Data Structure

**Session : Data types, data structures, Abstract data types**

* **Introduction – topics outline**
  + Uses of data, information.
  + Introduction to Data Structure.
  + Introduction to abstract data types.
* **Data**
  + What is the most important factor for any event/task to take place?
* Any task or job done involves the movement/ processing of data.
* Definition : data is a collection of raw facts and figures. Ex. Number(10,20), Word(“apple”).
* What is Data?
  + Data means unprocessed information.
  + It doesn’t have meaning of its own until it is organized.
  + Ex. 90, 85, 70 – just numbers (data).
  + When we say “marks of student” id becomes information.
* Uses of Data.
  + To store information in computer (like student records, photos, videos).
  + To analyse and make decisions (sales data – which product sells more).
  + To reduce errors
  + To build applications
* Data is Expensive !!!
* In the context of an organization, data are the building blocks that, when aggregated, analyzed, and interpreted, becomes information.
* **Data management**
  + Definition : data management is the process of collecting, storing, organizing, and using data efficiently.
  + It makes sure data is accurate, available, and secure.
* **Why important?**
  + Without management -> data becomes messy and useless.
  + With management -> we can easily find, update, and use data.
* **Use of data management**
  + Helps id decision making.
  + Reduces redundancy (avoids storing the same data many times).
  + Increases security (only authorized people can access data).
  + Improves efficiency (faster access and updates).
* Ex.
  + **In a Banking Ststem:**
    - Customer info, transactions, loans –> all managed in a database.
    - Data management ensures no duplication, safe storage, and quick access.
* **Data structure in programming**
  + **Need of knowledge of data structures.**
    - Programs handle a lot of data -> we need a systematic way to store and use it.
    - Good data structures make programs **faster, memory-efficient, and easier to manage.**
    - Ex. Searching in a stored list (binary search) is faster than in an unsorted one.
  + **Data representation**
    - Data can be stored in **different formats** inside memory:
      * Primitive types : int float, char,etc.
      * Non-primitive types: arrays, linked list, trees, graphs.
    - Representation decides **how data is stored and accessed.**
  + **Relationships Among Data:**
    - Data items are not always isolated -> they often relate.
      * Ex. Student -> roll no, name, marks
    - Data structures define **how elements are connected:**
      * Sequential (array, list).
      * Hierarchical(tree).
      * Networked(graph).
  + **Organization of data:**
    - Organizing data properly = easier operations.
    - Common organizations:
      * **Linear –** elements in a sequence (array, stack, queue).
      * **Non-liner –** elements connected different ways (tree, graph).
    - Proper organization improves **search, insertion, deletion** performance.
  + **Definition of data structure**
    - A data structure is a way to store, organize, and manage data in computer memory so it can used efficiently.
    - Ex.
      * Array – stores elements In continuous memory.
      * Linked list – stores elements with pointers to next node.
  + **Rules for Data Access**
    - Each data structure has its own **rules to access and modify data:**
      * Array – access by index.
      * Stack – LIFO (Last in, First out).
      * Queue – FIFO (first in, first out).
      * Tree – follow parent-child links.
      * Graph – follow edges between nodes.

->>> Data Structure = Organized Data + Allowed Operations