**HABIBON, FAIZER A. COURSE & SECTION: BSIT-3C**

1. **Entity Relationship (ER) model**

The conceptual architecture of a database may be built using the Entity Relationship (ER) model, but not its physical structure.  
  
Think about the case when we build a database for a school. In this database, the student will be an entity with attributes like address, name, id, age, etc. The address could be connected to another item that has attributes like a city, street name, pin code, etc.

For instance, a structure. There are certain floors in every building. As an illustration, imagine a firm building with a number of rooms on each level that have labels indicating what staff and workers are responsible for. There is a space where the firm CEO, employees, and others may hold meetings. A student is another illustration. As characteristics, a student object may have a name, age, and class.

1. **Relational model**

Relational model is the most often used one for data in the form of tables, sometimes known as relations since they include columns and rows. Each column contains a list of the entity in question's attributes. One instance of the object in question is described by one row, often known as a tuple.

A relational database is exactly what its name suggests: a tool for storing various types of information that are related to each other in various ways. For example, a relational database for an online store might maintain customer data and maintain related information as well, such as their various addresses, wish lists, orders, etc.

A relational model user is someone who teaches in a university or school and keeps track of their students' names, birthdays, addresses, phone numbers, and email addresses using tables with rows and columns or excel spreadsheets. This makes it easier for teachers and students to communicate, especially when using online resources.

Pros:

* structural independence
* heightened mental clarity
* A database should be simple to create, manage, and utilize.
* Redundancy in data is seldom present.

Cons:

* the potential for poor execution and design.
* It is more expensive than other models.

1. **Network model**

The network model improves on the hierarchical approach by allowing many to many relationships between linked records, which implies countless parent records.  
  
pros:

* It is fast data access with a network model.
* It is more adaptive since it recognizes many-to-many connections.
* Searching is facilitated with multidirectional pointers.

cons:

* The database design is complex.
* Lack of structural independence

Example: Finance Department. Department of Finance (DOF) is the government's steward of sound fiscal policy. It formulates revenue policies that will ensure funding of critical government programs

Department store conytais employees

1. **Hierarchical Model**

The earliest model is the Hierarchical Model, which was created by IBM. The hierarchical approach arranges data into a tree-like structure, with each record having a single parent or root. Many relationships in the actual world may be accurately described using this paradigm.  
  
pros:

* It is among the simplest database models.
* If a parent is known, searching is quick and simple.
* It supports both one-to-one and one-to-many partnerships.

cons:

* It is a dated, conventional design.
* It does not support many-to-many relationships.
* Due to interdependence, when a parent node is destroyed, all of the child nodes are also erased.

Order processing real life example is Jollibee and mcdonalds. As an instance, the parent table is the Customer table. The client table, which displays which customers are utilizing which computers, is referred to in the computer table.