

Data Warehouse  
&  
Data Mining  
Assignment 1

Question 1 :-

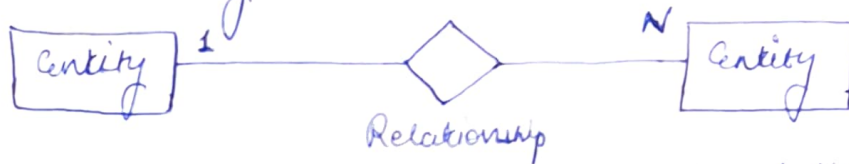
Solution:-

- ① Attributes : They are the characteristics or properties which describes an entity they belong to. Some attributes can have multiple values for a given entity. For example:- a customer could provide several phone numbers, so the telephone number attribute is multivalued. Attributes also help distinguish one entity from other entities of the same type. Here key attribute comes into picture.
- ② Domain : Domains are user-defined types or "attribute templates". They are reusable. User can construct a domain, specify a name, datatype properties, default values and validation rules and then they can be reused.
- ③ Entity : defined as tables that holds specific information are the backbones of an entity - Relationship model. It is an object in the real world that is distinguishable from other objects. For example: Manager of toy department.
- ④ Relationship : It is the way two or more data sets are linked. There is ~~three types~~ three type of relationships that can be found in DBMS:
  - One to One
  - One to Many
  - Many to One

① Entity set: Group of entities collectively is known as an entity set. For example: An entity set of bank accounts.

② Relationship set: A set of relationships of same type. It can also be a unary relationship. For example:- In case of social media friendship: a person is connected with a different person which is in same entity.

③ One to many relationship: One to many relationship between instances of an entity with more than one instance of another entity.



④ Many-to-many: When multiple records in a table are associated with multiple records in another table. For example, customers and products.

⑤ Participation constraint: Specifies the presence of an entity when it is related to another entity in a relationship type. It is also called the minimum cardinality constraint. There are two types of participation constraint:-

- Total Participation.
- Partial Participation.

⑥ Weak entity: The entity sets which do not have sufficient attributes to form a primary key are known as weak entity sets. For example: Employees take insurance policy for their family members.

Question 2 →

Solution:- 1.) If two entities have attributes with exact same name it can create a lot of confusion when someone works on the same. So, a better approach would be to change the name of one of the attributes in any diff of the entity.



- 2) • X is the primary key for A but not B: In this case, we can use X as a foreign key in set B which will connect set B to set A. Also known as referential integrity.
- X is the primary key for both A and B: If two sets have same primary key, it's better to join them together with the help of common primary key.
- X is not the primary key for A nor for B: If X is not primary key for A or B that means there is redundancy in the data which is not a good way to store the data.

Question-3 →

Solution:- • According to my point of view, a weak entity can have a one-to-one relationship. For example: If in a company, employee takes insurance policy for his/her family. If there are multiple dependents of an employee then it's one to many but if there is only one dependent such as spouse it's one-to-one. But, at last it does not make any sense to concretely define the relationship to be one-to-one as there can be multiple dependents.

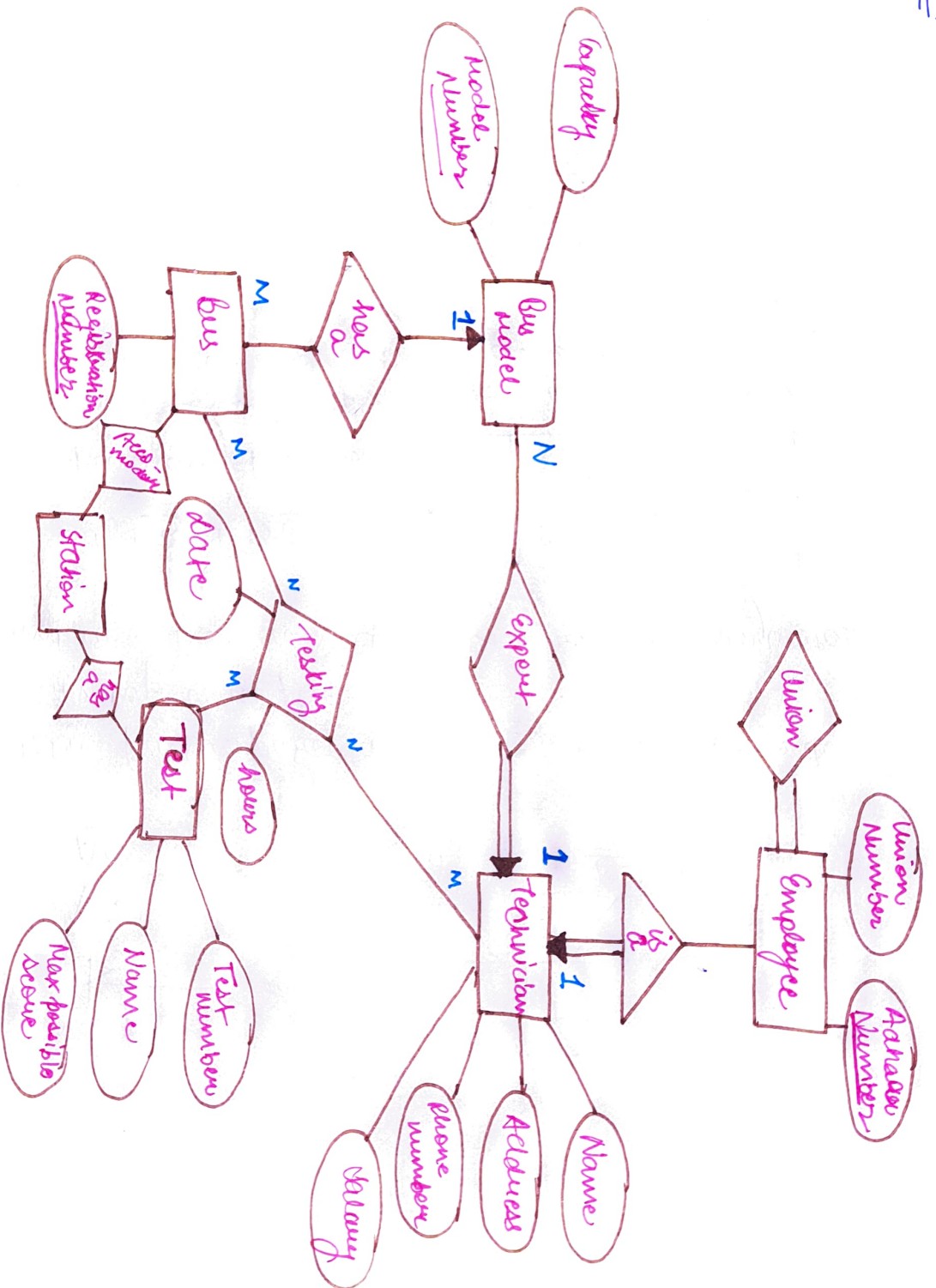
- The concept of weak entity is there in the E-R diagram to encourage the concept of dependency and to reduce the redundancy in the ~~model~~ model. Because if we increase the number of attributes in the weak entity, it increases the chances of data redundancy which is not a good practice.

Question 4 →

Solutions:-

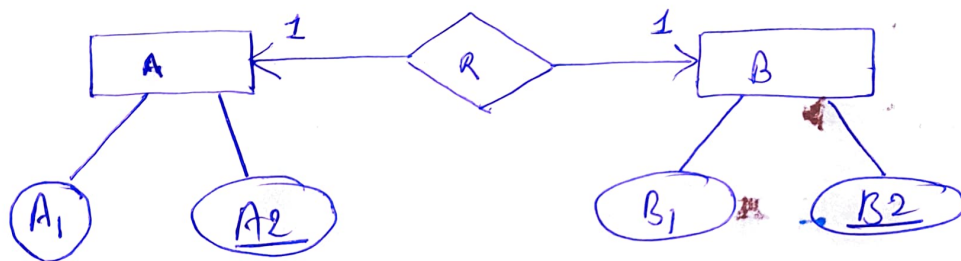
# Question - 4

1.)



2.) In our previous part where we made our E-R diagram we have already shown 'is a' relationship which is a special relationship and it shows that technical is already an expert who will do the testing for the required test cases.

Question - 5. Part A)  
Solution: Case 1:



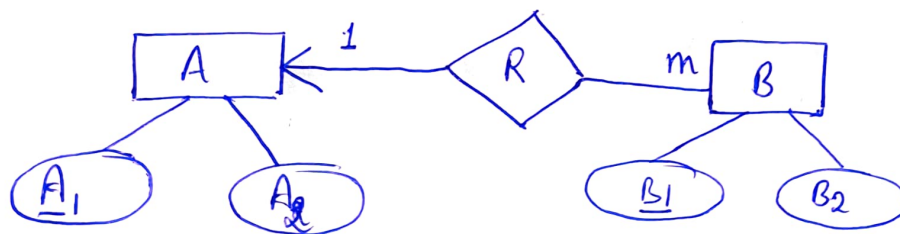
- When total participation is given, we know, one to one relationship is in play.  
 So in that case we can represent schema by single table.

ARB (A1, A2, B1, B2)

A1	A2	B1	B2

⇒ A2 & B2 are primary keys. So, anyone can be chosen as primary key and given that it's unique for every row and others can be foreign key.

Case 2:



Now we are given one-to-many relationship.  
 On combining the entity set B & relationship set R





$A(\underline{A_1}, A_2)$

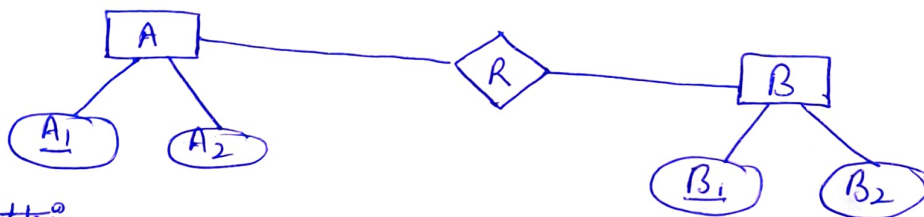
<u>A<sub>1</sub></u>	A <sub>2</sub>

$BR(\underline{B_1}, B_2, A_1)$   $\rightarrow$  Foreign Key

<u>B<sub>1</sub></u>	B <sub>2</sub>	A <sub>1</sub>

Here the total participation will be in play. So, Primary key & Foreign key acquire NOT Null constraint.

Case 3



In this case, we are dealing with many-to-many relationship so three tables will be created.

$A(\underline{A_1}, A_2)$   $\rightarrow$  Primary key

<u>A<sub>1</sub></u>	A <sub>2</sub>

$R(\underline{A_1}, \underline{B_1})$   $\rightarrow$  Primary key

<u>A<sub>1</sub></u>	<u>B<sub>1</sub></u>

$B(\underline{B_1}, B_2)$   $\rightarrow$  Primary key

<u>B<sub>1</sub></u>	B <sub>2</sub>

In this case also, binary key cannot be null

Part-B)

When there is a not total participation from both sides i.e. when there is partial participation from both ends then that means that the foreign key can be null in that case.