containers: - container is a object that stores a collection of other objects (elements).

Containers manages the storage space for its elements & provides members functions to access them(mostly through iterators(objects that behaves like pointers ) )

example: - stack,queue,vector etc.

type of containers: -

* **sequence containers**- array, vector(dynamic array that grows in size ), dequeue, forward list , list .
* **associative containers**- set, map , multiset, multimap
* **unordered associative containers** -unordered set, unordered map , unordered multiset, unordered multiset.
* **containers adaptors** – stack,queue,prority queue

**Array**

All STL containers are passed by value.

To pass a stl object we’ll have to pass it by reference: -array<int,6> &ar

//**const** used for creating read only functions/data members and make sure function doesn’t end up modify data.

**//for each loop**

for(int a: arr){

cout<<a <<" ";

}