

Assignment No: 2

1. What is difference between weak entity set & Strong entity set?

→

Strong Entity Set

- | Strong Entity Set | Weak Entity Set. |
|---|---|
| i) The Strong entity has a primary key. | i) weak entity set has a partial discriminator key |
| ii) The Strong entity set is denoted by single rectangle. | ii) Weak entity set is denoted by double rectangle. |
| iii) The Strong entity is independent of any other entity in a schema. | iii) weak entity dependent on the Strong entity for its existence. |
| iv) The relation between two strong entities is denoted by a single diamond simply called relationship. | iv) The relationship between a weak & strong entity is denoted by identifying relationship denoted with double diamond. |
| v) Strong entity may or may not have total participation in a relationship. | v) weak entity always has total participation in the identifying relationship shown by close double line. |

2. What are the types of Attributes?

1. Simple Attribute:-

An attribute that cannot be further subdivided into components is a simple attribute.

ex:- The roll number of a student, the ID numbers of an employee & many more.

2. Composite Attribute:-

An attribute that can split into components is a composite attribute.

ex:- The address can be further split in house number.

3. Single - Valued Attribute:-

The attribute which takes up only a single value for each entity instance is a single-valued attribute.

4. Multi - Valued attribute:-

The attribute which takes up more than a single value of for each entity instance is a multi-valued attribute. And it is represented by double oval shape.

5. Stored Attribute:-

The stored attribute are those attribute which + which doesn't require any type further update since they are stored in the database.

6. Derived Attributes:-

An attribute that can be derived from other attributes is derived attributes. And it is represented by dedotted oval shape.

7. Complex Attribute:-

Those attributes, which can be formed by the nesting of composite and multi-valued attributes, are called complex attributes.

Q. What is domain & entity.

Domain :-

- It is set of allowable values for attribute.
- It candidates area or field.
- It is also indicated indicates set of values of same data type.
- A domain refers to the set of valid values that a specific attribute can take.

Entity :-

- An individual object in a system system is called as Entity.
- An entity is an object or thing in the real world that is distinguishable from other objects.

4. Define Super key:-
→

Super Key: It is a set of one or more attributes that, taken collectively, to identify uniquely an item in the entity set.

ex- OSGN is a Super key.

① STUD STUD NO, (STUD NO, STUD NAME), etc.

5. Define the term. i) candidate key, ii) primary key.
→

i) Candidate key:-

The minimal set of attributes that can uniquely identify a tuple is known as Candidate key.

Syntax-

Create table table_name (column name data type (size) constraint constraint_name unique column name data type (size));

ex-

STUD NO, e in STUDENT relation.

ii) Primary Key:-

Primary key denotes a candidate key that is chosen by the database designer as the principal means of identifying items within an entity set.

The primary key should be chosen such that

its attributes are never, or very rarely changed.

Syntax:-

```
create table table_name (column_name data_type (size) constraint constraint_name primary key, column_name data_type (size));
```

ex:-

Project Number, EnrolNo, etc.

- Q. Define attribute and entity.
→

Attribute :-

The properties of an Entity in the called as Attributes.

Entity :-

Individual object in a system called as Entity Set.

- Q. Explain Single value and multivalue attribute of ER model.
→ In an Entity Relationship [ER] model attributes represent the properties or characteristics of an entity. These attributes can be categorized into two types:
Single - valued & multi valued attributes.



1. Single - Valued Attributes:

- A single valued attribute is one that holds a single value for each entity instance.
- For example, consider the attribute Age of a person entity. A person typically has only one age at a time, making Age a single-valued attribute.

Example:-

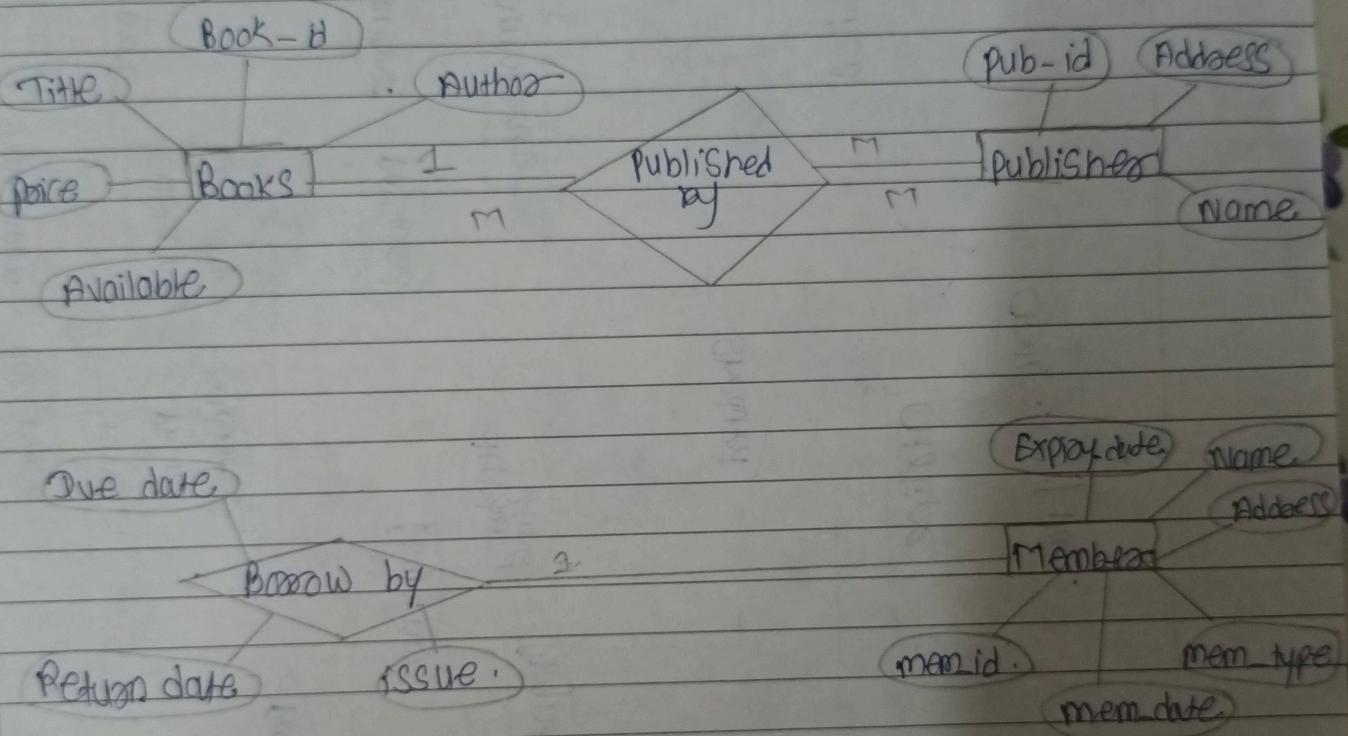
- Entity : person
- Attribute : Age.

2. Multi - Valued Attributes:

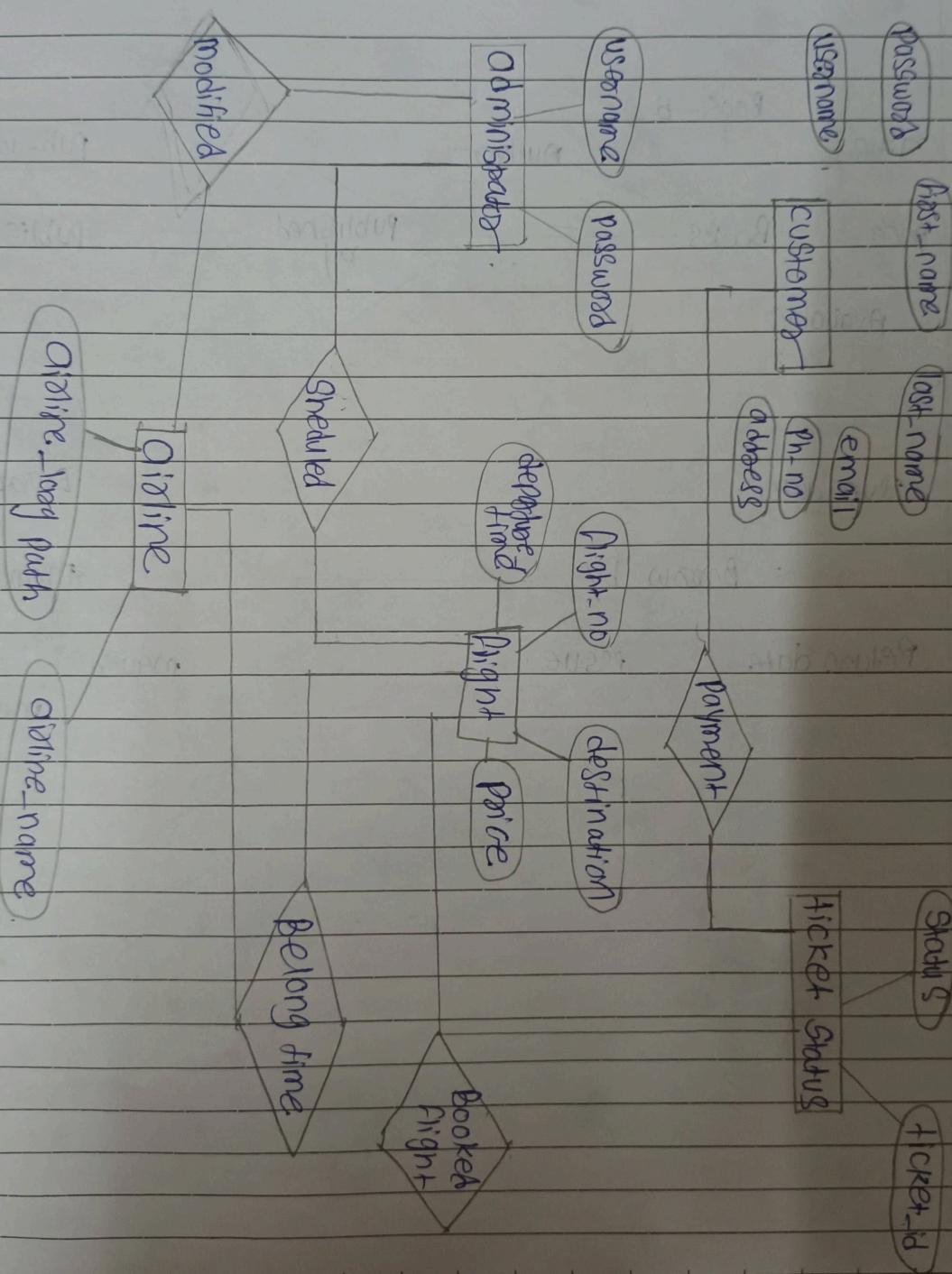
- A multi-valued attribute can hold the attribute multiple values for a single entity instance.
- For ex
- Example:-
- Entity : person
- Attribute : phone # Numbers.

22

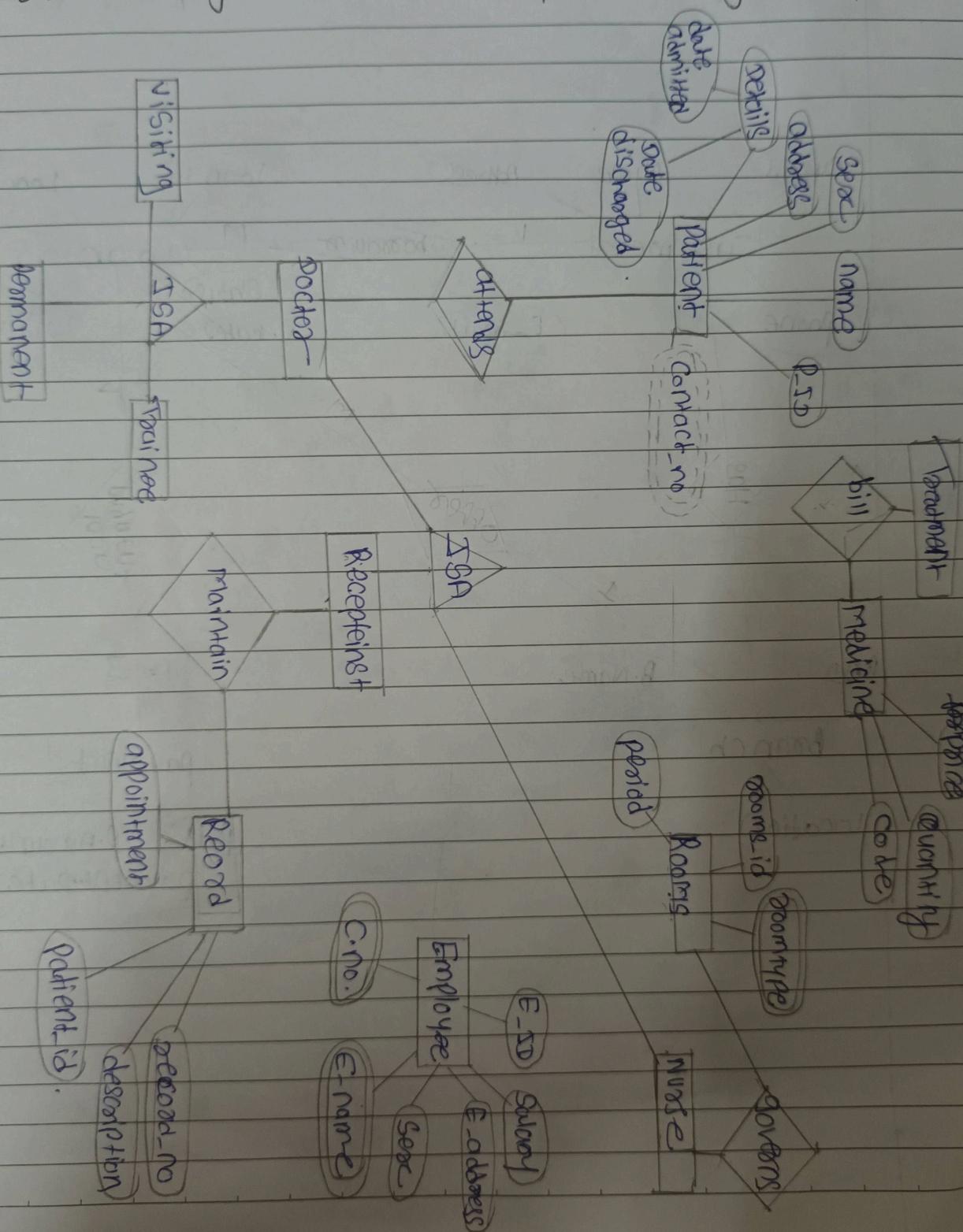
Q. Draw an ER diagram for library management system.



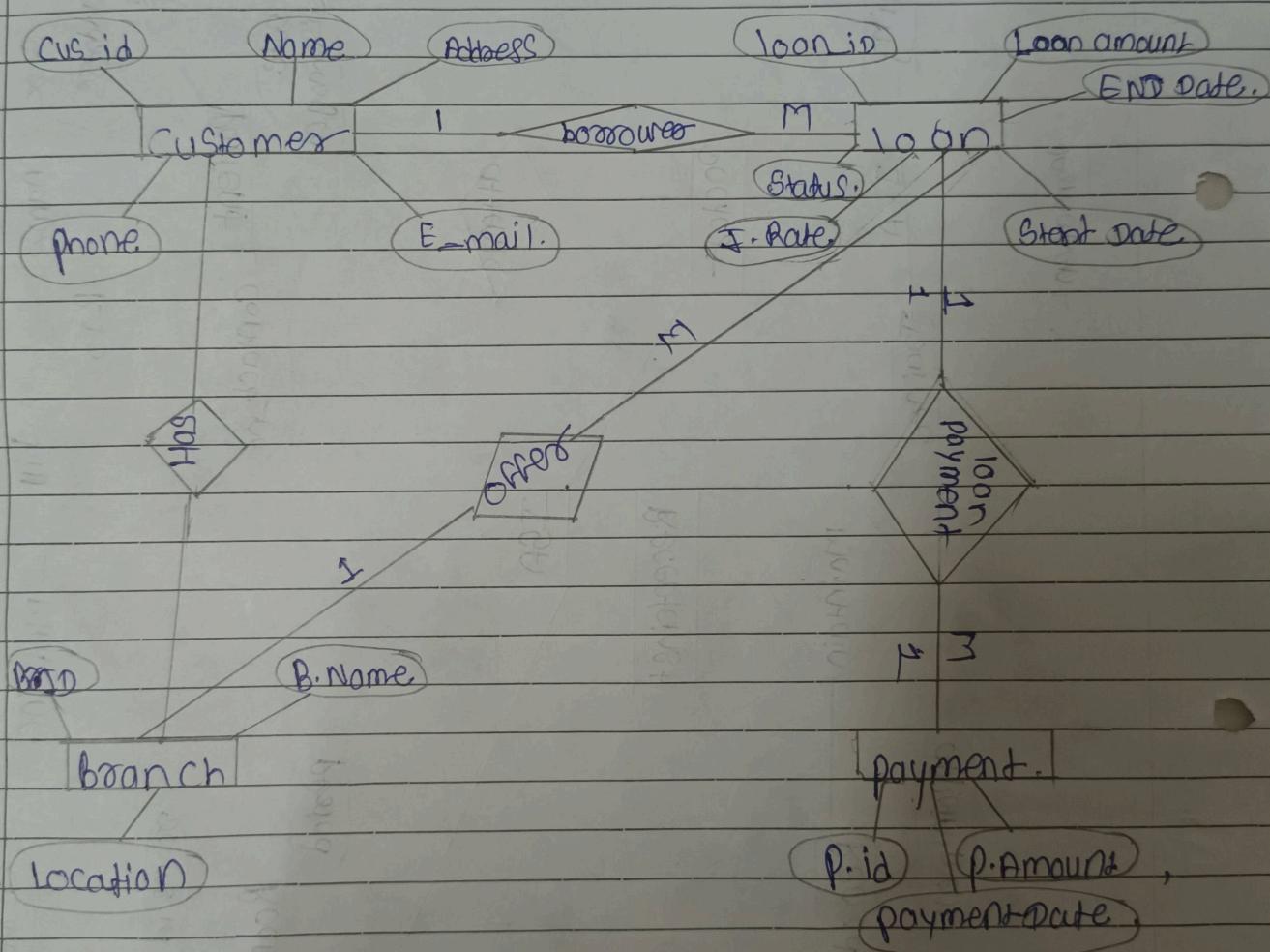
g. E-R diagram for Airline reservation System.



11. ER diagram for Hospital management system.

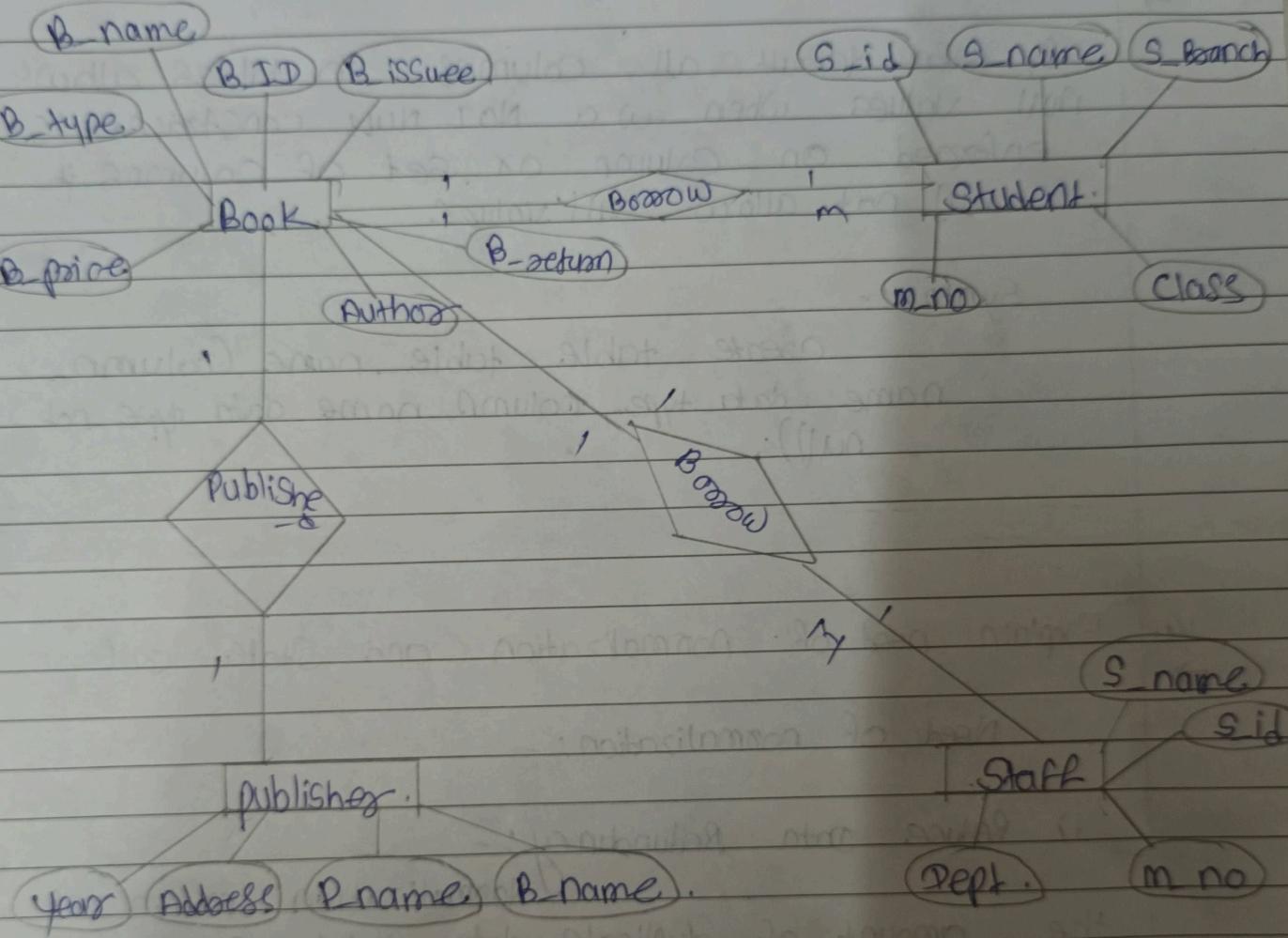


10. ER diagram for Customer & loan System



12. E-R diagram for library management system.

31



How to apply NOT NULL constraint at the table creation? Give syntax.

By default all columns in tables allows null values when a NOT NULL constraint is enforced on column or set of columns it will not allow null values.

Syntax:-

```
create table table_name (column_name data_type, column_name data_type not null);
```

Explain need of normalization and State 3NF

Need of normalization:-

i) Reduce data redundancy:-

Normalization minimizes the amount of duplicate data, which save storage space & enhance data consistency.

ii) Improve data integrity:-

By organizing data into related tables, normalization helps maintain accuracy and integrity, making it easier to enforce data constraints.

iii) facilities Maintenance:-

A well normalized database is easier to update and manage. Changes in one part of the database do not require changes in multiple places.

• Third Normal Form (3NF):

A table is an 3NF if:

- i) It is in 3NF if it is Second Normal Form (2NF).
- ii) It has no transitive dependencies; that is, all non-key attributes are dependent only on the primary key and NOT on other non-key attributes.
- In simpler term, every piece of information in a 3NF table should relate directly to the primary key, ensuring that there are no indirect relationships that could lead to data anomalies.

15. State and explain 1NF & 2NF with example.

1NF definition:-

The term first normal form (1NF) describes the tabular format in which:

- All the key attributes are defined.
- There are no repeating groups in the table.
- All attributes are dependent on the primary key.

Ex:-

1NF:

→ This rules defines that all the attributes in a relation must have atomic domains.

Course	Content
programming	Java, C++
web	HTML, PHP, ASP

unorganized relation

We re-arrange the relation (table) as below, to convert it to first Normal form.

Course	Content
programming	Java
programming	C++
web	HTML
web	PHP
web	ASP

Each attribute must contain only single value from its pre-defined domain.

2NF →

conversion to second normal form.

Starting with the 1NF format, the database can be converted into the 2NF format by writing each key component on a separate line and then writing the original key on the last line and writing the dependent attributes after each new key.

eg:-

let's assume, a school can store the data of teachers and the subjects they teach. In a school, a teacher can teach more than one subject.

Teacher table.

Teacher ID	Subject	Teacher Age
25		
25	Chemistry	30
25	Biology	30
47	English	35
83	Math	38
83	computer	38

In the given table, non prime attribute Teacher age is dependent on Teacher id which is proper subset of candidate key. That's why it violates the rule of 2NF.
To convert the given table into 2NF, we decompose it into two tables.

Teacher detail table:

Teacher_id	Teacher_age.
25	30
47	35
83	38

Teacher subject table:

Teacher_id	Subject.
25	chemistry
25	Biology
47	English
83	Math
83	Computer

16. Explain 3NF with example:

→ A table is in 3NF if:

- It is in 2NF and
- It contains no transitive dependencies.

Student detail

Stud_stuid	Stu_name	city	zip
------------	----------	------	-----

primary key

Relation Not in 3NF.

- we find that in above depicted Student detail relation, stu_id is key and only primary key attribute.
- we find that city can be identified by stu_id as well as zip itself. Neither zip is a Super key or nor city is a prime attribute.
- Additionally, stu_id \rightarrow city \rightarrow zip, so there exists transitive dependency.

Student_detail

stu_id	stu_name	zip
zipcodes		
zip	city	
		Relation in 3NF

we broke the relation as above depicted two relations to bring it into 3NF.

17. Explain normalization with example.

- Normalization is a process of assigning attributes to entities. It reduces data redundancies and helps eliminate the data anomalies.
- Normalization works through a set series of stages called normal forms.
 - First normal form
 - Second normal form.
 - Third normal form.
- Normalization is the process of minimizing redundancy from a relation or set of relations.

Redundancy form a relation or a set of relations or a set of relations. Redundancy in relation may cause insertion, deletion and update anomalies. So, it helps to minimize the redundancy in relation.

- Normalization is the process of organizing the data in the database.

- Normalization is used to minimize the redundancy from a relation or a set of relations. It is also used to eliminate undesirable characteristics like insertion, update and delete anomalies.

18

18. Explain domain constraint with example.
(NOTNULL, check)

→ NOT NULL:

By default all columns in table allows null values when a NOT NULL constraint is enforced on column or set of column it will not allow null values.

e.g.: SQL > create table Student (Roll_no number(2), Name varchar(20) NOT NULL);

Check:

The constraint defines a condition that each row must satisfy. A single column can have a multiple check condition.

19. List & explain the types of integrity constraints in detail.

→

i) primary key constraint:-

It is used to avoid redundancy duplicate value entry within the row of specified column in table. It restrict null values too.

ii) unique constraint:-

The unique constraint uniquely identifies each record in database table. The unique & primary key constraint both provide a guarantee for uniqueness for a column or set of column.

iii) foreign key constraint:-

A foreign key is a ~~flat~~ column that is used to join a table to other tables to ensure referential integrity of the data the primary key column on a specific dimension table.

iv) check:-

The constraint defines a condition that each row must satisfy. A single column can have a multiple check condition.

v) Null & NOT null:-

By default a column can hold null values the Null constraint enforces a column to not accept null values constraint is enforced on column or set of column it will not allow null values.