DSA in JavaScript

- ❖ **Data Types**

  Data types are classifications of data that determine what kind of value a variable can hold and what operations can be performed

1. Primitive Data Structures

- **String**: Sequence of characters, e.g., "hello".
- **Number**: Numeric value, e.g., 42, 3.14.
- **Boolean**: true or false.
- **Undefined**: A variable declared but not initialized.
- **Null**: Represents an explicitly empty or non-existent value.
- **Symbol**: Unique and immutable values (ES6).
- **BigInt**: For large integers beyond Number limits (ES11).

2. Non-Primitive (Complex) Data Structures

- **Arrays**: Ordered collection of elements, accessible by index.
- **Objects**: Key-value pairs that represent real-world entities.
- **Sets**: Collection of unique values, with no duplicates.
- **Maps**: Key-value pairs with the ability to store any type of key.
- **WeakSets**: Like sets, but stores only objects and allows garbage collection.
- **WeakMaps**: Like maps, but only stores objects as keys and allows garbage collection.

3. Custom Data Structures

- **Linked List**: A sequence of nodes where each node contains data and a reference to the next node.
- **Stack**: Follows the Last In, First Out (LIFO) principle; operations are done from one end (top).
- **Queue**: Follows the First In, First Out (FIFO) principle; elements are added to the rear and removed from the front.
- **Hash Table**: Stores key-value pairs using a hash function for efficient lookups.
- **Binary Tree**: A hierarchical data structure where each node has at most two children.
- **Trie**: A tree-like data structure used for storing strings or sequences where nodes represent characters.
- **Graph**: A collection of nodes (vertices) connected by edges; can be directed or undirected, weighted or unweighted.