



Dhaka International University

Department of Computer Science and Engineering

Project Title: DIU Bus Management System.

Course Name: Database management system Lab.

Course Code: 0612-304

Group member

MD Emon Sarkar (08)

Farjana Hoque Remo (09)

Showrov Shahriar Pran (11)

Sidratul Montaha Ayshi (32)*

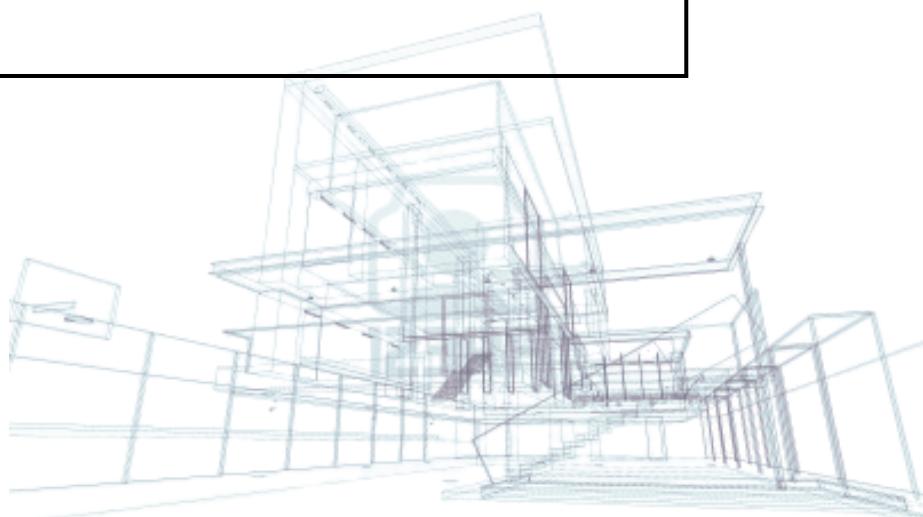


Table of Contents

Overview.....	1
Description.....	2
Motivation.....	2
Problem Statement.....	2
Solution Statement.....	3
Project Objective.....	3
Application.....	4
Bus Management System – with strong and weak entities	5 - 6
Bus Management System – Diagram with strong and weak entities.....	7 - 9
DIU Bus management system ER Diagram.....	10
DIU Bus management system ER Diagram to relation mapping	11
Table Creation.....	12 - 17
Code Implementation.....	18 - 21
Discussion.....	22
References.....	22
GitHub Link.....	22

Overview

The DIU Bus Management System is a web-based database application developed for the Database Management System course. The current university bus service operates entirely manually, with no fixed schedules, seat assignments, or maintenance records. As a result, students frequently stand during travel, teachers and staff lose priority seating, buses arrive late or skip stops, and unexpected breakdowns occur due to poor upkeep. These problems affect thousands of daily commuters, leading to wasted time, frustration, and reduced academic focus.

This project introduces a complete digital solution to organize campus transportation. The system manages bus routes, assigns vehicles to specific paths, maintains accurate schedules, and records every maintenance activity with dates, costs, and issues. It ensures that teachers and staff have reserved seats while allowing students to check available seats before boarding. All operations are handled through a central database using MySQL, with separate access for students, faculty, and administrators. Students can view their assigned bus and timing, teachers receive seat confirmation, and the admin controls routes, schedules, and repair logs.

The system is built on a strong database foundation. Data is stored across related tables for buses, routes, students, teachers, schedules, and maintenance. Relationships are defined using primary and foreign keys to maintain accuracy and prevent errors. The design follows normalization principles to eliminate redundancy and ensure efficient data handling. Role-based dashboards provide clear, real-time information to each user group.

By replacing paper-based processes with a digital platform, the system brings order and reliability to campus travel. Waiting times decrease, seating conflicts end, and vehicle reliability improves through planned maintenance. The result is a smoother, more comfortable, and punctual commuting experience for the entire university community. Future enhancements may include mobile access, automated notifications, and integration with campus services, but the current version already delivers a fully functional, database-driven management solution.

Description:

In our university, the existing bus transportation system is disorganized and inefficient. Students, teachers, and staff often face several problems:

There is no proper schedule or route tracking for buses. Students frequently have to stand during the journey due to a lack of available seats. Teachers and staff do not have designated seating arrangements, sometimes resulting in students occupying their seats. Buses occasionally miss stops or arrive late, causing students to miss classes. There is no way to know which bus is on which road, or how long it will take to reach a certain stop. Maintenance records of the buses are not tracked systematically, which may lead to unexpected breakdowns. Due to these issues, commuting becomes a daily hassle for everyone in the university, affecting punctuality, comfort, and overall satisfaction.

Motivation:

The motivation behind this project comes from the real challenges faced by DIU students, teachers and staff during daily bus travel. Long waiting times, no fixed seating system and confusion about schedule or route often lead to stress and late arrivals. Since the transportation system is mostly manual, there is no clear way to check seat availability or track buses in real time. We wanted to build a system that brings discipline, fairness and comfort to university transport. By digitalizing bus routes, seating priority and time management, we aim to reduce hassle and improve punctuality. This project is driven by the vision of turning a disorganized bus service into a smooth, smart and well-managed system. In short, our motivation is to make university travel easier, faster, secure, manageable, hassle free and more reliable for both students and teachers.

Problem Statement:

The existing bus system is not well organized and lacks proper schedules. Students often have to stand because there are not enough seats available. Teachers also face difficulties as they do not have any reserved seats. Sometimes buses miss their stops or arrive late, causing inconvenience to everyone. This situation creates problems for both students and teachers every day. Daily travel becomes stressful, time-consuming and inefficient. There is no proper system to check or record the maintenance of buses. Because of this, buses are not always in good condition and may face unexpected issues. The overall management of the bus system is poor and needs proper planning. A more organized and efficient transportation system is required to solve these problems.

Solution Statement:

- A smart and well-organized bus management system will be introduced to solve existing problems.
- The system will include proper schedules for all buses to ensure timely arrivals and departures.
- Students and teachers can book or reserve seats in advance to prevent overcrowding.
- Teachers will have reserved seats for more comfortable travel.
- A digital record will be kept for every bus's maintenance and servicing.
- The system will help manage repairs and maintenance on time.
- It will reduce stress and save time for both students and teachers.
- The system will make daily travel easier, safer, and more reliable.
- Overall, it will create an efficient and systematic bus service for the institution.

Project Objectives:

To solve these problems, we aim to develop a University Bus Management System that will help in:

- Monitoring which bus is assigned to which route.
- Tracking bus timings, stoppages, and estimated arrival times.
- Maintaining records of each bus's maintenance history.
- Ensuring smooth seating arrangements for students, teachers, and staff.
- Allowing users to check seat availability beforehand.
- Creating an organized route system that minimizes confusion and improves efficiency.

This system will bring transparency, reduce travel stress, and ensure a hassle-free journey for all university members. By digitalizing and organizing the bus service, we hope to improve daily transportation and make campus life more convenient.

Application:

- Automates seat allocation for students, teachers, and staff.
- Helps avoid overcrowding and ensures smoother travel experience.
- Improves time management through organized pickup and drop schedules.
- Provides clear route information to reduce confusion at bus stops.
- Allows students and staff to check seat availability before booking.
- Maintains bus maintenance history for improved safety and efficiency.
- Supports priority-based seating (e.g., teachers get reserved seats).
- Reduces manual work and paperwork through digital management.
- Generates useful reports (e.g., student count per route, seat utilization).
- Enhances campus transportation with a well-structured system.

Bus Management System – with strong and weak entities :

Entities:

1. Bus

```
--bus_id (PK)  
--bus_number  
--bus_type  
--capacity  
--registration_number  
--bus_name
```

2. Route

```
--route_id (PK)  
--route_name  
--start_point  
--end_point  
--total_distance  
--bus_id (FK)  
--estimate_time
```

3.Student

```
--student_id (PK)  
--s_name  
--roll_number  
--department  
--phone_number  
--email  
--address  
--bus_id (FK)  
--route_id (FK)  
--pickup_point
```

4. Teacher

```
--teacher_id (PK)  
--t_name  
--department  
--phone_number  
--email  
--address  
--pickup_point  
--bus_id (FK)  
--route_id (FK)
```

5. Bus_Schedule

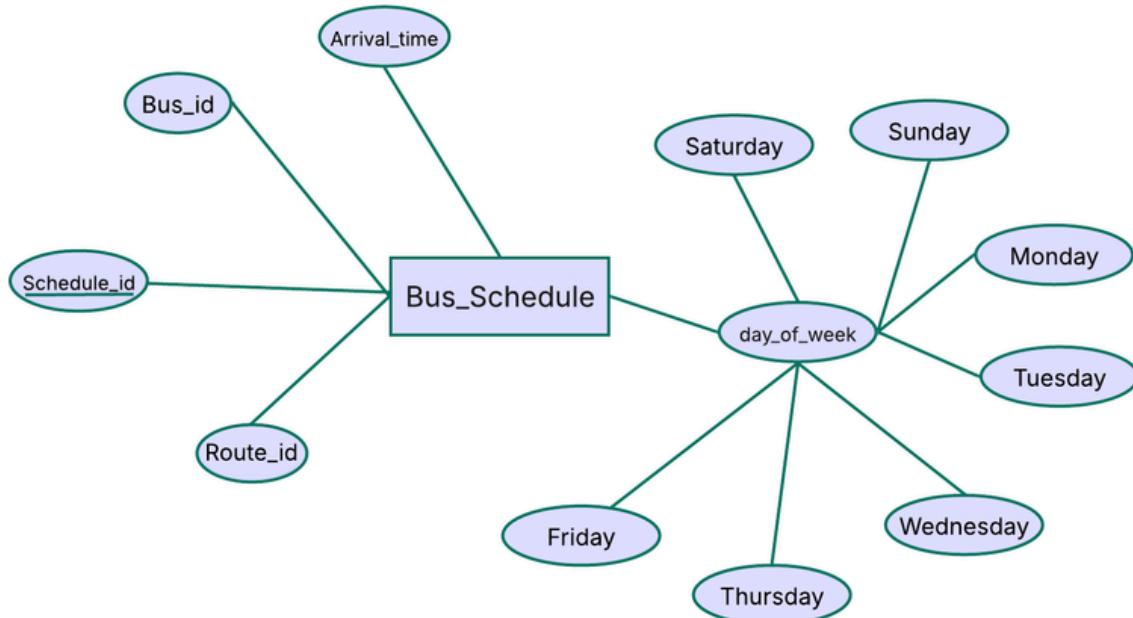
```
--schedule_id (PK)  
--bus_id (FK)  
--route_id (FK)  
--arrival_time  
--day_of_week
```

6. Maintenance (Weak Entity)

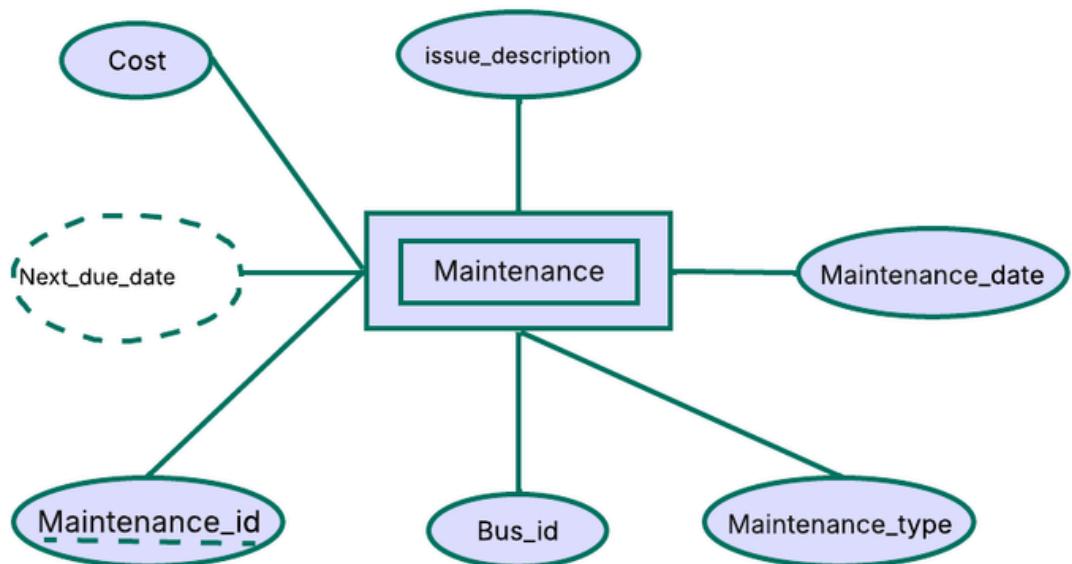
```
--bus_id (FK)  
--maintenance_date  
--maintenance_type  
--issue_description  
--cost  
--maintenance_id  
--next_due_date
```

Bus Management System – Diagram with strong and weak entities :

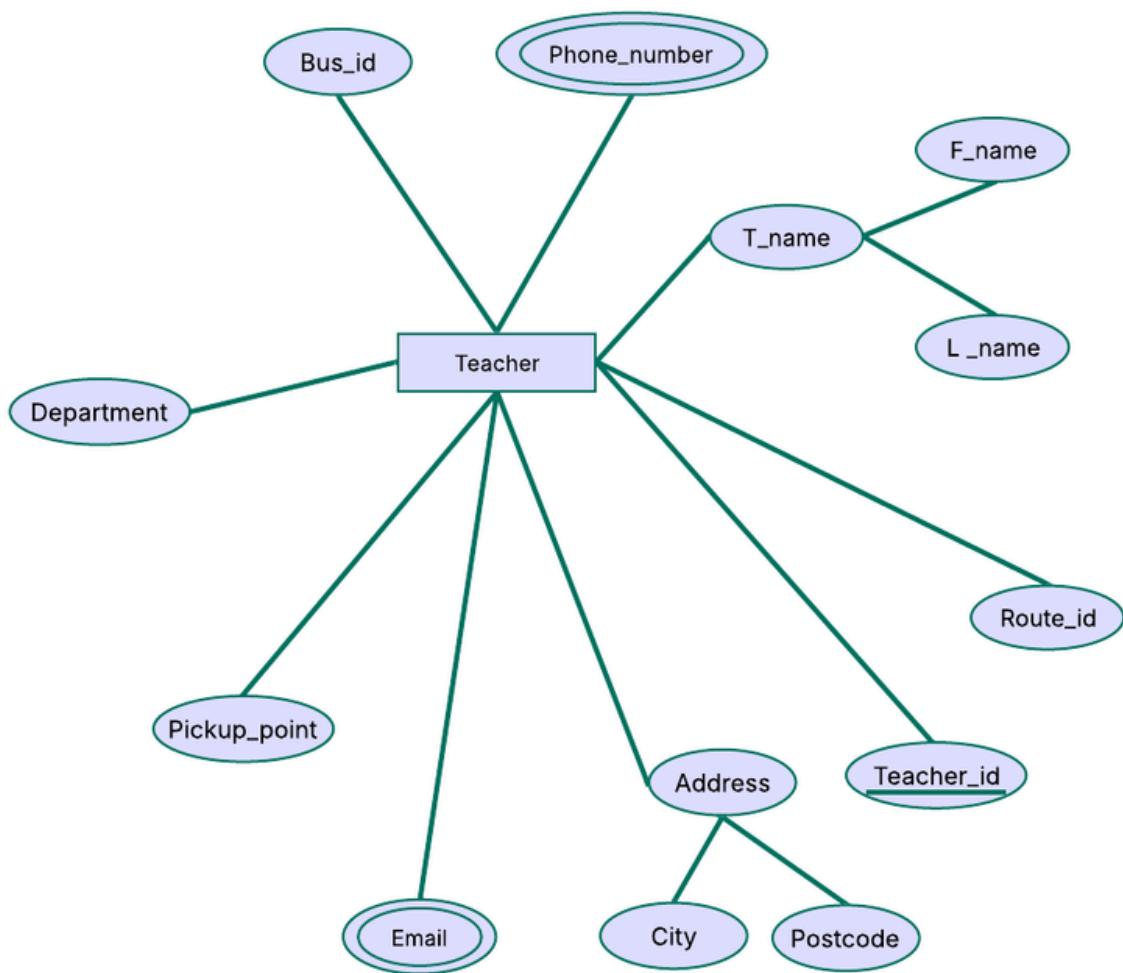
Bus_Schedule:



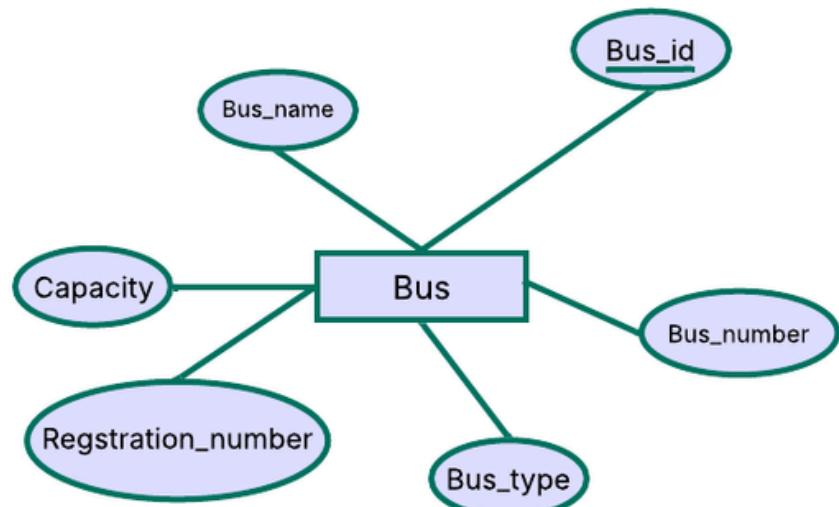
Maintenance:



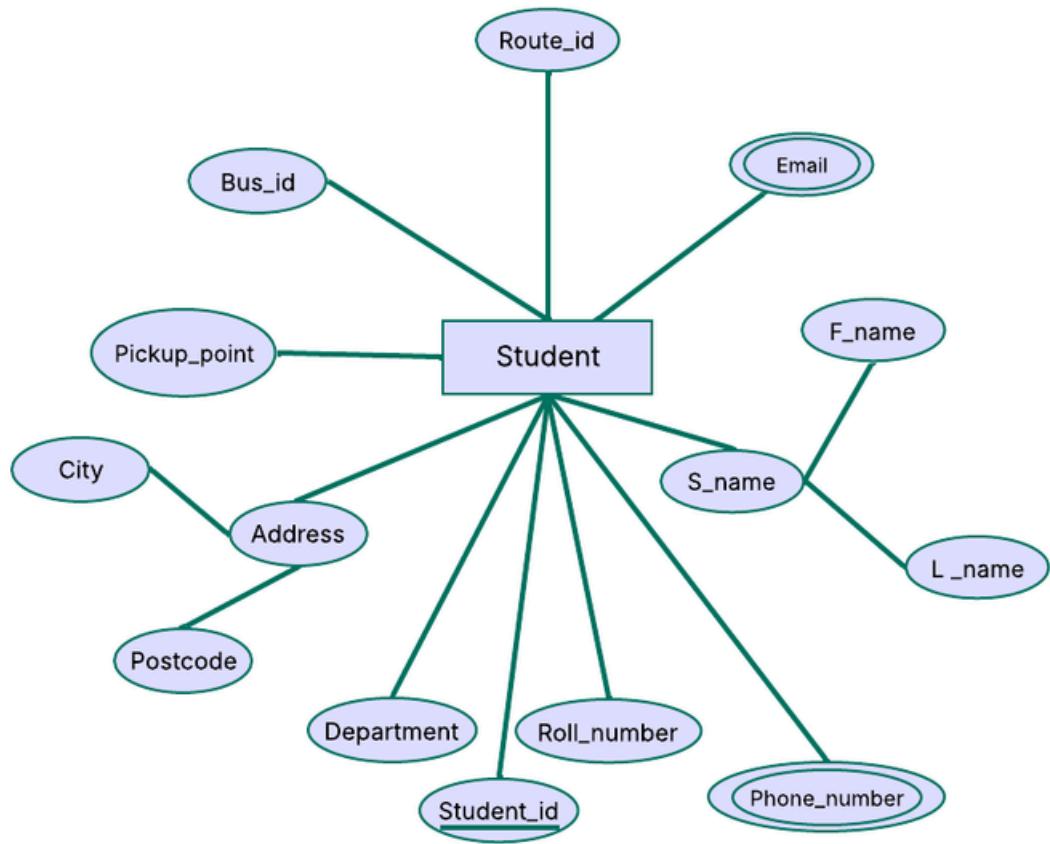
Teacher:



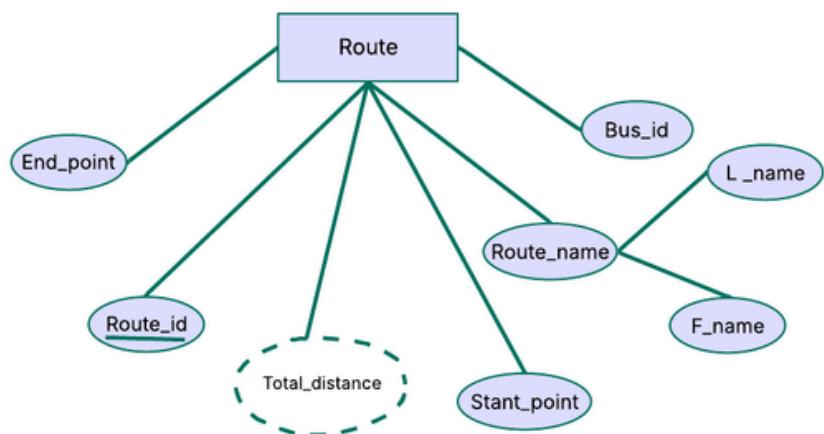
Bus:



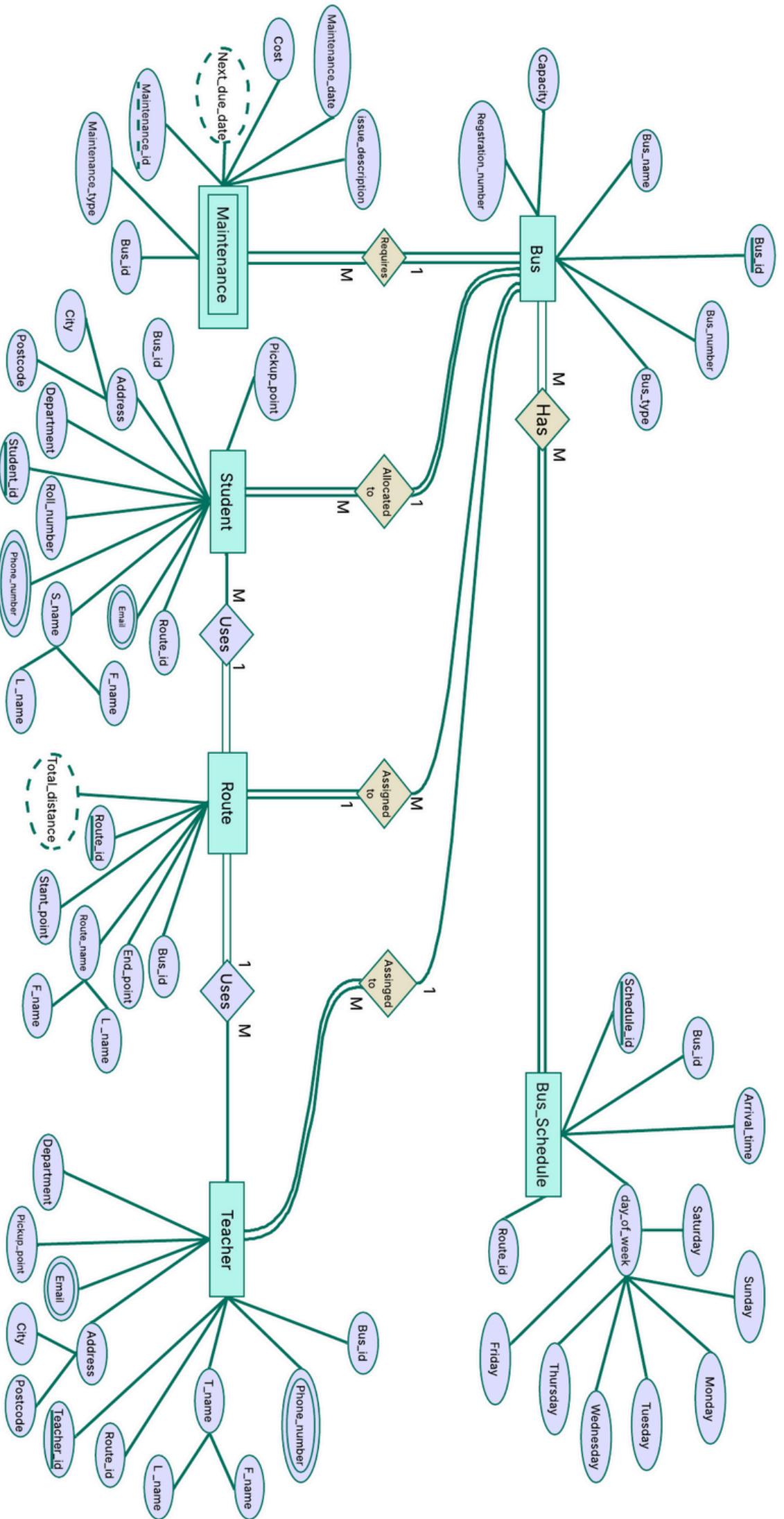
Student:



Route:



DIU Bus Management system ER Diagram



DIU Bus Management

(ER to relation mapping)

Step 3: Mapping Binary 1:N relationship
Bus Schedule:

Schedule_id	Bus_id	Arrival_time	Saturday	Sunday	Monday
Tuesday	Wednesday	Thursday	Friday	Route_id	Monday

Step 1: Mapping regular entity types

Bus_id	bus_name	Capacity	Registration_number	Bus_number	Bus_type

Bus Schedule:

Schedule_id	Bus_id	Arrival_time	Saturday	Sunday	Monday	Tuesday

Student:

Student_id	Bus_id	City	Postcode	Department	Roll_number	F_name

Route:

Route_id	Start_point	End_point	F_name	L_name	Bus_id

Teacher:

Teacher_id	Department	Pickup_point	City	Postcode	Route_id

Route:

Route_id	Start_point	End_point	F_name	L_name	Bus_id

Step 2: Mapping weak entity types

Maintenance:

Bus_id	Maintenance_id	Issue_description	Maintenance_date	Cost

Maintenance_type

Step 5: Mapping multivalued attributes

Student - Phone Number:

Student_id	S_phone_number

Teacher - Phone Number:

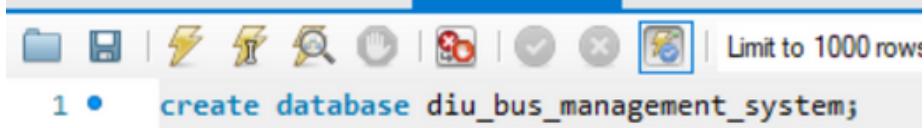
Teacher_id	T_phone_number

Teacher - Email:

Teacher_id	T_email

Table Creation

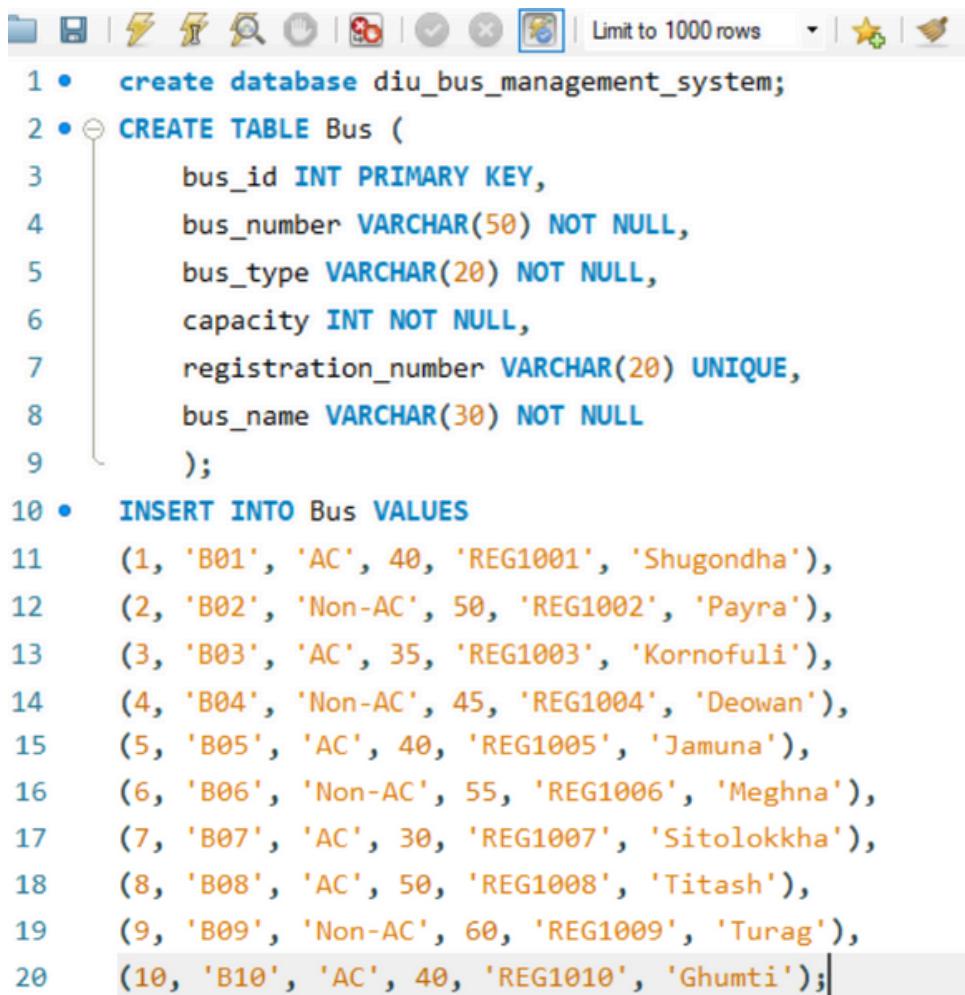
Create Database:



A screenshot of the MySQL Workbench interface. The toolbar at the top has icons for file, database, schema, table, search, and other database management functions. A dropdown menu says "Limit to 1000 rows". Below the toolbar, the code editor shows a single line of SQL: "1 • create database diu_bus_management_system;". The line is highlighted with a light gray background.

Create Bus Entity Table And Insert 10 values:

Code:



A screenshot of the MySQL Workbench interface. The toolbar at the top has icons for file, database, schema, table, search, and other database management functions. A dropdown menu says "Limit to 1000 rows". Below the toolbar, the code editor shows a multi-line SQL script:

```
1 • create database diu_bus_management_system;
2 • CREATE TABLE Bus (
3     bus_id INT PRIMARY KEY,
4     bus_number VARCHAR(50) NOT NULL,
5     bus_type VARCHAR(20) NOT NULL,
6     capacity INT NOT NULL,
7     registration_number VARCHAR(20) UNIQUE,
8     bus_name VARCHAR(30) NOT NULL
9 );
10 • INSERT INTO Bus VALUES
11     (1, 'B01', 'AC', 40, 'REG1001', 'Shugondha'),
12     (2, 'B02', 'Non-AC', 50, 'REG1002', 'Payra'),
13     (3, 'B03', 'AC', 35, 'REG1003', 'Kornofuli'),
14     (4, 'B04', 'Non-AC', 45, 'REG1004', 'Deowan'),
15     (5, 'B05', 'AC', 40, 'REG1005', 'Jamuna'),
16     (6, 'B06', 'Non-AC', 55, 'REG1006', 'Meghna'),
17     (7, 'B07', 'AC', 30, 'REG1007', 'Sitolokkha'),
18     (8, 'B08', 'AC', 50, 'REG1008', 'Titash'),
19     (9, 'B09', 'Non-AC', 60, 'REG1009', 'Turag'),
20     (10, 'B10', 'AC', 40, 'REG1010', 'Ghumti');
```

The code is numbered from 1 to 20. Lines 1 and 2 are highlighted with a light gray background. Lines 10 through 20 represent the insert statements for 10 different buses.

Output:

The screenshot shows a MySQL Workbench interface with a 'Result Grid' window. The grid displays 10 rows of data from the 'Bus' table. The columns are: bus_id, bus_number, bus_type, capacity, registration_number, and bus_name. The data is as follows:

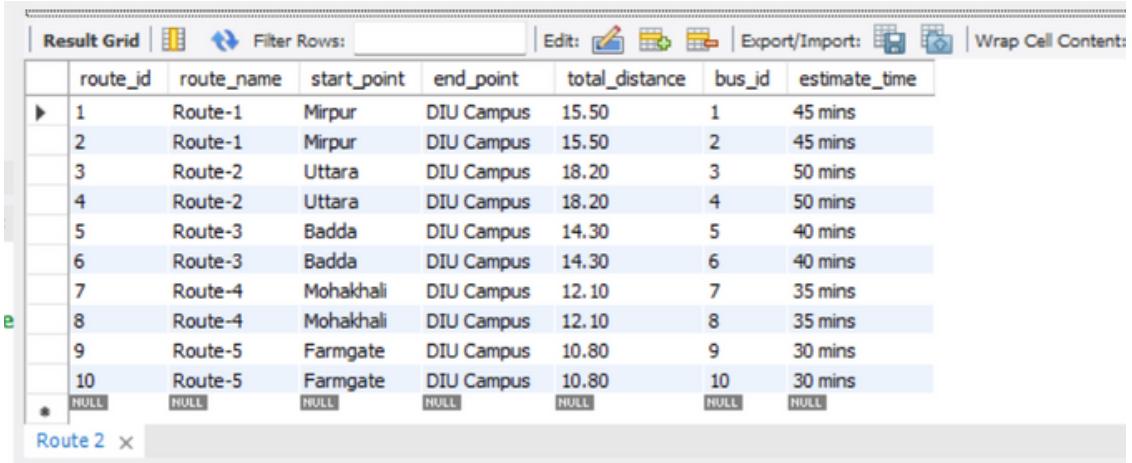
	bus_id	bus_number	bus_type	capacity	registration_number	bus_name
1	B01	AC	40	REG1001	Shugondha	
2	B02	Non-AC	50	REG1002	Payra	
3	B03	AC	35	REG1003	Kornofuli	
4	B04	Non-AC	45	REG1004	Deowan	
5	B05	AC	40	REG1005	Jamuna	
6	B06	Non-AC	55	REG1006	Meghna	
7	B07	AC	30	REG1007	Sitolokkha	
8	B08	AC	50	REG1008	Titash	
9	B09	Non-AC	60	REG1009	Turag	
10	B10	AC	40	REG1010	Ghumti	
*	NULL	NULL	NULL	NULL	NULL	NULL

Create Route Entity Table And Insert 10 values:

Code:

```
24 • CREATE TABLE Route (
25     route_id INT PRIMARY KEY,
26     route_name VARCHAR(30) NOT NULL,
27     start_point VARCHAR(30) NOT NULL,
28     end_point VARCHAR(30) NOT NULL,
29     total_distance DECIMAL(5,2) NOT NULL,
30     bus_id INT,
31     estimate_time VARCHAR(20) NOT NULL,
32     FOREIGN KEY (bus_id) REFERENCES Bus(bus_id)
33 );
34
36 • INSERT INTO Route VALUES
37     (1, 'Route-1', 'Mirpur', 'DIU Campus', 15.5, 1, '45 mins'),
38     (2, 'Route-1', 'Mirpur', 'DIU Campus', 15.5, 2, '45 mins'),
39     (3, 'Route-2', 'Uttara', 'DIU Campus', 18.2, 3, '50 mins'),
40     (4, 'Route-2', 'Uttara', 'DIU Campus', 18.2, 4, '50 mins'),
41     (5, 'Route-3', 'Badda', 'DIU Campus', 14.3, 5, '40 mins'),
42     (6, 'Route-3', 'Badda', 'DIU Campus', 14.3, 6, '40 mins'),
43     (7, 'Route-4', 'Mohakhali', 'DIU Campus', 12.1, 7, '35 mins'),
44     (8, 'Route-4', 'Mohakhali', 'DIU Campus', 12.1, 8, '35 mins'),
45     (9, 'Route-5', 'Farmgate', 'DIU Campus', 10.8, 9, '30 mins'),
46     (10, 'Route-5', 'Farmgate', 'DIU Campus', 10.8, 10, '30 mins');
```

Output:



The screenshot shows a database result grid titled "Result Grid". The grid has columns: route_id, route_name, start_point, end_point, total_distance, bus_id, and estimate_time. The data consists of 10 rows, each representing a route. The routes are: Route-1 (Mirpur to DIU Campus), Route-2 (Uttara to DIU Campus), Route-3 (Badda to DIU Campus), Route-4 (Mohakhali to DIU Campus), Route-5 (Farmgate to DIU Campus). The total distance for Route-1 is 15.50 km, and for Route-5 it is 10.80 km. The estimated time for Route-1 is 45 mins, and for Route-5 it is 30 mins.

	route_id	route_name	start_point	end_point	total_distance	bus_id	estimate_time
▶	1	Route-1	Mirpur	DIU Campus	15.50	1	45 mins
▶	2	Route-1	Mirpur	DIU Campus	15.50	2	45 mins
▶	3	Route-2	Uttara	DIU Campus	18.20	3	50 mins
▶	4	Route-2	Uttara	DIU Campus	18.20	4	50 mins
▶	5	Route-3	Badda	DIU Campus	14.30	5	40 mins
▶	6	Route-3	Badda	DIU Campus	14.30	6	40 mins
▶	7	Route-4	Mohakhali	DIU Campus	12.10	7	35 mins
▶	8	Route-4	Mohakhali	DIU Campus	12.10	8	35 mins
▶	9	Route-5	Farmgate	DIU Campus	10.80	9	30 mins
▶	10	Route-5	Farmgate	DIU Campus	10.80	10	30 mins
◀	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Create Student Entity Table And Insert 10 values:

Code:

```

49 • CREATE TABLE Student (
50     student_id INT PRIMARY KEY,
51     s_name VARCHAR(40) NOT NULL,
52     roll_number VARCHAR(20) NOT NULL UNIQUE,
53     department VARCHAR(30) NOT NULL,
54     phone_number VARCHAR(15) NOT NULL UNIQUE,
55     email VARCHAR(40) NOT NULL UNIQUE,
56     address VARCHAR(50) NOT NULL,
57     bus_id INT NOT NULL,
58     route_id INT NOT NULL,
59     pickup_point VARCHAR(40) NOT NULL,
60     FOREIGN KEY (bus_id) REFERENCES Bus(bus_id),
61     FOREIGN KEY (route_id) REFERENCES Route(route_id)
62 );

```



```

17 •     INTO Student (student_id, s_name, roll_number, department, phone_number, email, address, bus_id, route_id, pickup_point) VALUES
18     'Farjana Hoque', 'CSE221001', 'CSE', '01711111111', 'farjana@diu.edu.bd', 'Mirpur', 1, 1, 'Mirpur 10'),
19     'Sidratul Montaha', 'EEE221002', 'EEE', '01712222222', 'sidratul@diu.edu.bd', 'Mirpur', 1, 1, 'Sector 10'),
20     'Sahriar Pran', 'CSE221003', 'CSE', '01713333333', 'sahriar@diu.edu.bd', 'Mirpur', 1, 1, 'Badda Link Road'),
21     'MD.Emon', 'BBA221004', 'BBA', '01714444444', 'emon@diu.edu.bd', 'Mirpur', 2, 1, 'Wireless Gate'),
22     'Farhan Rahaman', 'CSE221005', 'CSE', '01715555555', 'farhan@diu.edu.bd', 'Mirpur', 2, 1, 'Green Road'),
23     'Imran Hossain', 'TEXT221006', 'Textile', '01716666666', 'imran@diu.edu.bd', 'Mohakhali', 3, 2, 'Chairman Bari'),
24     'Mim Akter', 'CSE221007', 'CSE', '01717777777', 'mim@diu.edu.bd', 'Mohakhali', 3, 2, 'Wireless Gate'),
25     'Tanjila Akter', 'CSE221008', 'CSE', '01718888888', 'tanjila@diu.edu.bd', 'Mohakhali', 4, 2, 'Banani'),
26     'Sabbir Alam', 'BBA221009', 'BBA', '01719999999', 'sabbir@diu.edu.bd', 'Mohakhali', 4, 2, 'Kakoli'),
27     'Rina Akter', 'CSE221010', 'CSE', '01710000000', 'rina@diu.edu.bd', 'Dhanmondi', 5, 3, 'Road 27'),
28     'Arif Hossain', 'CSE221011', 'CSE', '01720000001', 'arif@diu.edu.bd', 'Dhanmondi', 5, 3, 'Road 28'),
29     'Nabilo Akter', 'EEE221012', 'EEE', '01720000002', 'nabilo@diu.edu.bd', 'Dhanmondi', 6, 3, 'Road 32'),
30     'Tanvir Ahmed', 'CSE221013', 'CSE', '01720000003', 'tanvir@diu.edu.bd', 'Dhanmondi', 6, 3, 'Sobhanbag'),
31     'Rafiq Hossain', 'BBA221014', 'BBA', '01720000004', 'rafiq@diu.edu.bd', 'Khilgaon', 7, 4, 'Taltola'),
32     'Anika Rahman', 'TEXT221015', 'Textile', '01720000005', 'anika@diu.edu.bd', 'Khilgaon', 7, 4, 'Railgate'),
33     'Samiul Alam', 'CSE221016', 'CSE', '01720000006', 'samiul@diu.edu.bd', 'Khilgaon', 8, 4, 'Bashabo'),
34     'Tania Akter', 'BBA221017', 'BBA', '01720000007', 'tania@diu.edu.bd', 'Khilgaon', 8, 4, 'Nandipara'),
35     'Rina Sultana', 'CSE221018', 'CSE', '01720000008', 'rina2@diu.edu.bd', 'Rampura', 9, 5, 'TV Center'),
36     'Tanvir Rahman', 'CSE221019', 'CSE', '01720000009', 'tanvir2@diu.edu.bd', 'Rampura', 9, 5, 'East Rampura'),
37     'Nafisa Akter', 'EEE221020', 'EEE', '01720000010', 'nafisa@diu.edu.bd', 'Rampura', 10, 5, 'Ulan Road'),
38     'Sabbir Hossain', 'CSE221021', 'CSE', '01720000011', 'sabbir2@diu.edu.bd', 'Rampura', 10, 5, 'TV Center 2');

```

Output:

Result Grid		Filter Rows:		Edit:		Export/Import:		Wrap Cell Content:	
student_id	s_name	roll_number	department	phone_number	email	address	bus_id	route_id	pickup_point
101	Farjana Hoque	CSE221001	CSE	01711111111	farjana@diu.edu.bd	Mirpur	1	1	Mirpur 10
102	Sidratul Montaha	EEE221002	EEE	01712222222	sidratul@diu.edu.bd	Mirpur	1	1	Sector 10
103	Sahriar Pran	CSE221003	CSE	01713333333	sahriar@diu.edu.bd	Mirpur	1	1	Badda Link Road
104	MD.Emon	BBA221004	BBA	01714444444	emon@diu.edu.bd	Mirpur	2	1	Wireless Gate
105	Farhan Rahman	CSE221005	CSE	01715555555	farhan@diu.edu.bd	Mirpur	2	1	Green Road
106	Imran Hossain	TEXT221006	Textile	01716666666	imran@diu.edu.bd	Mohakhali	3	2	Chairman Bari
107	Mim Akter	CSE221007	CSE	01717777777	mim@diu.edu.bd	Mohakhali	3	2	Wireless Gate
108	Tanjila Akter	CSE221008	CSE	01718888888	tanjila@diu.edu.bd	Mohakhali	4	2	Banani
109	Sabbir Alam	BBA221009	BBA	01719999999	sabbir@diu.edu.bd	Mohakhali	4	2	Kakoli
110	Rina Akter	CSE221010	CSE	01710000000	rina@diu.edu.bd	Dhamondi	5	3	Road 27
111	Anif Hossain	CSE221011	CSE	01720000001	anif@diu.edu.bd	Dhamondi	5	3	Road 28
112	Nabila Akter	EEE221012	EEE	01720000002	nabila@diu.edu.bd	Dhamondi	6	3	Road 32
113	Tanvir Ahmed	CSE221013	CSE	01720000003	tanvir@diu.edu.bd	Dhamondi	6	3	Sobhanbag
114	Rafiq Hossain	BBA221014	BBA	01720000004	rafiq@diu.edu.bd	Khilgaon	7	4	Taltola
115	Anika Rahman	TEXT221015	Textile	01720000005	anika@diu.edu.bd	Khilgaon	7	4	Railgate
116	Samiul Alam	CSE221016	CSE	01720000006	samiul@diu.edu.bd	Khilgaon	8	4	Bashabo
117	Tania Akter	BBA221017	BBA	01720000007	tania@diu.edu.bd	Khilgaon	8	4	Nandipara
118	Rina Sultana	CSE221018	CSE	01720000008	rina2@diu.edu.bd	Rampura	9	5	TV Center
119	Tanvir Rahman	CSE221019	CSE	01720000009	tanvir2@diu.edu.bd	Rampura	9	5	East Rampura
120	Tanvir Rahman	CSE221019	CSE	01720000009	tanvir2@diu.edu.bd	Rampura	9	5	East Rampura
121	Nafisa Akter	EEE221020	EEE	01720000010	nafisa@diu.edu.bd	Rampura	10	5	Ulan Road
122	Sabbir Hossain	CSE221021	CSE	01720000011	sabbir2@diu.edu.bd	Rampura	10	5	TV Center 2
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Create Teacher Entity Table And Insert 10 values:

Code:

```
CREATE TABLE Teacher (
    teacher_id INT PRIMARY KEY,
    t_name VARCHAR(40) NOT NULL,
    department VARCHAR(30) NOT NULL,
    phone_number VARCHAR(15) NOT NULL UNIQUE,
    email VARCHAR(40) NOT NULL UNIQUE,
    address VARCHAR(50) NOT NULL,
    pickup_point VARCHAR(40) NOT NULL,
    bus_id INT NOT NULL,
    route_id INT NOT NULL,
    FOREIGN KEY (bus_id) REFERENCES Bus(bus_id),
    FOREIGN KEY (route_id) REFERENCES Route(route_id)
);
```

```
INSERT INTO Teacher (teacher_id, t_name, department, phone_number, email, address, pickup_point, bus_id, route_id) VALUES
(201, 'Ayon Mia', 'CSE', '01811111111', 'ayon@diu.edu.bd', 'Mirpur', 'Mirpur 11', 1, 1),
(202, 'Shifat Jahan', 'EEE', '01812222222', 'sifat@diu.edu.bd', 'Mirpur', 'Sector 12', 1, 1),
(203, 'Mr. Hasan Ali', 'CSE', '01813333333', 'hasan@diu.edu.bd', 'Mirpur', 'Badda Link Road', 2, 1),
(204, 'Dr. Rubina Islam', 'BBA', '01814444444', 'rubina@diu.edu.bd', 'Mirpur', 'Kazipara', 2, 1),
(205, 'Mr. Kamrul Hasan', 'CSE', '01815555555', 'kamrul@diu.edu.bd', 'Mohakhali', 'Wireless Gate', 3, 2),
(206, 'Ms. Anika Rahman', 'Textile', '01816666666', 'anika@diu.edu.bd', 'Mohakhali', 'Chairman Bari', 3, 2),
(207, 'Dr. Rashedul Karim', 'CSE', '01817777777', 'rashedul@diu.edu.bd', 'Mohakhali', 'Banani', 4, 2),
(208, 'Ms. Laila Sultana', 'CSE', '01818888888', 'laila@diu.edu.bd', 'Mohakhali', 'Kakoli', 4, 2),
(209, 'Dr. Rezaul Haque', 'BBA', '01819999999', 'rezaul@diu.edu.bd', 'Dhamondi', 'Road 28', 5, 3),
(210, 'Mr. Tanvir Ahmed', 'CSE', '01810000000', 'tanvir@diu.edu.bd', 'Dhamondi', 'Road 28', 5, 3),
(211, 'Ms. Nabila Akter', 'BBA', '01820000001', 'nabila@diu.edu.bd', 'Dhamondi', 'Road 32', 6, 3),
(212, 'Mr. Rafiq Hossain', 'CSE', '01820000002', 'rafiq@diu.edu.bd', 'Dhamondi', 'Sobhanbag', 6, 3),
(213, 'Ms. Tanvir Rahman', 'Textile', '01820000003', 'tanvir@diu.edu.bd', 'Khilgaon', 'Taltola', 7, 4),
(214, 'Dr. Samiul Alam', 'CSE', '01820000004', 'samiul@diu.edu.bd', 'Khilgaon', 'Railgate', 7, 4),
(215, 'Ms. Rina Sultana', 'CSE', '01820000005', 'rina@diu.edu.bd', 'Khilgaon', 'Bashabo', 8, 4),
(216, 'Mr. Anwar Hossain', 'BBA', '01820000006', 'anwar@diu.edu.bd', 'Khilgaon', 'Nandipara', 8, 4),
(217, 'Ms. Lamiya Akter', 'CSE', '01820000007', 'lamiya@diu.edu.bd', 'Rampura', 'TV Center', 9, 5),
(218, 'Dr. Sajid Khan', 'BBA', '01820000008', 'sajid@diu.edu.bd', 'Rampura', 'East Rampura', 9, 5),
(219, 'Ms. Tania Akter', 'CSE', '01820000009', 'tania@diu.edu.bd', 'Rampura', 'Ulan Road', 10, 5),
(220, 'Mr. Tanvir Rahman', 'CSE', '01820000010', 'tanvir2@diu.edu.bd', 'Rampura', 'TV Center', 10, 5);
```

Output:

Result Grid		Filter Rows:		Edit:		Export/Import:		Wrap Cell Content:	
teacher_id	t_name	department	phone_number	email	address	pickup_point	bus_id	route_id	
201	Ayon Mia	CSE	01811111111	ayon@diu.edu.bd	Mirpur	Mirpur 11	1	1	
202	Shifat Jahan	EEE	01812222222	sifat@diu.edu.bd	Mirpur	Sector 12	1	1	
203	Mr. Hasan Ali	CSE	01813333333	hasan@diu.edu.bd	Mirpur	Badda Link Road	2	1	
204	Dr. Rubina Islam	BBA	01814444444	rubina@diu.edu.bd	Mirpur	Kazipara	2	1	
205	Mr. Kamrul Hasan	CSE	01815555555	kamrul@diu.edu.bd	Mohakhali	Wireless Gate	3	2	
206	Ms. Anika Rahman	Textile	01816666666	anika@diu.edu.bd	Mohakhali	Chairman Bari	3	2	
207	Dr. Rashedul Karim	CSE	01817777777	rashedul@diu.edu.bd	Mohakhali	Banani	4	2	
208	Ms. Laila Sultana	CSE	01818888888	laila@diu.edu.bd	Mohakhali	Kakoli	4	2	
209	Dr. Rezaul Haque	BBA	01819999999	rezaul@diu.edu.bd	Dhanmondi	Road 27	5	3	
210	Mr. Tanvir Ahmed	CSE	01810000000	tanvir@diu.edu.bd	Dhanmondi	Road 28	5	3	
211	Ms. Nabilah Akter	BBA	01820000001	nabilah@diu.edu.bd	Dhanmondi	Road 32	6	3	
212	Mr. Rafiq Hossain	CSE	01820000002	rafiq@diu.edu.bd	Dhanmondi	Sobhanbag	6	3	
213	Ms. Tanvi Rahman	Textile	01820000003	tanvi@diu.edu.bd	Khilgaon	Taltola	7	4	
214	Dr. Samiul Alam	CSE	01820000004	samiul@diu.edu.bd	Khilgaon	Railgate	7	4	
215	Ms. Rina Sultana	CSE	01820000005	rina2@diu.edu.bd	Khilgaon	Bashabo	8	4	
216	Mr. Anwar Hossain	BBA	01820000006	anwar@diu.edu.bd	Khilgaon	Nandipara	8	4	
217	Ms. Lamia Akter	CSE	01820000007	lamia@diu.edu.bd	Rampura	TV Center	9	5	
218	Dr. Sajid Khan	BBA	01820000008	sajid@diu.edu.bd	Rampura	East Rampura	9	5	
219	Ms. Tania Akter	CSE	01820000009	tania@diu.edu.bd	Rampura	Ulan Road	10	5	
220	Mr. Tanvir Rahman	CSE	01820000010	tanvir2@diu.edu.bd	Rampura	TV Center	10	5	
*	HULL	HULL	HULL	HULL	HULL	HULL	HULL	HULL	

Create BUS_SCHEDULE Entity Table And Insert 10 values:

Code:

```

CREATE TABLE Bus_Schedule (
    schedule_id INT PRIMARY KEY,
    bus_id INT NOT NULL UNIQUE,
    route_id INT NOT NULL,
    arrival_time TIME NOT NULL,
    day_of_week VARCHAR(15) NOT NULL,
    FOREIGN KEY (bus_id) REFERENCES Bus(bus_id),
    FOREIGN KEY (route_id) REFERENCES Route(route_id)
);

INSERT INTO Bus_Schedule VALUES
(301, 1, 1, '07:30:00', 'Sunday'),
(302, 2, 1, '07:40:00', 'Monday'),
(303, 3, 2, '07:35:00', 'Tuesday'),
(304, 4, 2, '07:45:00', 'Wednesday'),
(305, 5, 3, '07:50:00', 'Thursday'),
(306, 6, 3, '07:55:00', 'Sunday'),
(307, 7, 4, '08:00:00', 'Monday'),
(308, 8, 4, '08:05:00', 'Tuesday'),
(309, 9, 5, '08:10:00', 'Wednesday'),
(310, 10, 5, '08:15:00', 'Thursday');

```

Output:

	schedule_id	bus_id	route_id	arrival_time	day_of_week
▶	301	1	1	07:30:00	Sunday
	302	2	1	07:40:00	Monday
	303	3	2	07:35:00	Tuesday
	304	4	2	07:45:00	Wednesday
	305	5	3	07:50:00	Thursday
	306	6	3	07:55:00	Sunday
	307	7	4	08:00:00	Monday
	308	8	4	08:05:00	Tuesday
	309	9	5	08:10:00	Wednesday
	310	10	5	08:15:00	Thursday
*	HULL	HULL	HULL	HULL	HULL

Create MAINTENANCE Entity(Weak Entity) Table And Insert 10 values:

Code:

```

162 • CREATE TABLE Maintenance (
163     maintenance_id INT PRIMARY KEY,
164     bus_id INT NOT NULL,
165     maintenance_date DATE NOT NULL,
166     maintenance_type VARCHAR(30) NOT NULL,
167     issue_description VARCHAR(100) NOT NULL,
168     cost DECIMAL(10,2) NOT NULL,
169     next_due_date DATE NOT NULL,
170     FOREIGN KEY (bus_id) REFERENCES Bus(bus_id),
171     UNIQUE (bus_id, maintenance_date)
172 );
173
174 • INSERT INTO Maintenance VALUES
175     (401, 1, '2024-01-05', 'Engine Check', 'Engine oil replaced', 2500.00, '2024-07-05'),
176     (402, 2, '2024-02-10', 'Tire Change', 'Front tire replaced', 1800.00, '2024-08-10'),
177     (403, 3, '2024-03-12', 'AC Repair', 'Cooling problem fixed', 3000.00, '2024-09-12'),
178     (404, 4, '2024-04-15', 'Brake Check', 'Brake pads changed', 2200.00, '2024-10-15'),
179     (405, 5, '2024-05-18', 'Battery Change', 'New battery installed', 2700.00, '2024-11-18'),
180     (406, 6, '2024-06-20', 'Tire Change', 'Rear tires replaced', 1900.00, '2024-12-20'),
181     (407, 7, '2025-01-22', 'Engine Tune-up', 'Full engine service', 3500.00, '2025-07-22'),
182     (408, 8, '2025-02-25', 'AC Check', 'Filter cleaned', 1500.00, '2025-08-25'),
183     (409, 9, '2025-03-28', 'Body Wash', 'Full body paint & clean', 1200.00, '2025-09-28'),
184     (410, 10, '2025-04-10', 'Light Repair', 'Headlights replaced', 800.00, '2025-10-10');

```

Output:

	maintenance_id	bus_id	maintenance_date	maintenance_type	issue_description	cost	next_due_date
▶	401	1	2024-01-05	Engine Check	Engine oil replaced	2500.00	2024-07-05
	402	2	2024-02-10	Tire Change	Front tire replaced	1800.00	2024-08-10
	403	3	2024-03-12	AC Repair	Cooling problem fixed	3000.00	2024-09-12
	404	4	2024-04-15	Brake Check	Brake pads changed	2200.00	2024-10-15
	405	5	2024-05-18	Battery Change	New battery installed	2700.00	2024-11-18
	406	6	2024-06-20	Tire Change	Rear tires replaced	1900.00	2024-12-20
	407	7	2025-01-22	Engine Tune-up	Full engine service	3500.00	2025-07-22
	408	8	2025-02-25	AC Check	Filter cleaned	1500.00	2025-08-25
	409	9	2025-03-28	Body Wash	Full body paint & clean	1200.00	2025-09-28
	410	10	2025-04-10	Light Repair	Headlights replaced	800.00	2025-10-10
*	HULL	HULL	HULL	HULL	HULL	HULL	HULL

Code Implementation

Requirement 1: Show buses that are almost full (less than 5 seats available):

Query:

```
SELECT b.bus_name, b.capacity, COUNT(s.student_id) AS
filled_seats, (b.capacity - COUNT(s.student_id)) AS
available_seats
FROM Bus b
LEFT JOIN Student s ON b.bus_id = s.bus_id
GROUP BY b.bus_name
HAVING available_seats < 5;
```

Output:

	bus_name	capacity	filled_seats	available_seats
▶	Shugondha	40	36	4

Requirement 2: Show buses that have teachers assigned:

Query:

```
SELECT b.bus_name
FROM Bus b
WHERE EXISTS
(SELECT 1
FROM Teacher t
WHERE t.bus_id = b.bus_id
);
```

Output:

	bus_name
▶	Shugondha
	Payra
	Kornofuli
	Deowan
	Jamuna
	Meghna
	Sitolokkha
	Titash
	Turag
	Ghumti

Requirement 3: Count how many stops each bus makes in a week?

Query:

```
SELECT bs.bus_id, COUNT(*) AS
total_stops
FROM Bus_Schedule bs
GROUP BY bs.bus_id;
```

Output :

Result Grid		Filter Rows:
	bus_id	total_stops
▶	1	1
	2	1
	3	1
	4	1
	5	1
	6	1
	7	1
	8	1
	9	1
	10	1

Requirement 4: List names and departments of teachers assigned to AC buses.

Query:

```
SELECT t.t_name, t.department, b.bus_name, b.bus_type
FROM Teacher t
JOIN Bus b ON t.bus_id = b.bus_id
WHERE b.bus_type = 'AC';
```

Output:

	t_name	department	bus_name	bus_type
▶	Ayon Mia	CSE	Shugondha	AC
	Mr. Hasan Ali	CSE	Kornofuli	AC
	Mr. Kamrul Hasan	CSE	Jamuna	AC
	Dr. Rashedul Karim	CSE	Sitolokkha	AC
	Ms. Laila Sultana	CSE	Titash	AC
	Mr. Tanvir Ahmed	CSE	Ghumti	AC

Requirement 5: Which buses spent more than the average maintenance cost?

Query:

```
SELECT b.bus_name, m.cost
FROM Maintenance m
JOIN Bus b ON m.bus_id = b.bus_id
WHERE m.cost >
(SELECT AVG(cost) FROM Maintenance);
```

Output:

Result Grid		Filter Rows:
	bus_name	cost
▶	Shugondha	2500.00
	Kornofuli	3000.00
	Deowan	2200.00
	Jamuna	2700.00
	Sitolokkha	3500.00

Requirement 6: Find the first and last bus scheduled each day.

Query:

```
SELECT  
    MIN(arrival_time) AS first_bus,  
    MAX(arrival_time) AS last_bus  
FROM Bus_Schedule;
```

Output:

Result Grid		Filter Rows:
	first_bus	last_bus
▶	07:30:00	08:15:00

Requirement 7: Filter buses that cover long routes and have AC.

Query:

```
SELECT      b.bus_name,          b.bus_type,  
r.total_distance  
FROM Bus b  
JOIN Route r ON b.bus_id = r.bus_id  
WHERE r.total_distance > 15 AND  
b.bus_type = 'AC';
```

Output:

Result Grid				Filter Rows:
	bus_name	bus_type	total_distance	
▶	Shugondha	AC	15.50	
	Kornofuli	AC	18.20	

Requirement 8: Find the busiest route (most students assigned)

Query:

```
SELECT  
    r.route_name,  
    COUNT(s.student_id) AS total_students  
FROM Route r  
JOIN Student s ON r.route_id = s.route_id  
GROUP BY r.route_name  
ORDER BY total_students DESC  
LIMIT 1;
```

Output:

	route_name	total_students
▶	Route-1	36

Requirement 9: Find the route with the highest total bus capacity

Query:

```
SELECT
    r.route_name,
    SUM(b.capacity) AS total_capacity
FROM Route r
JOIN Bus b ON r.bus_id = b.bus_id
GROUP BY r.route_name;
```

Output:

Result Grid		Filter Rows:
	route_name	total_capacity
▶	Route-1	90
	Route-2	80
	Route-3	95
	Route-4	80
	Route-5	100

Requirement 10: Find buses that are used by both students and teachers .

Query:

```
SELECT
    b.bus_id,
    b.bus_name,
    COUNT(DISTINCT s.student_id) AS student_count,
    COUNT(DISTINCT t.teacher_id) AS teacher_count
FROM Bus b
JOIN Student s ON b.bus_id = s.bus_id
JOIN Teacher t ON b.bus_id = t.bus_id
GROUP BY b.bus_id, b.bus_name;
```

Output:

Result Grid					Filter Rows:	Export
	bus_id	bus_name	student_count	teacher_count		
▶	1	Shugondha	36	3		

Discussion:

The University Bus Management System aims to solve the major transportation challenges faced by students, teachers, and staff in our campus. By digitalizing and organizing bus routes, schedules, and maintenance records, this system will ensure timely, hassle-free, and efficient transportation for all. With seat management, and proper maintenance logs, it will reduce confusion at stops, eliminate unnecessary delays, and enhance the overall commuting experience. Additionally, clear seating arrangements will help maintain order and comfort, especially during rush hours. Overall, this project is a step towards smarter campus management, offering a reliable and user-friendly solution to one of the most common issues in university life — daily transportation.

References:

- 1. W3Schools:** MySQL Tutorial. Retrieved from <https://www.w3schools.com/mysql/>
(Used for creating databases, tables, and understanding MySQL data types and commands.)
- 2. GeeksforGeeks:** MySQL Tutorial – Introduction, Commands, and Examples. Retrieved from <https://www.geeksforgeeks.org/mysql-tutorial/>
(Used for practical examples of MySQL database management and relational queries.)
- 3. MySQL Documentation:** MySQL Reference Manual. Oracle. Retrieved from <https://dev.mysql.com/doc/>
(Used as a primary reference for MySQL syntax, functions, and database administration.)
- 4. TutorialsPoint:** MySQL Tutorial. Retrieved from <https://www.tutorialspoint.com/mysql/index.htm>
(Used for examples on table creation, foreign keys, and SQL relationships.)

GitHub Links:

https://github.com/Ayshi32/DIU_Bus_Management _System

1. <https://github.com/itsemon007>
2. <https://github.com/farzanahoque430>
3. <https://github.com>ShowrovShahariar>
4. <https://github.com/Ayshi32>