_Student SubmitAnswer	Stored Procedure	Allows the student to submit an answer for a question in an exam (defaults IsCorrect=0, MarksObtained=0 until evaluated).

DATABASE OBJECTS

OBJECT NAME	TYPE	DESCRIPTION
sp_Instructor_AddExam	Stored Procedure	Allows the instructor to add a new exam to one of their own courses. Includes authorization check to ensure the course belongs to them.
sp_Instructor_AssignStudent>ToExam	Stored Procedure	Allows the instructor to assign a student to an exam they own (and initializes the student's mark as 0).
v_Instructor_Courses	View	Lists the courses taught by the currently logged-in instructor.
v_Instructor_StudentsGrades	View	Shows the students of the current instructor and their grades in the instructor's courses.
v_Instructor_Exams	View	Displays the exams that belong to the courses of the current instructor
v_Instructor_Questions	View	Retrieves the questions related to the courses of the current instructor.
v_Manager_Branches	View	Lists all branches managed by the currently logged-in Training Manager.
v_Manager_Students	View	Displays all students belonging to the Training Manager's branches.

DATABASE OBJECTS

OBJECT NAME	TYPE	DESCRIPTION
v_Manager_Courses	View	Shows all courses in the Training Manager's branches, with instructor and branch details.
v_Manager_Exams	View	Retrieves all exams in the Training Manager's branches, including course, instructor, and branch info.
v_Student_Exams	View	Lists all exams available to the currently logged-in student, with course and instructor details.
v_Student_Exam_Questions	View	Displays the exam questions assigned to the student, including possible choices if applicable.
v_Student_Answers	View	Shows the student's submitted answers, correctness, and obtained marks.
v_Student_CourseGrades	View	Provides the student's transcript: courses, total score, and final grade.
v_Student_AssignedExams	View	Lists all exams assigned to the student, including assignment date and exam schedule.
v_Student_Assigned_Exams_Friendly	View	Displays upcoming, finished, or in-progress exams for the student, with days remaining until the start.

DATABASE OBJECTS

OBJECT NAME	TYPE	DESCRIPTION
trg_Students_Audit	Trigger	Logs all INSERT, UPDATE, DELETE operations on Students table into AuditLog with the action type and the SQL Server user (SUSER_SNAME()).
trg_Exams_Audit	Trigger	Logs all INSERT, UPDATE, DELETE operations on Exams table into AuditLog.
trg_Manager_Branches	Trigger	Logs all manager actions on Branches (insert, update, delete) into ManagerLog, linked to the responsible TrainingManager
trg_Manager_Tracks	Trigger	Logs manager actions on Tracks into ManagerLog. Currently stores ManagerID as NULL
trg_Manager_Intakes	Trigger	Logs manager actions on Intakes into ManagerLog. Stores ManagerID as NULL.
TRG_Exams_AutoEndTime	Trigger	Automatically calculates and sets EndDateTime for an exam when inserted/updated, based on StartDateTime + ExamDuration. Ensures consistency even when values change.
FN_CheckTextAnswer	Function	Checks a student's text answer against the accepted pattern for the question and returns whether it is correct (1) or not (0).
FN_CalculateExamResult	Function	Calculates the total score of a student for a given exam.

DATABASE OBJECTS

OBJECT NAME	TYPE	DESCRIPTION
FN_GetTotalEnrollmentStudent	Function	Returns the total number of enrolled students in the system.
FN_CheckTrueFalse	Function	Validates a student's True/False answer and returns a text result (Correct - Grade: X or Wrong - Grade: 0).
FN_GetExamDetails	Function	Retrieves exam details, including exam info, course name, and instructor name.
dashboardOfExam	Function	Provides exam statistics: total students, highest grade, lowest grade, and average grade.

EXECUTION

USER : INSTRUCTOR

The screenshot shows the execution of SQL queries in three different environments:

- Top Environment:** A Microsoft SQL Server Management Studio (SSMS) instance connected to 'inst_rahma'. The Object Explorer shows the database structure. The Query Editor window contains the following T-SQL code:


```

1 SELECT * FROM dbo.v_Instructor_Courses
2 SELECT * FROM dbo.v_Instructor_Exams
3 SELECT * FROM dbo.v_Instructor_Questions
4 SELECT * FROM dbo.v_Instructor_StudentsGrades
      
```

 The Results pane displays four tables:

	CourseID	CourseName	CourseDescription	CourseHour	MaxDegree	MinDegree	IsActive	TrackName	InstructorName
1	7	Python for Data Science	Python programming for data analysis and visual...	60	100	60	1	Data Science	Rahma Mohammed
2	5	Python Data Science Quiz	Quiz	50	60	2025-05-05 14:00:00.000	2025-05-15 00:00:00.000	Python for Data Science	
3	14	Python Data Science Mid	Mid	50	60	2025-05-20 14:00:00.000	2025-05-20 15:00:00.000	Python for Data Science	

	QuestionID	QuestionText	QuestionType	CourseName
3	12	Explain list comprehension in Python.	Text	Python for Data Science
4	13	Which keyword is used to define a function in Python?	MultipleChoice	Python for Data Science
5	14	What is the output of print(type([]))?	MultipleChoice	Python for Data Science
6	25	What is a p-value in statistics	MultipleChoice	Python for Data Science
7	57	What is a p-value in statistics	MultipleChoice	Python for Data Science
8	89	What is a p-value in statistics	MultipleChoice	Python for Data Science
9	121	What is a p-value in statistics	MultipleChoice	Python for Data Science
10	122	What is a p-value in statistics	MultipleChoice	Python for Data Science

	StudentID	StudentName	CourseName	TotalScore	FinalGrade
1	2	Sara Gamal	Python for Data Science	70.25	B
2	13	Fatma Omar	Python for Data Science	89.25	A
- Middle Environment:** Another SSMS instance connected to 'inst_rahma'. The Object Explorer shows the database structure. The Query Editor window contains the following T-SQL code:


```

1 SELECT * FROM dbo.v_Instructor_Courses
2 SELECT * FROM dbo.v_Instructor_Exams
3 SELECT * FROM dbo.v_Instructor_Questions
4 SELECT * FROM dbo.v_Instructor_StudentsGrades
5
6
7 EXEC dbo.sp_Instructor_AddExam 7, 'Python Data Science Final', 'Final', 100, 120, '2025-06-20'
8 EXEC dbo.sp_Instructor_AssignStudentToExam 14, 2, '2025-06-20 14:00:00.000', '2025-06-20
9
10
      
```

 The Messages pane shows the execution results:


```

Operation logged in AuditLog table.
Operation logged in AuditLog table.

Completion time: 2025-06-20T23:08:00.4486600+03:00
      
```
- Bottom Environment:** A Microsoft Visual Studio instance showing a Solution Explorer with a file named 'SQLQuery1.sql'. The Object Explorer shows the database structure. The Query Editor window contains the same T-SQL code as the middle environment:


```

1 SELECT * FROM dbo.v_Instructor_Courses
2 SELECT * FROM dbo.v_Instructor_Exams
3 SELECT * FROM dbo.v_Instructor_Questions
4 SELECT * FROM dbo.v_Instructor_StudentsGrades
5
6
7 EXEC dbo.sp_Instructor_AddExam 7, 'Python Data Science Final', 'Final', 100, 120, '2025-06-20'
8 EXEC dbo.sp_Instructor_AssignStudentToExam 15, 2, '2025-06-20 14:00:00.000', '2025-06-20
9
10
      
```

 The Messages pane shows the execution results:


```

Commands completed successfully.
Completion time: 2025-06-20T23:16:39.0074760+03:00
      
```

EXECUTION

USER : STUDENT

Object Explorer

localhost (SQL Server 16.0.1000.6 - stud_ahmed)

Databases

Tables

Views

System Views

dbo.v_Student_Answers

dbo.v_Student_AssignedExams

dbo.v_Student_AssignedExams_Friendly

dbo.v_Student_CourseGrades

dbo.v_Student_ExamQuestions

dbo.v_Student_Exams

dbo.v_StudentUpcomingExams

Dropped Ledger Views

External Resources

Synonyms

Programmability

Stored Procedures

System Stored Procedures

dbo.sp_Student_AssignedExams

dbo.sp_StudentSubmitAnswer

Functions

Database Triggers

Assemblies

Types

Rules

Defaults

Ready

Scenario.sql ...d_ahmed (53) views&spStude_stem (sa (65))

```
--Log in by username : stud_ahmed Password : stud123
SELECT * FROM V_Student_Exams;
SELECT * FROM V_Student_ExamQuestions;
SELECT * FROM V_Student_Answers;
SELECT * FROM V_Student_CourseGrades;
SELECT * FROM V_Student_AssignedExams;
SELECT * FROM V_Student_AssignedExams_Friendly;
```

Result Messages

ExamID	ExamTitle	ExamType	TotalDegrees	ExamDuration	StartTime	EndTime	CourseName	InstructorName
1	HTML & CSS Final Exam	Final	100	120	2025-03-15 09:00:00.000	2025-03-15 11:00:00.000	HTML & CSS Fundamentals	Rawha Abdulrahman

ExamID	QuestionID	QuestionText	QuestionType	ChoiceLetter	ChoiceTextQ
1	3	HTML is a programming language.	TrueFalse	A	True
2	3	HTML is a programming language.	TrueFalse	B	False
3	3	Explain the box model in CSS.	Text	NULL	NULL
4	3	Which HTML tag is used for line breaks?	MultipleChoice	A	
5	3	Which HTML tag is used for line breaks?	MultipleChoice	B	<break>
6	3	Which HTML tag is used for line breaks?	MultipleChoice	C	<lb>
7	3	Which HTML tag is used for line breaks?	MultipleChoice	D	<newline>
8	3	What is the correct way to declare a variable in...	MultipleChoice	A	let variableName:

AnswerID	ExamID	StudentID	QuestionText	StudentAnswer	IsCorrect	MarksObtained
1	6	3	HTML is a programming language.	False	0	0.00
2	7	3	Explain the box model in CSS.	The box model includes margin, border, padding...	0	1.00
3	8	3	Which HTML tag is used for line breaks?	 	1	0.00
4	44	3	Explain the box model in CSS.	let variableName:	0	0.00
5	46	3	What is the correct way to declare a var...	let variableName:	0	0.00
6	48	3	HTML is a programming language.	B	0	0.00

StudentID	StudentName	CourseName	TotalScore	FinalGrade
1	Ahmed Mohamed	DevOps Fundamentals	62.00	C

ExamID	StudentID	AssignedDate	StartTime	EndTime	StudentMark
1	3	2025-03-01 08:00:00.000	2025-03-15 09:00:00.000	2025-03-15 11:00:00.000	85.00

ExamID	ExamTitle	CourseName	AssignedDate	StartTime	EndTime	StudentMark	TimeMessage	ExamStatus
1	HTML & CSS Final Exam	HTML & CSS Fundamentals	2025-03-01 08:00:00.000	2025-03-15 09:00:00.000	2025-03-15 11:00:00.000	85.00	Finished 160 days ago	Finished

Query executed successfully.

localhost (16.0 RTM) stud_ahmed (53) ITIExaminationSystem 00:00:00 38 rows

Object Explorer

localhost (SQL Server 16.0.1000.6 - sa)

localhost (SQL Server 16.0.1000.6 - stud_ahmed)

Databases

Tables

Views

System Views

dbo.v_Student_Answers

dbo.v_Student_AssignedExams

dbo.v_Student_AssignedExams_Friendly

dbo.v_Student_CourseGrades

dbo.v_Student_ExamQuestions

dbo.v_Student_Exams

Ready

Scenario.sql ...d_ahmed (53) views&spStude_stem (sa (65))

```
EXEC sp_StudentSubmitAnswer
    @ExamID = 3,
    @QuestionID = 2,
    @StudentAnswer = 'B';
```

Result Messages

Commands completed successfully.

Completion time: 2025-03-22T20:59:46.420Z 2025-03-22T20:59:46.420Z

EXECUTION

USER : TRAININGMANAGER

Object Explorer

```
--Log in by username : tm_mohammed Password : tm456
SELECT * from dbo.v_Manager_Branches
SELECT * from dbo.v_Manager_Courses
SELECT * from dbo.v_Manager_Students
```

Results

BranchID	BranchName	BranchEmail	BranchPhone	ManagerName
1	Alexandria Branch	alexandria@company.com	0334567890	Mohamed Ali
2	Minia Branch	minia@company.com	0862456789	Mohamed Ali
3	Fayoum	Fayoum123@gmail.com	01098765420	Mohamed Ali

CourseID CourseName CourseDescription ModDegree MinDegree CourseHours InstructorName BranchName

CourseID	CourseName	CourseDescription	ModDegree	MinDegree	CourseHours	InstructorName	BranchName
1	React Development	Modern frontend framework for building user int.	100	60	45	Youssif Mahmoud	Minia Branch
2	Node.js Backend	Server-side JavaScript development	100	60	50	Bishoy Samir	Minia Branch

StudentID FirstName LastName Email Phone BranchName

StudentID	FirstName	LastName	Email	Phone	BranchName
1	Youself	Ibrahim	youssef.ibrahim@student.com	01000000003	Alexandria Branch
2	Fardaa	Mohamed	fardaa.mohamed@student.com	01000000006	Minia Branch
3	Fatma	Omar	fatma.omar@student.com	01000000013	Alexandria Branch
4	Dina	Youself	dina.youssef@student.com	01000000015	Minia Branch
5	Youself	Ibrahim	youssef.ibrahim@student.com	01000000003	Alexandria Branch
6	Fardaa	Mohamed	fardaa.mohamed@student.com	01000000006	Minia Branch
7	Fatma	Omar	fatma.omar@student.com	01000000013	Alexandria Branch
8	Dina	Youself	dina.youssef@student.com	01000000015	Minia Branch

Query executed successfully.

Object Explorer

```
EXEC dbo.AddStudentByManager
@ManagerUserID = 3,
@Username = 'kenzy_stud',
@Password = '123456',
@NationalID = '82345678901234',
@FirstName = 'kenzy',
@LastName = 'ahmed',
@email = 'kenzy@student.com',
@Phone = '01000000000',
@Address = 'giza',
@EnrollmentDate = '2025-08-18',
@BranchIntakeTrackID = 1;
```

Messages

Operation logged in AuditLog table.
Student added successfully by Training Manager.
Completion time: 2025-08-22T21:11:35.9179410+03:00

No issues found

Query executed successfully.

Object Explorer

```
EXEC AddInstructorByManager
@ManagerUserID = 3,
@Username = 'Ehab.inst',
@Password = 'inst0987',
@NationalID = '98765432101234',
@FirstName = 'Ehab',
@LastName = 'Hassan',
@email = 'Ehab.hassan@example.com',
@Phone = '01000000002',
@HireDate='',
@Specialization = 'Database',
@BranchIntakeTrackID = 1;
```

```
EXEC dbo.AddStudentByManager
```

Instructor added successfully by Training Manager.
Completion time: 2025-08-22T21:12:28.9489729+03:00

No issues found

Query executed successfully.

EXECUTION

USER : TRAININGMANAGER

The screenshot shows the Object Explorer on the left and the Results pane on the right. The results pane displays the output of a stored procedure execution:

```
EXEC sec.sp_ManagerAddBranch  
'Nasr_City' , 'NasrCity@gmail.com' , '01999643501'  
EXEC sec.sp_ManagerAddIntake  
'Intake 47' , '2026-07-30' , '2027-07-30' , 2027  
EXEC sec.sp_ManagerDeleteBranch 7
```

Below the results, a message box shows:

Manager action logged in ManagerLog table
Completion timer: 2025-08-22T21:13:27.2967174+03:00

The screenshot shows the Object Explorer on the left and the SQL Editor on the right. The SQL Editor contains the following T-SQL code:

```
55    @BranchName = 'Al-Med',
56    @Email = 'kenzy@student.com',
57    @Phone = '01000000000',
58    @Address = 'giza',
59    @EnrollmentDate = '2025-08-18',
60    @BranchIntakeTrackID = 1;
61
62
63 EXEC sec.sp_ManagerAddBranch
64      'Nasr_City' , 'NasrCity@gmail.com' , '01999643501'
65
66 EXEC sec.sp_ManagerAddIntake
67      'Intake 48' , '2027-07-30' , '2028-07-30' , 2028
68
69
70
```

The line `EXEC sec.sp_ManagerAddIntake` is highlighted with a red box. Below the code, the message pane shows:

Commands completed successfully.
Completion time: 2025-08-22T21:14:00.5026934+03:00

The status bar at the bottom indicates "Query executed successfully."

EXECUTION

USER : ADMIN

The screenshot shows the Object Explorer on the left and the Results pane on the right. In the Object Explorer, a red arrow points to the 'Stored Procedures' node under the 'sec' schema. In the Results pane, the command `EXEC sec.sp_ManagerDeleteBranch 8` is run, and the output shows:

```
1 EXEC sec.sp_ManagerDeleteBranch 8
146 % No issues found
Messages
Manager action logged in ManagerLog table
Completion time: 2025-08-22T21:42:09.3360467+03:00
Ln: 1 Ch: 38 SPC CRLF
```

FUNCTIONS

The screenshot shows the Object Explorer on the left and the Results pane on the right. In the Object Explorer, the 'Functions' node under the 'dbo' schema is expanded. In the Results pane, the command `SELECT * FROM dbo.FN_GetExamDetails(3);` is run, and the output shows:

```
96
97 --Functions
98 SELECT dbo.FN_CalculateExamResult(2, 4) AS StudentTotalMark;
99 SELECT dbo.FN_GetTotalEnrollmentStudent() AS TotalStudents;
100 SELECT dbo.FN_CheckTextAnswer('The CSS box model', 4, 1) AS CheckText;
101 SELECT dbo.FN_CheckMultipleChoiceAnswer(4, 8, 2) AS TF_Result;
102 SELECT * FROM dbo.FN_GetExamDetails(3);
103 SELECT * FROM dbo.dashboardOfExam(4);
```

Results

StudentTotalMark
4.00

TotalStudents
31

CheckText
1

TF_Result
Correct - Grade: 1.00

ExamID	ExamTitle	ExamType	TotalDegrees	ExamDuration	StartTime	EndTime	IsRandom...	QuestionsNumbers	CounselID	InstructorID	BranchIntakeTrackID	ExamMark	CourseName	InstructionName
1	HTML & CSS	Final	100	120	2025-03-15 09:00:00...	2025-03-15 11:00:00...	1	10	3	2	1	100	HTML & CSS Final Exam	Rawha Abdulrahman

Total_Students	Highest_Grade	Lowest_Grade	AverageMark
1	91.00	76.00	85.000000

DAILY FULL BACKUP

Steps Implemented

1. Enable SQL Server Agent

- Open SQL Server Configuration Manager
- Start SQL Server Agent from SQL Server Management Studio.

2. Create Backup Directory

- Created a folder on the server to store backups
- C:\SQLBackups\

3. Test Manual Backup (T-SQL)

```
1 BACKUP DATABASE [ITIExaminationSystem]
2 TO DISK = 'C:\SQLBackups\ExaminationSYS_20250821_141741.bak'
3 WITH INIT, NAME = 'Manual Test Backup';
```

Processed 920 pages for database 'ITIExaminationSystem', file 'ITIExaminationSystem' on file 1.
Processed 1 pages for database 'ITIExaminationSystem', file 'ITIExaminationSystem_log' on file 1.
BACKUP DATABASE successfully processed 921 pages in 0.050 seconds (143.779 MB/sec).

4. Create SQL Server Agent Job (Daily Backup Job)

