8 days:

**Day 1**

**Data structure and algorithm using JavaScript and Database using MySQL**

Data structure : Data Structure are specialized format for organizing and storing data(it can be number, string, Boolean, collection of data, object data) to perform operation (add, remove, search, sort, iterate etc) in efficient way.

In JS data types. Data types is a type of data which tells what type of data it can hold.

Primitive data types : it can hold only value

number(int and float), boolean type, character

Non primitive or reference data types: it can hold value as well as reference of another data types.

Array, stack, queue, list, set, map etc.

Reference data types

Linear data structure or data type

It is use to store the data in sequence manner. Each element connected with each other.

Example : array, stack, list and queue

Simple relationship

Non Linear data types : in Non linear data structure elements are not arranged in sequentially.

Example : Tree and Graph

Complex relationship

Algorithms

Searching technique

Sorting technique

**array:**

Array is reference data types. In JS array is known as dynamic memory. Array use index concept to store the data.

Array store same types values in C,C++ ,C# and Java etc

But in JS and Python it allow store same as wells different data types.

In JS we can create the array using

Literal style

Using new keyword with pre defined objects.

**Set :** Set is a pre defined object also known as linear data structure. Set is use to store unique values. Set doesn’t provide index position concept.

**Map :** Map is pre defined object also known as linear data structure. Map is uses to store key-value pairs. Key must be unique and value can be duplicate.

**Object :** it is a type of reference or non primitive data types. Which allow to more than information in key-value pairs.

Person

Bank

Customer

Product

Order

etc

**Stack :** A Stack is a linear data structure which support features as First In Last Out (FILO) or LIFO(Last In First Out) concept.

In Stack we add the elements (using push) concept from top to bottom.

Stack Operation

* Push : to Add the element
* Pop: to remove top most element or last element added in stack
* Peek : view top most element or last element added
* isEmpty: Stack is empty (**under flow**)
* size -🡪 check number of element in stack container
* base upon size we can say stack full : **overflow**

Example : Books, Plate, CD player etc.

Real time example :

Web Browser history :forward and backward option

Word file : undo and re-undo

Expression execution : pre-fix, post-fix, in-fix etc.

In JS if want to achieve Stack function we take the help **array** and we can perform stack operation.