CREATE DATABASE ECCDATA;

USE ECCDATA;

CREATE TABLE Students (

studentNum INT PRIMARY KEY IDENTITY(1,1),

Firstname VARCHAR(50),

Lastname VARCHAR(50),

Email VARCHAR(50),

Password VARCHAR(50)

);

ALTER TABLE Students

ALTER COLUMN Password VARCHAR(500);

select \* from Students;

-- Insert prepopulated data

INSERT INTO Students (Firstname, Lastname, Email, Password)

VALUES

('Mbasa', 'Sitsheke', 's@gmail.com', 'password123'),

('Zanele', 'Ngcobo', 'zanele.n@gmail.com', 'pass456');

select \* from Students;

drop table Students;

CREATE TABLE Administrator (

adminID INT PRIMARY KEY IDENTITY(1,1),

Firstname VARCHAR(50),

Lastname VARCHAR(50),

Email VARCHAR(50),

Password VARCHAR(50)

);

CREATE TABLE Mentor (

mentorID INT PRIMARY KEY IDENTITY(1,1),

Firstname VARCHAR(50),

Lastname VARCHAR(50),

Email VARCHAR(50),

Password VARCHAR(50)

);

Select \* from Mentor;

-- Insert Mentor 1

INSERT INTO Mentor (Firstname, Lastname, Email, Password)

VALUES ('John', 'Doe', 'john.doe@example.com', 'password123');

-- Insert Mentor 2

INSERT INTO Mentor (Firstname, Lastname, Email, Password)

VALUES ('Jane', 'Smith', 'jane.smith@example.com', 'securepass456');

CREATE TABLE FundingGuide (

FundingGuideId INT PRIMARY KEY IDENTITY(1,1),

studentNum int ,

FundingPurpose VARCHAR(2000),

AmountRequested INT,

BussinessOverview VARCHAR(2000),

BussinessName VARCHAR(2000),

Mission VARCHAR(2000),

BussinessModel VARCHAR(2000),

TotalFunding VARCHAR(2000),

UseOfFunds VARCHAR(2000),

Expenses VARCHAR(2000),

Profitability VARCHAR(2000),

Industry VARCHAR(2000),

Competitors VARCHAR(2000),

MarketTrends VARCHAR(2000),

KeyMembersAndRoles VARCHAR(2000),

KeyMilestones VARCHAR(2000),

Timeline VARCHAR(2000),

Risks VARCHAR(2000),

RiskPlan VARCHAR(2000),

Summary VARCHAR(2000),

Name VARCHAR(2000),

Email VARCHAR(2000),

PhoneNumber VARCHAR(2000),

Foreign key (studentNum) references Students(studentNum),

);

CREATE TABLE BusinessProposal (

BusinessProposalId INT PRIMARY KEY IDENTITY(1,1),

studentNum int ,

ExecutiveSummary VARCHAR(2000),

Introduction VARCHAR(2000),

ProposedSolution VARCHAR(2000),

ProblemSolution VARCHAR(2000),

BudgetAndPenalties VARCHAR(2000),

Benefits VARCHAR(2000),

TermsAndConditions VARCHAR(2000),

Appendix VARCHAR(2000)

Foreign key (studentNum) references Students(studentNum),

);

select \* from BusinessProposal;

select \* from Students;

select \* from FundingGuide;

select \* from Administrator;

select \* from FundingGuide;

select \* from Mentor;