**Course 2 - Backend and Database Development**

**Day 10: 3 Jan 2025**

**Collection Framework**

Collection framework provided set of pre defined classes and interfaces which help to add the collection of elements or data or item or object of any type like int, float, char, string or any user defined objects.

It provided lot of pre defined which help to add, remove, search, iterate very easily.

Normal variable.

int a=10;

a=20;

array

int abc[]={10,20,30,40}

int num[]=new int[10]; array is known as fixed in memory. It allow to store same data type values.

class Employee {

int id;

String name;

float salary;

}

Employee emp= new Employee();

emp.id=100;

emp.name=”Steven”;

emp.salary=14000;

array objects

Employee employees[]=new Employee[100];

employees[0]=new Employee();

employees[1]=new Employee();

Collection Framework Hierarchy : all classes and interfaces part util package.

Collection --🡪 interface

Set List Queue Map

All four are interfaces. Set, List and Queue internally extends Collections. But Map doesn’t extends Collection.

Set : it is use to store set of element or item or data. Set doesn’t allow duplicate. The element under the set may be maintain order, unorder or sorted. Set doesn’t provide index position.

Set classes

HashSet, LinkedHashSet and TreeSet. These are classes internally extends Set interface.

HashSet : it store the element in unorder format.

LinkedHashSet : it maintain the order. LinkedHashSet internally extends HashSet. It doesn’t provide any its own methods. Only it maintain the order.

TreeSet : sorted by default. In TreeSet we need to store same data type values. TreeSet internally implements SortedSet interface and that interface internally extends Set interface.

List : it is use to store list of items. It maintain the order using index position. List allow duplicate.

List classes

ArrayList

ArrayList is a type of List class. which provides set of methods which help to add, remove, search and iterate very easily.

Normal Array

* Fixed in memory size.
* It allow to store homogeneous elements or same data types.
* It doesn’t provide any pre defined method to add element in between or remove element in between.

ArrayList

* Dynamic in memory.
* It allow to store homogeneous as well as heterogeneous elements.
* It provided lot of pre defined methods we can add the element in between as well as we can remove the elements.

LinkedList : LinkedList is a type of data structure which internally use node concept to store the data.

10 nextRef 20 nextRef 3 nextRef🡪null

preRef 10 nexRef preRef 20 nextRef preRef 30nextRef

Types of linked list

1. Single linked list
2. Double linked list
3. Circular single linked list
4. Circular double linked list

Arraylist is good if we want to retrieve the element using loop

But linked list is good if we do more insertion and deletion operation.

Vector : Vector is known as legacy class. By default all methods in Vector class are synchronized. Thread safe (because only one thread execute). But slow in performance.

Stack

These all classes internally implements List interface. Stack is a type of data structure which provide a features as FILO or LIFO

First in Last out of Last In First Out.

Stack operation

Push 🡪 Add the elements

Pop 🡪 Remove top most elements

Peek 🡪 Check top most elements but not remove from a stack.

Search 🡪 To check the elements.

400 push

300 push

200 push

100 push

Pop() : it remove top most or last added element ie 400. Peek() it display top most element ie 300 it doesn’t remove.

Search element search from top to bottom. If element present it display the position of element else it return -1.

Queue: First in First out. It allows duplicate.

PriorityQueue : First In First Out base upon priority ie lower value.

LinkedList : First In First Out no priority.

Map : it is use to store the information in the form of key-value pairs. Key must be unique and value may be duplicate. Using key we can get the value.

HashMap, LinkedHashMap, TreeMap and Hashtable

These classes internally implements Map interface.

HashMap -🡪 doesn’t maintain the order. We can store different data types values.

LinkedHashMap 🡪 maintain the order

TreeMap 🡪 Sorted using key by default ascending order. So key must be same data types.

Hashtable 🡪 by default methods are synchronized.