HOSTEL MANAGEMENT SYSTEM

A Case Study Submitted to

DEPARTMENT
of
COMPUTER SCIENCE AND SYSTEMS ENGINEERING

Submitted by

C.SAI CHARAN	**20121A1519**
K.SUMANTH	**20121A1552**

Under the Guidance of
M. P. Yogendra Prasad, M.Tech
Assistant Professor
Dept. of CSSE, SVEC

Department of Computer Science and Systems Engineering
Sree Vidyanikethan Engineering College (Autonomous)
Sree Sainath Nagar, Tirupati – 517 102
(2021-2022)

SREE VIDYANIKETHAN ENGINEERING COLLEGE

(AUTONOMOUS)
Sree Sainath Nagar, Tirupati

DEPARTMENT OF COMPUTER SCIENCE AND SYSTEMS ENGINEERING

CERTIFICATE

This is to certify that the project report entitled

HOSTEL MANAGEMENT SYSTEM

is the Bonafide work done by

C.SAI CHARAN	**20121A1519**
K.SUMANTH	**20121A1552**

in the Department of **Computer Science and Systems Engineering**, and submitted to Computer Science and Systems Engineering during the academic year 2021-2022. This work has been carried out under my supervision.

Guide:

P. Yogendra Prasad,
Assistant Professor
Dept. of CSSE

Head:

Dr. K. Ramani
Professor & Head
Dept. of CSSE

INTERNAL EXAMINER

EXTERNAL EXAMINER

DEPARTMENT OF COMPUTER SCIENCE AND SYSTEMS ENGINEERING

VISION

To become a centre of excellence in Computer Sciences and Systems Engineering through teaching, training, research, and innovation to create quality engineering professionals who can solve the growing complex problems of the society.

MISSION

- ✓ Established with the cause of development of technical education in advanced computer sciences and engineering with applications to systems there by serving the society and nation.
- ✓ Transfer of Knowledge through contemporary curriculum and fostering faculty and student development.
- ✓ Create keen interest for research and innovation among students and faculty by understanding the needs of the society and industry.
- ✓ Skill development among diversity of students in technical domains and profession for development of systems and processes to meet the demands of the industry and research.
- ✓ Imbibing values and ethics in students for prospective and promising engineering profession and develop a sense of respect for all.

PROGRAM EDUCATIONAL OBJECTIVES

1. Demonstrate competencies in the Computer Science domain and Management with an ability to comprehend, analyze, design, and create software systems for pursuing advanced studies in the areas of interest.
2. Evolve as entrepreneurs or be employed by acquiring required skill sets for developing computer systems and solutions in multi-disciplinary areas.
3. Exhibit progression and professional skill development in Computer programming and systems development with ethical attitude through life-long learning.

PROGRAM SPECIFIC OUTCOMES

PSO1: Employ Systems Approach to model the solutions for real life problems, design and develop software systems by applying Modern Tools.

PSO2: Develop solutions using novel algorithms in High Performance Computing and Data Science.

PSO3: Use emerging technologies for providing security and privacy to design, deploy and manage network systems.

PROGRAM OUTCOMES

1. Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. Identify, formulate, research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4. Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.
6. Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal, and cultural issues, and the consequent responsibilities relevant to the professional engineering practice.
7. Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8. Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
10. Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
11. Demonstrate knowledge and understanding of the engineering and management principles and apply these to one’s own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
12. Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

II B. Tech. – II Semester

(20BT40531) DATABASE MANAGEMENT SYSTEMS LAB

COURSE OUTCOMES

CO1. Analyze the requirements of a given database problem and design viable ER-Models for implementation of database.

CO2. Create database schemas, select, and apply suitable integrity constraints for querying databases using SQL interface.

CO3. Develop and interpret PL/SQL blocks to centralize database applications for maintainability and reusability.

CO4. Develop database applications for societal applications such as ticket reservation system, employee payroll system using modern tools.

CO5. Work independently and communicate effectively in oral and written forms.

ABSTRACT

This Project “HOSTEL MANAGEMENT INFORMATION SYSTEM” targeted for the College Hostel integrates the transaction management of the Hostel for better control and timely response. This eliminates time delay and paper transactions being marked. The warden is provided with a better control over the transactions like adding the details of new students in the hostel, modifying the details of the students, deleting the students, viewing the students details in the Hostel. This project’s main motto is to reduce the effort of Wardens and provide better service to the students. The goal of this project is to develop a system for the computerization of the Hostel. The common transactions of the hostel includes the maintenance of mess bills, information about students in the hostel, enrolling of new students and their payments and dues etc are stored into the databases and reports are generated according to the user requirements.

Hostel Management System Database Design is software developed for managing various activities in the hostel. For the past few years, the number of travellers around the world is increasing rapidly. Thereby the number of hotels is also increasing for the accommodation of travelers around the world.

TABLE OF CONTENTS

Title **Page No.**

CHAPTER 1. INTRODUCTION

1.1 Introduction to the topic:

In our current era of automated systems with it being either software or hardware, it’s not advisable to be using manual system. Hostels without a management system are usually done manually. Registration forms verification to other data saving processes are done manually and most at times, they are written on paper. Thus, a lot of repetitions can be avoided with an automated system. The drawbacks of existing systems lead to the design of a computerized system that will help reduce a lot of manual inputs. With this system in place, we can improve the efficiency of the system, thus overcome the drawbacks of the existing manual system.

This system is designed in favor of the hostel management which helps them to save the records of the students about their rooms and other things. It helps them from the manual work from which it is very difficult to find the record of the students and the mess bills of the students, and the information of about those ones who had left the hostel years before.

This system gives an idea about how a student and fee details, room allocation, mess expenditure is maintained in a better way. The hostel management system will also contain special features like how many students are in a room, student’s id and free rooms or space available. The administration has a unique identity for each member as well as student’s details

Description:

This Database for the Hostel Management system is a tedious the process by manual way, since it involves workload time consumption. With Complete Management System, we can easily manage the customer details, room/room status, booking, hostel branch, employees, payments, and transactions. Thus, there are a lot of repetitions that can be easily evaded which has reduced the data redundancy.

Hostel Management System Database Design is software developed for managing various activities in the hostel. For the past few years, the number of travellers around the world is increasing rapidly. Thereby the number of hotels is also increasing for the accommodation of travelers around the world.

1.2 Problem Statement:

There are a lot of drawbacks in keeping and maintaining a hostel. Especially with a manual system. Since most hostels are being run by only one hostel manager, the number of students in a room are sometimes not known by the officer. He must go room by room to ensure that a room is occupied or not. Sometimes people may be owing in the hostel, and they are saved on papers or huge notebooks, and sometimes receipts. If the books should go missing or stolen, one would never be able to know if a student is owing or not. Room allocation also becomes a problem as the officer might not know which rooms are available or not. And some hostels have a lot of rooms or have mare stores and it would be very tedious to go through all stores in search of a free room for an applicant. Also, the officer might not know the number of students in a room or know if a room is full or not.

Now a days there are many students chooses hostel, for safety and security purposes we need to maintain their details. Manual management will be difficult for the manager to maintain the hostel. There are students from a different department, and branches. The hostel management system is designed to manage all hostel activities like hostel admissions, fees, room, mess allotment, and hostel stores & generates related reports for smooth transactions. It is also used to manage monthly mess bill calculation, hostel staff payroll, student certificates.

Thus, there are a lot of repetitions that can be easily evaded which has reduced the data redundancy.

1.3 OBJECTIVE OF THE PROJECT:

The main objectives of this project are:

- To make it easier for data collection, storage and referencing reliable.
- To maintain the students as hostellers and waiting list students separately.
- To process allotment list.
- For easy accessing of the students list.
- Makes easier to access room status.
- To reduce data redundancy.
- To maintain fee status easily.

CHAPTER 2. DATABASE DESIGN

2.1 List of Attributes, entities and relationship:

1. student (s_id, sname, mobile,gender, mess, address, department, year)

2. hostel (hostel_id, hostel_name, no_of_rooms, no_of_students)

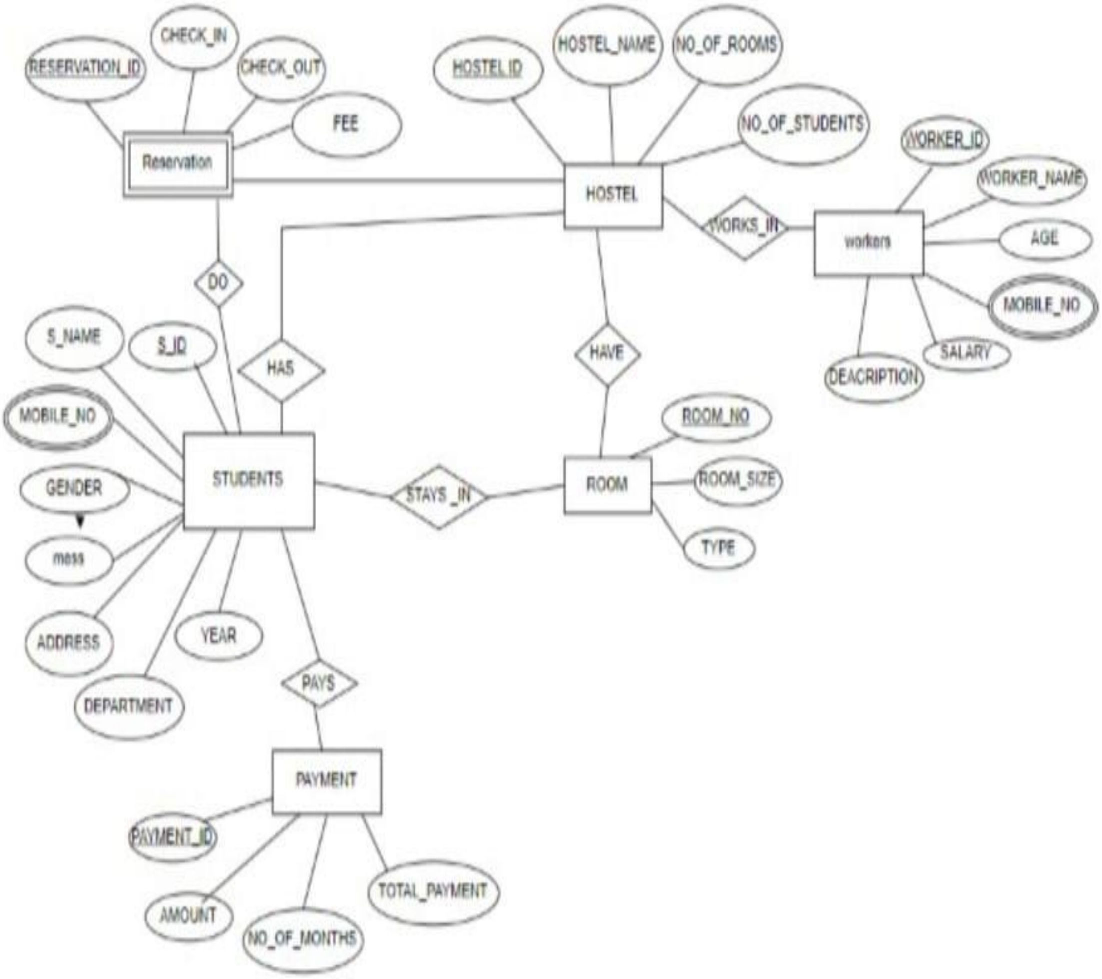
3. room (room_no, hostel_id, room type, room_size)

4. reservation (reservation_id, student_id, hostel_id, room_no, check_in_year, check_out_year, fee)

5. worker (worker_id, name, hostel_id, salary, gender, city, mobile, designation)

6. payment (payment_id, student_id, amount, years,fee_due)

2.2 Er diagram:

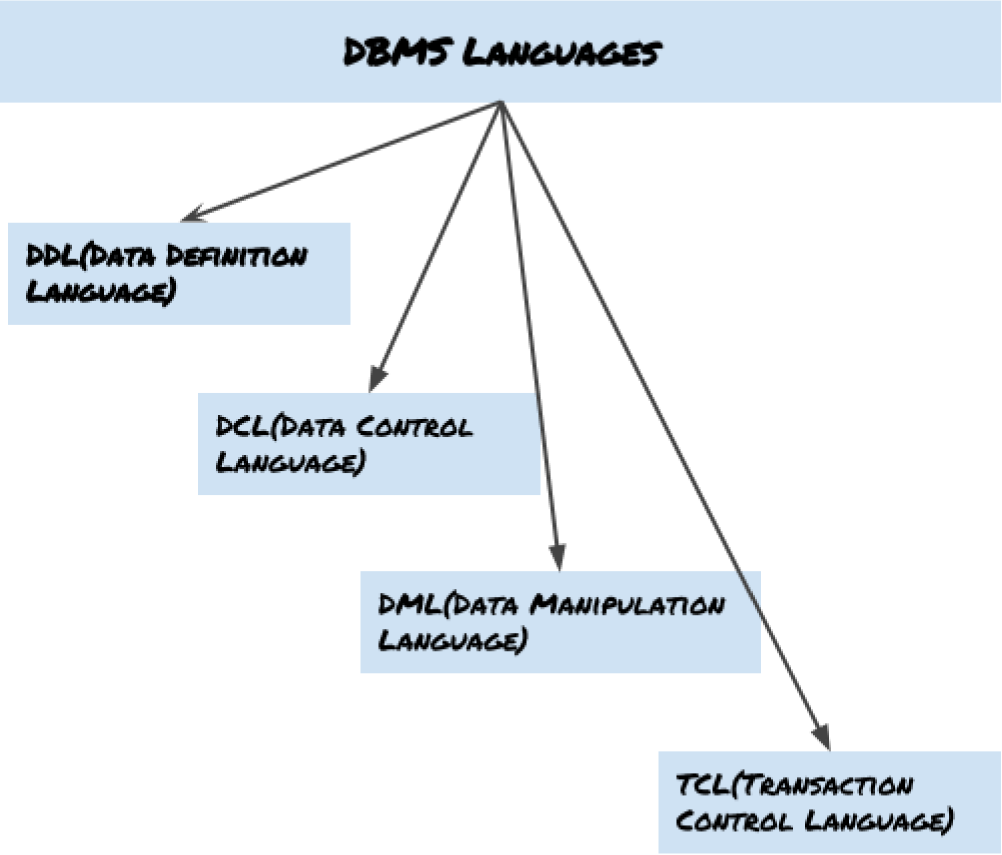


CHAPTER 3. RELATIONAL MODEL

3.1 Database languages:

Database languages are used to read, update and store data in a database. There are several such languages that can be used for this purpose; one of them is SQL (Structured Query Language).

Types of DBMS languages:


DBMS Languages
DDL(Data Definition Language)
DCL(Data Control Language)
DML(Data Manipulation Language)
TCL(Transaction Control Language)

Data Definition Language (DDL)

DDL is used for specifying the database schema. It is used for creating tables, schema, indexes, constraints etc. in database. Lets see the operations that we can perform on database using DDL:

- To create the database instance – **CREATE**

- To alter the structure of database – **ALTER**
- To drop database instances – **DROP**
- To delete tables in a database instance – **TRUNCATE**
- To rename database instances – **RENAME**
- To drop objects from database such as tables – **DROP**
- To Comment – **Comment**

All these commands either defines or update the database schema that's why they come under Data Definition language.

Data Manipulation Language (DML)

DML is used for accessing and manipulating data in a database. The following operations on database comes under DML:

- To read records from table(s) – **SELECT**
- To insert record(s) into the table(s) – **INSERT**
- Update the data in table(s) – **UPDATE**
- Delete all the records from the table – **DELETE**

Data Control language (DCL)

DCL is used for granting and revoking user access on a database –

- To grant access to user – GRANT
- To revoke access from user – REVOKE

In practical data definition language, data manipulation language and data control languages are not separate language, rather they are the parts of a single database language such as SQL.

Transaction Control Language(TCL)

The changes in the database that we made using DML commands are either performed or rollbacked using TCL.

- To persist the changes made by DML commands in database – COMMIT
- rollback the changes made to the database – ROLLBACK

3.2 Table Description:

Following are the tables along with constraints used in Bank Management System Database.

1. **Student:**

This table consists of details about the various students in the hostel. The information stored in this table includes the department, ID, name, address, location, mobile number, and the year he/she studies.

Constraints:

Student id will be unique for all students, and it is a primary key.

2. **Hostel:**

This table consists of details about the various hostels in the institution. The information stored in this table includes Hostel name, Hostel ID, number of rooms, and total number of students available in that Hostel.

Constraints:

Hostel Id will be unique for all hostels, and it is a primary key.

3. **Room:**

This table consists of details about the various rooms in hostels. The information stored in this table includes room number, Hostel ID, room size, and type of room (ac/non-ac) of that Hostel.

Constraints:

Room number will be unique for all rooms and corresponding hostel ID's should exist in the room table.

4. **Reservations:**

This table consists of details about the Reservations reserved by the students. The information stored in this table includes the student ID, the room he is staying, hostel, the year he check-in and check-out and the fee of that room.

Constraints:

On this table student ID, the hostel ID he is staying, and the room number should be mentioned on table as for the reservations they must and should.

5. **Workers:**

This table consists of details about the various workers working in the hostel. The information stored in this table includes the worker's ID, name, address, mobile number, age, salary, and designation.

Constraints:

Worker id will be unique for all workers, it will be the primary key of that table.

6. **Payment:**

This table contains the details of the amount paid by the students in the hostel. It stores the amount paid and the total amount of the student so that we can calculate the due.

Constraints:

Student ID should exist in this table, the student will pay the amount, so the student ID is required in the table.

3.3 Relational Database Schema for Case Study:

The relational database schema for bank Management database is as follow:

1. student (s_id, sname, mobile,gender, mess, address, department, year)

Student table:

SNO	Attributes	Datatype	constraint
1	S_id	int	Primary key
2	Sname	Varchar(20)	Not null
3	Mobile	Varchar(10)	Not null
4	Gender	Char(1)	Not null check(gender='f' or gender='m')
5	Mess	Varchar(10)	Not null check(mess='veg' or mess='non-veg')
6	Address	Varchar(10)	Not null
7	department	Varchar(10)	Not null

2. hostel (hostel_id, hostel_name, no_of_rooms, no_of_students)

Hostel table:

Sno	Attributes	Datatype	Constraint
1	Hostel_id	int	Primary key
2	Hostel_name	varchar(20)	Not null
3	No_of_rooms	int	Not null
4	No_of_students	Int	Not null

3. room (room_no, hostel_id, room type, room_size)

Room table:

S.No	**Attribute**	**Datatype**	**Constraint**
1	Room_no	int	Primary key
2	Hostel_id	int	Foreign key
3	Type	varchar(10)	Not null
4	Room_size	int	Not null

4. reservation (reservation_id, student_id, hostel_id, room_no, check_in_year, check_out_year, fee)

Accounts table:

Sno	**Attributes**	**Datatype**	**Constraints**
1	Reservation_id	Int	Primary key
2	Student_id	Int	Foreign key
3	Hostel_id	Int	Foreign key
4	Room_no	Int	Foreign key
5	Check_in_year	Int	Not null
6	Check_out_year	Int	
7	fee	Int	Not null

5. worker (worker_id, name, hostel_id, salary, gender, city, mobile, designation)

Worker table:

Sno	Attributes	Datatype	Constraint
1	Worker_id	Int	Primary key
2	Name	Varchar(20)	Not null
3	Hostel_id	Int	Foreign key
4	Salary	Int	Not null
5	Gender	Char(1)	Not null check(gender='f' or gender='m')
6	City	Varchar(20)	Not null
7	Mobile	Varchar(10)	Not null
8	Designation	Varchar(20)	Not null

6. payment (payment_id, student_id, amount, year, fee_due)

Payment table:

Sno	Attributes	Datatype	Constraint
1	Payment_id	Int	Primary key
2	Student_id	Int	Foreign key
3	Amount	Int	Not null
4	Year	Int	Not null
5	Fee due	Int	Not null

3.4 Relational Queries

Database creation

```
create database hostel
use hostel
```

Student Table:

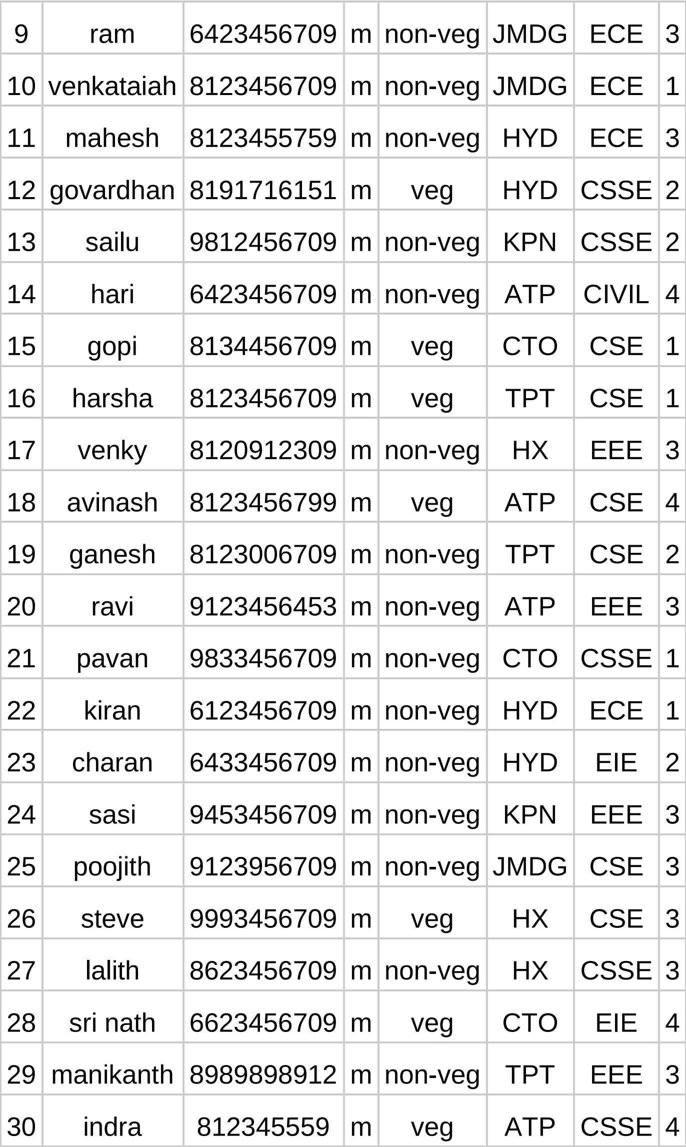
```
create table student(
        s_id    int                 primary key,
        sname  varchar(20)       not null,
        mobile varchar(10)       not null,
        gender char(1)           not null        check(gender='f' or gender='m'),
        mess    varchar(10)       not null    check(mess='veg' or mess='non-veg'),
        addressvarchar(10)      not null,
        department      varchar(10)      not null,
        year    int
);

insert into student values(01,'sumanth',9177252309,'m','veg','NLR','CSSE',2);
insert into student values(02,'sathish',9573703266,'m','non-veg','ATP','CSSE',2);
insert into student values(03,'ashiq',9123456786,'m','veg','ATP','CSSE',2);
insert into student values(04,'karthik',8123456709,'m','non-veg','HX','EEE',3);
insert into student values(05,'mahesh',8317555382,'m','non-veg','ATP','EEE',3);
insert into student values(06,'venkat',1234569709,'m','non-veg','KRNT','CSE',2);
insert into student values(07,'hemanth',8123686709,'m','non-veg','HYD','CSE',3);
insert into student values(08,'vishnu',8123456799,'m','non-veg','KPN','ECE',3);
insert into student values(09,'ram',6423456709,'m','non-veg','JMDG','ECE',3);
insert into student values(10,'venkataiah',8123456709,'m','non-veg','JMDG','ECE',1);
insert into student values(11,'mahesh',8123455759,'m','non-veg','HYD','ECE',3);
insert into student values(12,'govardhan',8191716151,'m','veg','HYD','CSSE',2);
insert into student values(13,'sailu',9812456709,'m','non-veg','KPN','CSSE',2);
insert into student values(14,'hari',6423456709,'m','non-veg','ATP','CIVIL',4);
insert into student values(15,'gopi',8134456709,'m','veg','CTO','CSE',1);
insert into student values(16,'harsha',8123456709,'m','veg','TPT','CSE',1);
insert into student values(17,'venky',8120912309,'m','non-veg','HX','EEE',3);
insert into student values(18,'avinash',8123456799,'m','veg','ATP','CSE',4);
insert into student values(19,'ganesh',8123006709,'m','non-veg','TPT','CSE',2);
insert into student values(20,'ravi',9123456453,'m','non-veg','ATP','EEE',3);
insert into student values(21,'pavan',9833456709,'m','non-veg','CTO','CSSE',1);
insert into student values(22,'kiran',6123456709,'m','non-veg','HYD','ECE',1);
insert into student values(23,'charan',6433456709,'m','non-veg','HYD','EIE',2);
insert into student values(24,'sasi',9453456709,'m','non-veg','KPN','EEE',3);
insert into student values(25,'poojith',9123956709,'m','non-veg','JMDG','CSE',3);
insert into student values(26,'steve',9993456709,'m','veg','HX','CSE',3);
insert into student values(27,'lalith',8623456709,'m','non-veg','HX','CSSE',3);
insert into student values(28,'sri nath',6623456709,'m','veg','CTO','EIE',4);
insert into student values(29,'manikanth',8989898912,'m','non-veg','TPT','EEE',3);
insert into student values(30,'indra',812345559,'m','veg','ATP','CSSE',4);

select * from student;
```

Table:

1	sumanth	9177252309	m	veg	NLR	CSSE	2
2	sathish	9573703266	m	non-veg	ATP	CSSE	2
3	ashiq	9123456786	m	veg	ATP	CSSE	2
4	karthik	8123456709	m	non-veg	HX	EEE	3
5	mahesh	8317555382	m	non-veg	ATP	EEE	3
6	venkat	1234569709	m	non-veg	KRNT	CSE	2
7	hemanth	8123686709	m	non-veg	HYD	CSE	3
8	vishnu	8123456799	m	non-veg	KPN	ECE	3



9	ram	6423456709	m	non-veg	JMDG	ECE	3
10	venkataiah	8123456709	m	non-veg	JMDG	ECE	1
11	mahesh	8123455759	m	non-veg	HYD	ECE	3
12	govardhan	8191716151	m	veg	HYD	CSSE	2
13	sailu	9812456709	m	non-veg	KPN	CSSE	2
14	hari	6423456709	m	non-veg	ATP	CIVIL	4
15	gopi	8134456709	m	veg	CTO	CSE	1
16	harsha	8123456709	m	veg	TPT	CSE	1
17	venky	8120912309	m	non-veg	HX	EEE	3
18	avinash	8123456799	m	veg	ATP	CSE	4
19	ganesh	8123006709	m	non-veg	TPT	CSE	2
20	ravi	9123456453	m	non-veg	ATP	EEE	3
21	pavan	9833456709	m	non-veg	CTO	CSSE	1
22	kiran	6123456709	m	non-veg	HYD	ECE	1
23	charan	6433456709	m	non-veg	HYD	EIE	2
24	sasi	9453456709	m	non-veg	KPN	EEE	3
25	poojith	9123956709	m	non-veg	JMDG	CSE	3
26	steve	9993456709	m	veg	HX	CSE	3
27	lalith	8623456709	m	non-veg	HX	CSSE	3
28	sri nath	6623456709	m	veg	CTO	EIE	4
29	manikanth	8989898912	m	non-veg	TPT	EEE	3
30	indra	812345559	m	veg	ATP	CSSE	4

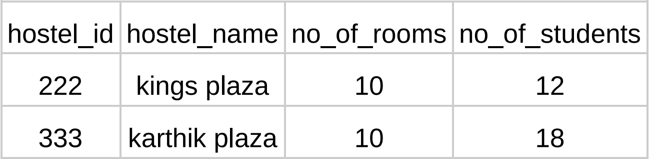
Hostel Table:

```
create table hostel(
            hostel_id         int         primary key,
            hostel_name    varchar(20)      not null,
            no_of_rooms   int         not null,
            no_of_students            int         not null
);

insert into hostel values(222,'kings plaza',10,12);
insert into hostel values(333,'karthik plaza',10,18);

select * from hostel;
```

Table:



hostel_id	hostel_name	no_of_rooms	no_of_students
222	kings plaza	10	12
333	karthik plaza	10	18

Room Table:

```
create table  room(
        room_no int primary key,
        hostel_id       int foreign key references hostel(hostel_id),
        type varchar(10) check(type='ac' or type='non-ac'),
        room_size       int
);

insert into room values(211,222,'ac',2);
insert into room values(212,222,'non-ac',2);
insert into room values(213,222,'non-ac',2);
insert into room values(214,222,'non-ac',2);
insert into room values(215,222,'ac',2);
insert into room values(221,222,'ac',3);
insert into room values(222,222,'non-ac',3);
insert into room values(223,222,'non-ac',3);
insert into room values(224,222,'ac',3);
insert into room values(225,222,'ac',3);
insert into room values(311,333,'non-ac',2);
insert into room values(312,333,'ac',2);
insert into room values(313,333,'non-ac',2);
insert into room values(314,333,'ac',2);
insert into room values(315,333,'non-ac',2);
insert into room values(321,333,'ac',3);
insert into room values(322,333,'non-ac',3);
insert into room values(323,333,'ac',3);
insert into room values(324,333,'non-ac',3);
insert into room values(325,333,'ac',3);

select * from room;
```

Table:

room_no	hostel_id	type	room_size
211	222	ac	2
212	222	non-ac	2
213	222	non-ac	2
214	222	non-ac	2
215	222	ac	2
221	222	ac	3
222	222	non-ac	3
223	222	non-ac	3
224	222	ac	3
225	222	ac	3
311	333	non-ac	2
312	333	ac	2
313	333	non-ac	2
314	333	ac	2
315	333	non-ac	2
321	333	ac	3
322	333	non-ac	3
323	333	ac	3
324	333	non-ac	3
325	333	ac	3

Reservation Table:

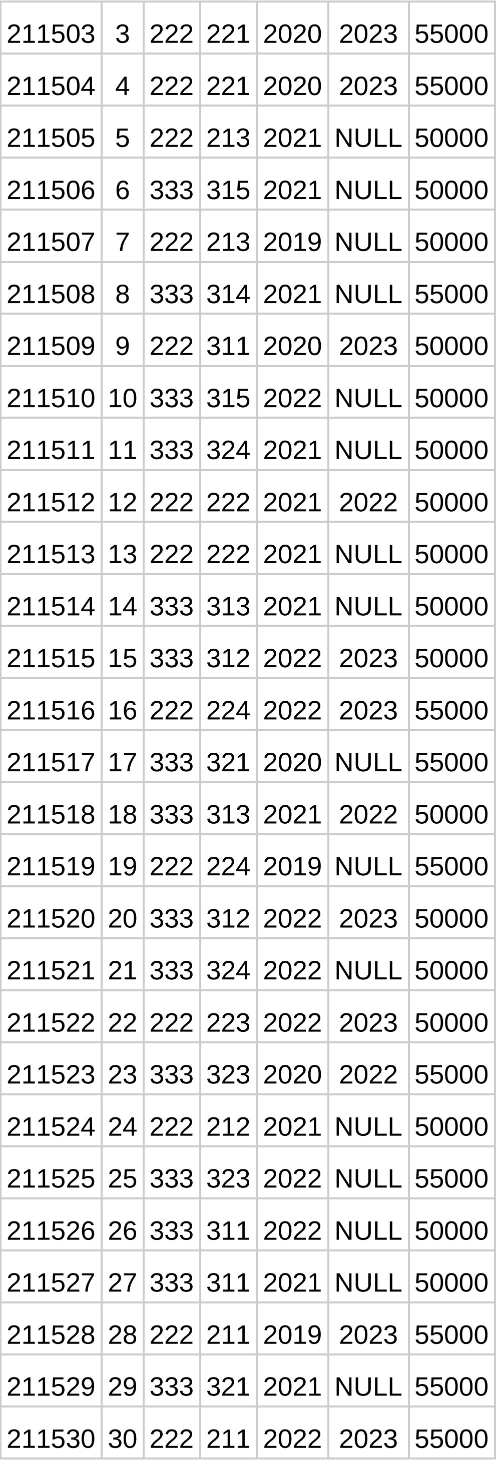
```
create table reservation(
        reservation_id int      primary key,
        student_id      int foreign key references student(s_id),
        hostel_id       int foreign key references hostel(hostel_id),
        room_no int foreign key references room(room_no),
        check_in_year int not null,
        check_out_year          int,
        fee     int     not null,
);
```

```
insert into reservation values(211515,15,333,312,2022,2023,50000);
insert into reservation values(211501,01,333,311,2021,2022,50000);
insert into reservation values(211502,02,333,311,2021,null,50000);
insert into reservation values(211503,03,222,221,2020,2023,55000);
insert into reservation values(211504,04,222,221,2020,2023,55000);
insert into reservation values(211505,05,222,213,2021,null,50000);
insert into reservation values(211506,06,333,315,2021,null,50000);
insert into reservation values(211507,07,222,213,2019,null,50000);
insert into reservation values(211508,08,333,314,2021,null,55000);
insert into reservation values(211509,09,222,311,2020,2023,50000);
insert into reservation values(211510,10,333,315,2022,null,50000);
insert into reservation values(211511,11,333,324,2021,null,50000);
insert into reservation values(211512,12,222,222,2021,2022,50000);
insert into reservation values(211513,13,222,222,2021,null,50000);
insert into reservation values(211514,14,333,313,2021,null,50000);
insert into reservation values(211516,16,222,224,2022,2023,55000);
insert into reservation values(211517,17,333,321,2020,null,55000);
insert into reservation values(211518,18,333,313,2021,2022,50000);
insert into reservation values(211519,19,222,224,2019,null,55000);
insert into reservation values(211520,20,333,312,2022,2023,50000);
insert into reservation values(211521,21,333,324,2022,null,50000);
insert into reservation values(211522,22,222,223,2022,2023,50000);
insert into reservation values(211523,23,333,323,2020,2022,55000);
insert into reservation values(211524,24,222,212,2021,null,50000);
insert into reservation values(211525,25,333,323,2022,null,55000);
insert into reservation values(211526,26,333,311,2022,null,50000);
insert into reservation values(211527,27,333,311,2021,null,50000);
insert into reservation values(211528,28,222,211,2019,2023,55000);
insert into reservation values(211529,29,333,321,2021,null,55000);
insert into reservation values(211530,30,222,211,2022,2023,55000);

select * from reservation;
```

Table:

211501	1	333	311	2021	2022	50000
211502	2	333	311	2021	NULL	50000



211503	3	222	221	2020	2023	55000
211504	4	222	221	2020	2023	55000
211505	5	222	213	2021	NULL	50000
211506	6	333	315	2021	NULL	50000
211507	7	222	213	2019	NULL	50000
211508	8	333	314	2021	NULL	55000
211509	9	222	311	2020	2023	50000
211510	10	333	315	2022	NULL	50000
211511	11	333	324	2021	NULL	50000
211512	12	222	222	2021	2022	50000
211513	13	222	222	2021	NULL	50000
211514	14	333	313	2021	NULL	50000
211515	15	333	312	2022	2023	50000
211516	16	222	224	2022	2023	55000
211517	17	333	321	2020	NULL	55000
211518	18	333	313	2021	2022	50000
211519	19	222	224	2019	NULL	55000
211520	20	333	312	2022	2023	50000
211521	21	333	324	2022	NULL	50000
211522	22	222	223	2022	2023	50000
211523	23	333	323	2020	2022	55000
211524	24	222	212	2021	NULL	50000
211525	25	333	323	2022	NULL	55000
211526	26	333	311	2022	NULL	50000
211527	27	333	311	2021	NULL	50000
211528	28	222	211	2019	2023	55000
211529	29	333	321	2021	NULL	55000
211530	30	222	211	2022	2023	55000

Worker Table:

```
create table worker(
        worker_id       int             primary key,
        name    varchar(20)     not null,
        hostel_id       int foreign key references hostel(hostel_id),
        salary          int             not null,
        gender char(1)          not null        check(gender='f' or gender='m'),
        age     int not null,
        city    varchar(20),
        mobile varchar(10),
        designation     varchar(20)
);


insert into worker values(201,'suresh','222',100000,'m',39,'TPT',9000380380,'manager');
insert into worker values(202,'madav','222',50000,'m',30,'HX',9100760705,'chef');
insert into worker values(203,'sunil','222',50000,'m',28,'JMDG',9848766552,'chef');
insert into worker values(204,'prasd','222',30000,'m',40,'CTO',6301213212,'asst chef');
insert into worker values(205,'pavani','222',20000,'f',45,'TPT',9080706050,'receptionist');
insert into worker values(206,'manisha','222',15000,'f',48,'TPT',6123459870,'cleaner');
insert into worker values(207,'sunitha','222',15000,'f',39,'TPT',9000380380,'cleaner');
insert into worker values(208,'kumari','222',5000,'F',45,'TPT',8565785468,'laundry_worker');
insert into worker values(209,'mallesh','222',5500,'m',43,'TPT',9000380380,'laundry_worker');
insert into worker values(301,'anil','333',130000,'m',50,'TPT',9018262525,'manager');
insert into worker values(302,'sumanth','333',55000,'m',46,'KPN',6574382901,'chef');
insert into worker values(303,'kiranmayi','333',45000,'f',32,'HX',5464758742,'chef');
insert into worker values(304,'sravan','333',25000,'m',27,'NLR',8473029486,'asst chef');
insert into worker values(305,'kumari','333',20000,'f',31,'TPT',8182838485,'receptionist');
insert into worker values(306,'lalitha','333',15000,'f',42,'TPT',9192939495,'cleaner');
insert into worker values(307,'latha','333',15000,'f',51,'TPT',9637649393,'cleaner');
insert into worker values(308,'aparao','222',6000,'m',37,'CTO',6093838789,'laundry_worker');
insert into worker values(309,'anusha','222',6000,'f',45,'TPT',6789789456,'laundry_worker');


select * from worker;
```

Table:

worker_id	name	hostel_id	salary	gender	age	city	mobile	designation
201	suresh	222	100000	m	39	TPT	9000380380	manager
202	madav	222	50000	m	30	HX	9100760705	chef

203	sunil	222	50000	m	28	JMDG	9848766552	chef
204	prasd	222	30000	m	40	CTO	6301213212	asst chef
205	pavani	222	20000	f	45	TPT	9080706050	receptionist
206	manisha	222	15000	f	48	TPT	6123459870	cleaner
207	sunitha	222	15000	f	39	TPT	9000380380	cleaner
208	kumari	222	5000	F	45	TPT	8565785468	laundry_worker
209	mallesh	222	5500	m	43	TPT	9000380380	laundry_worker
301	anil	333	130000	m	50	TPT	9018262525	manager
302	sumanth	333	55000	m	46	KPN	6574382901	chef
303	kiranmayi	333	45000	f	32	HX	5464758742	chef
304	sravan	333	25000	m	27	NLR	8473029486	asst chef
305	kumari	333	20000	f	31	TPT	8182838485	receptionist
306	lalitha	333	15000	f	42	TPT	9192939495	cleaner
307	latha	333	15000	f	51	TPT	9637649393	cleaner
308	aparao	222	6000	m	37	CTO	6093838789	laundry_worker
309	anusha	222	6000	f	45	TPT	6789789456	laundry_worker

Payment Table:

```
create table payment(
        payment_id    int     primary key,
        student_id    int     foreign key references student(s_id),
        amount        int     not null,
        years         int     not null,
        fee_due       int     not null,
);

insert into payment values(76859654,01,10000,2022,40000);
insert into payment values(67564739,02,10000,2022,40000);
insert into payment values(04846435,03,5000,2022,50000);
insert into payment values(53643466,04,55000,2021,0);
insert into payment values(84428236,04,25000,2022,35000);
insert into payment values(48468846,05,10000,2022,40000);
insert into payment values(49875977,06,10000,2022,40000);
insert into payment values(53123446,07,55000,2020,0);
insert into payment values(59873466,07,55000,2021,0);
insert into payment values(94857578,07,10000,2022,40000);
insert into payment values(20384865,08,15000,2022,40000);
insert into payment values(53993466,09,50000,2021,0);
insert into payment values(00085757,09,10000,2022,40000);
insert into payment values(34758699,10,5000,2023,45000);
insert into payment values(55586864,11,10000,2022,40000);
insert into payment values(45757755,12,10000,2022,40000);
insert into payment values(01988477,13,10000,2022,40000);
insert into payment values(25355464,14,10000,2022,40000);
insert into payment values(85857585,15,10000,2022,40000);
insert into payment values(85758586,16,15000,2022,40000);
insert into payment values(74646444,17,25000,2022,30000);
insert into payment values(44485753,18,10000,2022,40000);
insert into payment values(21152319,19,35000,2022,20000);
insert into payment values(95957646,20,20000,2022,30000);
insert into payment values(85845275,21,10000,2022,40000);
insert into payment values(09876773,22,25000,2022,25000);
insert into payment values(23443344,23,45000,2022,10000);
insert into payment values(98788788,24,10000,2022,40000);
insert into payment values(76765666,25,35000,2022,20000);
insert into payment values(92884975,26,10000,2022,40000);
insert into payment values(54308597,27,20000,2022,30000);
insert into payment values(98546527,28,45000,2022,10000);
insert into payment values(23564525,29,25000,2022,30000);
insert into payment values(43515455,30,45000,2022,10000);
```

```
select * from reservation;
```

Table:

211501	1	333	311	2021	2022	50000
211502	2	333	311	2021	NULL	50000
211503	3	222	221	2020	2023	55000
211504	4	222	221	2020	2023	55000
211505	5	222	213	2021	NULL	50000
211506	6	333	315	2021	NULL	50000
211507	7	222	213	2019	NULL	50000

211508	8	333	314	2021	NULL	55000
211509	9	222	311	2020	2023	50000
211510	10	333	315	2022	NULL	50000
211511	11	333	324	2021	NULL	50000
211512	12	222	222	2021	2022	50000
211513	13	222	222	2021	NULL	50000
211514	14	333	313	2021	NULL	50000
211515	15	333	312	2022	2023	50000
211516	16	222	224	2022	2023	55000
211517	17	333	321	2020	NULL	55000
211518	18	333	313	2021	2022	50000
211519	19	222	224	2019	NULL	55000
211520	20	333	312	2022	2023	50000
211521	21	333	324	2022	NULL	50000
211522	22	222	223	2022	2023	50000
211523	23	333	323	2020	2022	55000
211524	24	222	212	2021	NULL	50000
211525	25	333	323	2022	NULL	55000
211526	26	333	311	2022	NULL	50000
211527	27	333	311	2021	NULL	50000
211528	28	222	211	2019	2023	55000
211529	29	333	321	2021	NULL	55000
211530	30	222	211	2022	2023	55000

--**QUERIES**

--1 select the list of students

select * from student;

Output:

s_id	sname	mobile	gender	mess	address	department	year
1	sumanth	9177252309	m	veg	NLR	CSSE	2
2	sathish	9573703266	m	non-veg	ATP	CSSE	2
3	ashiq	9123456786	m	veg	ATP	CSSE	2
4	karthik	8123456709	m	non-veg	HX	EEE	3
5	mahesh	8317555382	m	non-veg	ATP	EEE	3

6	venkat	1234569709	m	non-veg	KRNT	CSE	2
7	hemanth	8123686709	m	non-veg	HYD	CSE	3
8	vishnu	8123456799	m	non-veg	KPN	ECE	3
9	ram	6423456709	m	non-veg	JMDG	ECE	3
10	venkataiah	8123456709	m	non-veg	JMDG	ECE	1
11	mahesh	8123455759	m	non-veg	HYD	ECE	3
12	govardhan	8191716151	m	veg	HYD	CSSE	2
13	sailu	9812456709	m	non-veg	KPN	CSSE	2
14	hari	6423456709	m	non-veg	ATP	CIVIL	4
15	gopi	8134456709	m	veg	CTO	CSE	1
16	harsha	8123456709	m	veg	TPT	CSE	1
17	venky	8120912309	m	non-veg	HX	EEE	3
18	avinash	8123456799	m	veg	ATP	CSE	4
19	ganesh	8123006709	m	non-veg	TPT	CSE	2
20	ravi	9123456453	m	non-veg	ATP	EEE	3
21	pavan	9833456709	m	non-veg	CTO	CSSE	1
22	kiran	6123456709	m	non-veg	HYD	ECE	1
23	charan	6433456709	m	non-veg	HYD	EIE	2
24	sasi	9453456709	m	non-veg	KPN	EEE	3
25	poojith	9123956709	m	non-veg	JMDG	CSE	3
26	steve	9993456709	m	veg	HX	CSE	3
27	lalith	8623456709	m	non-veg	HX	CSSE	3
28	sri nath	6623456709	m	veg	CTO	EIE	4
29	manikanth	8989898912	m	non-veg	TPT	EEE	3
30	indra	812345559	m	veg	ATP	CSSE	4

--2 list the students of 4th year

select * from student where year=4;

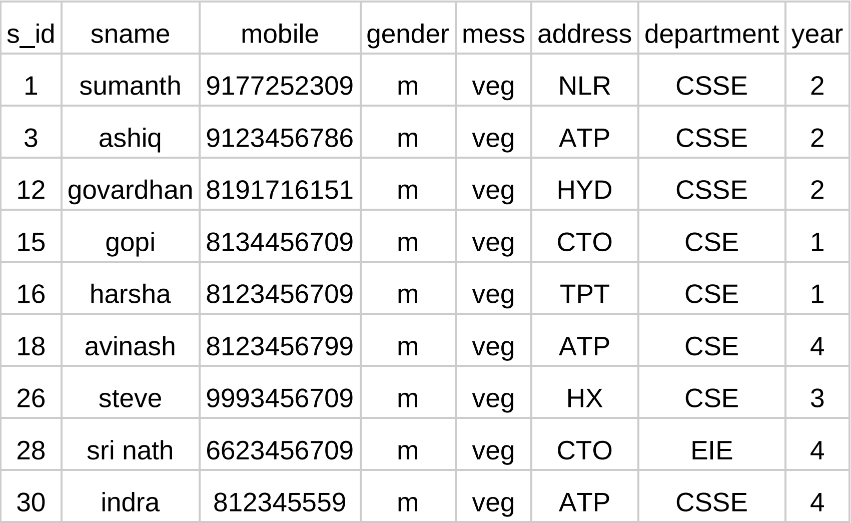
Output:

s_id	sname	mobile	gender	mess	address	department	year
14	hari	6423456709	m	non-veg	ATP	CIVIL	4
18	avinash	8123456799	m	veg	ATP	CSE	4
28	sri nath	6623456709	m	veg	CTO	EIE	4
30	indra	812345559	m	veg	ATP	CSSE	4

--3 display students who prefer veg

select * from student where mess='veg';

Output:

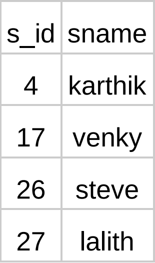


s_id	sname	mobile	gender	mess	address	department	year
1	sumanth	9177252309	m	veg	NLR	CSSE	2
3	ashiq	9123456786	m	veg	ATP	CSSE	2
12	govardhan	8191716151	m	veg	HYD	CSSE	2
15	gopi	8134456709	m	veg	CTO	CSE	1
16	harsha	8123456709	m	veg	TPT	CSE	1
18	avinash	8123456799	m	veg	ATP	CSE	4
26	steve	9993456709	m	veg	HX	CSE	3
28	sri nath	6623456709	m	veg	CTO	EIE	4
30	indra	812345559	m	veg	ATP	CSSE	4

--4 list the id and name of the students from kadapa(HX)

select s_id,sname from student where address='HX';

Output:

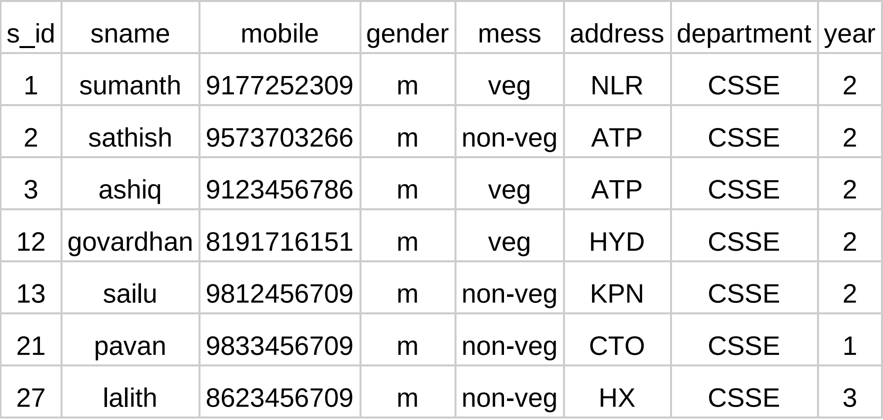


s_id	sname
4	karthik
17	venky
26	steve
27	lalith

--5 list the students of department csse

select * from student where department='CSSE';

Output:



s_id	sname	mobile	gender	mess	address	department	year
1	sumanth	9177252309	m	veg	NLR	CSSE	2
2	sathish	9573703266	m	non-veg	ATP	CSSE	2
3	ashiq	9123456786	m	veg	ATP	CSSE	2
12	govardhan	8191716151	m	veg	HYD	CSSE	2
13	sailu	9812456709	m	non-veg	KPN	CSSE	2
21	pavan	9833456709	m	non-veg	CTO	CSSE	1
27	lalith	8623456709	m	non-veg	HX	CSSE	3

30	indra	812345559	m	veg	ATP	CSSE	4

--6 list the data of the hostels

select * from hostel;

Output:

hostel_id	hostel_name	no_of_rooms	no_of_students
222	kings plaza	10	12
333	karthik plaza	10	18

--7 find number of students stays in kings plaza

select no_of_students from hostel where hostel_name='kings plaza';

Output

no_of_students
12

--8 find id and name of the students who are staying in hostel having id=333

select s.s_id,s.sname from student s,reservation r where s.s_id=r.student_id and r.hostel_id=333;

Output:

s_id	sname
1	sumanth
2	sathish
6	venkat
8	vishnu
10	venkataiah
11	mahesh
14	hari
15	gopi
17	venky
18	avinash
20	ravi
21	pavan
23	charan
25	poojith
26	steve

27	lalith
29	manikanth

--9 list the data of rooms

select * from room;

Output:

room_no	hostel_id	type	room_size
211	222	ac	2
212	222	non-ac	2
213	222	non-ac	2
214	222	non-ac	2
215	222	ac	2
221	222	ac	3
222	222	non-ac	3
223	222	non-ac	3
224	222	ac	3
225	222	ac	3
311	333	non-ac	2
312	333	ac	2
313	333	non-ac	2
314	333	ac	2
315	333	non-ac	2
321	333	ac	3
322	333	non-ac	3
323	333	ac	3
324	333	non-ac	3
325	333	ac	3

--10 find the names of student who are staying in ac rooms

select s.sname from student s,reservation r,room ro where s.s_id=r.student_id and r.room_no=ro.room_no and ro.type='ac';

Output:

sname
ashiq
karthik

vishnu
gopi
harsha
venky
ganesh
ravi
charan
poojith
sri nath
manikanth
indra

--11 Some students have not given check_out_year, list the students

select s.s_id,s.sname from student s,reservation r where s.s_id=r.student_id and r.check_out_year is null;

Output:

s_id	sname
2	sathish
5	mahesh
6	venkat
7	hemanth
8	vishnu
10	venkataiah
11	mahesh
13	sailu
14	hari
17	venky
19	ganesh
21	pavan
24	sasi
25	poojith
26	steve
27	lalith
29	manikanth

--12 list the students who reserved room this year select * from student where s_id in(select s.s_id from student s, reservation r where s.s_id=r.student_id and r.check_in_year=2022);

Output:

s_id	sname	mobile	gender	mess	address	department	year
10	venkataiah	8123456709	m	non-veg	JMDG	ECE	1
15	gopi	8134456709	m	veg	CTO	CSE	1
16	harsha	8123456709	m	veg	TPT	CSE	1
20	ravi	9123456453	m	non-veg	ATP	EEE	3
21	pavan	9833456709	m	non-veg	CTO	CSSE	1
22	kiran	6123456709	m	non-veg	HYD	ECE	1
25	poojith	9123956709	m	non-veg	JMDG	CSE	3
26	steve	9993456709	m	veg	HX	CSE	3
30	indra	812345559	m	veg	ATP	CSSE	4

--13 list the details of the students who are in room no 221

select * from student where s_id in(select s.s_id from student s, reservation r where s.s_id=r.student_id and r.room_no=221);

Output:

s_id	sname	mobile	gender	mess	address	department	year
3	ashiq	9123456786	m	veg	ATP	CSSE	2
4	karthik	8123456709	m	non-veg	HX	EEE	3

--14 list the students whose name starts with s

select * from student where sname like 's%';

Output:

s_id	sname	mobile	gender	mess	address	department	year
1	sumanth	9177252309	m	veg	NLR	CSSE	2
2	sathish	9573703266	m	non-veg	ATP	CSSE	2
13	sailu	9812456709	m	non-veg	KPN	CSSE	2
24	sasi	9453456709	m	non-veg	KPN	EEE	3
26	steve	9993456709	m	veg	HX	CSE	3
28	sri nath	6623456709	m	veg	CTO	EIE	4

--15 list all the addresses from where students are staying

select distinct address from student;

Output:

address
ATP
CTO
HX
HYD
JMDG
KPN
KRNT
NLR
TPT

```
--16 find how many students in kings plaza
select count(r.student_id) as no_of_students from reservation r,room ro where
r.room_no=ro.room_no and ro.hostel_id=333;
```

Output:

no_of_students
18

```
--17 query the details of the workers in the hostel karthik plaza
select * from worker where hostel_id=333;
```

Output:

worker_id	name	hostel_id	salary	gender	age	city	mobile	designation
301	anil	333	130000	m	50	TPT	9018262525	manager
302	sumanth	333	55000	m	46	KPN	6574382901	chef
303	kiranmayi	333	45000	f	32	HX	5464758742	chef
304	sravan	333	25000	m	27	NLR	8473029486	asst chef
305	kumari	333	20000	f	31	TPT	8182838485	receptionist
306	lalitha	333	15000	f	42	TPT	9192939495	cleaner
307	latha	333	15000	f	51	TPT	9637649393	cleaner

```
--18 list the top 2 chefs according to salary
select top 2 * from worker where designation='chef' order by salary desc;
```

Output:

worker_id	name	hostel_id	salary	gender	age	city	mobile	designation

302	sumanth	333	55000	m	46	KPN	6574382901	chef
203	sunil	222	50000	m	28	JMDG	9848766552	chef

```
--19 Find total salary paid to the workers
select sum(salary) as amount from worker;
```

Output:

amount
607500

```
--20 list the names,salary,designation of the worker who are taking more than 50000
select name,salary,designation from worker where salary>50000;
```

Output:

name	salary	designation
suresh	100000	manager
anil	130000	manager
sumanth	55000	chef

```
--21 anil wants to change his mobile number to 9177252308 update the details
update worker set mobile=9177252308 where name='anil'
select name,mobile from worker where name='anil';
```

Output:

(1 row affected)

name	mobile
anil	9177252308

```
--22 Find the students who are in room no 211,212,213,214,215
select s.* from student s where s.s_id in(select r.student_id from reservation r where r.room_no in(211,212,213,214,215) );
```

Output:

s_id	sname	mobile	gender	mess	address	department	year
5	mahesh	8317555382	m	non-veg	ATP	EEE	3
7	hemanth	8123686709	m	non-veg	HYD	CSE	3

24	sasi	9453456709	m	non-veg	KPN	EEE	3
28	sri nath	6623456709	m	veg	CTO	EIE	4
30	indra	812345559	m	veg	ATP	CSSE	4

--23 find the salary of workers from each city

```
select sum(salary) as salary,city from worker group by city;
```

Output:

salary	city
36000	CTO
95000	HX
50000	JMDG
55000	KPN
25000	NLR
346500	TPT

--24 find the no of sudents who are paying this year

```
select count(*) as students_no from payment  where years=2022;
```

Output:

students_no
29

--25 find the due of the student whose id is 23

```
select fee_due from payment where student_id=23;
```

Output:

fee_due
10000

--26 find the total amount paid by the students this year

```
select sum(amount) as amount from payment where years=2022;
```

Output:

amount
530000

--27 list the students who paid less than 25000 select s.*,p.amount,p.fee_due from student s, payment p where p.student_id=s.s_id and p.amount<25000;

Output:

s_id	sname	mobile	gender	mess	address	department	year	amount	fee_due
9	ram	6423456709	m	non-veg	JMDG	ECE	3	10000	40000
13	sailu	9812456709	m	non-veg	KPN	CSSE	2	10000	40000
3	ashiq	9123456786	m	veg	ATP	CSSE	2	5000	50000
8	vishnu	8123456799	m	non-veg	KPN	ECE	3	15000	40000
14	hari	6423456709	m	non-veg	ATP	CIVIL	4	10000	40000
10	venkataiah	8123456709	m	non-veg	JMDG	ECE	1	5000	45000
18	avinash	8123456799	m	veg	ATP	CSE	4	10000	40000
12	govardhan	8191716151	m	veg	HYD	CSSE	2	10000	40000
5	mahesh	8317555382	m	non-veg	ATP	EEE	3	10000	40000
6	venkat	1234569709	m	non-veg	KRNT	CSE	2	10000	40000
27	lalith	8623456709	m	non-veg	HX	CSSE	3	20000	30000
11	mahesh	8123455759	m	non-veg	HYD	ECE	3	10000	40000
2	sathish	9573703266	m	non-veg	ATP	CSSE	2	10000	40000
1	sumanth	9177252309	m	veg	NLR	CSSE	2	10000	40000
16	harsha	8123456709	m	veg	TPT	CSE	1	15000	40000
21	pavan	9833456709	m	non-veg	CTO	CSSE	1	10000	40000
15	gopi	8134456709	m	veg	CTO	CSE	1	10000	40000
26	steve	9993456709	m	veg	HX	CSE	3	10000	40000
7	hemanth	8123686709	m	non-veg	HYD	CSE	3	10000	40000
20	ravi	9123456453	m	non-veg	ATP	EEE	3	20000	30000
24	sasi	9453456709	m	non-veg	KPN	EEE	3	10000	40000

--28 some students have not given check_out_year, update its value for students who are not given by one year of the check in year

update reservation set check_out_year=check_in_year+1 where check_out_year is null;

select * from reservation;

Output:

(17 rows affected)

reservation_id	student_id	hostel_id	room_no	check_in_year	check_out_year	fee
211501	1	333	311	2021	2022	50000
211502	2	333	311	2021	2022	50000
211503	3	222	221	2020	2023	55000

211504	4	222	221	2020	2023	55000
211505	5	222	213	2021	2022	50000
211506	6	333	315	2021	2022	50000
211507	7	222	213	2019	2020	50000
211508	8	333	314	2021	2022	55000
211509	9	222	311	2020	2023	50000
211510	10	333	315	2022	2023	50000
211511	11	333	324	2021	2022	50000
211512	12	222	222	2021	2022	50000
211513	13	222	222	2021	2022	50000
211514	14	333	313	2021	2022	50000
211515	15	333	312	2022	2023	50000
211516	16	222	224	2022	2023	55000
211517	17	333	321	2020	2021	55000
211518	18	333	313	2021	2022	50000
211519	19	222	224	2019	2020	55000
211520	20	333	312	2022	2023	50000
211521	21	333	324	2022	2023	50000
211522	22	222	223	2022	2023	50000
211523	23	333	323	2020	2022	55000
211524	24	222	212	2021	2022	50000
211525	25	333	323	2022	2023	55000
211526	26	333	311	2022	2023	50000
211527	27	333	311	2021	2022	50000
211528	28	222	211	2019	2023	55000
211529	29	333	321	2021	2022	55000
211530	30	222	211	2022	2023	55000

--29 find top three students who more fees this year

select top 3 s.*,p.amount,p.fee_due from student s, payment p where p.student_id=s.s_id and p.years=2022 order by p.amount desc;

Output:

s_id	sname	mobile	gender	mess	address	department	year	amount	fee_due
30	indra	812345559	m	veg	ATP	CSSE	4	45000	10000
23	charan	6433456709	m	non-veg	HYD	EIE	2	45000	10000

28	sri nath	6623456709	m	veg	CTO	EIE	4	45000	10000

```
--30 list the details of the hostel manager
select * from worker where designation='manager';
```

Output:

worker_id	name	hostel_id	salary	gender	age	city	mobile	designation
201	suresh	222	100000	m	39	TPT	9000380380	manager
301	anil	333	130000	m	50	TPT	9177252308	manager

CHAPTER 4. CONCLUSION AND FUTUREWORK

4.1 Conclusion:

Hostel management system is designed to manage all hostel activities like hostel admissions, fees, room, mess allotment, hostel stores & generates related reports for smooth transactions. It is also used to manage monthly mess bill calculation, hostel staff payroll, student certificates, etc.

To conclude the description about the project, the project, developed using PHP with MySQL & XAMPP is based on the requirement specification of the user and the analysis of the existing system, with flexibility for future enhancement. HOSTEL MANAGEMENT SYSTEM is very useful for hostel allotment and mess fee calculation This hostel management software is designed for people who want to manage various activities in the hostel. For the past few years the numbers of educational institutions are increasing rapidly Thereby the numbers of hostels are also increasing for the accommodation of the students studying in this institution. And hence there is a lot of strain on the person who are running the hostel and software's are not usually used in this contex. This project deals with the problems on managing a hostel and avoids the problems which occur when carried manually. Identification of the drawbacks of the existing system leads to the designing of computerized system that will be compatible to the existing system with the system which is more user friendly and more GUI oriented.

4.2 Future Work:

This project is aimed at developing a system for keeping records and showing information about a hostel. This system will help the hostel officer to be able to manage the affairs of the hostel. This system will provide full information about a student in the hostel. It will show rooms available or not and number of people in a particular room. This will also provide information on students who have paid in full or are still owing. This system will also provide a report on the summary detail regarding fees and bills students are owing. Also included is a user module for employees or the hostel officer.

With this database we can develop a web application or android applications with some modifications. Moreover, combining fieldwork and lectures to produce a comprehensive research report the time limit was tiresome. Nevertheless, the quality of this study was not compromised.