➢ Table of Contents:

Sr. No	Name	Page No.
1	Abstract	2
2	List of Abbreviations	3
3	List of Figures	4
4	List of Tables	4
5	Database Design: ER Diagram	5
6	Database Schema	6
7	DDL	7
8	DML along with the results of the queries	14
9	DCL	21
10	Triggers	23
11	PL SQL Procedures and Functions	25
12	Frontend GUI Screenshots	27
13	Conclusion	34
14	References in IEEE Format	34

1) Abstract:

a) Introduction:

Attendance Management System is a software developed for daily student attendance in schools, colleges and institutes. It allows the faculty to access the attendance information of a particular student in a particular class. The information is sorted by the operators, which will be provided by the teacher for a particular class. This system will also help in evaluating attendance eligibility criteria of a student.

b) Purpose:

The purpose of developing attendance management system is to computerize the traditional way of taking attendance. Another purpose for developing this software is to generate the report automatically at the end of the session or in between the session

c) Scope:

The scope of the project is the system on which the software is installed, i.e. the project is developed as a desktop application, and it will work for a particular institute. But later on the project can be modified to operate it online.

d) Tools and Technologies used:

Backend: MySQL

Frontend: JAVA

e) Overview or Problem Definition:

Attendance Management System basically has two main modules for proper functioning:

i) First module is admin which has the right for creating space for new batch. Any entry of new faculty, updation in subject if necessary and sending notice is done by the admin.

ii) Second module is handled by the user which can be a faulty or an operator. User has a right of making daily attendance, generating report.

Attendance can be taken in two ways:

i) On the basis of Subject and month.

ii) On the basis of Class.

2) List of Abbreviations:

f_id - Faculty ID

fF_name - Faculty First Name

fL_name - Faculty Last Name

f_email - Faculty Email ID

d_id - Department ID

d_name - Department Name

c_id - Course ID

c_name - Course Name

s_id - Student ID

f_name - Student First Name

l_name - Student Last Name

dob - Date of Birth

att_date - Attendance Date

s_time - Start Time

e_time - End Time

3) List of Figures:

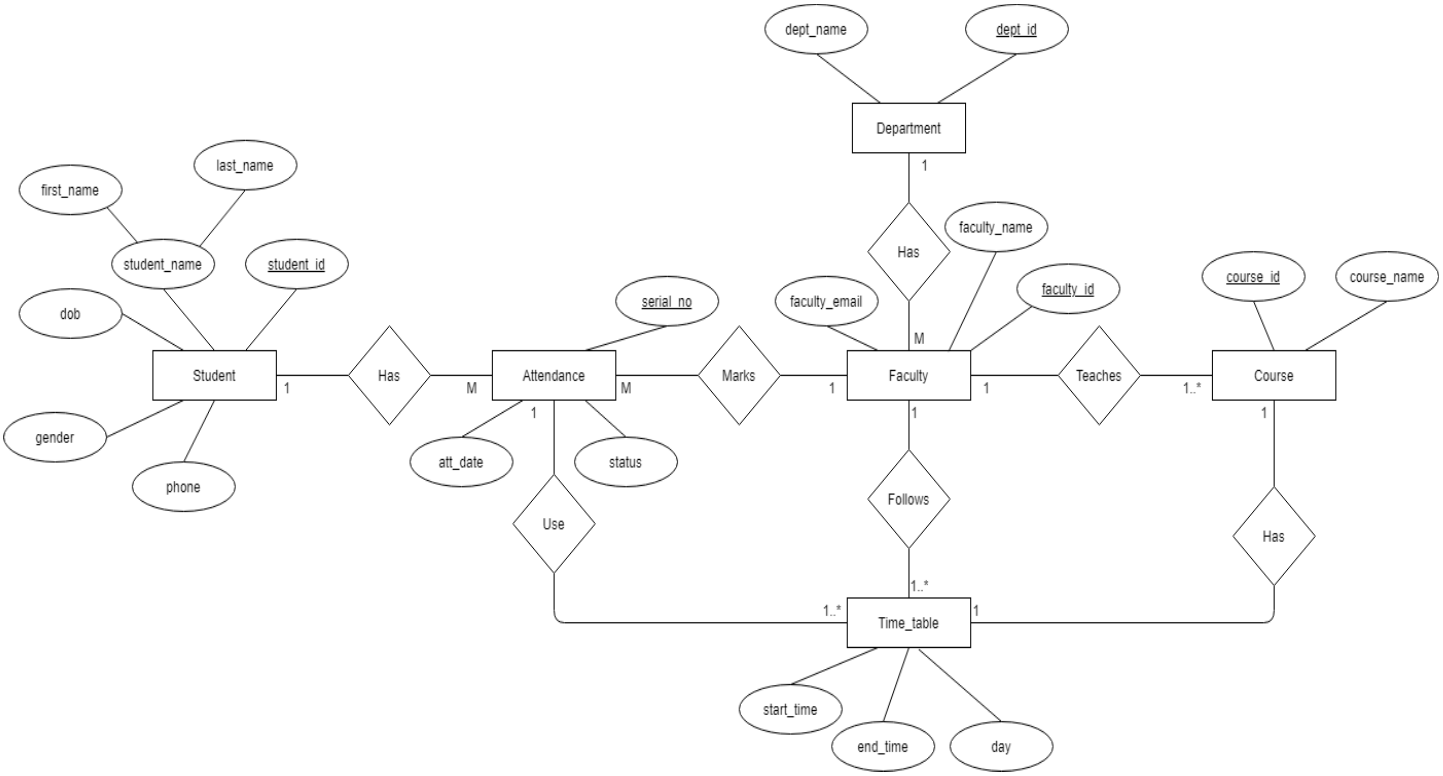
5.1 - ER Diagram For Attendance Management System

6.1 - Schema Diagram For Attendance Management System

4) List of Tables:

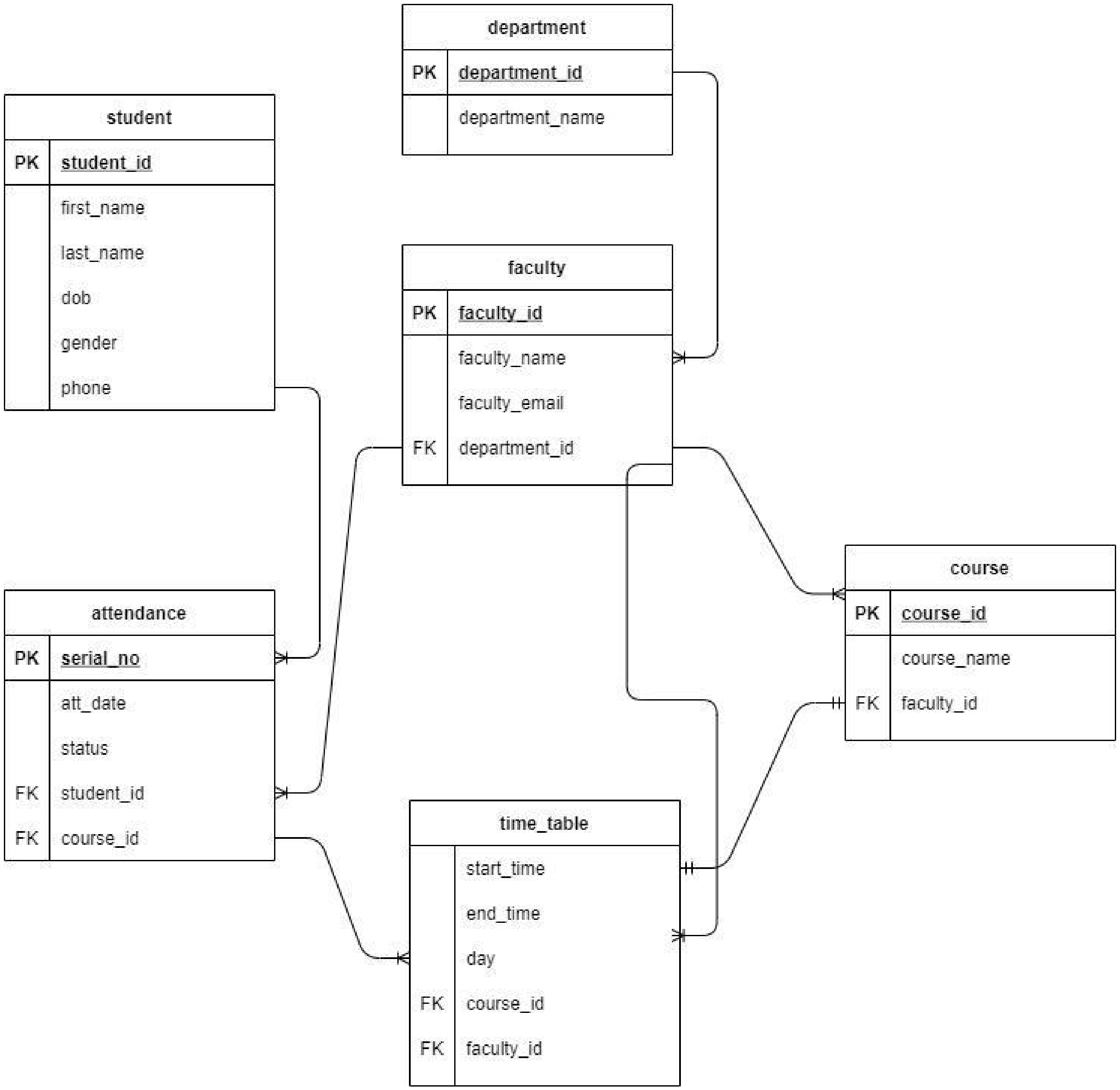
- Faculty
- Student
- Course
- Attendance
- Department
- Time Table

5) Database Design: ER Diagram



5.1 ER Diagram For Attendance Management System

6) Database Schema:



6.1 Schema Diagram For Attendance Management System

7) DDL Commands:

```
mysql> use attendance_management;

Database changed

mysql> show tables;

+--------------------------------+

| Tables_in_attendance_management |

+--------------------------------+

| faculty                         |

+--------------------------------+

1 row in set (0.03 sec)


mysql> desc faculty;

+---------+-------------+------+-----+---------+-------+

| Field   | Type        | Null | Key | Default | Extra |

+---------+-------------+------+-----+---------+-------+

| f_id    | int         | NO   | PRI | NULL    |       |

| fF_name | varchar(15) | NO   |     | NULL    |       |

| fL_name | varchar(15) | YES  |     | NULL    |       |

| f_email | varchar(35) | YES  |     | NULL    |       |

+---------+-------------+------+-----+---------+-------+

4 rows in set (0.02 sec)


mysql> create table department(d_id int, d_name varchar(35) not null, constraint
d_pk primary key(d_id));

Query OK, 0 rows affected (0.08 sec)
```

```
mysql> alter table faculty add column d_id int;

Query OK, 0 rows affected (0.07 sec)

Records: 0  Duplicates: 0  Warnings: 0

mysql> alter table faculty add constraint f_fk foreign key(d_id) references
department(d_id);

Query OK, 5 rows affected (0.14 sec)

Records: 5  Duplicates: 0  Warnings: 0


mysql> desc faculty;

+---------+-------------+------+-----+---------+-------+

| Field   | Type        | Null | Key | Default | Extra |

+---------+-------------+------+-----+---------+-------+

| f_id    | int         | NO   | PRI | NULL    |       |

| fF_name | varchar(15) | NO   |     | NULL    |       |

| fL_name | varchar(15) | YES  |     | NULL    |       |

| f_email | varchar(35) | YES  |     | NULL    |       |

| d_id    | int         | YES  | MUL | NULL    |       |

+---------+-------------+------+-----+---------+-------+

5 rows in set (0.01 sec)


mysql> desc department;

+--------+-------------+------+-----+---------+-------+

| Field  | Type        | Null | Key | Default | Extra |

+--------+-------------+------+-----+---------+-------+
| d_id   | int         | NO   | PRI | NULL    |       |

| d_name | varchar(35) | NO   |     | NULL    |       |

+--------+-------------+------+-----+---------+-------+

2 rows in set (0.01 sec)


mysql> create table course(c_id int, c_name varchar(35), f_id int, constraint c_pk primary key(c_id), constraint c_fk foreign key(f_id) references faculty(f_id));

Query OK, 0 rows affected (0.10 sec)


mysql> desc course;

+--------+-------------+------+-----+---------+-------+

| Field  | Type        | Null | Key | Default | Extra |

+--------+-------------+------+-----+---------+-------+

| c_id   | int         | NO   | PRI | NULL    |       |

| c_name | varchar(35) | YES  |     | NULL    |       |

| f_id   | int         | YES  | MUL | NULL    |       |

+--------+-------------+------+-----+---------+-------+

3 rows in set (0.03 sec)


mysql> create table student(s_id int, f_name varchar(15) not null, l_name varchar(15), dob date not null, gender char not null, phone int not null, constraint check_gender check(gender='M' or gender='F'), constraint s_pk primary key(s_id));

Query OK, 0 rows affected (0.04 sec)
```

```
mysql> desc student;

+--------+-------------+------+-----+---------+-------+

| Field  | Type        | Null | Key | Default | Extra |

+--------+-------------+------+-----+---------+-------+

| s_id   | int         | NO   | PRI | NULL    |       |

| f_name | varchar(15) | NO   |     | NULL    |       |

| l_name | varchar(15) | YES  |     | NULL    |       |

| dob    | date        | NO   |     | NULL    |       |

| gender | char(1)     | NO   |     | NULL    |       |

| phone  | int         | NO   |     | NULL    |       |

+--------+-------------+------+-----+---------+-------+

6 rows in set (0.02 sec)
```

```
mysql> create table attendance(serial_no int auto_increment, att_date date not null, status char not null, s_id int, c_id int, constraint a_pk primary key(serial_no), constraint check_status check(status='P' or status='A'), constraint a_fk1 foreign key(s_id) references student(s_id), constraint a_fk2 foreign key(c_id) references course(c_id));

Query OK, 0 rows affected (0.07 sec)
```

```
mysql> desc attendance;

+-----------+---------+------+-----+---------+----------------+

| Field     | Type    | Null | Key | Default | Extra          |

+-----------+---------+------+-----+---------+----------------+

| serial_no | int     | NO   | PRI | NULL    | auto_increment |
| att_date  | date    | NO   |     | NULL    |                |
| status    | char(1) | NO   |     | NULL    |                |
| s_id      | int     | YES  | MUL | NULL    |                |
| c_id      | int     | YES  | MUL | NULL    |                |
+-----------+---------+------+-----+---------+----------------+
5 rows in set (0.03 sec)
```

```
mysql> create table time_table(s_time time not null, e_time time not null, day varchar(20) not null, c_id int not null, f_id int not null, constraint t_fk1 foreign key(c_id) references course(c_id), constraint t_fk2 foreign key(f_id) references faculty(f_id), constraint check_day check(day='Monday' or day='Tuesday' or day='Wednesday' or day='Thursday' or day='Friday' or day='Saturday' or day='Sunday'));
Query OK, 0 rows affected (0.09 sec)
```

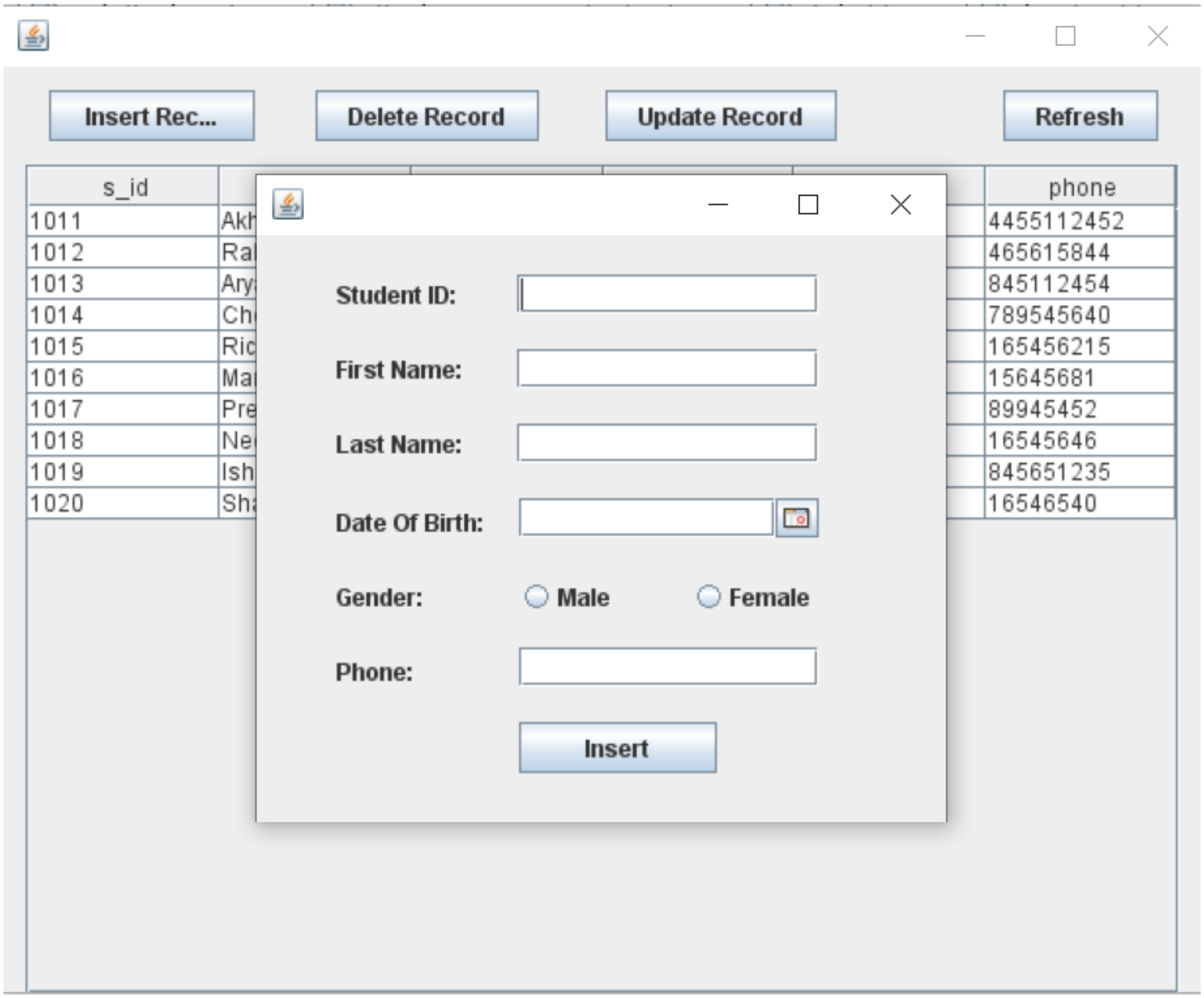
```
mysql> desc time_table;
+--------+-------------+------+-----+---------+-------+
| Field  | Type        | Null | Key | Default | Extra |
+--------+-------------+------+-----+---------+-------+
| s_time | time        | NO   |     | NULL    |       |
| e_time | time        | NO   |     | NULL    |       |
| day    | varchar(20) | NO   |     | NULL    |       |
| c_id   | int         | NO   | MUL | NULL    |       |
| f_id   | int         | NO   | MUL | NULL    |       |
+--------+-------------+------+-----+---------+-------+
5 rows in set (0.03 sec)
```

```
mysql> commit;

Query OK, 0 rows affected (0.00 sec)


mysql> alter table student modify column phone int(10);

Query OK, 0 rows affected, 1 warning (0.08 sec)

Records: 0  Duplicates: 0  Warnings: 1


mysql> alter table student modify column phone int(15);

Query OK, 0 rows affected, 1 warning (0.04 sec)

Records: 0  Duplicates: 0  Warnings: 1


mysql> desc student;

+--------+-------------+------+-----+---------+-------+

| Field  | Type        | Null | Key | Default | Extra |

+--------+-------------+------+-----+---------+-------+

| s_id   | int         | NO   | PRI | NULL    |       |

| f_name | varchar(15) | NO   |     | NULL    |       |

| l_name | varchar(15) | YES  |     | NULL    |       |

| dob    | date        | NO   |     | NULL    |       |

| gender | char(1)     | NO   |     | NULL    |       |

| phone  | int         | YES  |     | NULL    |       |

+--------+-------------+------+-----+---------+-------+

6 rows in set (0.01 sec)
```

```
mysql> alter table student modify column phone bigint;

Query OK, 0 rows affected (0.08 sec)

Records: 0  Duplicates: 0  Warnings: 0


mysql> desc student;

+--------+-------------+------+-----+---------+-------+

| Field  | Type        | Null | Key | Default | Extra |

+--------+-------------+------+-----+---------+-------+

| s_id   | int         | NO   | PRI | NULL    |       |

| f_name | varchar(15) | NO   |     | NULL    |       |

| l_name | varchar(15) | YES  |     | NULL    |       |

| dob    | date        | NO   |     | NULL    |       |

| gender | char(1)     | NO   |     | NULL    |       |

| phone  | bigint      | YES  |     | NULL    |       |

+--------+-------------+------+-----+---------+-------+

6 rows in set (0.00 sec)
```

8) DML along with the results of the queries:

Note: Insert, Delete, Update Using FrontEnd eg.,



s_id		phone
1011	Akh	4455112452
1012	Ra	465615844
1013	Ary	845112454
1014	Ch	789545640
1015	Ric	165456215
1016	Ma	15645681
1017	Pre	89945452
1018	Ne	16545646
1019	Ish	845651235
1020	Sh	16546540

Student ID:

First Name:

Last Name:

Date Of Birth:

Gender: ○ Male ○ Female

Phone:

```
mysql> insert into department values(001, 'CSE');

Query OK, 1 row affected (0.01 sec)


mysql> select * from faculty;

+------+----------+-------------+---------------------------------+------+

| f_id | fF_name  | fL_name     | f_email                         | d_id |

+------+----------+-------------+---------------------------------+------+
|  101 | lalit    | kulkarni    | lalitkulkarni@mitwpu.edu.in       | NULL |
|  102 | vaishali | suryawanshi | vaishalisuryawanshi@mitwpu.edu.in | NULL |
|  103 | sajida   | shigalkar   | sajidashigalkar@mitwpu.edu.in     | NULL |
|  104 | ruhi     | patankar    | ruhipatankar@mitwpu.edu.in        | NULL |
|  105 | sampada  | kale        | sampadakale@mitwpu.edu.in         | NULL |
+------+----------+-------------+-----------------------------------+------+
5 rows in set (0.02 sec)

mysql> update faculty set d_id=001;
Query OK, 5 rows affected (0.01 sec)
Rows matched: 5  Changed: 5  Warnings: 0

mysql> select * from faculty;
+------+----------+-------------+-----------------------------------+------+
| f_id | fF_name  | fL_name     | f_email                           | d_id |
+------+----------+-------------+-----------------------------------+------+
|  101 | lalit    | kulkarni    | lalitkulkarni@mitwpu.edu.in       |    1 |
|  102 | vaishali | suryawanshi | vaishalisuryawanshi@mitwpu.edu.in |    1 |
|  103 | sajida   | shigalkar   | sajidashigalkar@mitwpu.edu.in     |    1 |
|  104 | ruhi     | patankar    | ruhipatankar@mitwpu.edu.in        |    1 |
|  105 | sampada  | kale        | sampadakale@mitwpu.edu.in         |    1 |
+------+----------+-------------+-----------------------------------+------+
5 rows in set (0.00 sec)
```

```
mysql> select d_name from department d inner join faculty f where f.d_id= d.d_id
and f.fF_name='lalit';
+--------+
| d_name |
+--------+
| CSE    |
+--------+
1 row in set (0.00 sec)
mysql> insert into student values(1032191369, 'Mustafa', 'Mokashi', '2001-08-07',
'm', 7026713017);
Query OK, 1 row affected (0.01 sec)

mysql> select * from student;
+------------+---------+---------+------------+--------+------------+
| s_id       | f_name  | l_name  | dob        | gender | phone      |
+------------+---------+---------+------------+--------+------------+
| 1032191369 | Mustafa | Mokashi | 2001-08-07 | m      | 7026713017 |
+------------+---------+---------+------------+--------+------------+
1 row in set (0.00 sec)

mysql> insert into course values(311, 'Computer Networks', 101);
Query OK, 1 row affected (0.03 sec)

mysql> insert into course values(312, 'Database Management System', 102);
Query OK, 1 row affected (0.01 sec)
```

```
mysql> insert into course values(313, 'Software Modelling & Design', 103);
Query OK, 1 row affected (0.01 sec)
mysql> insert into course values(314, 'Theory of Computation', 104);
Query OK, 1 row affected (0.01 sec)

mysql> select * from course;
+------+-----------------------------+------+
| c_id | c_name                      | f_id |
+------+-----------------------------+------+
|  311 | Computer Networks           |  101 |
|  312 | Database Management System  |  102 |
|  313 | Software Modelling & Design |  103 |
|  314 | Theory of Computation       |  104 |
+------+-----------------------------+------+
4 rows in set (0.00 sec)

mysql> insert into time_table values('08:00', '08:50', 'Monday', 314, 104);
Query OK, 1 row affected (0.01 sec)

mysql> insert into time_table values('09:00', '09:50', 'Monday', 312, 102);
Query OK, 1 row affected (0.01 sec)

mysql> insert into time_table values('10:00', '10:50', 'Monday', 311, 101);
Query OK, 1 row affected (0.01 sec)
```

```
mysql> insert into time_table values('08:00', '08:50', 'Tuesday', 312, 102);
Query OK, 1 row affected (0.01 sec)

mysql> insert into time_table values('09:00', '09:50', 'Tuesday', 311, 101);
Query OK, 1 row affected (0.01 sec)

mysql> insert into time_table values('08:00', '08:50', 'Wednesday', 313, 103);
Query OK, 1 row affected (0.01 sec)

mysql> insert into time_table values('09:00', '09:50', 'Wednesday', 314, 104);
Query OK, 1 row affected (0.01 sec)

mysql> insert into time_table values('08:00', '08:50', 'Thursday', 312, 102);
Query OK, 1 row affected (0.01 sec)

mysql> insert into time_table values('09:00', '09:50', 'Thursday', 313, 103);
Query OK, 1 row affected (0.01 sec)

mysql> insert into time_table values('08:00', '08:50', 'Friday', 311, 101);
Query OK, 1 row affected (0.01 sec)

mysql> insert into time_table values('09:00', '09:50', 'Friday', 314, 104);
Query OK, 1 row affected (0.01 sec)
mysql> insert into time_table values('10:00', '10:50', 'Friday', 313, 103);
```

```
Query OK, 1 row affected (0.01 sec)

mysql> select * from time_table;
+----------+----------+-----------+------+------+
| s_time   | e_time   | day       | c_id | f_id |
+----------+----------+-----------+------+------+
| 08:00:00 | 08:50:00 | Monday    |  314 |  104 |
| 09:00:00 | 09:50:00 | Monday    |  312 |  102 |
| 10:00:00 | 10:50:00 | Monday    |  311 |  101 |
| 08:00:00 | 08:50:00 | Tuesday   |  312 |  102 |
| 09:00:00 | 09:50:00 | Tuesday   |  311 |  101 |
| 08:00:00 | 08:50:00 | Wednesday |  313 |  103 |
| 09:00:00 | 09:50:00 | Wednesday |  314 |  104 |
| 08:00:00 | 08:50:00 | Thursday  |  312 |  102 |
| 09:00:00 | 09:50:00 | Thursday  |  313 |  103 |
| 08:00:00 | 08:50:00 | Friday    |  311 |  101 |
| 09:00:00 | 09:50:00 | Friday    |  314 |  104 |
| 10:00:00 | 10:50:00 | Friday    |  313 |  103 |
+----------+----------+-----------+------+------+
12 rows in set (0.00 sec)

mysql> insert into attendance(att_date, status, s_id, c_id) values('2021-09-30', 'P',
1032191369, 312);
Query OK, 1 row affected (0.01 sec)
```

```
mysql> select * from attendance;

+-----------+------------+--------+------------+------+

| serial_no | att_date   | status | s_id       | c_id |

+-----------+------------+--------+------------+------+

|         1 | 2021-09-30 | P      | 1032191369 |  312 |

+-----------+------------+--------+------------+------+

1 row in set (0.00 sec)


mysql> select a.*,c.c_name from attendance a inner join course c where
a.c_id=c.c_id;

+-----------+------------+--------+------------+------+--------------------------+

| serial_no | att_date   | status | s_id       | c_id | c_name                   |

+-----------+------------+--------+------------+------+--------------------------+

|         1 | 2021-09-30 | P      | 1032191369 |  312 | Database Management System
|

+-----------+------------+--------+------------+------+--------------------------+

1 row in set (0.00 sec)
```

9) **DCL Commands:**

```
mysql> create user "faculty"@"localhost" identified by "pass123";

Query OK, 0 rows affected (0.05 sec)


mysql> grant insert, delete, update, select on attendance_management.* to
"faculty"@"localhost";

Query OK, 0 rows affected (0.02 sec)


mysql> create user "admin"@"localhost" identified by "admin123";

Query OK, 0 rows affected (0.07 sec)


mysql> grant all privileges on attendance_management.* to
"admin"@"localhost";

Query OK, 0 rows affected (0.02 sec)


mysql> revoke delete on attendance_management.* from "faculty"@"localhost";

Query OK, 0 rows affected (0.01 sec)


mysql> grant execute on procedure getstudent to "faculty"@"localhost";

Query OK, 0 rows affected (0.01 sec)


mysql> grant execute on procedure defaulterlist to "faculty"@"localhost";

Query OK, 0 rows affected (0.01 sec)


mysql> grant execute on procedure absentlist to "faculty"@"localhost";
```

```
Query OK, 0 rows affected (0.01 sec)


mysql> grant execute on procedure presentlist to "faculty"@"localhost";

Query OK, 0 rows affected (0.01 sec)


mysql> show grants for 'faculty'@'localhost';

+------------------------------------------------------------------------------------------+

| Grants for faculty@localhost                                                       |

+------------------------------------------------------------------------------------------+

| GRANT USAGE ON *.* TO `faculty`@`localhost`                                              |

| GRANT SELECT, INSERT, UPDATE ON `attendance_management`.* TO
`faculty`@`localhost`         |

| GRANT EXECUTE ON PROCEDURE `attendance_management`.`absentlist` TO
`faculty`@`localhost`    |

| GRANT EXECUTE ON PROCEDURE `attendance_management`.`defaulterlist` TO
`faculty`@`localhost` |

| GRANT EXECUTE ON PROCEDURE `attendance_management`.`getstudent` TO
`faculty`@`localhost`    |

| GRANT EXECUTE ON PROCEDURE `attendance_management`.`presentlist` TO
`faculty`@`localhost`   |

+------------------------------------------------------------------------------------------+

6 rows in set (0.01 sec)


mysql> show grants for 'admin'@'localhost';

+--------------------------------------------------------------------+

| Grants for admin@localhost                                         |
+----------------------------------------------------------------------+
| GRANT USAGE ON *.* TO `admin`@`localhost`                            |
| GRANT ALL PRIVILEGES ON `attendance_management`.* TO `admin`@`localhost`
|
+----------------------------------------------------------------------+
2 rows in set (0.00 sec)
```

10) Triggers:

```
mysql> create trigger pa_no_trig1
    -> after insert on student
    -> for each row
    -> begin
    -> insert into pa_no
    -> set stud_id=new.s_id,
    -> present=0,
    -> absent=0;
    -> end//
Query OK, 0 rows affected (0.02 sec)
```

```
mysql> create trigger pa_no_trig2
    -> after insert on attendance
    -> for each row
    -> if(new.status='P')
    -> then
    -> update pa_no
    -> set present=present+1 where stud_id=new.s_id;
    -> end if;
    -> end//
Query OK, 0 rows affected (0.04 sec)

mysql> create trigger pa_no_trig3
    -> after insert on attendance
    -> for each row
    -> if(new.status='A')
    -> then
    -> update pa_no
    -> set absent=absent+1 where stud_id=new.s_id;
    -> end if;
    -> end//
Query OK, 0 rows affected (0.03 sec)
```

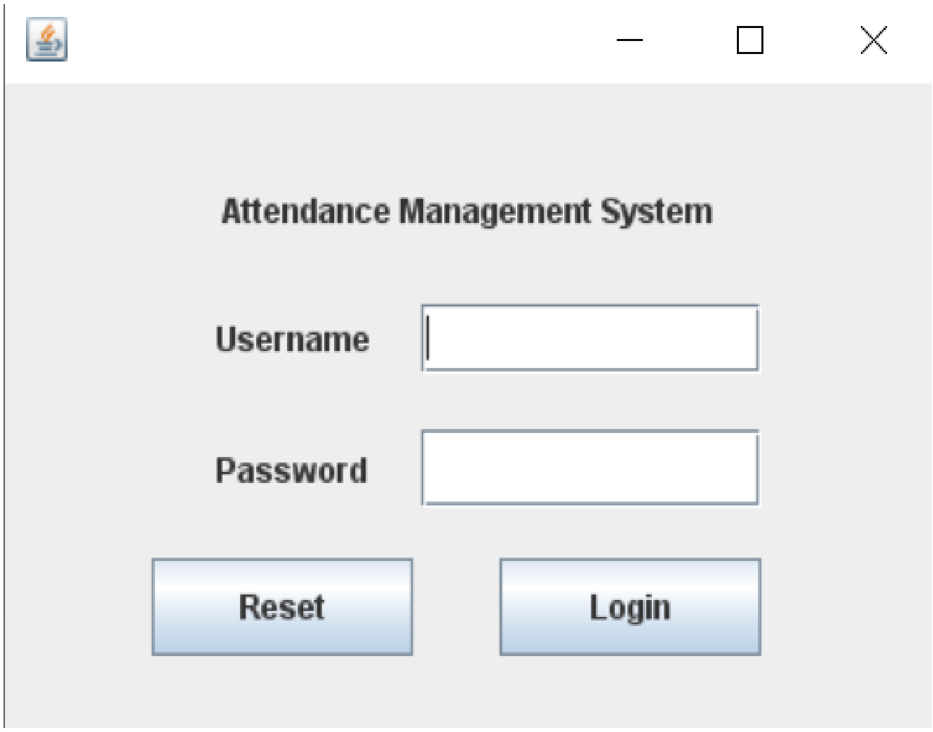
11) **PL SQL Procedures and Functions:**

```
mysql> create procedure updatepercent()
    -> begin
    -> declare done int default 0;
    -> declare p1 int;
    -> declare a1 int;
    -> declare id int;
    -> declare pa_select cursor for select stud_id, present, absent from pa_no;
    -> declare continue handler for not found set done=1;
    -> open pa_select;
    -> repeat
    -> fetch pa_select into id, p1, a1;
    -> update pa_no
    -> set percentage=(p1/(p1+a1))*100 where stud_id=id;
    -> until done
    -> end repeat;
    -> close pa_select;
    -> end//
Query OK, 0 rows affected (0.02 sec)
mysql> create procedure getstudent(in sidd int)
    -> begin
    -> select s.s_id as Student_ID, s.f_name as First_Name, s.l_name as Last_Name,
s.phone as Phone_Number,
p.percentage as Attendance_Percentage from student s inner join pa_no p where
p.stud_id=sidd and s.s_id=sidd;
```

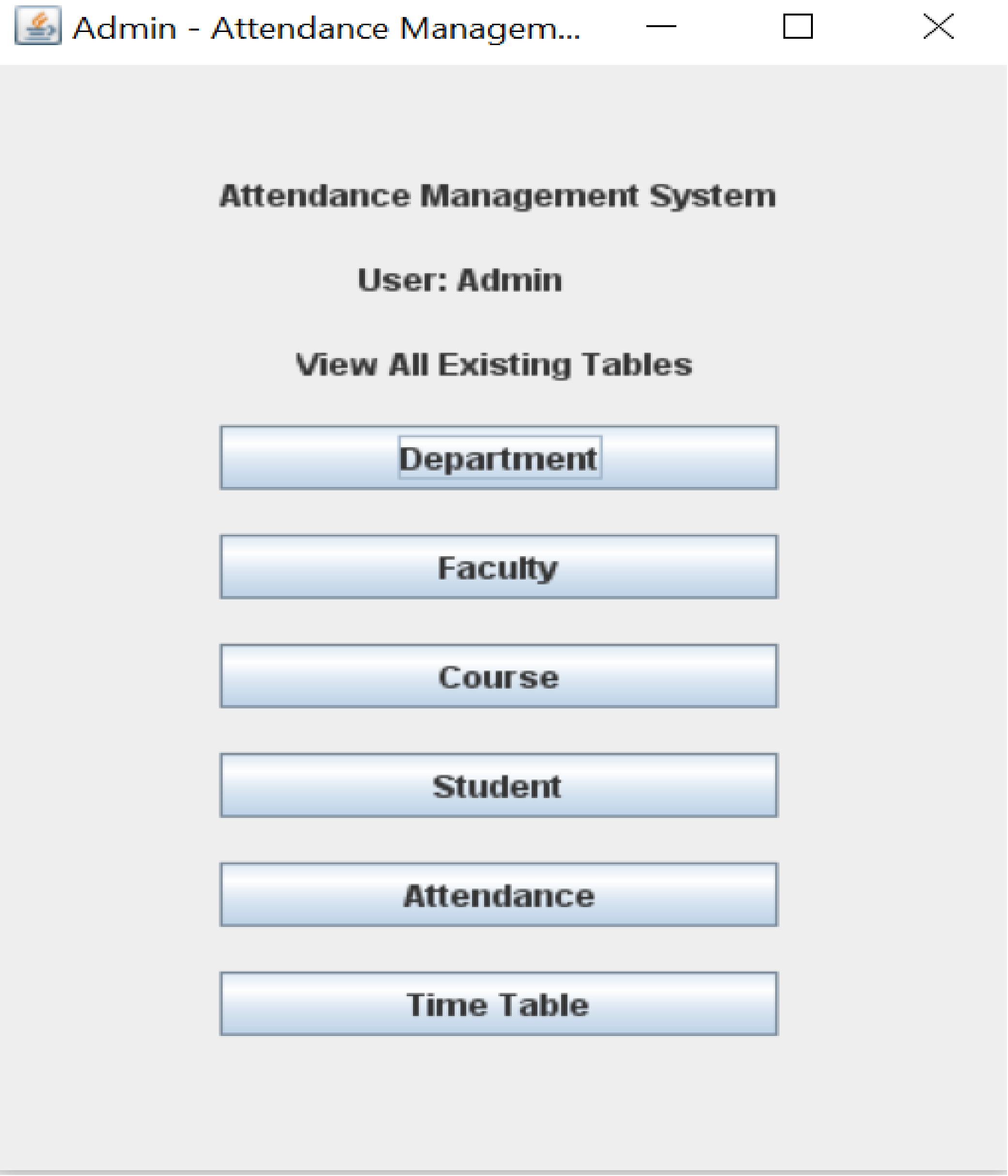
```
    -> end//
Query OK, 0 rows affected (0.01 sec)


mysql> create procedure defaulterlist()
    -> begin
    -> select s.s_id as Student_ID, s.f_name as First_Name, s.l_name as Last_Name,
s.phone as Phone_Number, p.percentage as Attendance_Percentage from student
s inner join pa_no p where p.percentage<80 and p.stud_id=s.s_id;
    -> end//
Query OK, 0 rows affected (0.01 sec)


mysql> create procedure presentlist(in atdat date, in coid int)
    -> begin
    -> select s.s_id as Student_ID, s.f_name as First_Name, s.l_name as Last_Name,
a.status as Status from student s inner join attendance a where a.att_date=atdat
and a.status='P' and a.c_id=coid and a.s_id=s.s_id;
    -> end//
Query OK, 0 rows affected (0.01 sec)


mysql> create procedure absentlist(in atdat date, in coid int)
    -> begin
    -> select s.s_id as Student_ID, s.f_name as First_Name, s.l_name as Last_Name,
a.status as Status from student s inner join attendance a where a.att_date=atdat
and a.status='A' and a.c_id=coid and a.s_id=s.s_id;
    -> end//
Query OK, 0 rows affected (0.01 sec)
```

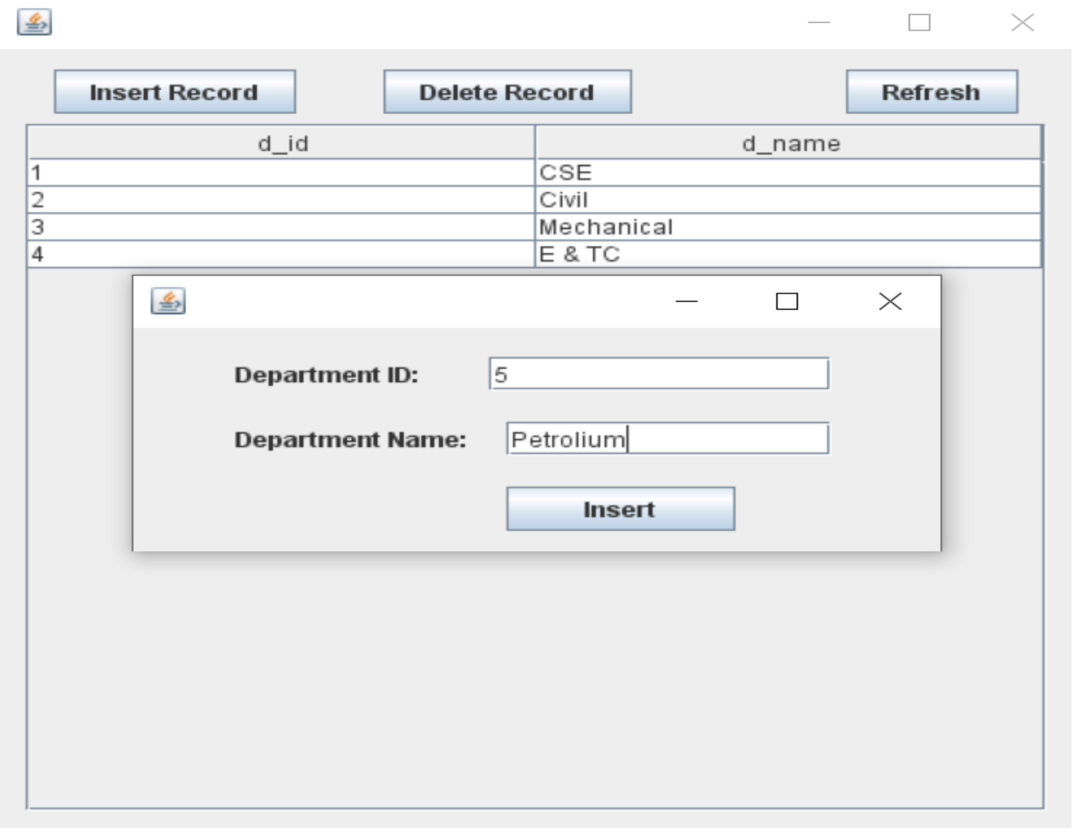
12) **Frontend GUI Screenshots:**

- ➢ Login Page:



➢ Admin Interface:

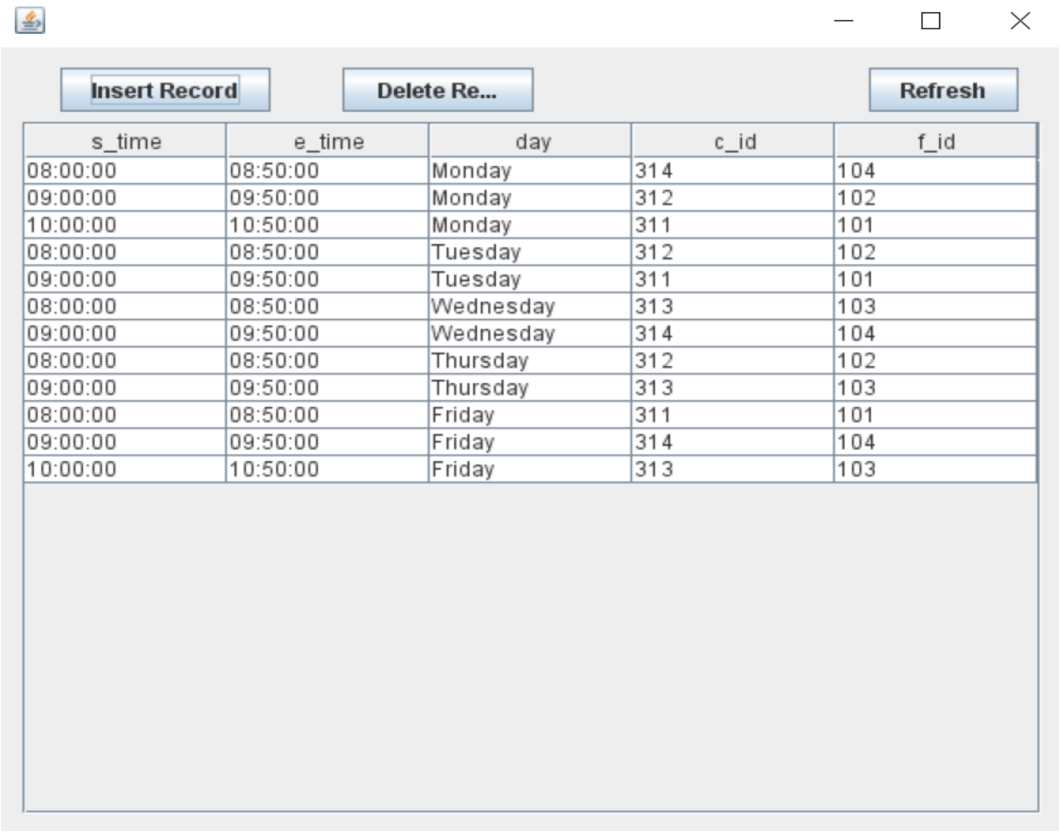




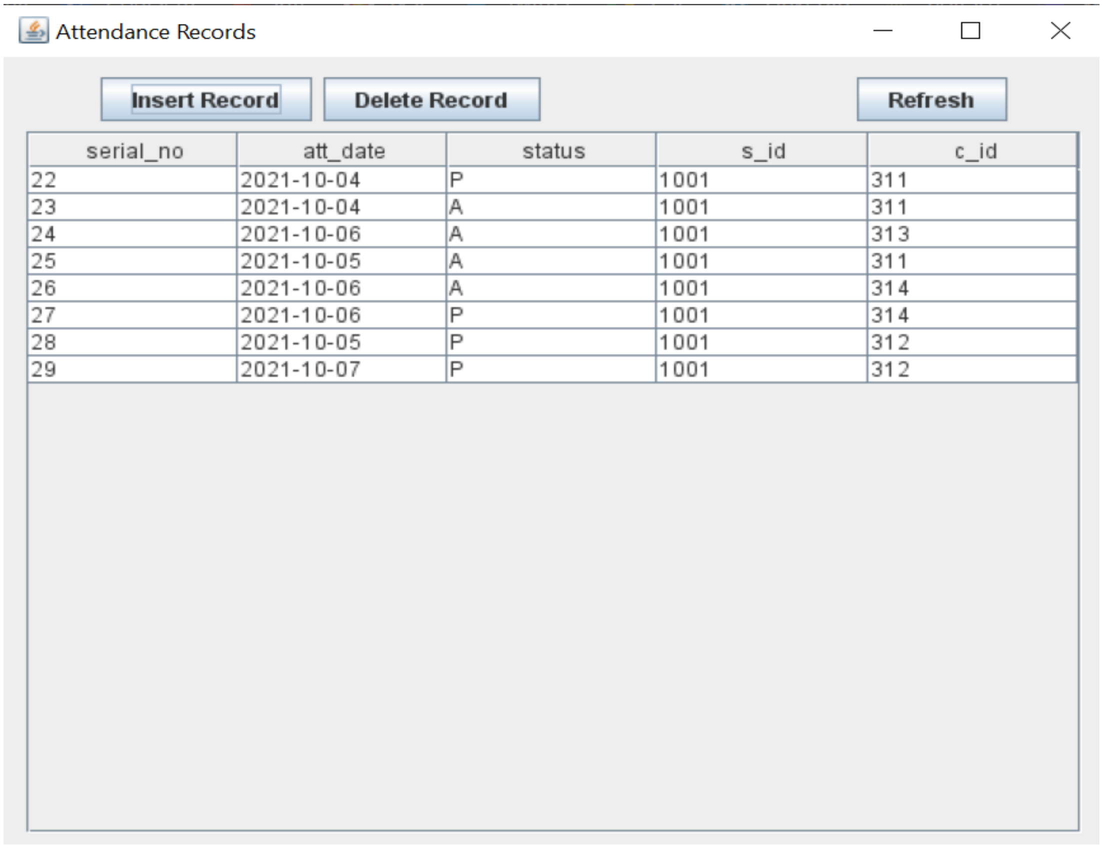
d_id	d_name
1	CSE
2	Civil
3	Mechanical
4	E & TC

Department ID: 5

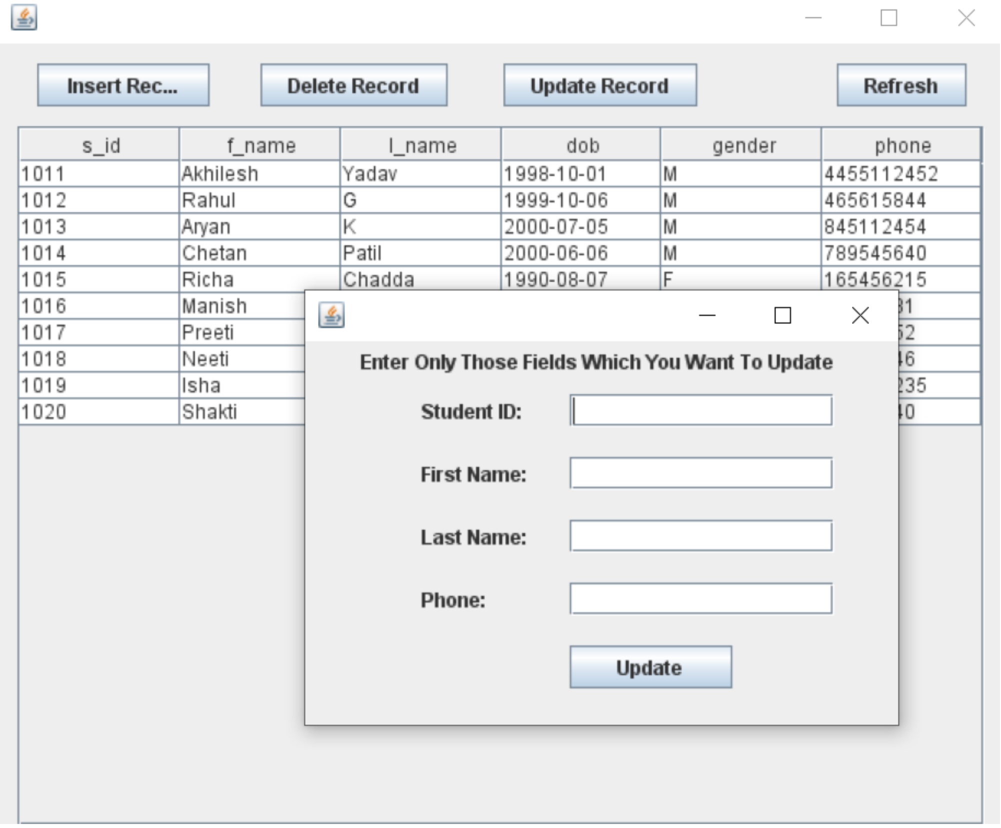
Department Name: Petrolium



s_time	e_time	day	c_id	f_id
08:00:00	08:50:00	Monday	314	104
09:00:00	09:50:00	Monday	312	102
10:00:00	10:50:00	Monday	311	101
08:00:00	08:50:00	Tuesday	312	102
09:00:00	09:50:00	Tuesday	311	101
08:00:00	08:50:00	Wednesday	313	103
09:00:00	09:50:00	Wednesday	314	104
08:00:00	08:50:00	Thursday	312	102
09:00:00	09:50:00	Thursday	313	103
08:00:00	08:50:00	Friday	311	101
09:00:00	09:50:00	Friday	314	104
10:00:00	10:50:00	Friday	313	103



serial_no	att_date	status	s_id	c_id
22	2021-10-04	P	1001	311
23	2021-10-04	A	1001	311
24	2021-10-06	A	1001	313
25	2021-10-05	A	1001	311
26	2021-10-06	A	1001	314
27	2021-10-06	P	1001	314
28	2021-10-05	P	1001	312
29	2021-10-07	P	1001	312



s_id	f_name	l_name	dob	gender	phone
1011	Akhilesh	Yadav	1998-10-01	M	4455112452
1012	Rahul	G	1999-10-06	M	465615844
1013	Aryan	K	2000-07-05	M	845112454
1014	Chetan	Patil	2000-06-06	M	789545640
1015	Richa	Chadda	1990-08-07	F	165456215
1016	Manish				31
1017	Preeti				52
1018	Neeti				46
1019	Isha				235
1020	Shakti				40

Enter Only Those Fields Which You Want To Update

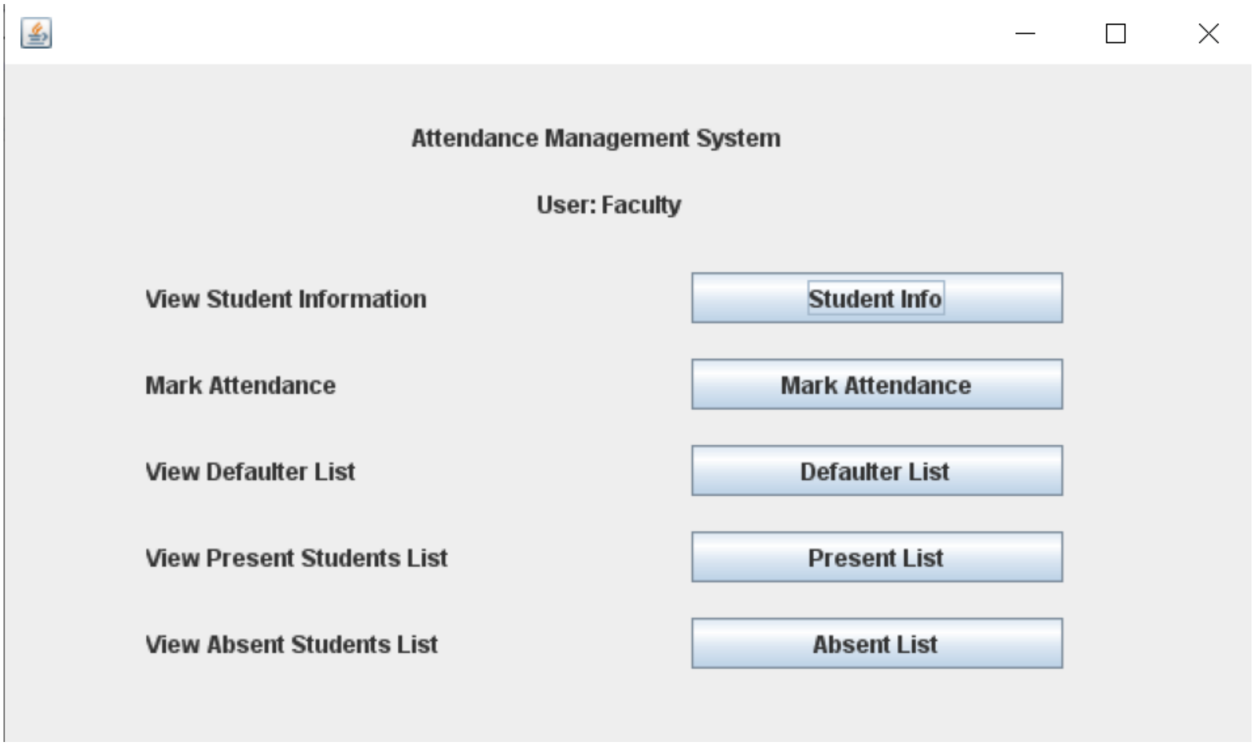
Student ID:

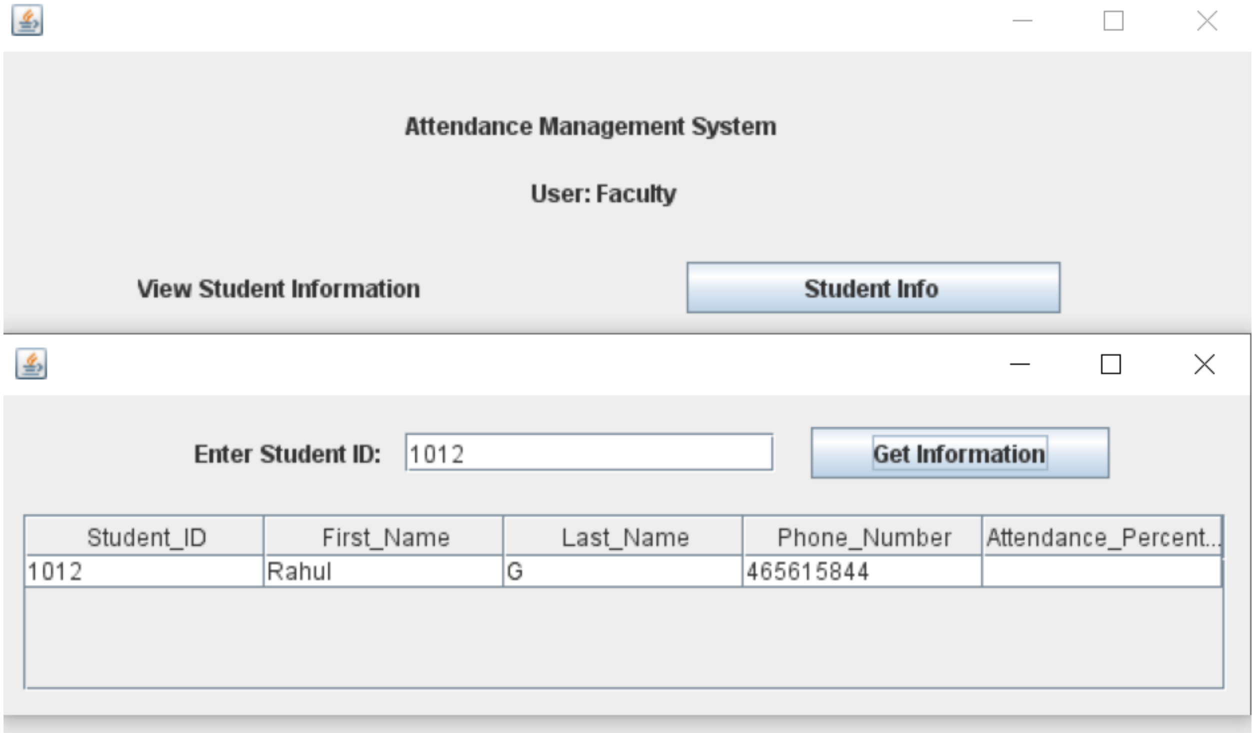
First Name:

Last Name:

Phone:

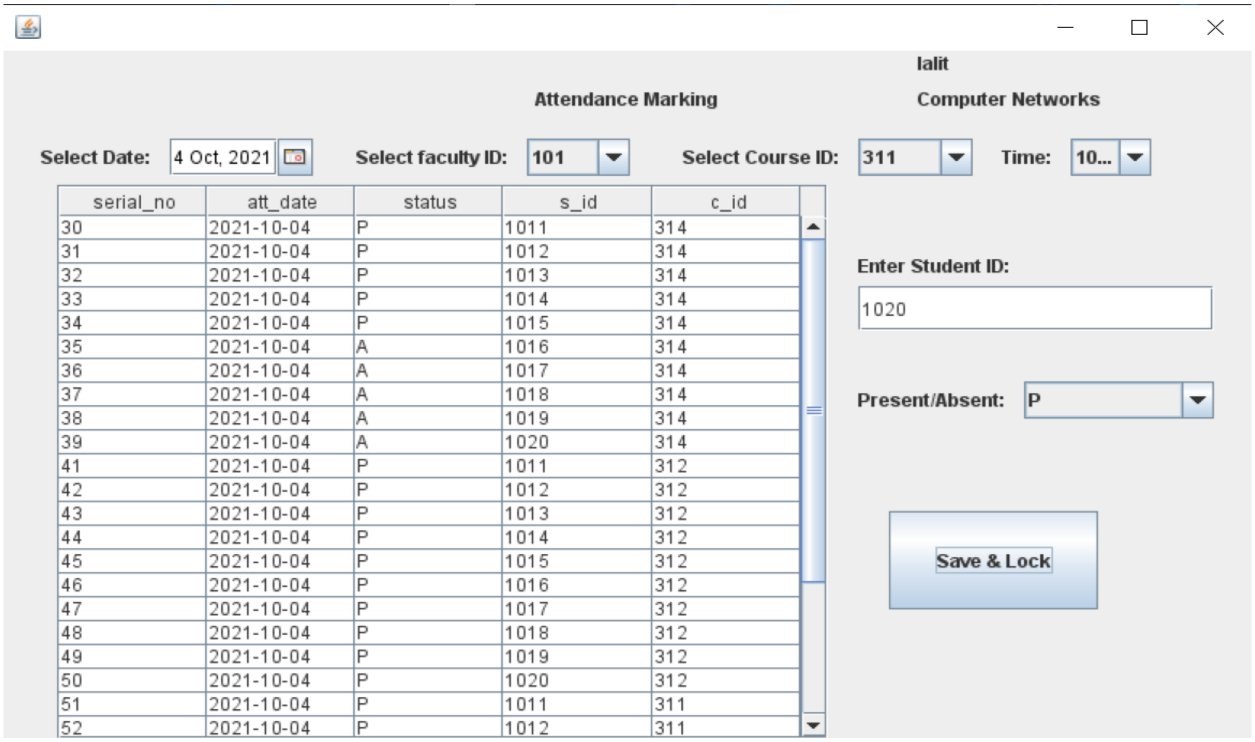
➢ Faculty Interface:





Enter Student ID: 1012

Student_ID	First_Name	Last_Name	Phone_Number	Attendance_Percent...
1012	Rahul	G	465615844	



lalit

Attendance Marking

Computer Networks

Select Date: 4 Oct, 2021

Select faculty ID: 101

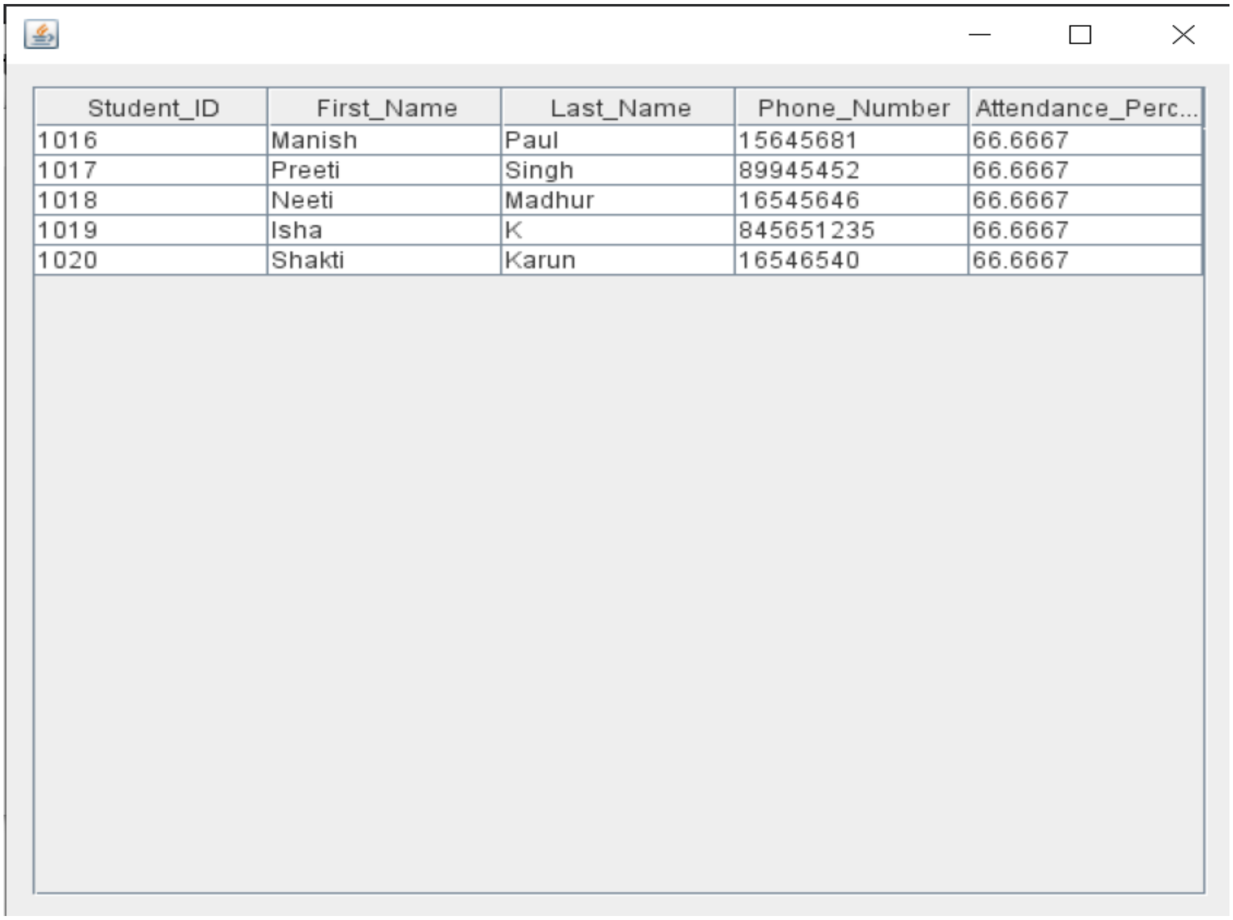
Select Course ID: 311

Time: 10...

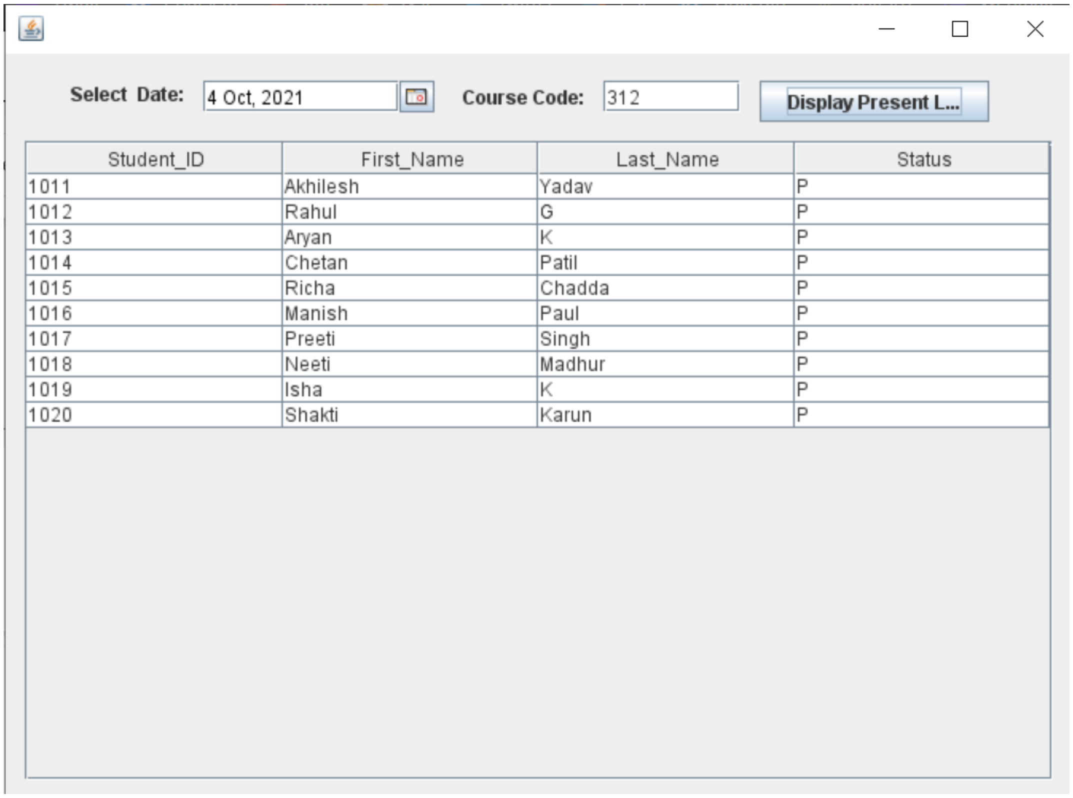
serial_no	att_date	status	s_id	c_id
30	2021-10-04	P	1011	314
31	2021-10-04	P	1012	314
32	2021-10-04	P	1013	314
33	2021-10-04	P	1014	314
34	2021-10-04	P	1015	314
35	2021-10-04	A	1016	314
36	2021-10-04	A	1017	314
37	2021-10-04	A	1018	314
38	2021-10-04	A	1019	314
39	2021-10-04	A	1020	314
41	2021-10-04	P	1011	312
42	2021-10-04	P	1012	312
43	2021-10-04	P	1013	312
44	2021-10-04	P	1014	312
45	2021-10-04	P	1015	312
46	2021-10-04	P	1016	312
47	2021-10-04	P	1017	312
48	2021-10-04	P	1018	312
49	2021-10-04	P	1019	312
50	2021-10-04	P	1020	312
51	2021-10-04	P	1011	311
52	2021-10-04	P	1012	311

Enter Student ID: 1020

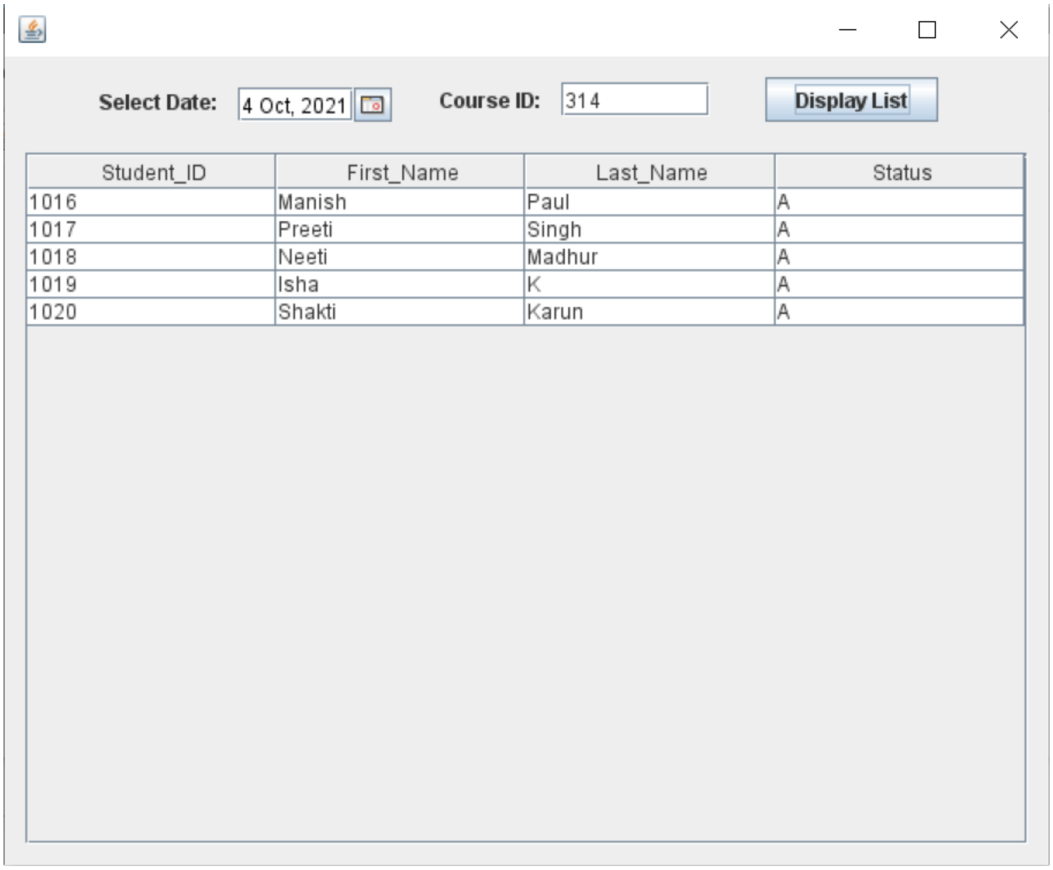
Present/Absent: P



Student_ID	First_Name	Last_Name	Phone_Number	Attendance_Perc...
1016	Manish	Paul	15645681	66.6667
1017	Preeti	Singh	89945452	66.6667
1018	Neeti	Madhur	16545646	66.6667
1019	Isha	K	845651235	66.6667
1020	Shakti	Karun	16546540	66.6667



Student_ID	First_Name	Last_Name	Status
1011	Akhilesh	Yadav	P
1012	Rahul	G	P
1013	Aryan	K	P
1014	Chetan	Patil	P
1015	Richa	Chadda	P
1016	Manish	Paul	P
1017	Preeti	Singh	P
1018	Neeti	Madhur	P
1019	Isha	K	P
1020	Shakti	Karun	P



Student_ID	First_Name	Last_Name	Status
1016	Manish	Paul	A
1017	Preeti	Singh	A
1018	Neeti	Madhur	A
1019	Isha	K	A
1020	Shakti	Karun	A

13) **Conclusion:**

The Attendance Management System helps the user to work with the attendance, fees, course update. It reduces the amount of manual data entry and gives greater efficiency. The user interface of it is very friendly and can be easily used by anyone. It also decreases the amount of the time taken to write details. All the details can be seen only by verified users. The Attendance management system is a solution to all the problems related to attendance,fee status. It is performing all the tasks accurately and is doing the work for which it is made and this system can be implemented in various colleges and schools.

14) **References in IEEE Format:**

➔ Connecting to a MySQL Database - Apache NetBeans-

http://netbeans.apache.org/kb/docs/ide/mysql.html

➔ JFrame (Java Platform SE 7) - Oracle Help Center

https://docs.oracle.com/javase/7/docs/api/javax/swing/JFrame.html

➔ SWING - JFrame Class - Tutorialspoint

https://www.tutorialspoint.com/swing/swing_jframe.html

➔ javax.swing.JFrame java code examples | Tabnine

https://www.tabnine.com/code/java/classes/javax.swing.JFrame

➔ Designing a Swing GUI in NetBeans IDE

https://netbeans.apache.org/kb/docs/java/quickstart-gui.html

➔ How to Create a JDBC Application in NetBeans | Developer.com

https://www.developer.com/database/creating-a-jdbc-application-in-netbeans-a-step-by-step-guide/