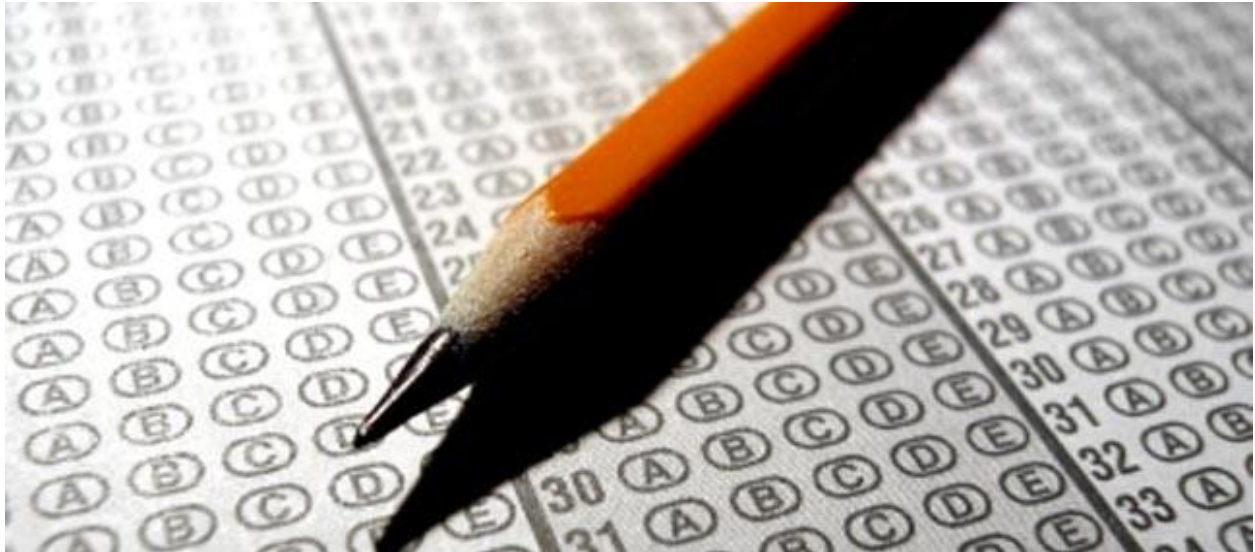# DATABASE MANAGEMENT SYSTEMS PROJECT

on

Coaching Institute Database

Name-Akash Kumar   Roll no.-202202

ABOUT THE PROJECT

In this project we are designing a database management of an educational coaching institute.

Many educational coaching institutes require a robust database for daily management of resources and classes.

The database gives a complete overview of students, departments, teachers, fee structure and also connects them in an appropriate way as to reduce data redundancy. It also keeps a record of the institute alumni for giving exposure to the student about various colleges and opportunities ahead.
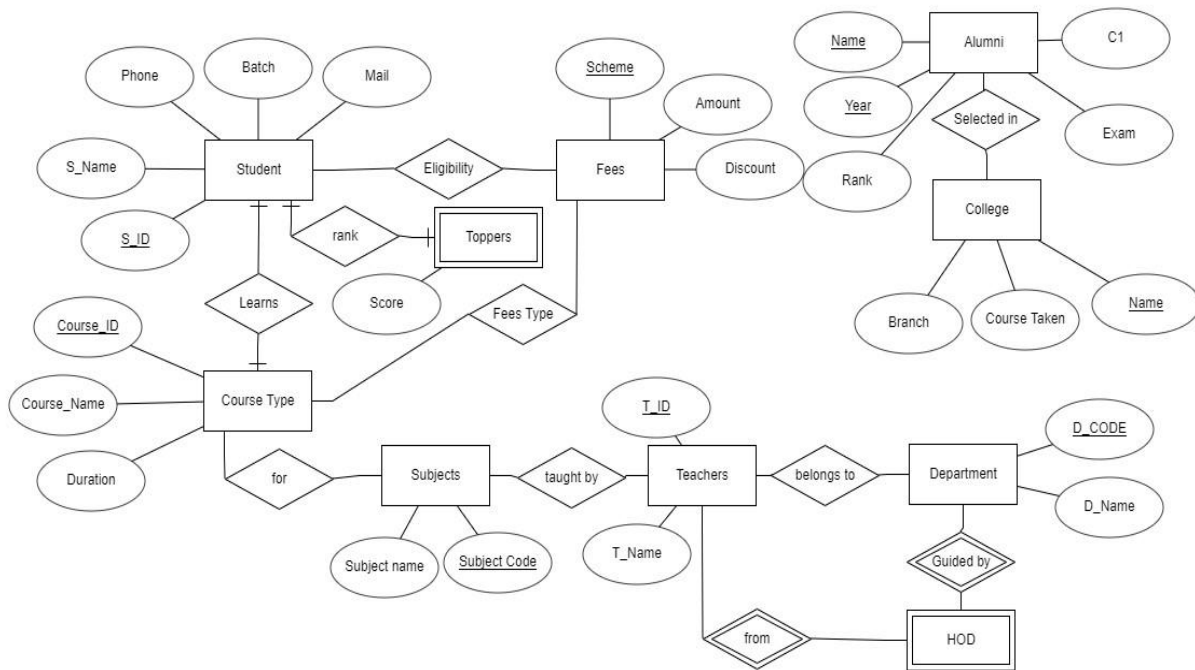
This database also keeps a list of students who are high achievers, who ought to be given special classes with advanced concepts. This institute also gives students scholarships on the basis of their academic excellence and the database keeps complete track of their fee structures.

The educational institute also provides many different courses offering different subjects, the database keeps track of different students taking different courses and segregates them accordingly.
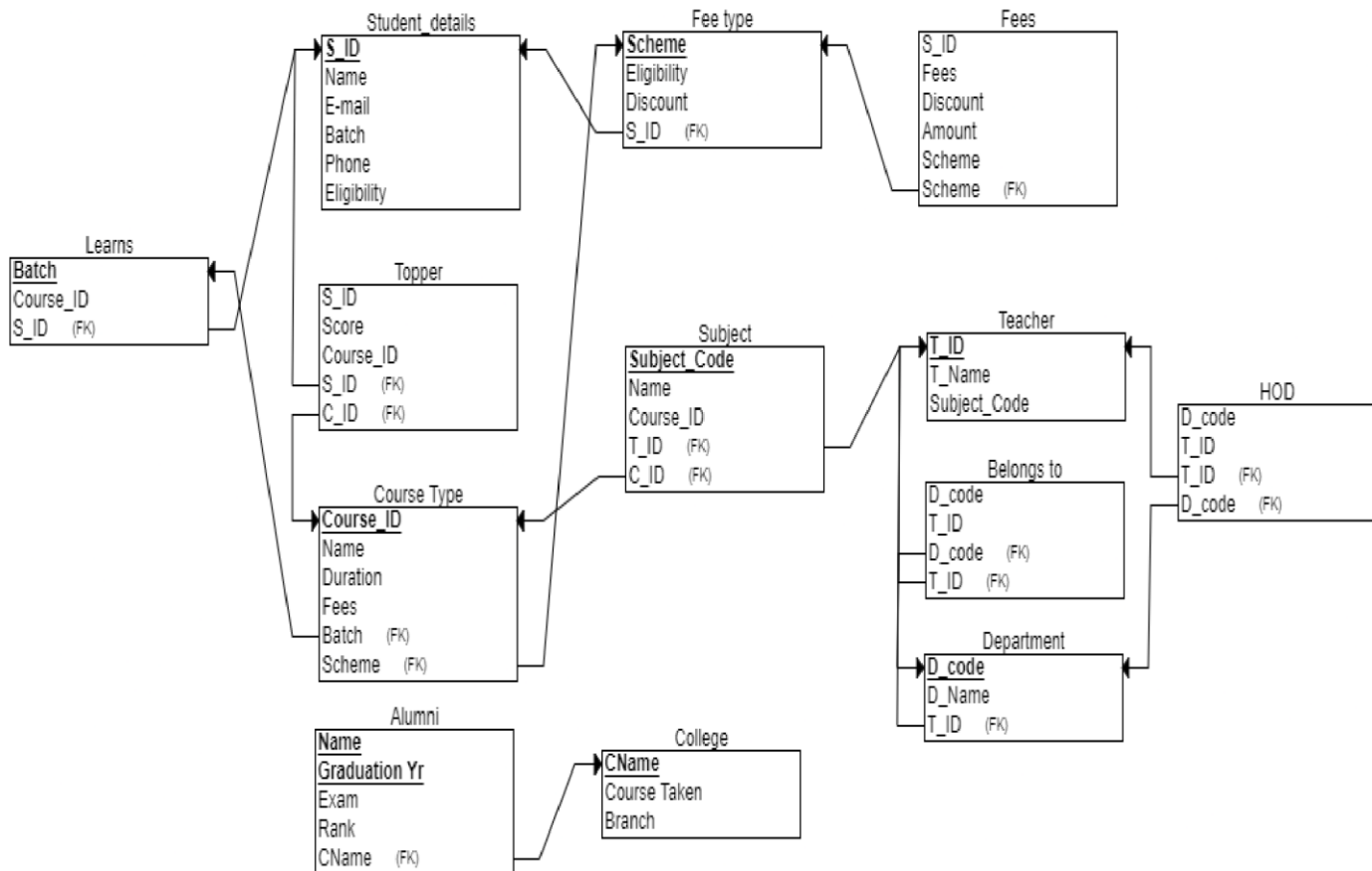
Normalization has been done on each table to the highest normal form possible to remove redundancies and optimize the code.

Each entity and related functional dependencies are illustrated in the project document for better understanding of the database design.

Entity-Relationship Diagram

Relational Schema

Student_details
- S_ID
- Name
- E-mail
- Batch
- Phone
- Eligibility

Fee type
- Scheme
- Eligibility
- Discount
- S_ID   (FK)

Fees
- S_ID
- Fees
- Discount
- Amount
- Scheme
- Scheme   (FK)

Learns
- Batch
- Course_ID
- S_ID   (FK)

Topper
- S_ID
- Score
- Course_ID
- S_ID   (FK)
- C_ID   (FK)

Subject
- Subject_Code
- Name
- Course_ID
- T_ID   (FK)
- C_ID   (FK)

Teacher
- T_ID
- T_Name
- Subject_Code

HOD
- D_code
- T_ID
- T_ID   (FK)
- D_code   (FK)

Course Type
- Course_ID
- Name
- Duration
- Fees
- Batch   (FK)
- Scheme   (FK)

Belongs to
- D_code
- T_ID
- D_code   (FK)
- T_ID   (FK)

Department
- D_code
- D_Name
- T_ID   (FK)

Alumni
- Name
- Graduation Yr
- Exam
- Rank
- CName   (FK)

College
- CName
- Course Taken
- Branch

Entity: **Student**

Attributes:
- **<u>S_ID</u>**
- S_name
- Email
- Batch
- Phone
- Eligibility

Functional Dependencies:
- S_ID → S_name
- S_ID → email ● S_ID → batch
- S_ID → phone
- S_ID → Eligibility

Candidate key: S_ID

In this table, the candidate key is able to uniquely identify all the other attributes. There are no partial or transitive dependencies.Hence the student table is in **BCNF.**

Entity: **Topper**

Attributes:
- S_ID
- Score
- Course_ID

Functional Dependencies:
- S_ID → Score
- S_ID → Course_ID

Candidate key: S_ID

In this table, the candidate key is able to uniquely identify all the other attributes. There are no partial or transitive dependencies.Hence the topper table is in **BCNF.**

Entity: **Fees**
Attributes:
- **S_ID**
- Fees
- Discount
- Amount
- Scheme

Functional Dependencies:
- S_ID → Fees
- S_ID→ Discount
- Fees,Discount → Amount
- S_ID → Scheme

Candidate key: S_ID

In this table, a transitive dependency exists between Fees,Discount and Amount as they are all non-prime attributes. Hence it does not satisfy the conditions for 3NF. Since there is no partial dependency the Fees table is in **2NF**.

Entity: **Course type**
Attributes:
- **C_ID**
- Course Duration
- Course Name

Functional Dependencies:
1. C_ID → Course Name

2. C_ID → Course Duration
3. Course Name → Course Duration
4. C_ID → Fees

Candidate key: C_ID
In this table, a transitive dependency exists between course name and course duration as they are both non-prime attributes. Hence it does not satisfy the conditions for 3NF.There is no partial dependency, hence the table is in **2NF.**

Entity: **Subjects**
Attributes:
- **Sub_code**
- Sub_name

Functional Dependencies:
- Sub_code → Sub_name

Candidate key: Sub_code

In this table, there are only 2 attributes, so only one functional dependency from the candidate key to the non-prime key.
Hence the subjects table is in **BCNF.**

Entity: **Teachers**
Attributes:
- **T_ID**
- T_name
- subject_code

Functional Dependencies:
- T_ID → T_name

- T_ID → subject_code

Candidate key: T_ID

In this table, the candidate key is able to uniquely identify all the other attributes. There are no partial or transitive dependencies.Hence the teachers table is in **BCNF.**

Entity: **Department**
Attributes:
- **D_code**
- D_name

Functional Dependencies:
- D_code → D_name

Candidate key: D_code

In this table, there are only 2 attributes, so only one functional dependency from the candidate key to the non-prime key.
Hence the subjects table is in **BCNF.**

Entity: **HOD**
Attributes:
- **D_code**
- **T_ID**
- D_code (foreign key)
- T_ID (foreign key)

Functional Dependencies:
- D_code,T_ID → D_code
- D_code,T_ID → T_ID

Candidate key: D_code,T_ID

In this table, there's a functional dependency from the candidate key to all the other attributes. There is no partial or transitive dependency. Hence the HOD table is in **BCNF.**

Entity: **Alumni**
Attributes:
- **Name**
- **Graduation Year**
- Exam
- Rank
- c1

Functional Dependencies:
- Name,graduation year → exam
- Name, graduation year → rank
- Name, graduation year → c1

Candidate key: name,graduation year
In this table,name and graduation year together form the candidate key and they are able to uniquely identify all the other attributes. There are no transitive or partial dependencies, hence the alumni table is in **BCNF**.

Entity: **College**
Attributes:
- **C_name**
- Course
- branch

Functional Dependencies:
- C_name → course

- Course → branch

Candidate key: C_name

In this table, a transitive dependency exists between course and branch as they are both non-prime attributes, so it doesn't satisfy the conditions for 3NF. There are no partial dependencies, hence the College table is in **2NF**.

Relationships among Entities

➢Selected In

A one-to-one relationship exists between the alumni table and the college table where the alumni table has total participation and the primary key of the college table(college_name) is added as a foreign key(c1) to the alumni table. Entity: **College** Attributes:

- **C_name (primary key)**
- Course
- branch

Entity: **Alumni**
Attributes:

- **Name   (primary**
- **Graduation Year   key)**
- Exam
- Rank
- C1 **(foreign key)**

➢Fee Type

A one to one relationship between fees table and course type table displaying the discount of each course. Scheme is a primary key which is taken as foreign key for the fees table.

Entity: **Fees**
Attributes:

- **S_ID (primary key)**
- Fees
- Discount
- Amount
- Scheme

Entity: **Course type**
Attributes:

- **C_ID (primary key)**
- Course Duration
- Course Name

➢Learns

A one to one relationship between student table and course type table displaying the batch and respective course ID.

Entity: **Student**
Attributes:

- **S_ID (primary key)**
- S_name
- Email
- Batch
- Phone
- Eligibility

Entity: **Course type**
Attributes:

- **C_ID (primary key)**
- Course Duration ● Course Name

➢Eligibility

A relationship between student table and fees table.The amount is calculated based on the discount availed from eligibility.

Entity: **Student**
Attributes:
- **S_ID (primary key)**
- S_name
- Email
- Batch
- Phone
- Eligibility

Entity: **Fees**
Attributes:
- **S_ID (primary key)**
- Fees
- Discount
- Amount
- Scheme

➢Taught_By

A 1 : n relationship between entities teachers and Subjects. Sub_code is a foreign key in entity Teacher which acts as a foreign key and references entity Subjects.

Entity: **Teachers**
Attributes:

- **T_ID**
- T_name
- Sub_code (foreign key)

Entity: **Subjects**
Attributes:
- **Sub_code**
- Sub_name

➢Belongs_to

A 1 : 1 relationship between Department and Teacher . It is used to link a teacher with their respective department. It uses a combination of (D_code,T_Id) as an identifying key.

Entity: **Teachers**
Attributes:
- **T_ID**
- T_name
- Sub_code (foreign key)

Entity: **Department**
Attributes:
- **D_code**
- D_name

➢For_Relation
Sub_code
Course_id
Identifying key (sub_code,course_id)

It is a m : n relation.

Entity: **Course type**

Attributes:

- **C_ID (primary key)**
- Course Duration
- Course Name

Entity: **Subjects**

Attributes:

- **Sub_code**
- Sub_name

➢Rank

A relation between student table and toppers. The toppers table gets an S_ID as foreign key from the Student table and score and course_ID are entered.

Entity: **Student**

Attributes:

- **S_ID (primary key)**
- S_name
- Email
- Batch
- Phone
- Eligibility

Entity: **Topper**

Attributes:

- S_ID (foreign key)
- Score
- Course_ID (foreign key)

➢Relationship

A 1 : 1 relationship between teacher and HOD.

Entity: **Teachers**
Attributes:
- **T_ID**
- T_name
- Sub_code (foreign key)

Entity: **HOD**
 Attributes:
- **D_code**
- **T_ID**
- D_code (foreign key) ● T_ID (foreign key)

➢ **Guided by**

A 1 : 1 relationship between HOD and department.

Entity: **Department**
Attributes:
- **D_code**
- D_name

Entity: **HOD**
 Attributes:
- **D_code**
- **T_ID**
- D_code (foreign key)
- T_ID (foreign key)

SQL codes

Creation
```
create table Student_Details(
Sname varchar(50),
 S_ID char(10) primary key,
email varchar(50), phone
number,
batch char(10),
eligibility char(10));

create table course_type( course_ID
char(50) PRIMARY KEY,
Course_name varchar(50), duration
number, fees number);

create table fee_type( scheme
varchar(50) primary key,
eligibility char(10), discount
number)

create table fees( S_ID char(10), fees number,
discount number, amount number, scheme
varchar(50), foreign key (scheme) references
fee_type(scheme));

create table toppers( score number, S_ID char(10),
course_ID char(10), foreign key(course_ID) references
course_Type(course_ID)); create table learns( batch
char(10), course_ID char(50), foreign key(course_ID)
references course_Type(course_ID));

create table subject(
sub_code int, sub_name
varchar(30), primary
key(sub_code));

create table For_relation(
    sub_code int, course_ID char(50), primary
    key(sub_code,course_ID), foreign key(sub_code)
```

```
    references subject(sub_code), foreign key(course_ID)
    references course_type(course_ID)
);

create table teachers( T_ID int primary key, T_name
varchar(30), subject_code int, foreign key(subject_code)
references subject(sub_code));

create table department(
D_code int primary key,
D_name varchar(30));

create table belongs_to(
D_code int, T_id int, primary key(D_code,T_id), foreign key(D_code)
references department(D_code) on delete cascade, foreign key(T_id)
references Teachers(T_ID) on delete cascade);

create table HOD(
D_code int, T_id int, primary key(D_code,T_id),
foreign key(D_code) references department(D_code),
foreign key(T_id) references Teachers(T_id));


create table college
( college_name varchar(30) primary key,
  course_taken varchar(30), branch
  varchar(30));

create table alumni
( alumni_name varchar(30), graduation_year int,
  exam varchar(30), rank1 int, c1 varchar(20),
  foreign key(c1) references college(college_name),
  primary key(alumni_name,graduation_year));
```

Insertion

```
insert into  fee_type values('SDA','DA',40);
insert into  fee_type values('SSC','SC',20);
insert into  fee_type values('SST','ST',30);
insert into  fee_type values('SET','ET',50);
insert into  fee_type values('SLI','LI',40);
```

```sql
insert into course_type values('E101','JEE',2); insert into
course_type values('E102','BITSAT',2); insert into course_type
values('E103','GATE',1); insert into course_type
values('M101','AIIMS',3); insert into course_type
values('M102','NEET',4); insert into course_type
values('MB101','CAT',1); update course_type set fees=150000
where course_ID='E101'; update course_type set fees=100000
where course_ID='E102'; update course_type set fees=100000
where course_ID='E103'; update course_type set fees=200000
where course_ID='M101'; update course_type set fees=400000
where course_ID='M102'; update course_type set fees=100000
where course_ID='MB101';

insert into learns values('AE101019','E101');
insert into learns values('AE102019','E102');
insert into learns values('AE103019','E103');
insert into learns values('AB101020','MB101');
insert into learns values('AB101021','MB101');
insert into learns values('AM101019','M101');
insert into learns values('AM101020','M101');
insert into learns values('AM102019','M102');
insert into learns values('AM102020','M102');

insert into student_details
values('Ankit','A01901','A01901@mail.com',9876543201,'AE101019',NULL);
insert into student_details
values('Bnanda','A01902','A01902@mail.com',9876543202,'AE101019','ST');
insert into student_details
values('Danswrang','A01903','A01903@mail.com',9876543203,'AE101019','ST');

insert into student_details
values('Didwm','A01904','A01904@mail.com',9876543204,'AE102019',NULL);
insert into student_details
values('Doli','A01905','A01905@mail.com',9876543205,'AE102019','SC'); insert
into student_details
values('Jitendra','A01906','A01906@mail.com',9876543206,'AE102019','SC');

insert into student_details
values('Joy','A01907','A01907@mail.com',98765432047,'AE103019',NULL);
insert into student_details
values('Juvita','A01908','A01908@mail.com',9876543208,'AE103019','DA');
```

```
insert into student_details
values('Khalid','A01909','A01909@mail.com',9876543209,'AE103019','DA');

insert into student_details
values('K.Anchal','A02001','A02001@mail.com',9876543210,'AB101020',NULL);
insert into student_details
values('Lajoo','A02002','A02001@mail.com',9876543211,'AB101020','EM');
insert into student_details
values('Maithaisri','A02003','A02003@mail.com',9876543212,'AB101020','EM');

insert into student_details
values('Majoni','A02101','A02101@mail.com',9876543213,'AB101021',NULL);
insert into student_details
values('Mijink','A02102','A02102@mail.com',9876543214,'AB101021','LI'); insert
into student_details
values('Mritunjoy','A02103','A02103@mail.com',9876543215,'AB101021','LI');

insert into student_details
values('Nachiket','A01910','A01910@mail.com',9876543216,'AM101019',NULL);
insert into student_details
values('Nayan','A01911','A01911@mail.com',9876543217,'AM101019','LI'); insert
into student_details
values('Nur','A01912','A01912@mail.com',9876543218,'AM101019','LI');

insert into student_details
values('Parisha','A02004','A02004@mail.com',9876543219,'AM101020',NULL);
insert into student_details
values('Pawan','A02005','A02005@mail.com',9876543220,'AM101020','DA');
insert into student_details
values('raj','A02006','A02006@mail.com',9876543221,'AM101020','DA');

insert into student_details
values('Rakesh','A01913','A01913@mail.com',9876543222,'AM102019',NULL);
insert into student_details
values('Ranabir','A01914','A01914@mail.com',9876543223,'AM102019','SC');
insert into student_details
values('Simon','A01915','A01915@mail.com',9876543224,'AM102019','SC');

insert into student_details
values('Sonali','A02007','A02007@mail.com',9876543225,'AM102020',NULL);
insert into student_details
```

```sql
values('Sonalika','A02008','A02008@mail.com',9876543226,'AM102020','EM');
insert into student_details
values('Suraj','A02009','A02009@mail.com',9876543227,'AM102020','EM');

--for fees table
SET SERVEROUTPUT ON;
DECLARE
CURSOR CUR IS
SELECT * FROM student_details;
cid char(50); bill number; dis
number; temp char(10); amt
number; sch varchar(50);
BEGIN
FOR ITEM IN CUR
LOOP
select Course_ID into cid from learns where batch=item.batch; select fees into bill
from course_type where course_ID=cid; select discount,scheme into dis,sch from
fee_type where eligibility=item.eligibility;

dis:=(bill*dis/100); amt:=(bill-(dis)); insert into

fees values(item.S_ID,bill,dis,amt,sch);

END LOOP;
END;
/

insert into toppers values(99,'A01901','E101');
insert into toppers values(95,'A01905','E102');
insert into toppers values(98,'A01909','E103');
insert into toppers values(90,'A02001','MB101');
insert into toppers values(57,'A01911','M101');
insert into toppers values(75,'A02004','M101');
insert into toppers values(89,'A02102','MB101');
insert into toppers values(88,'A01915','M102');
insert into toppers values(99,'A02009','M102');

insert into subject values(001,'physics'); insert into
subject values(002,'Mathematics'); insert into
subject values(003,'Chemistry'); insert into subject
values(004,'Biology'); insert into subject
values(005,'Computer Science'); insert into subject
```

values(006,'English'); insert into subject
values(007,'GK'); insert into subject
values(008,'Hindi'); insert into subject
values(009,'Economics'); insert into subject
values(010,'Political Science');

insert into for_relation values(001,'E101');
insert into for_relation values(002,'E101');
insert into for_relation values(003,'E101');
insert into for_relation values(001,'E102');
insert into for_relation values(002,'E102');
insert into for_relation values(003,'E102');
insert into for_relation values(001,'M101');
insert into for_relation values(002,'M101');
insert into for_relation values(004,'M101');
insert into for_relation values(001,'M102');
insert into for_relation values(002,'M102');
insert into for_relation values(004,'M102');
insert into for_relation values(005,'E103');
insert into for_relation values(007,'UP001');
insert into for_relation values(006,'UP001');
insert into for_relation values(008,'UP001');
insert into for_relation values(010,'UP001');
insert into for_relation values(009,'MB101');
insert into teachers values(100,'Aditya',001);
insert into teachers values(101,'Nayan',002);
insert into teachers
values(102,'Himanshu',003); insert into
teachers values(103,'Rahul',004); insert into
teachers values(104,'Siddhart',005); insert into
teachers values(105,'Shubham',006); insert into
teachers values(106,'Arjun',007); insert into
teachers values(107,'Krishna',008); insert into
teachers values(108,'Ram',009); insert into
teachers values(109,'Shiva',010); insert into
teachers values(110,'Narayan',001); insert into
teachers values(111,'Gajendra',002); insert into
teachers values(112,'Ravi',003);

insert into department values(201,'Sciences'); insert
into department values(202,'Languages'); insert

into department values(203,'General'); insert into
department values(204,'Social Sciences');

insert into belongs_to values(201,100);
insert into belongs_to values(201,101);
insert into belongs_to values(201,102);
insert into belongs_to values(201,103);
insert into belongs_to values(201,104);
insert into belongs_to values(202,105);
insert into belongs_to values(203,106);

insert into belongs_to values(202,107);
insert into belongs_to values(204,108);
insert into belongs_to values(204,109);
insert into belongs_to values(201,110);
insert into belongs_to values(201,111);
insert into belongs_to values(201,112);

insert into HOD values(201,100);
insert into HOD values(202,107);
insert into HOD values(203,106);
insert into HOD values(204,108);
insert into college values('IIT
Delhi','Engineering','ECE'); insert into
college values('IIT
Bombay','Engineering','CSE'); insert
into college values('IIT
Madras','Engineering','ME'); insert
into college values('BITS
Pilani','Engineering','EEE'); insert into
college values('IIM
Bangalore','MBA','Finance'); insert
into college
values('SRCC','Commerce','Accountanc
y'); insert into college values('NIT
Warangal','Engineering','CSE'); insert
into college values('IIM
Ahmadabad','MBA','Marketing');
insert into college values('AIIMS
Delhi','Medical','MBBS'); insert into
college values('KMC-
Manipal','Medical','MBBS');

insert into alumni values('Ayush Mehra',2003,'JEE',601,'IIT Delhi'); insert into alumni values('Tejas Iyer',2000,'JEE',141,'IIT Bombay'); insert into alumni values('Abhinav Sharma',2005,'JEE',1067,'IIT Madras'); insert into alumni values('Arjun Verma',1999,'JEE',862,'BITS Pilani'); insert into alumni values('Kartik Gupta',2007,'CAT',58,'IIM Ahmadabad'); insert into alumni values('Jai Chaudhry',1997,'Boards',121,'SRCC'); insert into alumni values('Sanjana Ramesh',2004,'JEE',324,'NIT Warangal'); insert into alumni values('Vikram Sethi',2005,'CAT',110,'IIM Bangalore'); insert into alumni values('Ananya Singh',2000,'NEET',974,'KMC-Manipal'); insert into alumni values('Ashwati Nair',2004,'NEET',770,'AIIMS Delhi');