Create database Proj;

CREATE TABLE Campus (

campus\_id INT AUTO\_INCREMENT PRIMARY KEY,

campus\_name VARCHAR(50) UNIQUE,

physical\_location VARCHAR(100),

campus\_web\_address VARCHAR(100),

campus\_contact VARCHAR(20)

);

INSERT INTO Campus (campus\_name, physical\_location, campus\_web\_address, campus\_contact) VALUES

('AAiT', 'Amist kilo', 'http://www.aait.edu.et', '+251-11-232412'),

('6 kilo campus', '6 kilo', 'http://www.aau.edu.et', '+251-11-987654'),

('4 kilo campus', '4 kilo' , 'https://www.aau.edu.et', '+251-11-234564'),

('Sefere Selam Campus', 'Atena tera', 'http://www.aau.edu.et', '+251-11-543464'),

('EiBC Campus', 'Lideta', 'http://www.aau.edu.et', '+251-11-432574');

CREATE TABLE Proctor (

employee\_id INT AUTO\_INCREMENT PRIMARY KEY,

proctor\_password VARCHAR(50),

first\_name VARCHAR(50),

last\_name VARCHAR(50),

proctor\_contact VARCHAR(20),

proctor\_email VARCHAR(50),

employment\_date DATE,

campus\_id INT,

FOREIGN KEY (campus\_id) REFERENCES Campus(campus\_id)

);

INSERT INTO Proctor (proctor\_password, first\_name, last\_name, proctor\_contact, proctor\_email, employment\_date, campus\_id) VALUES

('\*\*\*\*\*\*\*\*', 'Samuel', 'Tessema', '0912345678', 'SamuelTessema@gmail.com', '2018-01-01', 1),

('\*\*\*\*\*\*\*\*', 'Tewodros', 'Abebe', '0913043666', 'Tewodrosabebe531@gmail.com', '2018-02-01', 1);

CREATE TABLE Room (

room\_number VARCHAR(20) PRIMARY KEY,

room\_type VARCHAR(20),

block VARCHAR(20),

payment\_per\_room DECIMAL(10, 2),

room\_status ENUM('occupied', 'vacant', 'under maintenance'),

number\_of\_students INT,

proctor\_id INT,

campus\_id INT,

FOREIGN KEY (proctor\_id) REFERENCES Proctor(employee\_id),

FOREIGN KEY (campus\_id) REFERENCES Campus(campus\_id)

);

INSERT INTO Room (room\_number, room\_type, block, payment\_per\_room, room\_status, number\_of\_students, proctor\_id, campus\_id) VALUES

('104', 'Proc-room', 'A', 25.00, 'occupied', 10, 1, 1),

('201', 'St-room', 'B', 20.00, 'vacant', 5, 2, 2);

CREATE TABLE Student (

student\_id VARCHAR(15) PRIMARY KEY,

first\_name VARCHAR(50),

last\_name VARCHAR(50),

department VARCHAR(20),

date\_of\_birth DATE,

email VARCHAR(50),

phone\_number VARCHAR(20),

emergency\_contact VARCHAR(20),

year\_of\_admission VARCHAR(4),

total\_debt DECIMAL(10, 2),

campus\_id INT,

employee\_id INT, -- This is a foreign key referencing Proctor(employee\_id)

room\_number VARCHAR(20), -- This is a foreign key referencing Room(room\_number)

FOREIGN KEY (campus\_id) REFERENCES Campus(campus\_id),

FOREIGN KEY (employee\_id) REFERENCES Proctor(employee\_id),

FOREIGN KEY (room\_number) REFERENCES Room(room\_number)

);

-- Inserting additional records into Student table

INSERT INTO Student (student\_id, first\_name, last\_name, department, date\_of\_birth, email, phone\_number, emergency\_contact, year\_of\_admission, total\_debt, campus\_id, employee\_id, room\_number) VALUES

('UGR/8267/15', 'Biruk', 'Seyoum', 'Software Engineering', '1997-08-20', 'bseyoum003@gmail.com', '0937454293', '0937373624', '2018', 0.00, 1, 1, '104'),

('UGR/8065/15', 'Yodit', 'Debebe', 'Computer Science', '1997-08-20', 'yodit@example.com', '0912345678', '0918345747', '2015', 0.00, 1, 1, '104'),

('UGR/2975/15', 'Talegeta', 'Belay', 'Physics', '1996-10-15', 'talegeta@example.com', '0912345678', '0918345747', '2015', 0.00, 1, 1, '104'),

('UGR/1559/15', 'Bemnet', 'Bisrat', 'Chemistry', '1998-03-25', 'bemnet@example.com', '0912345678', '0918345747', '2015', 0.00, 2, 2, '201'),

('UGR/1591/15', 'Abdulkerim', 'Neja', 'Mathematics', '1997-05-30', 'abdulkerim@example.com', '0912345678', '0918345747', '2015', 0.00, 2, 2, '201'),

('UGR/7623/15', 'Edilawit', 'Manaye', 'Biology', '1999-12-10', 'edilawit@example.com', '0912345678', '0918345747', '2015', 0.00, 1, 1, '104');

CREATE TABLE Payment (

payment\_id INT AUTO\_INCREMENT PRIMARY KEY,

student\_id VARCHAR(15),

amount DECIMAL(10, 2),

payment\_method VARCHAR(20),

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

campus\_id INT,

FOREIGN KEY (campus\_id) REFERENCES Campus(campus\_id)

);

INSERT INTO Payment (student\_id, amount, payment\_method, campus\_id) VALUES

('UGR/8267/15', 3500.00, 'cash', 1),

('UGR/1234/19', 2800.50, 'credit card', 2);

CREATE TABLE Facility (

facility\_id INT AUTO\_INCREMENT PRIMARY KEY,

facility\_name VARCHAR(50),

facility\_description VARCHAR(100),

created\_at DATE,

campus\_id INT,

FOREIGN KEY (campus\_id) REFERENCES Campus(campus\_id)

);

INSERT INTO Facility (facility\_name, facility\_description) VALUES

('Dormitory', 'Double rooms'),

('Library', 'Spacious study area');

-- Queries

-- Retrive firstname and proctor id from Room and Proctor table

select first\_name,proctor\_id from (Room join Proctor on proctor\_id=employee\_id);

-- Retrieve all rooms and their statuses for a specific campus

SELECT room\_number, room\_status FROM Room WHERE campus\_id = 1;

-- Update a payment record

UPDATE Payment SET amount = 3000.00 WHERE payment\_id = 1;

-- Add a new facility

INSERT INTO Facility (facility\_name, facility\_description, campus\_id) VALUES ('Gym', 'Fitness center with modern equipment', 2);

-- Delete a student record

DELETE FROM Student WHERE student\_id = 'UGR/1234/19';

-- Update proctor contact information

UPDATE Proctor SET proctor\_contact = '0912345000' WHERE employee\_id = 1;

-- Insert a new room record

INSERT INTO Room (room\_number, room\_type, block, payment\_per\_room, room\_status, number\_of\_students, proctor\_id, campus\_id)

VALUES ('301', 'single', 'C', 18.00, 'vacant', 2, 2, 1);

-- Retrieve all students along with their proctor's first and last names:

SELECT s.student\_id, s.first\_name AS student\_first\_name, s.last\_name AS student\_last\_name,

p.first\_name AS proctor\_first\_name, p.last\_name AS proctor\_last\_name

FROM Student s

JOIN Proctor p ON s.employee\_id = p.employee\_id;

-- Retrieve all payments along with the corresponding student's email for a specific campus:

SELECT p.amount, p.payment\_method, s.email AS student\_email

FROM Payment p

JOIN Student s ON p.student\_id = s.student\_id

WHERE p.campus\_id = 2;

-- Retrieve emain and amount from Student and Payment table

select email,amount from Student,Payment;