



HOSTEL MANAGEMENT SYSTEM

PROJECT REPORT DBMS



JANUARY 13, 2021

MUHAMMAD BILAL

3993-FBAS/BSCS/F18-B

Teacher : M Imran Saeed

CHAPTER #1: INTRODUCTION

1.1 INTRODUCTION :

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 computerised 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 favour 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 the 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 are 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 students details.

1.2 EXISTING SYSTEM:

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 has to 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 more storeys and it would be very tedious to go through all storeys 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.

DISADVANTAGES:

- More human power
- More strength and strain of manual labour needed
- Repetition of same procedure.
- Low security.
- Data redundancy.
- Difficulty to handle.
- Difficulty to update data.
- Record keeping is difficult.
- Backup data can be easily generated.

1.3 PROCESSED SYSTEM:

This project is aimed at developing a system for keeping records and showing information about or in 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.

ADVANTAGES

- Less human error
- Strength and strain of manual labour can be reduced
- High security
- Data redundancy can be avoided to some extent
- Data consistency
- Easy to handle
- Easy data updating
- Easy record keeping
- Backup data can be easily generated

CHAPTER #2 : DATA REQUIRMENT

This document is explaining about the entities we are using in the project, reason for choosing the entities their attributes how they will work ,data_type, size, primary key, foreign key , constraints (if exists) and flow of data that how the data will move between the entities.

Basically this is designed to make it more easy and understandable for everyone.

2.1 ENTITY CLASSES:

1. Hostel.
2. Students.
3. Employs.
4. Complaints.
5. Blocks.
6. Rooms.
7. Mess details.
8. Bill Reports.

2.2 DISCRIBE ENTITIES AND THEIR ATTRIBUTES:

HOSTEL

Many institutes provide the facility of hostel for boys and girls.

REASON:

We are taking the entity named as hostel because from here data will move to other entities and we will manage the database.

This entity will manage the data of students in hostel.

ATTRIBUTES:

If we talk about the attributes, there can be many but we have mention only few those will be more help full to make the project workable and simple as Following bellow.

1. **H ID**
2. H_Name
3. Adress
4. H_Type

DATATYPE AND SIZE:

Each attribute have the suitable data-type and size according to the requirement.

PRIMARY KEY

 **H-ID**

> STUDENT

Students will be in hostel or hostel has students.

REASON:

The reason for choosing this entity is that ,student is the main entity in the hostel and we will keep the database of students and logically related things with the students.

ATTRIBUTES:

These are selected so that hostel administration can easily manage the whole information about student and through different perspectives students can be accessed by hostel management.

1. **Reg No**
2. S_Name
3. Contact No
4. Address
5. S_DOB
6. Religion

DATATYPE AND SIZE:

Each attribute have the suitable data-type and size according to the requirement.

PRIMARY KEY:

- ❖ **Reg No**

EMPLOY

REASON:

There will be staff in mess and hostel as well.
Some will work in mess and others like sweepers in hostel and many more.

ATTRIBUTES:

1. **E_id.**
2. E_name.
3. E_phn nO.
4. E_mail.
5. E_role.

DATATYPE AND SIZE:

Each attribute have the suitable datatype and size according to the requirement.

PRIMARY KEY:

1. E_id is used as a primary key.

CONSTRAINTS:

Primary key should be unique and cannot be null.

H_ID will be foreign key in this table.

> BLOCK

Hostel have 1 or more Blocks

REASON:

To adjust more student no. of rooms are increased by building blocks.

ATTRIBUTES:

There can be many attribute in this entity as below.

1. B_id
2. B_name
3. no_of_floors
4. no_of_rooms

.

DATATYPE AND SIZE:

Each attribute have the suitable data-type and size according to the requirement.

PRIMARY KEY:

B_id is the primary key and it can't be null.

CONSTRAINTS:

Primary key should be unique and cannot be null.

H_ID will be foreign key in this table.

> ROOM

Students come in the hostel to get the room.

REASON:

Room will be allotted to different students, so the students can be accessed by the attributes of room as well.

ATTRIBUTES:

There can be many attribute in this entity

1. **R_No**
2. R_type
3. R_booking Status
4. R_capacity
5. R_condion

DATATYPE AND SIZE:

Each attribute have the suitable data-type and size according to the requirement.

PRIMARY KEY:

R No is the primary key and it can't be null.

CONSTRAINTS:

Primary key should be unique and cannot be null.

➤ MESS

Students will get the facility of mess

REASON:

Students will get the facility of mess. It can be considered as marketing strategy so more students will come in this hostel because every hostel do not give this facility.

ATTRIBUTES:

1. M No
2. M_Name
3. M_Type
4. PerDayCost

DATATYPE AND SIZE:

Each attribute have the suitable data-type and size according to the requirement.

PRIMARY KEY:

M No will be the primary key

CONSTRAINTS:

Primary key should be unique and cannot be null.

➤ **BILL REPORT**

Obviously when students will live in the hostel they will submit hostel fee.

REASON:

When students will live in hostel there will be fees charges on the monthly basis. So he must pay bill

ATTRIBUTES:

1. Bill No
2. Room_cost
3. Mess_cost
4. Electric bill
5. Caution_money

DATATYPE AND SIZE:

Each attribute have the suitable data-type and size according to the requirement.

PRIMARY KEY:

Bill No is the primary key and it can't be null.

CONSTRAINTS:

In which three foreign keys R_No ,M _No, Reg No

➤ COMPLAINTS

Student will have issues while living in hostel.

REASON:

This is weak entity here because if the student will exist then complaints will be submitted other-wise not.

When students will live in hostel they will have issues and complaints.

ATTRIBUTES:

1. Reg No
2. E id.
3. complaint
4. cdate

DATA TYPE AND SIZE:

Each attribute have the suitable data-type and size according to the requirement.

PRIMARY KEY:

E id and Reg No is the primary key and it can't be null. Their both keys work as a composite key

CONSTRAINTS:

2.3 BUSSINESS RULES

For the best understanding first we have to define the project scope or the scenario because different problem can be solve different design and more than one scenarios can be created for each problem. People design them according to their thinking.

We are also creating some type scenario so that our design can be bit specific for some kind of situation.

Our project is defined as;

As we can see that our university has the facility of hostel for boys and girls. We will focus on the boy's hostel only as almost all the things will be same in both hostel we will manage only boys so that it will be simple and easy to understand for everyone. Obviously many students will be living in the boy's hostel.

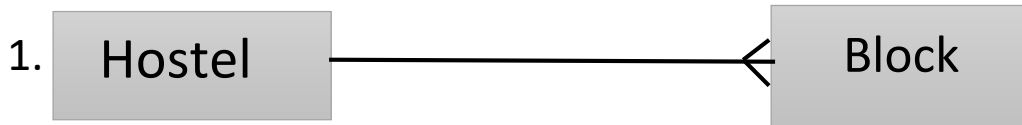
Boy's hostel has many rooms for the accommodation of the students in which more than one student accommodate their self.

Each room has assigned different types of furniture for the students so they can easily spend the time in room.

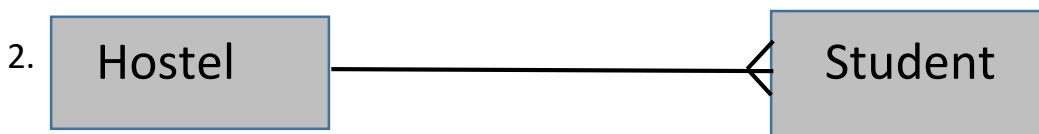
There are many workers in the boy's hostel who work in the mess and hostel as well (cleaning, washing etc);

Every student must submit his fees in-time so they can get the every facility in hostel. For the food service the hostel management is providing the facility of mess for students so students and easily get the meal in hostel. It is also possible that visitors can come to meet students those are living in hostel.

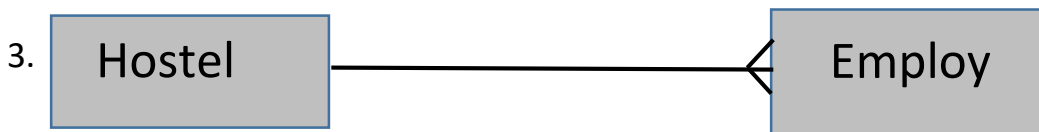
2.4 TYPE OF RELATIONSHIP



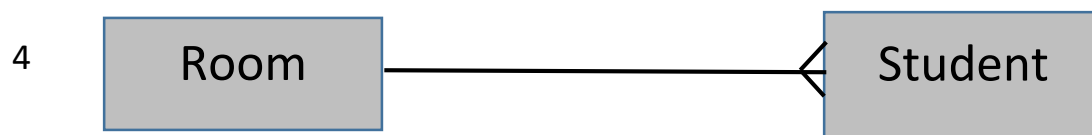
This relation is in **One_to_Many** Because One hostel have Many **BLOCKS** but one block created in One hostel.



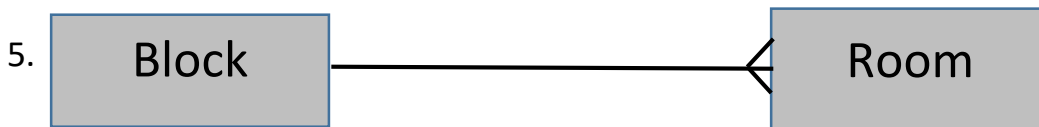
This relation is in **One_to_Many** Because One hostel have Many **SUDENT** but one student lived in One hostel.



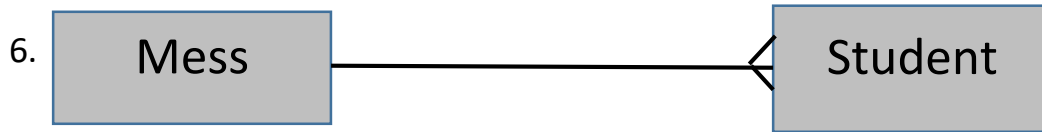
This relation is in **One_to_Many** Because One hostel have Many **EMPLOY**s but one EMPLOY work in One hostel.



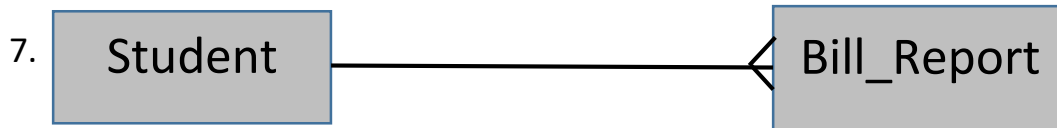
This relation is in **One_to_Many** Because One **ROOM** have Many **STUDENT** but one Student lived in One Room.



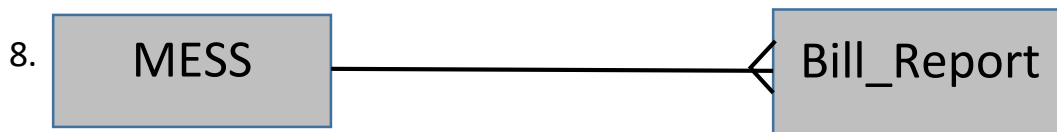
This relation is in **One_to_Many** Because One **BLOCK** have Many **ROOMS** but one Room made in One Block.



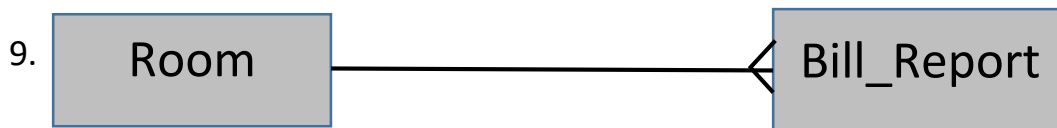
This relation is in **One_to_Many** Because One **MESS** Serve Food of many **STUDENT** but one Student can eat in One Mess.



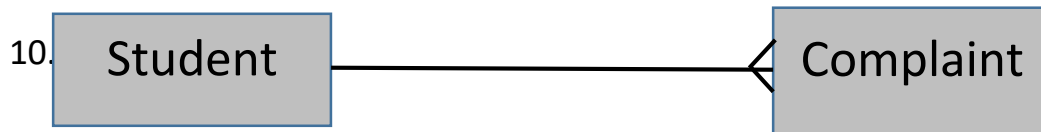
This relation is in **One_to_Many** Because One **STUDENT** can have many **BILL RECODS** but Bill_report is can only be of one student.



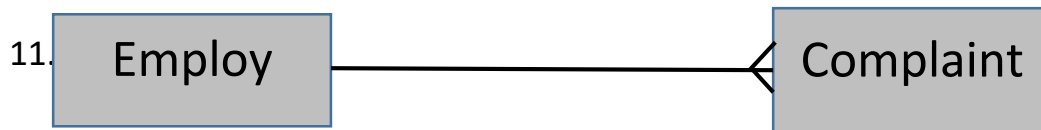
This relation is in **One_to_Many** Because One **MESS** can genrate many **BILL RECODS** but one Bill_report is can only be of one Mess



This relation is in **One_to_Many** Because One **ROOM** can genrate many **BILL** but one Bill_report is can only be of one room.

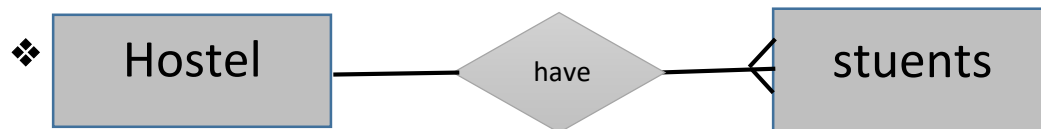


This relation is in **One_to_Many** Because One **STUDENT** can have many **COMPLAINS** but One complain is only of one student at one time.

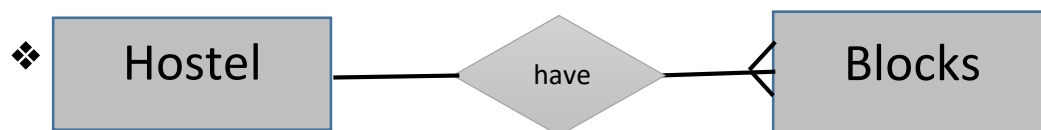


This relation is in **One_to_Many** Because One **EMPLOY** can collect many complains but One complain can be collected by employ at one time.

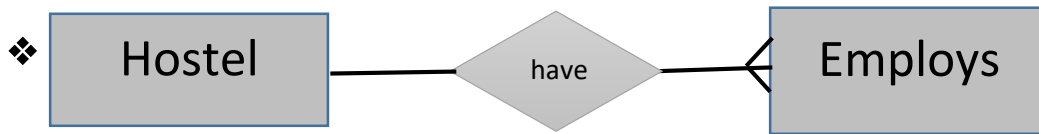
2.5 DEGREE OF RELATIONSHIPS:



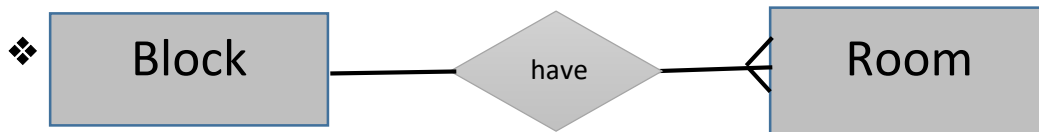
This relationship is in **Binary** and the degree of this relationship is **2**.



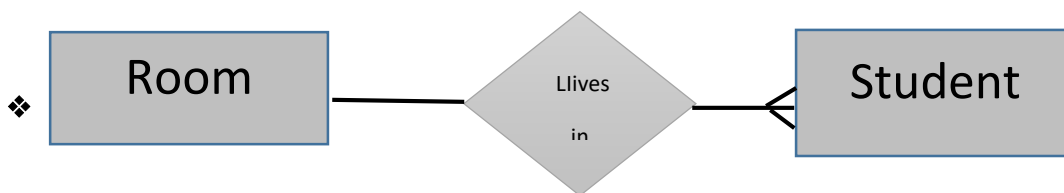
This relationship is in **Binary** and the degree of this relationship is **2**



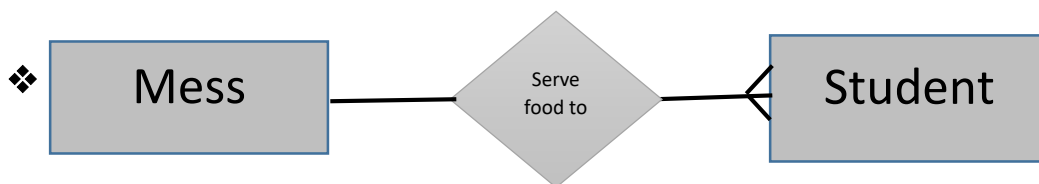
This relationship is in **Binary** and the degree of this relationship is **2**



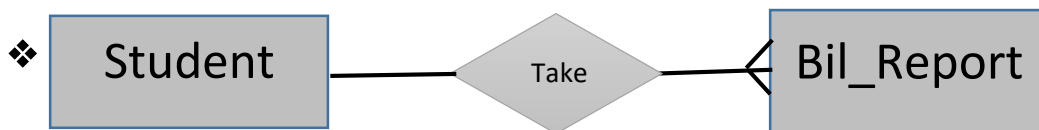
This relationship is in **Binary** and the degree of this relationship is **2**



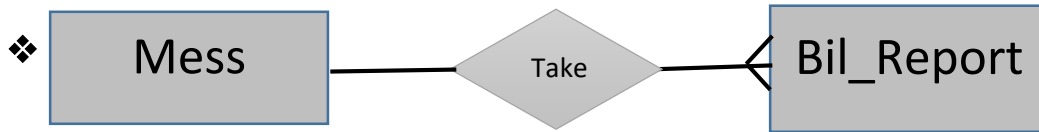
This relationship is in **Binary** and the degree of this relationship is **2**



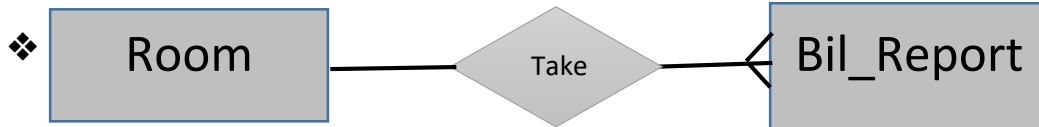
This relationship is in **Binary** and the degree of this relationship is **2**.



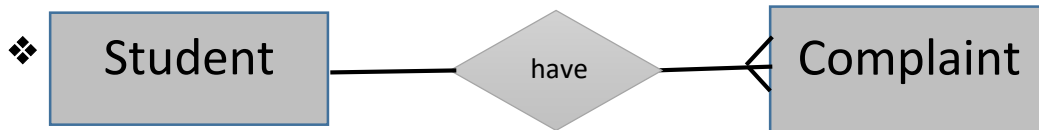
This relationship is in **Binary** and the degree of this relationship is **2**



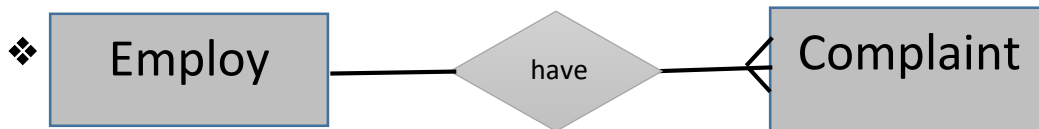
This relationship is in **Binary** and the degree of this relationship is **2**



This relationship is in **Binary** and the degree of this relationship is **2**



This relationship is in **Binary** and the degree of this relationship is **2**

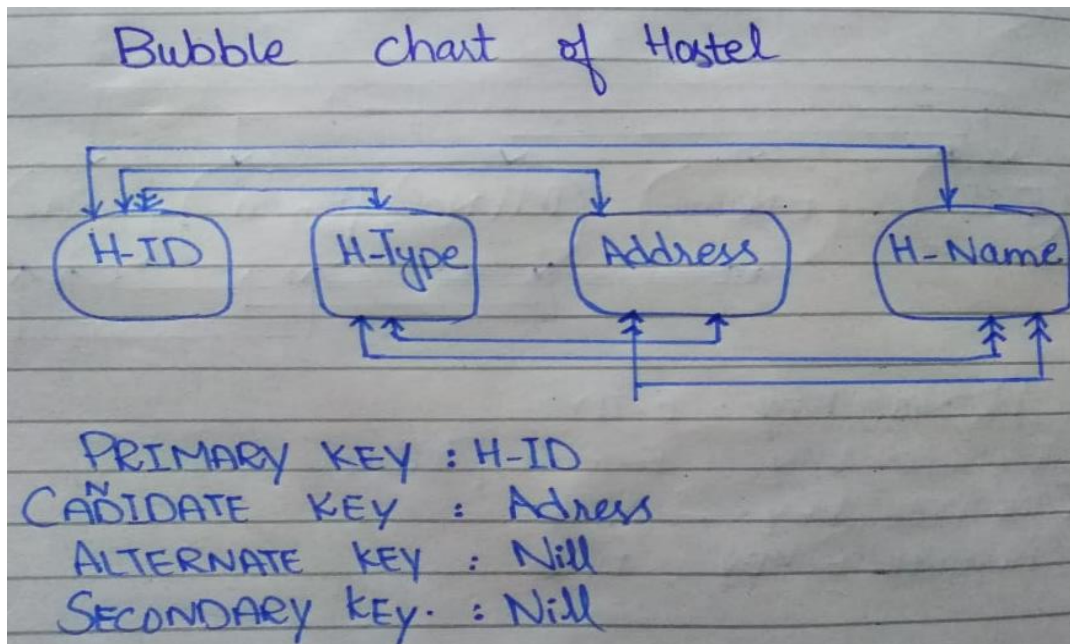


This relationship is in **Binary** and the degree of this relationship is **2**

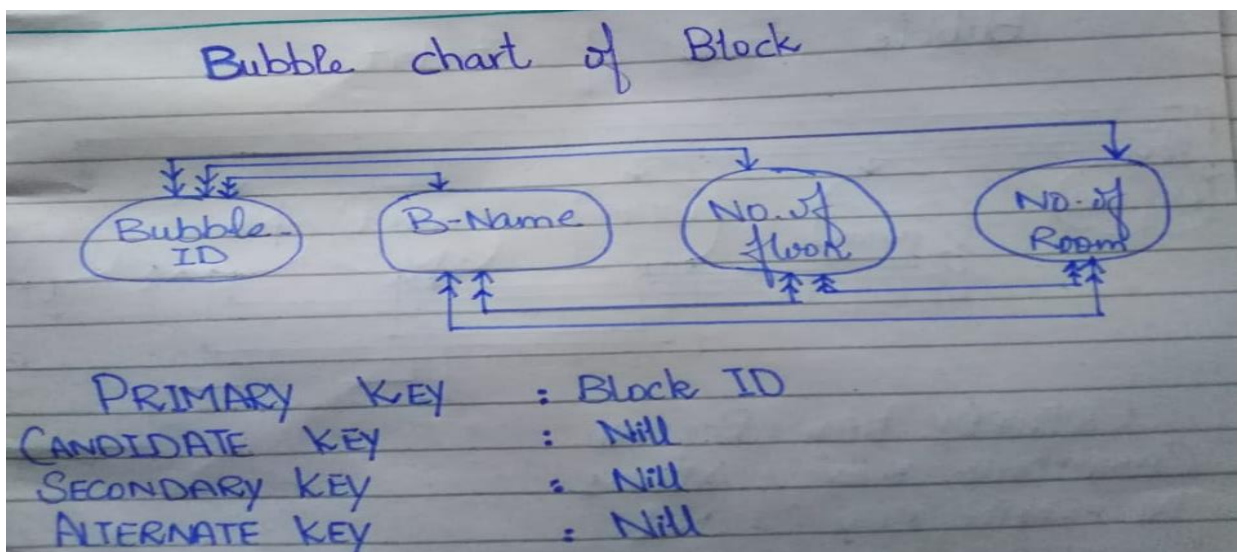
CHAPTER 3: DATA MODELING:

BUBBLE CHARTS:

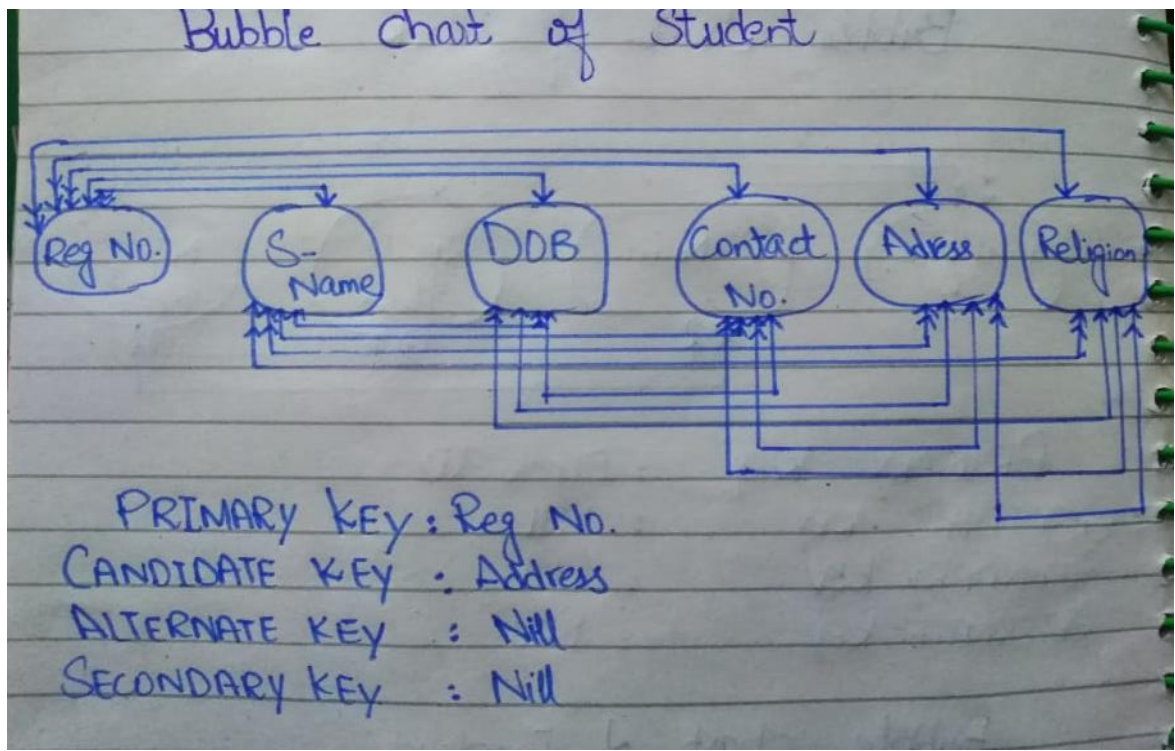
➤ HOSTEL



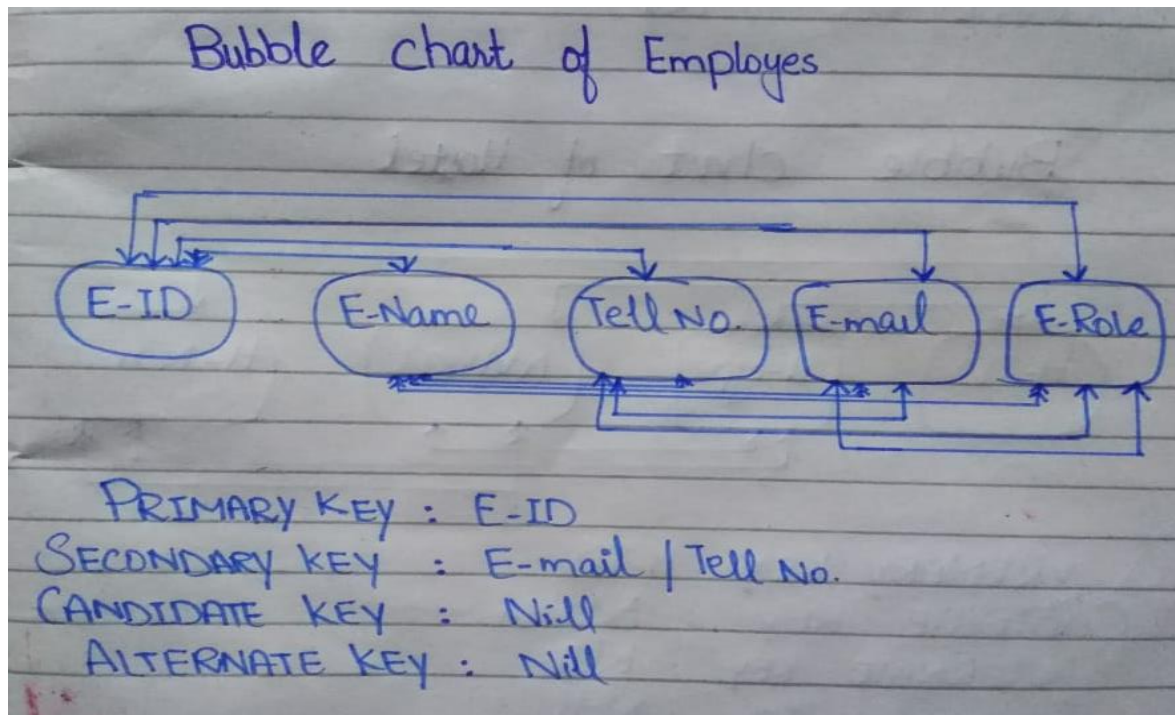
➤ BLOCK:



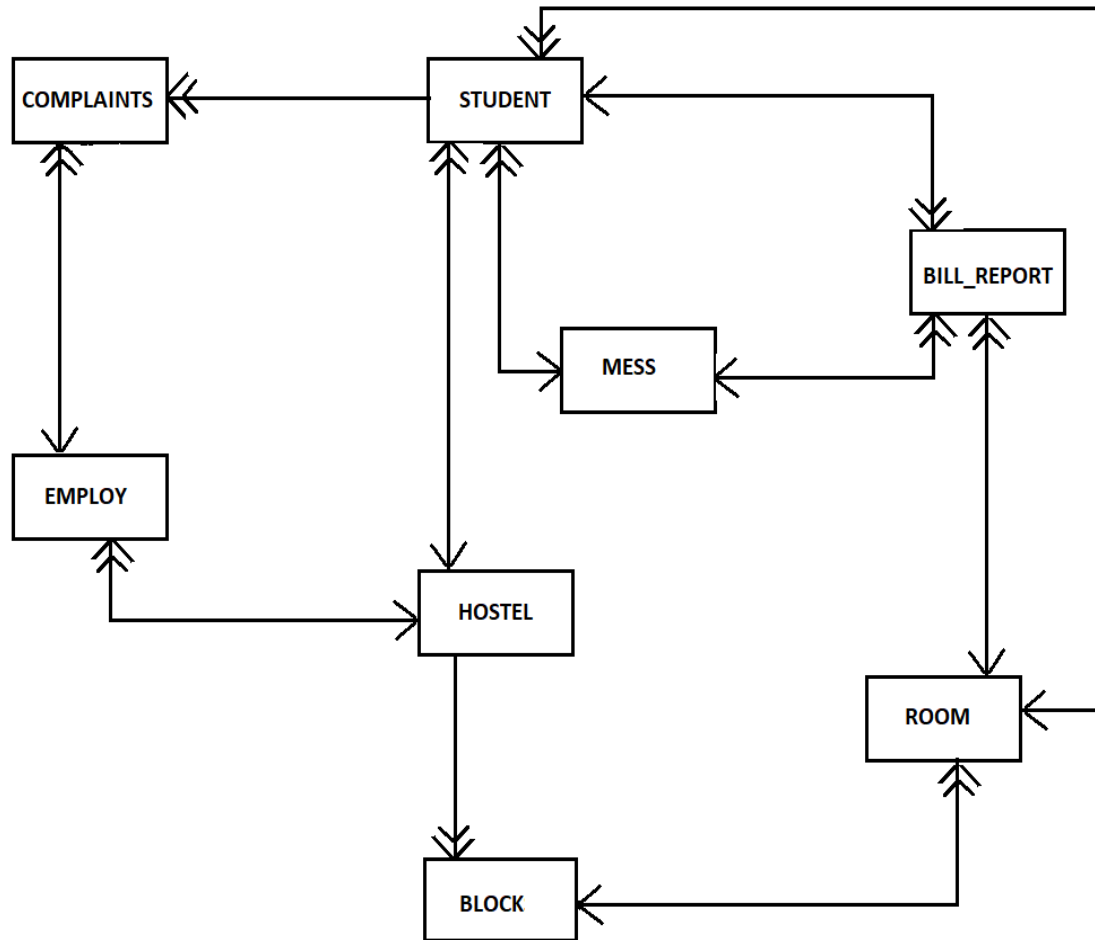
➤ STUDENT:



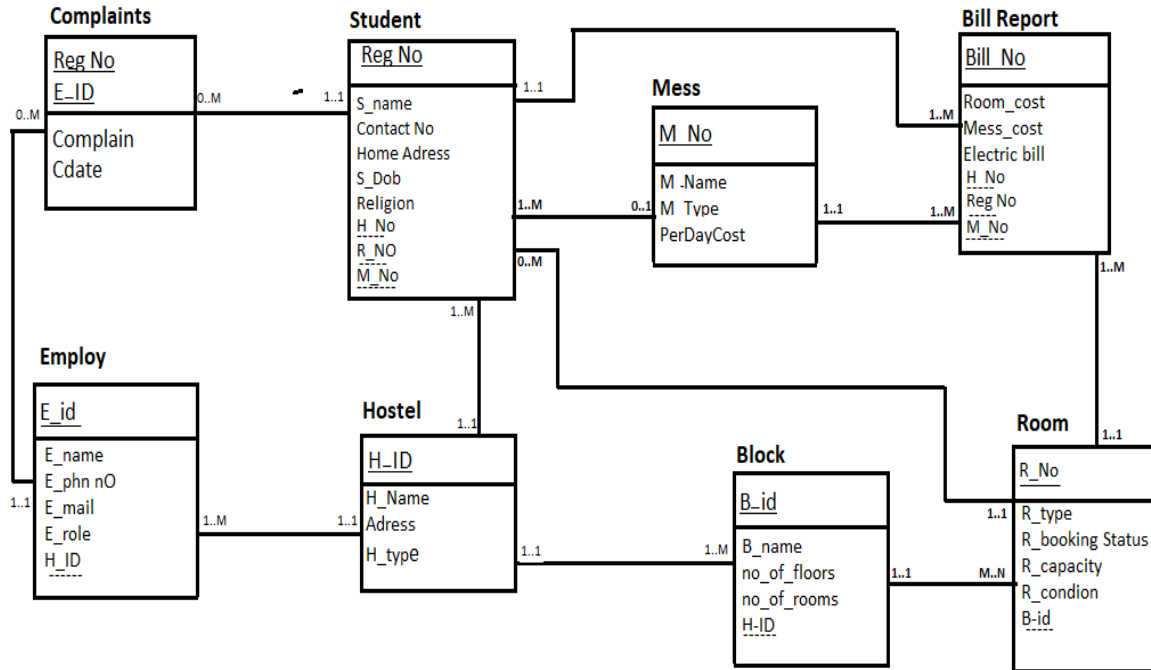
➤ EMPLOY:



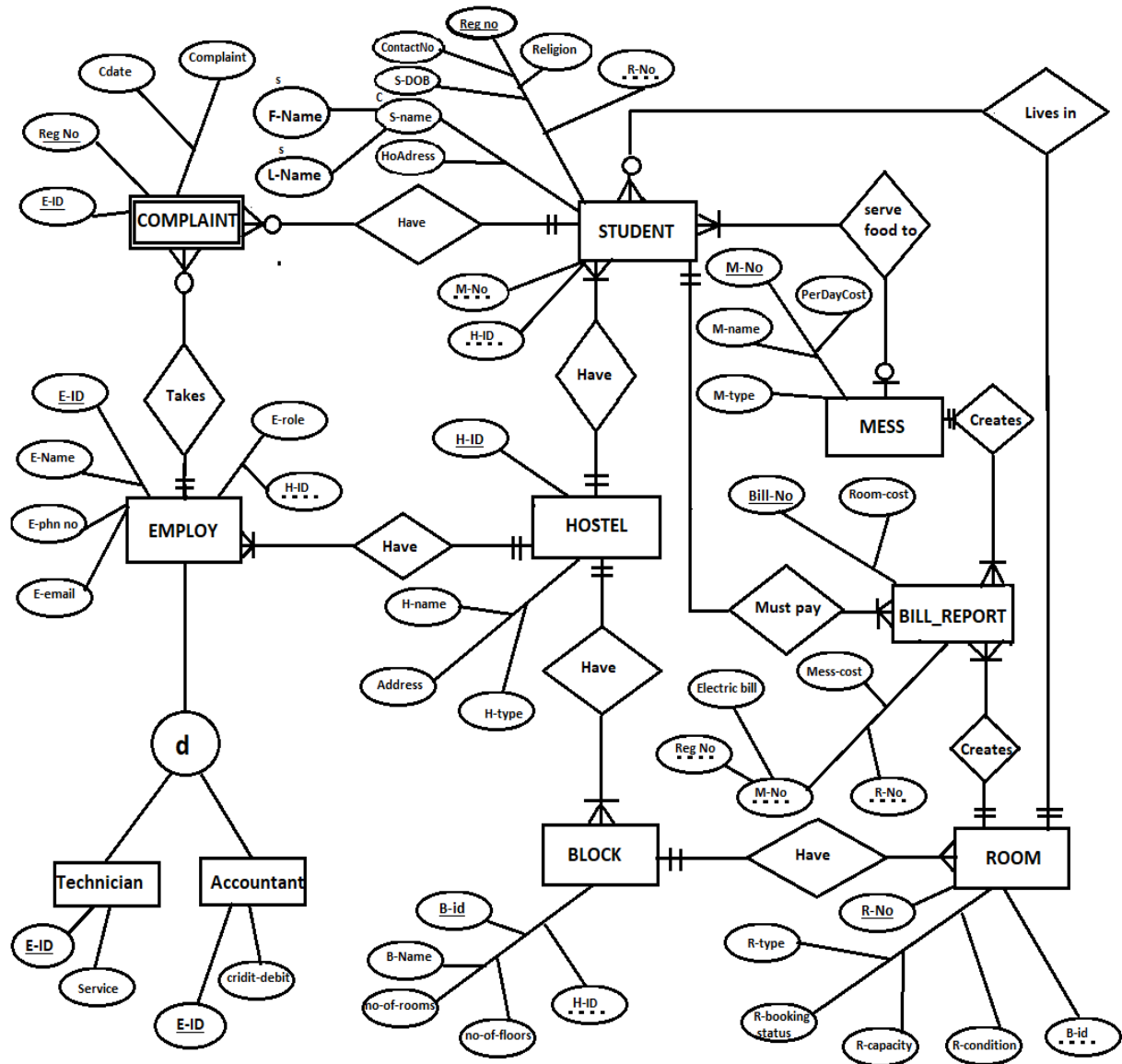
3.2 DATA STRUCTURE DIAGRAM



3.3 ENTITY RELATIONSHIP DIAGRAM(ERD):



3.4 ENHANCED ENTITY RELATIONSHIP DIAGRAM(EERD):



CHAPTER :4 LOGICAL DATA MODEL

In Chapter 4, we are going to discuss about Entity Classes, their Relationship and their Normal forms after removing all their anomalies.

4.1 REPRESENT ENTITY CLASS :

Firstly, we will represent all the entity classes of my project. Those Tables are as below;

HOSTEL

<u>H-ID</u>	H_Name	Adress	H_type
1111	Ibn_E_Hasham	F9ISD	BOYS
2222	Fatima	F8ISD	GIRLS
5555	Hamdani	I8ISD	BOYS
6666	Bilal	F5ISD	BOYS
7777	Umer	E7ISD	GIRLS

STUDENT

<u>Reg No</u>	S_Name	Contact No	Home Adress	S_Dob	Religion
3990	Jawad	0466464	ISB	03 JAN	Muslim
3991	Hamza	4444556	LHR	01 JAN	Muslim
3992	Awais	9495995	RWP	04 JAN	Muslim
3993	Bilal	0585676	GOJ	02 Feb	Muslim
4003	Zawar	0586776	FSD	01 Dec	Muslim

EMPLOY

E-id	E_name	E_phn nO	E_mail	E_role
5656	Imram Saeed	538856	imran.com	manager
6888	Ali	5767666	ali.com	other works
7676	Hamza	999999	hamza.com	assistant manager
8989	Awais	69878	awais.com	guard
9999	Ahmad	686687	ahmad.com	washer

BLOCK

B-id	B_name	no_of_floors	no_of_rooms
QR111	QAID-E-AZAM	2	120
QR222	ALLAMA IQBAL	3	122
QR333	FATIMA	4	238
QR444	BSCS	5	344
QR555	SHAHSA	3	200

ROOM

R-No	R_type	R_booking Status	R_capacity	R_condition
1	simple	Yes	4	Normal
2	luxury	No	3	Good
3	simple	No	5	extra ordinary
4	simple	yes	2	Normal
5	luxury	yes	4	Good

Complaint

<u>Reg No</u>	<u>E-ID</u>	Complain	Cdate
3990	5656	ABC	01 FEB
3991	6888	DDD	02 FEB
3992	7676	RRR	03 FEB
3993	8989	GGG	04 FEB
4003	9999	YYY	05 FEB

MESS

M_No	M_Name	M_Type	PerDayCost
01	MR1	Simple	1.41
02	MR2	Good	1.42
03	MR3	Simple	1.43
04	MR4	Good	1.44
05	MR5	Simple	1.45

Bill_report

Bill_No	Room_cost	Mess_cost	Electric bill
120	2300	2000	1250
122	2020	4000	1200
126	2490	1500	1400
133	2000	3000	1000
144	2200	5000	1300

4.2 REPRESENT RELATINSHIPS:

In which we represent the relationship between entity classes as below.

Hostel			
<u>H-ID</u>	H_Name	Adress	H_type
1111	Ibn_E_Hasham	F9ISD	BOYS
2222	Fatima	F8ISD	GIRLS
5555	Hamdani	I8ISD	BOYS
6666	Bilal	F5ISD	BOYS
7777	Umer	E7ISD	GIRLS

Student								
<u>Reg No</u>	S_name	Contact No	Home Adress	S_Dob	Religion	<u>H-ID</u>	<u>R-No</u>	<u>M-No</u>
3990	Jawad	0466464	ISB	03 JAN	Muslim	1111	1	01
3991	Hamza	4444556	LHR	01 JAN	Muslim	2222	2	02
3992	Awais	9495995	RWP	04 JAN	Muslim	5555	3	03
3993	Bilal	0585676	GOJ	02 FEB	Muslim	6666	4	04
4003	Zawar	0586776	FSD	01 DEC	Muslim	7777	5	05

Employ					
<u>E-id</u>	E_name	E_phn nO	E_mail	E_role	<u>H-ID</u>
5656	Imram Saeed	538856	imran.com	Manager	1111
6888	Ali	5767666	ali.com	Other Works	6666
7676	Hamza	999999	hamza.com	Assistant Manager	2222
8989	Awais	69878	awais.com	Guard	5555
9999	Ahmad	686687	ahmad.com	Washer	7777

Block

<u>B-id</u>	<u>B_name</u>	<u>no_of_floors</u>	<u>no_of_rooms</u>	<u>H-ID</u>
QR111	QAID E AZAM	2	120	1111
QR222	ALLAMA IQBAL	3	122	2222
QR333	FATIMA	4	238	5555
QR444	BS	5	344	6666
QR555	SHAHSA	3	200	7777

Room

<u>R-No</u>	<u>R_type</u>	<u>R_booking Status</u>	<u>R_capacity</u>	<u>R_condion</u>	<u>B-id</u>
1	Simple	Yes	4	Normal	QR111
2	Luxury	No	3	Good	QR222
3	Simple	No	5	Extra Ordinary	QR333
4	Simple	yes	2	Normal	QR444
5	Luxury	yes	4	Good	QR555

Complaints

<u>Reg No</u>	<u>E-ID</u>	<u>Complain</u>	<u>Cdate</u>
3990	5656	ABC	01 FEB
3991	6888	DDD	02 FEB
3992	7676	RRR	03 FEB
3993	8989	GGG	04 FEB
4003	9999	YYY	05 FEB

Mess

<u>M-No</u>	<u>M_Name</u>	<u>M_Type</u>	<u>PerDayCost</u>
01	MR1	Simple	1.41
02	MR2	Good	1.42
03	MR3	Simple	1.43
04	MR4	Good	1.44
05	MR5	Simple	1.45

Bill report

<u>Bill-No</u>	<u>Room_cost</u>	<u>Mess_cost</u>	<u>Electric bill</u>	<u>Reg No</u>	<u>R-NO</u>	<u>M-No</u>
120	2300	2000	1250	3990	1	01
122	2020	4000	1200	3991	2	02
126	2490	1500	1400	3992	3	03
133	2000	3000	1000	3993	4	04
144	2200	5000	1300	4003	5	05

4.3 NORMALIZATION:

IN which we discuss about These ENTITIES Have Anomalies or Not:

HOSTEL			
<u>H-ID</u>	H_Name	Adress	H_type

In Hostel table no anomalies are Present .We can update , delete and insert data . There is no repeating Group ,no partial dependency , no transitive dependency so its is already in 1NF,2NF,3NF.

Student								
<u>Reg No</u>	S_name	Contact No	Home Adress	S_Dob	Religion	<u>H-ID</u>	<u>R-No</u>	<u>M-No</u>

In student table no anomalies are Present .We can update , delete and insert data . There is no repeating Group ,no partial dependency , no transitive dependency so its is already in 1NF,2NF,3NF.

EMPLOY					
<u>E-id</u>	E_name	E_phn nO	E_mail	E_role	<u>H-ID</u>

In employ table no anomalies are Present .We can update , delete and insert data . There is no repeating Group ,no partial dependency , no transitive dependency so its is already in 1NF,2NF,3NF.

BLOCK

<u>B-id</u>	B_name	no_of_floors	no_of_rooms	H-ID

In block table no anomalies are Present .We can update , delete and insert data . There is no repeating Group ,no partial dependency , no transitive dependency so its is already in 1NF,2NF,3NF.

ROOM

<u>R-No</u>	R_type	R_booking Status	R_capacity	R_condion	<u>B-id</u>

In Room table no anomalies are Present .We can update , delete and insert data . There is no repeating Group ,no partial dependency , no transitive dependency so its is already in 1NF,2NF,3NF.

COMPLAINTS

<u>Reg No</u>	<u>E-ID</u>	Complain	Cdate

In complaintl table no anomalies are Present .We can update , delete and insert data . There is no repeating Group ,no partial dependency , no transitive dependency so its is already in 1NF,2NF,3NF.

MESS			
<u>M-No</u>	M_Name	M_Type	PerDayCost

In mess table no anomalies are Present .We can update , delete and insert data . There is no repeating Group ,no partial dependency , no transitive dependency so its is already in 1NF,2NF,3NF.

BILL REPORT						
<u>Bill-No</u>	Room_cost	Mess_cost	Electric bill	<u>Reg No</u>	<u>R-NO</u>	<u>M-No</u>

In bill report table no anomalies are Present .We can update , delete and insert data . There is no repeating Group ,no partial dependency , no transitive dependency so its is already in 1NF,2NF,3NF.

CHAPTER 5:IMPLEMENTATION

In this chapter I wil Attach some screen shorts of My MS Access.

5.1 CREATION OF TABLES:

I Create 8 tables of each entity class . Because of requirement I Attach 1 screen shot of design view out of 8 tables.

Field Name	Data Type	Description (Optional)
Reg No	Short Text	
S_name	Short Text	
Contact No	Short Text	
Home Adress	Short Text	
S_Dob	Short Text	
Religion	Short Text	
H_ID	Short Text	
R_No	Short Text	
M_No	Short Text	

Field Properties

General	
Field Size	255
Format	
Input Mask	
Caption	
Default Value	
Validation Rule	
Validation Text	
Required	Yes
Allow Zero Length	Yes
Indexed	Yes (No Duplicates)
Unicode Compression	Yes
IME Mode	No Control
IME Sentence Mode	None
Text Align	General

5.2 CREATION OF QUERY:

I Create 20 query of each entity class . Because of requirement I Attach 2 screen shot of design view out of 20 queries.

1. SIMPLE QUERY:

Mess_check_Type query

Mess

- * M_No
- M_Name
- M_Type
- PerDayCost
- Reg No

Field:	M_No	M_Name	M_Type	PerDayCost		
Table:	Mess	Mess	Mess	Mess		
Sort:						
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Criteria:			[Please Enter the Mes			
or:						

2. MULTIPLE TABLE QUERY:

Student_All Query

Student

- * Reg No
- S_name
- Contact No
- Home Adress
- S_Dob

Room

- * R_No
- R_type
- R_booking Stat
- R_capacity
- R_condition

Block

- * B_Id
- B_name
- no_of_floors
- no_of_rooms
- H_ID

Hostel

- * H_ID
- H_Name
- Adress
- H_type

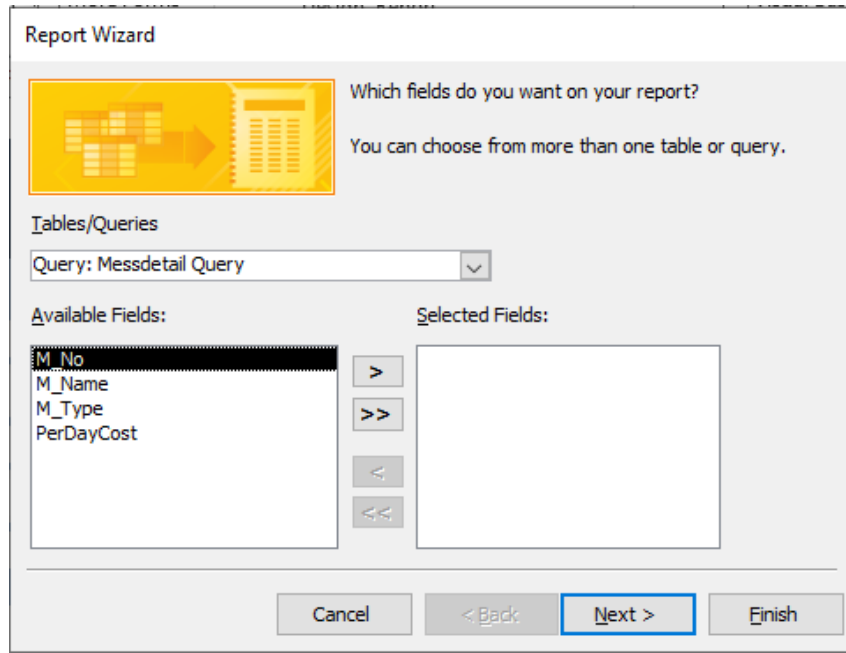
Relationships: Student (1) to Room (∞), Student (1) to Hostel (∞), Room (1) to Block (∞), Hostel (1) to Block (∞)

Field:	Reg No	S_name	H_ID	H_Name	B_id	B_name	R_No	R_type
Table:	Student	Student	Hostel	Hostel	Block	Block	Room	Room
Sort:								
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:								
or:								

5.3 CREATION OF REPORT:

I Create 20 Reports. Because of requirement I Attach screen shots of report wizard out of 20 Reports.

I.



Report Wizard

Which fields do you want on your report?
You can choose from more than one table or query.

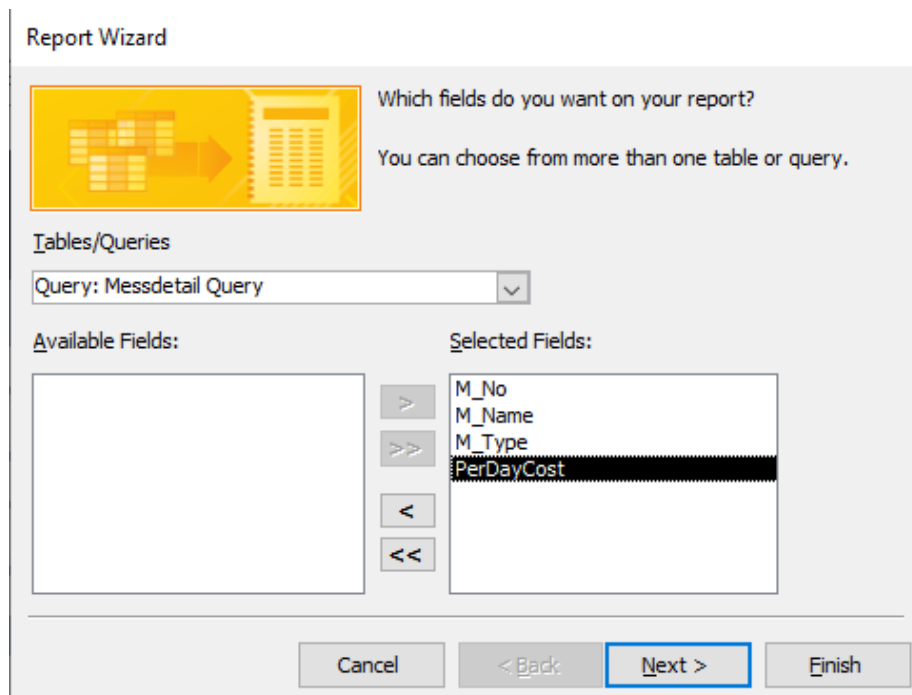
Tables/Queries
Query: Messdetail Query

Available Fields:
M_No
M_Name
M_Type
PerDayCost

Selected Fields:

Cancel < Back Next > Finish

II.



Report Wizard

Which fields do you want on your report?
You can choose from more than one table or query.

Tables/Queries
Query: Messdetail Query

Available Fields:

Selected Fields:
M_No
M_Name
M_Type
PerDayCost

Cancel < Back Next > Finish

III.

Report Wizard

Do you want to add any grouping levels?

M_No

M_Name

M_Type

PerDayCost

>

<

↑

↓

Priority

M_No, M_Name, M_Type, PerDayCost

Grouping Options ...

Cancel

< Back


Next >

Finish

IV.

Report Wizard

What sort order do you want for your records?



You can sort records by up to four fields, in either ascending or descending order.

1

Ascending

2

Ascending

3

Ascending

4

Ascending

Cancel


< Back

Next >

Finish

V.

Report Wizard



What title do you want for your report?

Messdetail Report

That's all the information the wizard needs to create your report.

Do you want to preview the report or modify the report's design?


☐ Preview the report.

☒ Modify the report's design.

Cancel < Back Next > Finish

VI.

Report Wizard



What title do you want for your report?

Messdetail Report

That's all the information the wizard needs to create your report.

Do you want to preview the report or modify the report's design?

☐ Preview the report.

☒ Modify the report's design.

Cancel < Back Next > Finish

VII.

Mess report

Report Header

Mess

Page Header

M_No	M_Name	M_Type	PerDayCost
------	--------	--------	------------

Detail

M_No	M_Name	M_Type	PerDayCost
------	--------	--------	------------

Page Footer

=Now()

=Page " & [Page] & " of " & [Pages]

Property Sheet

Selection type: Section

ReportHeader

Format Data Event Other All

Property	Value
Visible	Yes
Height	0.7083"
Back Color	Accent 1, Darker 50%
Special Effect	Flat
Auto Height	Yes
Can Grow	No
Can Shrink	No
Display When	Always
Keep Together	Yes
Force New Page	None
New Row Or Col	None

VIII.

Mess report

Mess

M_No	M_Name	M_Type	PerDayCost
02	MR2	Good	1.42
03	MR3	Simple	1.43
04	MR4	Good	1.44
05	MR5	Simple	1.45
01	MR1	Simple	1.41

Wednesday, January 13, 2021

Page 1 of 1

5.4 CREATION OF FORMS

I Create 19 forms. Because of requirement I Attach screen shots of form wizard out of 19 form.

Form Wizard

Which fields do you want on your form?
You can choose from more than one table or query.

Tables/Queries
Table: Room

Available Fields:

- R_No
- R_type
- R_booking Status
- R_capacity
- R_condition
- B_id

Selected Fields:

Buttons: > >> < <<

Buttons: Cancel < Back Next > Finish

Form Wizard

Which fields do you want on your form?
You can choose from more than one table or query.

Tables/Queries
Table: Room

Available Fields:

Selected Fields:

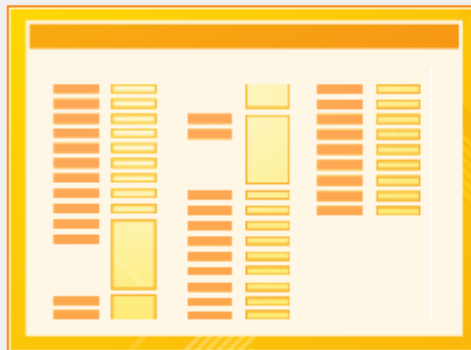
- R_No
- R_type
- R_booking Status
- R_capacity
- R_condition
- B_id

Buttons: > >> < <<

Buttons: Cancel < Back Next > Finish

Form Wizard

What layout would you like for your form?



- ☒ Columnar
- ☐ Tabular
- ☐ Datasheet
- ☐ Justified

Cancel

< Back

Next >

Finish

Form Wizard

What title do you want for your form?

Room

That's all the information the wizard needs to create your form.

Do you want to open the form or modify the form's design?

- ☐ Open the form to view or enter information.
- ☒ Modify the form's design.

Cancel

< Back

Next >

Finish

Room FORM

Form Header

Room

Detail

R_No

R_No

R_type

R_type

R_booking Status

R_booking Status

R_capacity

R_capacity

R_condition

R_condition

B_id

B_id

Property Sheet

Selection type: Section

Detail

Format

Data

Event

Other

All

Visible	Yes
Height	3.3333"
Back Color	Background 1
Alternate Back Color	Text 1
Special Effect	Flat
Auto Height	No
Can Grow	No
Can Shrink	No
Display When	Always
Keep Together	No
Force New Page	None
New Row Or Col	None

Room FORM

Room

R_No

R_type

R_booking Status

R_capacity

R_condition

B_id

simple

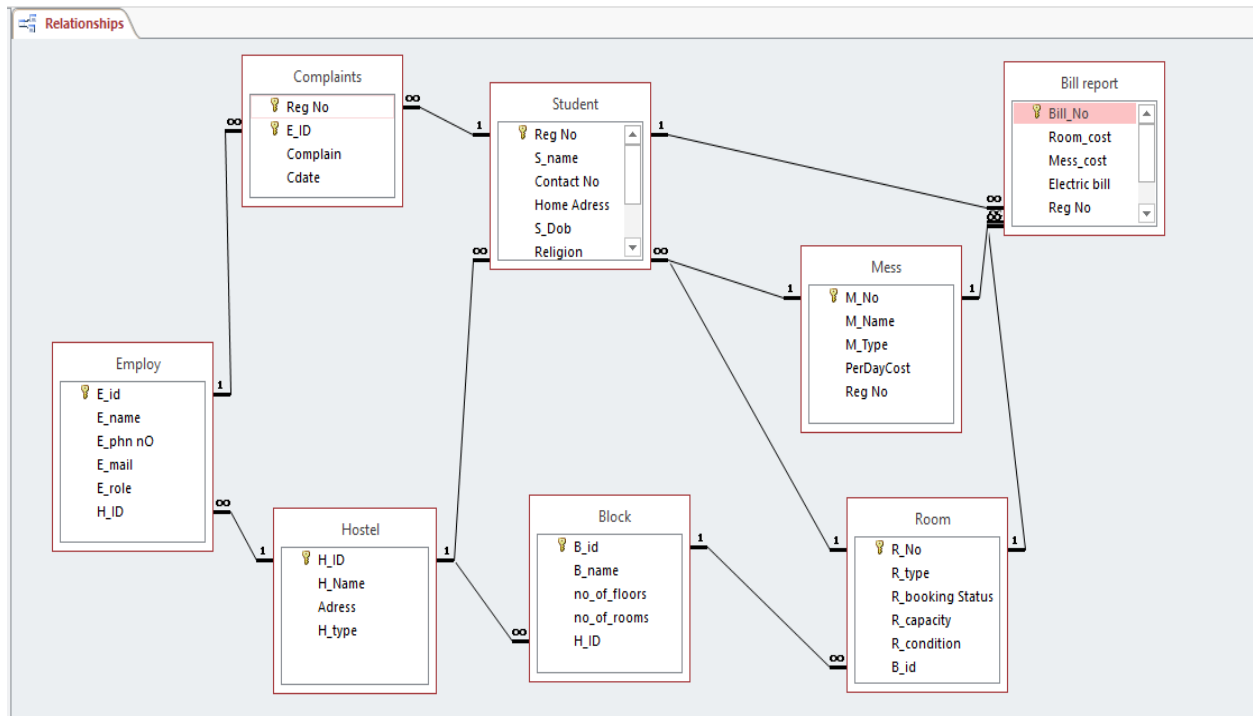
Available

4

Normal

QR111

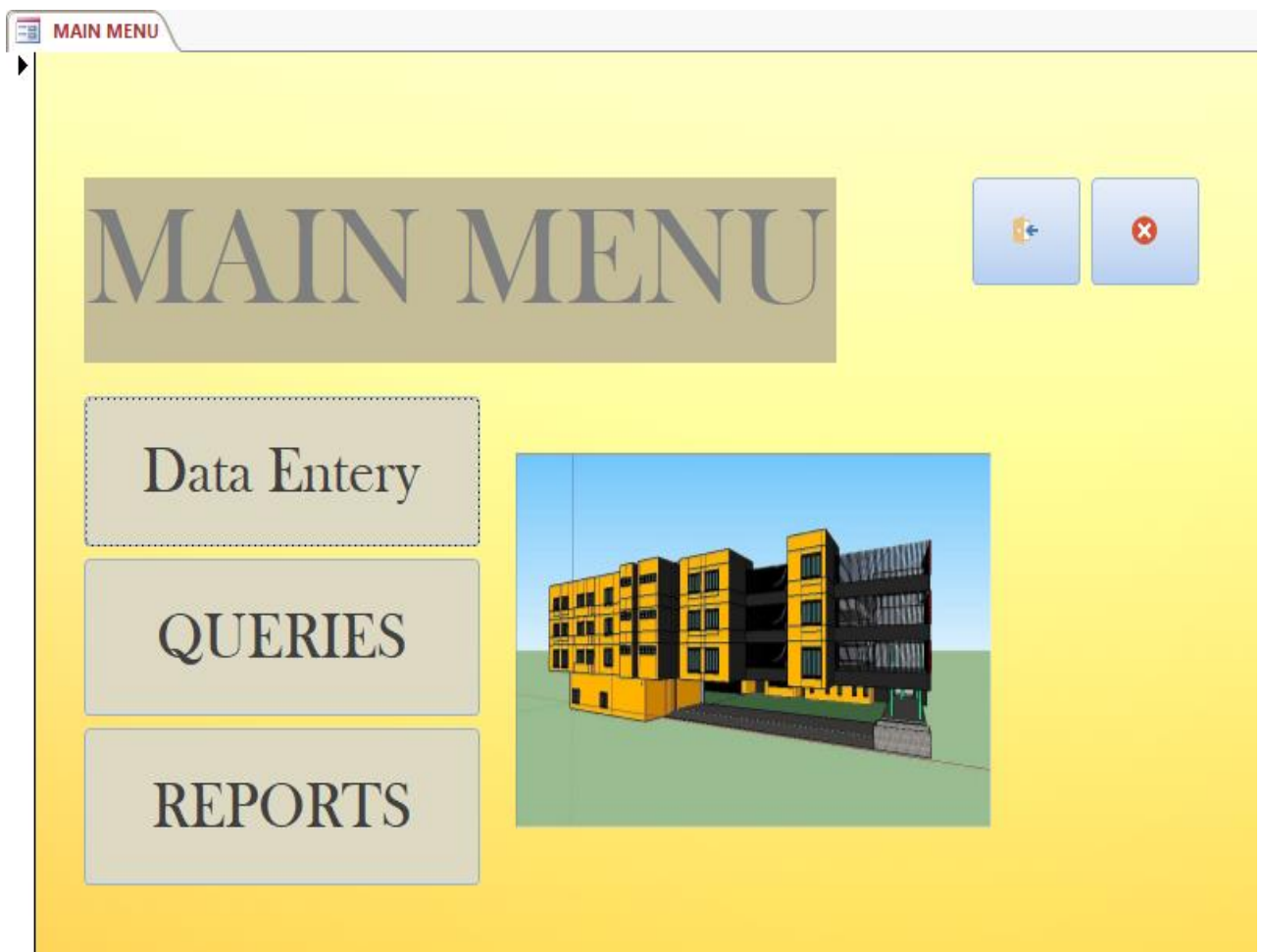
5.5 RELATIONSHIP DIAGRAM:



APPENDIX:

MAIN MENU

I create a Main menu form for Some function .In this form I create some button for some different functions like Data entry , to see All queries, to see All Reports and other for Exit and close .i Attach screen shot of this as bellow.



DATA ENTRY:

When you click data entry button then you go to Data entry form in which all the forms are present so we can entry data easily

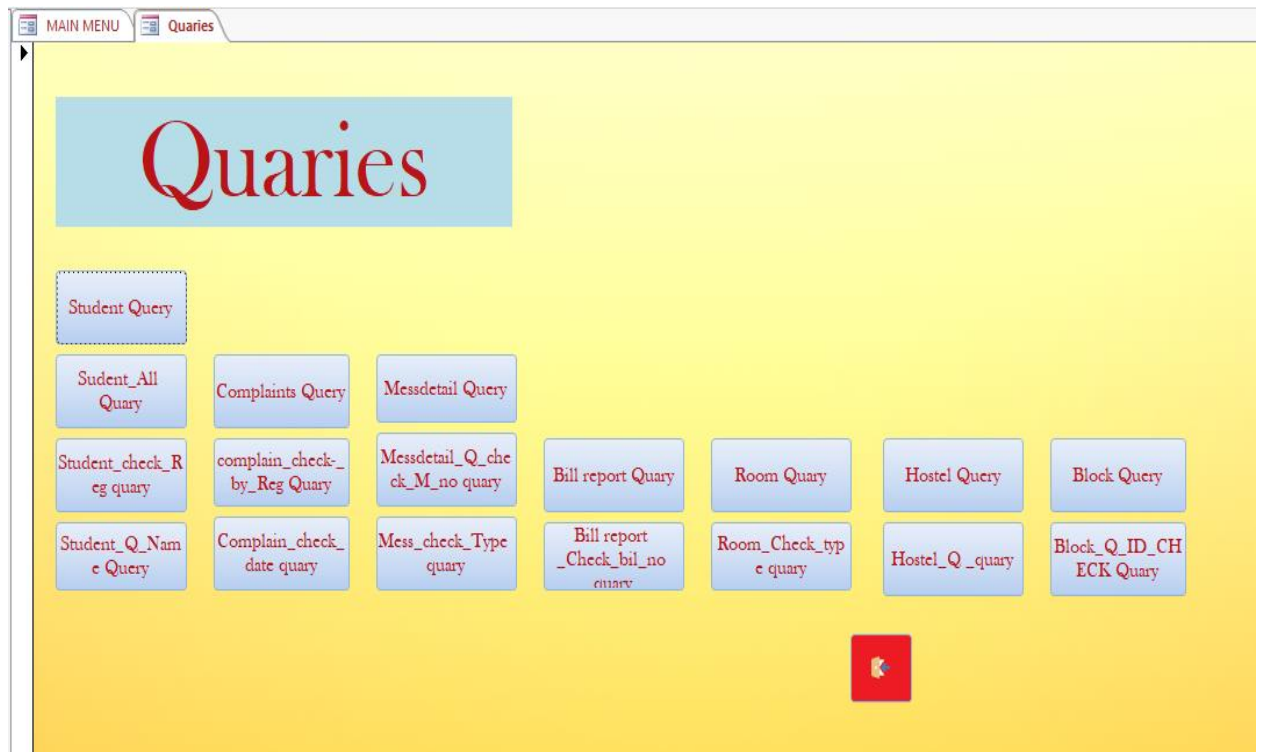
The screenshot shows a web application interface with a navigation bar at the top containing two tabs: "MAIN MENU" and "ALL FORMS". The "ALL FORMS" tab is active. Below the navigation bar is a large yellow banner with the text "Open Any Form To Enter Data" in a dark purple box. Underneath the banner, there is a grid of 12 purple buttons arranged in 4 rows and 3 columns. The buttons are labeled as follows:

Hostel Form	Employ Form	Bill report _Check_bil_no quarry
Hostel quarry form	Employ Query	Bill report Query FORM
Student form	Block Form	Room FORM
Student_All_By_Reg	Block Query FORM	Room Query
Complaints Query form	Messdetail Query form	Room_Check_type quarry form

In the bottom right corner of the yellow area, there is a small red square button with a white icon.

QUERIES FORM:

When we click to queries button then we go to queries form in which we can see all the queries of my database .



REPORTS FORM:

When we click on reports button in main menu then a reports form will be open in we can see all the reports.

