SE-Computer-Div-A	Roll number : 9913
Experiment no. : 1	Date of Implementation :30/1/2023
Related Course outcome : At the end of the course, Students will be able to design ER model and develop relational model	

Rubrics for assessment of Experiment:

Indicator	Poor	Average	Good
Timeliness ● Maintains assignment deadline (3)	Assignment not done (0)	One or More than One week late (1-2)	Maintains deadline (3)
Completeness and neatness ● Complete all parts of ER diagram(3)	N/A	< 80% complete (1-2)	100% complete (3)
Originality ● Extent of plagiarism(2)	Copied it from someone else(0)	At least few questions have been done without copying(1)	Assignment has been solved completely without copying (2)
Knowledge ● In depth knowledge of the assignment(2)	Unable to answer 2 questions(0)	Unable to answer 1 question (1)	Able to answer 2 questions (2)

Assessment Marks :

Timeliness	
Completeness and neatness	
Originality	
Knowledge	
Total	

Total : (Out of 10)

Teacher's Sign :

Name Student	Mark Lopes	Roll No.	9913
Lab Experiment No.	1	Date	30/1/2023
Expt. Title	Write Problem Definition and Draw ER /EER diagram		

Aim : Problem Definition and draw ER /EER diagram

Objective of the Experiment:

1. To design/draw ER/EER for the selected problem

Theory:

The **entity-relationship (E-R)** data model is based on a perception of a real world that consists of a set of basic objects called **entities**, and of **relationships** among these objects.

The model is intended primarily for the database-design process. It was developed to facilitate database design by allowing the specification of an **enterprise schema**. Such a schema represents the overall logical structure of the database. This overall structure can be expressed graphically by an **E-R diagram**.

An **entity** is an object that exists in the real world and is distinguishable from other objects. We express the distinction by associating with each entity a set of attributes that describes the object.

A **relationship** is an association among several entities. The collection of all entities of the same type is an **entity set**, and the collection of all relationships of the same type is a **relationship set**.

Mapping cardinalities express the number of entities to which another entity can be associated via a relationship set.

A **superkey** of an entity set is a set of one or more attributes that, taken collectively, allows us to identify uniquely an entity in the entity set. We choose a minimal superkey for each entity set from among its superkeys; the minimal superkey is termed the entity set's **primary key**. Similarly, a relationship set is a set of one or more attributes that, taken collectively, allows us to identify uniquely a relationship in the relationship set. Likewise, we choose a minimal superkey for each relationship set from among its superkeys; this is the relationship set's primary key.

- An entity set that does not have sufficient attributes to form a primary key is termed a **weak entity set**. An entity set that has a primary key is termed a **strong entity set**.
- **Specialization** and **generalization** define a containment relationship between a higher-level entity set and one or more lower-level entity sets. Specialization is the result of taking a subset of a higher-level entity set to form a lower-level entity set. Generalization is the result of taking the union of two or more disjoint (lower-level) entity sets to produce a higher-level entity set. The attributes of higher-level entity sets are inherited by lower-level entity sets.
- **Aggregation** is an abstraction in which relationship sets (along with their associated entity sets) are treated as higher-level entity sets, and can participate in relationships.
- The various features of the E-R model offer the database designer numerous choices in how to best represent the enterprise being modeled. Concepts and objects may, in certain cases, be represented by entities, relationships, or attributes. Aspects of the overall structure of the enterprise may be best described by using weak entity sets, generalization, specialization, or aggregation. Often, the designer must weigh the merits of a simple, compact model versus those of a more precise, but more complex, one.
- A database that conforms to an E-R diagram can be represented by a collection of tables. For each entity set and for each relationship set in the database, there is a unique table that is assigned the name of the corresponding entity set or relationship set. Each table has a number of columns, each of which has a unique name. Converting database representation from an E-R diagram to a table format is the basis for deriving a relational-database design from an E-R diagram.
- The **unified modeling language (UML)** provides a graphical means of modeling various components of a software system. The class diagram component of UML is based on E-R diagrams. However, there are some differences between the two that one must beware of.

(Sample Problem statement-BANKING)

The initial specification of user requirements may be based on interviews with the database users, and on the designer's own analysis of the enterprise. The description that arises from this design phase serves as the basis for specifying the conceptual structure of the database. Here are the major characteristics of the banking enterprise.

- The bank is organized into branches. Each branch is located in a particular city and is identified by a unique name. The bank monitors the assets of each branch.

- Bank customers are identified by their *customer-id* values. The bank stores each customer's name, and the street and city where the customer lives. Customers may have accounts and can take out loans. A customer may be associated with a particular banker, who may act as a loan officer or personal banker for that customer.
- Bank employees are identified by their *employee-id* values. The bank administration stores the name and telephone number of each employee, the names of the employee's dependents, and the *employee-id* number of the employee's manager. The bank also keeps track of the employee's start date and, thus, length of employment.
- The bank offers two types of accounts—savings and checking accounts. Accounts can be held by more than one customer, and a customer can have more than one account. Each account is assigned a unique account number. The bank maintains a record of each account's balance, and the most recent date on which the account was accessed by each customer holding the account. In addition, each savings account has an interest rate, and overdrafts are recorded for each checking account.
- A loan originates at a particular branch and can be held by one or more customers. A loan is identified by a unique loan number. For each loan, the bank keeps track of the loan amount and the loan payments. Although a loan-payment number does not uniquely identify a particular payment among those for all the bank's loans, a payment number does identify a particular payment for a specific loan. The date and amount are recorded for each payment.

In a real banking enterprise, the bank would keep track of deposits and withdrawals from savings and checking accounts, just as it keeps track of payments to loan accounts. Since the modeling requirements for that tracking are similar, and we would like to keep our example application small, we do not keep track of such deposits and withdrawals in our model.

Step 1- Identify entities of problem stmt

we begin to identify entity sets and their attributes:

- The *branch* entity set, with attributes *branch-name*, *branch-city*, and *assets*.
- The *customer* entity set, with attributes *customer-id*, *customer-name*, *customer-street*; and *customer-city*. A possible additional attribute is *banker-name*.
- The *employee* entity set, with attributes *employee-id*, *employee-name*, *telephone-number*, *salary*, and *manager*. Additional descriptive features are the multivalued attribute *dependent-name*, the base attribute *start-date*, and the derived attribute *employment-length*.

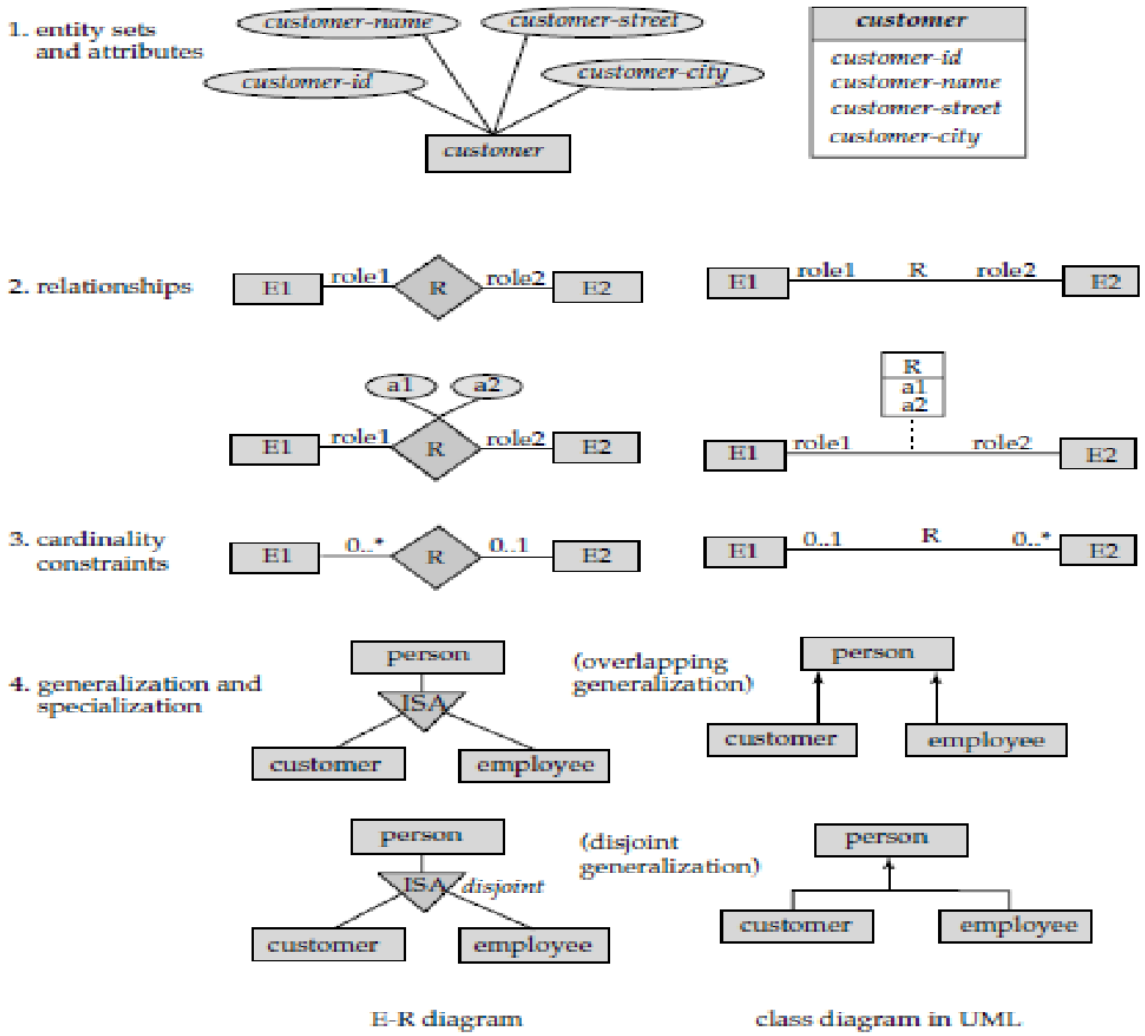
- Two account entity sets—*savings-account* and *checking-account*—with the common attributes of *account-number* and *balance*; in addition, *savings-account* has the attribute *interest-rate* and *checking-account* has the attribute *overdraft-amount*.
- The *loan* entity set, with the attributes *loan-number*, *amount*, and *originating-branch*.
- The weak entity set *loan-payment*, with attributes *payment-number*, *payment-date*, and *payment-amount*.

Step-2 identify relationship with cardinality, type, and participation

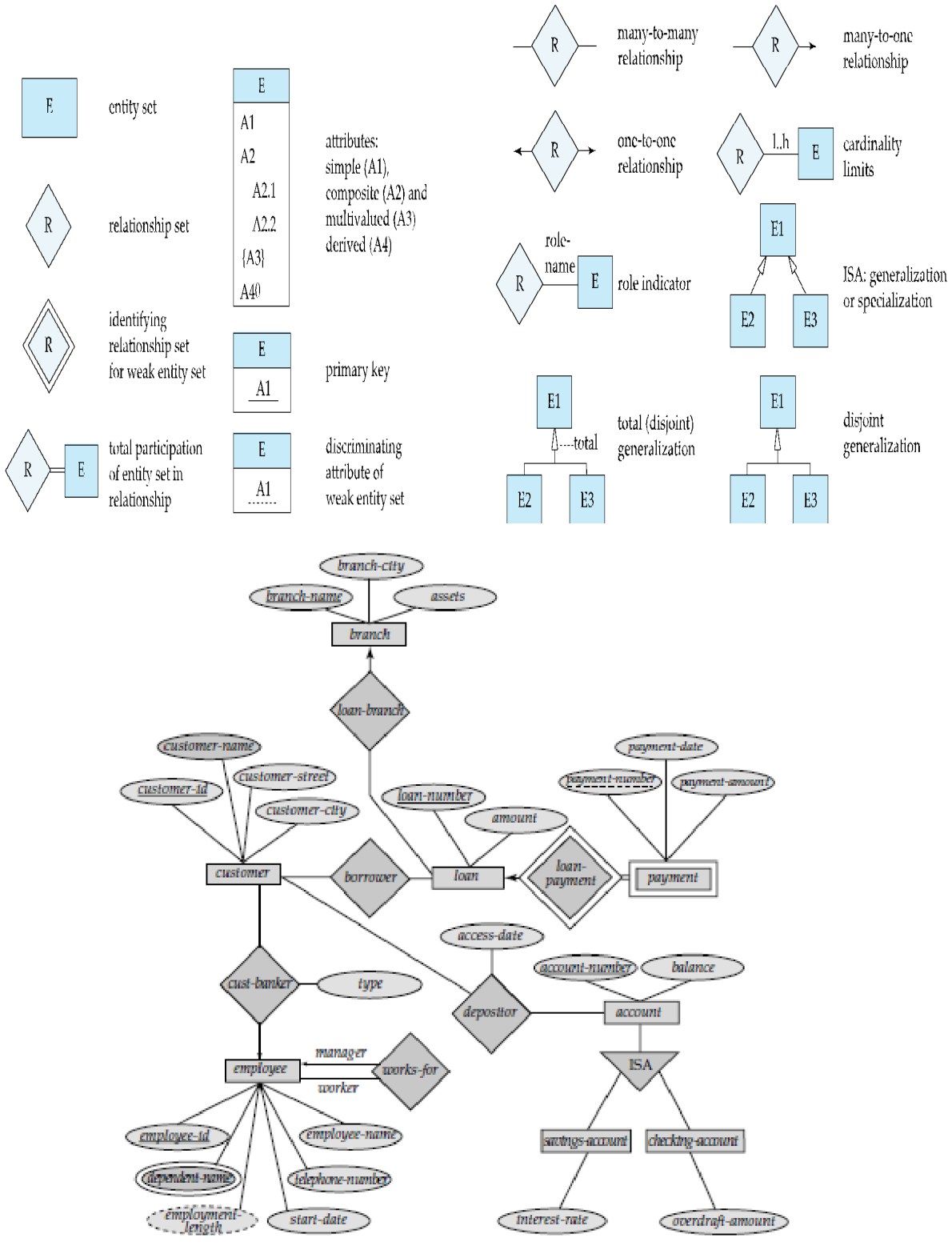
We now return to the rudimentary design scheme of Section 2.8.2.2 and specify the following relationship sets and mapping cardinalities. In the process, we also refine some of the decisions we made earlier regarding attributes of entity sets.

- *borrower*, a many-to-many relationship set between *customer* and *loan*.
- *loan-branch*, a many-to-one relationship set that indicates in which branch a loan originated. Note that this relationship set replaces the attribute *originating-branch* of the entity set *loan*.
- *loan-payment*, a one-to-many relationship from *loan* to *payment*, which documents that a payment is made on a loan.
- *depositor*, with relationship attribute *access-date*, a many-to-many relationship set between *customer* and *account*, indicating that a customer owns an account.
- *cust-banker*, with relationship attribute *type*, a many-to-one relationship set expressing that a customer can be advised by a bank employee, and that a bank employee can advise one or more customers. Note that this relationship set has replaced the attribute *banker-name* of the entity set *customer*.
- *works-for*, a relationship set between *employee* entities with role indicators *manager* and *worker*; the mapping cardinalities express that an employee works for only one manager and that a manager supervises one or more employees. Note that this relationship set has replaced the *manager* attribute of *employee*.

Step-3 use symbols to draw ER/EER model -<u>Old approach symbols-its representation in UML</u>



<u>New Approach notations-as per 6th edition of Korth</u>



ER-Diagram banking example –Old approach

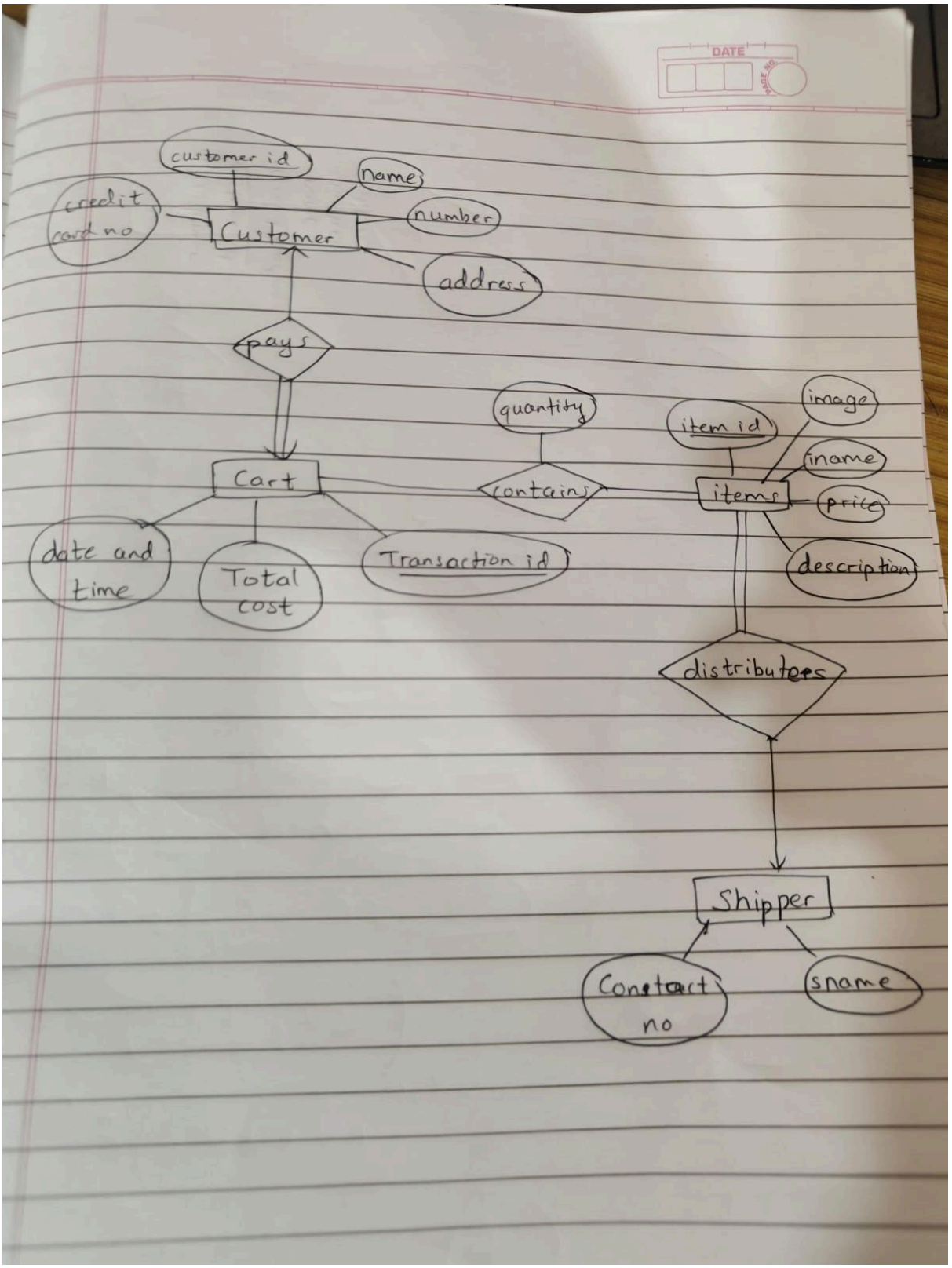
Convert this using New approach

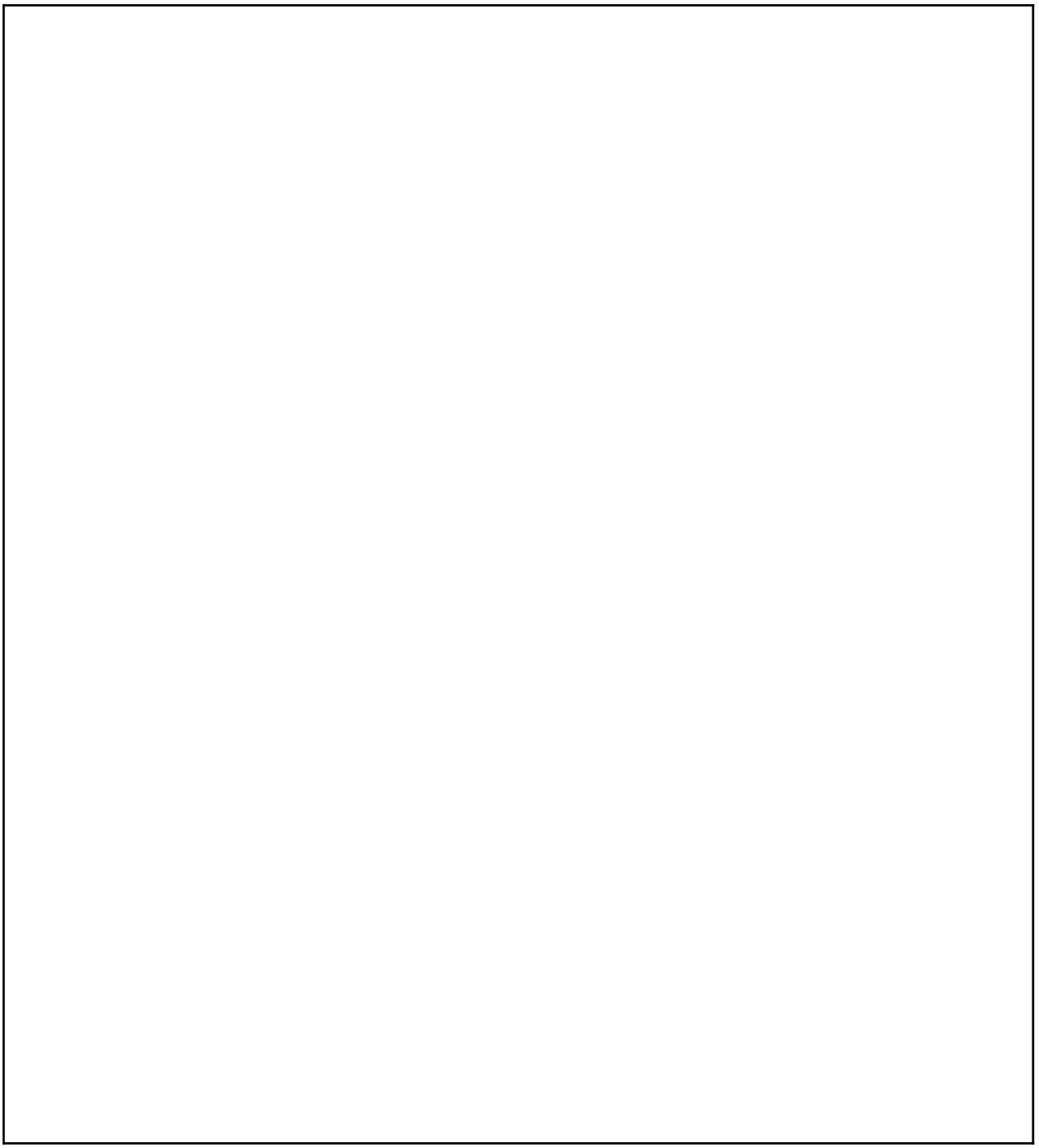
Description of Problem Statement:-NAME OF CASE STUDY- AMAZON

Amazon, a leading e-commerce platform, aims to streamline its order fulfillment process. To achieve this, the company requires a database system to manage customer information, track orders, handle payments, manage inventory, and coordinate shipping.

The system aims to optimize order processing by efficiently managing customer orders, tracking inventory availability, calculating total costs, and coordinating shipping through designated shippers.

E-R /EER diagram for the Problem to be implemented.





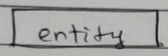
Post Lab Assignment:

1) Describe various symbols used in E-R Diagram and EER diagram

Postlab :-

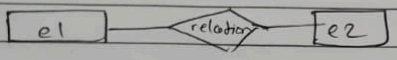
1) Entity

→ It is an object that exists and is distinguishable from other objects.



2) Relation

→ It is an association among several entities.



3) attributes

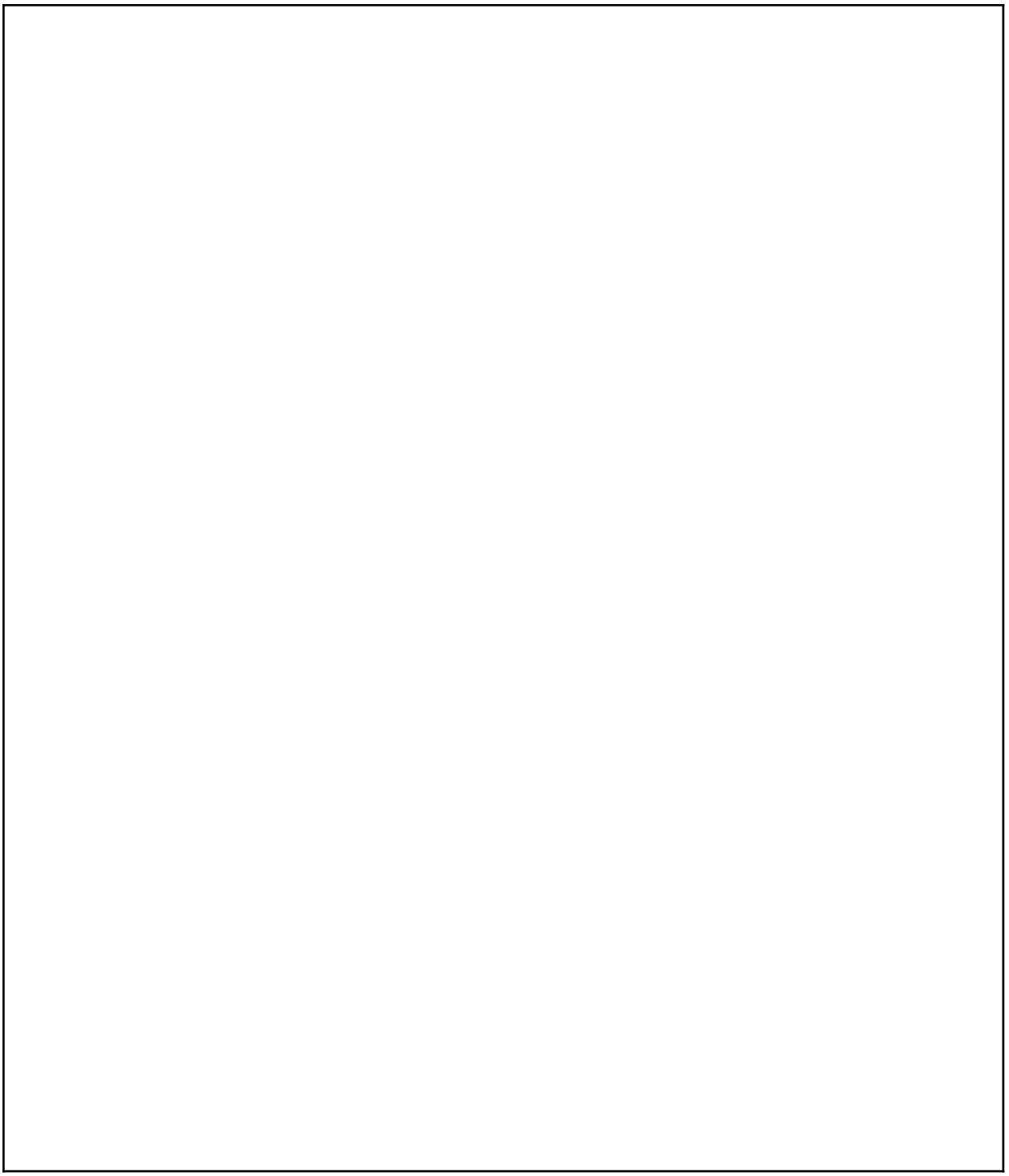
→ They are the properties or characteristic that describes an entity in a database.

4) Primary key

→ It provides a way to specify how entities and relations are distinguished.

5) Weak entity

→ A entity that does not have a primary key because of insufficient attributes is called a weak entity.



List of topics for ER diagram

1. Facebook system
2. Stock exchange syst
3. Wikipedia
4. Youtube
5. Traffic monitoring system
6. Dmart
7. Amazon
8. Twitter
9. Instagram
10. Olx.com
11. Hike
12. Whatsapp
13. Flipcart
14. Yahoo
15. Google search
16. Bio research
17. Bookmyshow
18. Election system- targeting the voters
19. Inventory management system
20. Library management system
21. College information management system
22. Banking system
23. Hospital management system
24. Airline reservation system
25. Railway reservation system
26. Ticket booking system
27. Hotel reservation system
28. Ola.
29. Other topics after the discussion and approval of subject teacher