**Course 2 - Backend and Database Development**

**Day 11: 7 Jan 2025**

**Collection Framework – Continue**

**Retrieve the element or data from collection classes.**

1. For each loop : type of loop
2. Iterator interface : only forward direction
3. ListIterator interface : forward as well as backward direction.

Set family

HashSet hs = new HashSet();

hs.add(10); // auto-boxing : converting primitive to object.

hs.add(20);

hs.add(30);

System.out.println(hs);

for(Object obj:hs) {

System.out.println(obj);

}

Iterator is a interface which provided set of method which help to retrieve the element from set, list or queue one by one. The internal implementation of for each loop is base upon Iterator.

Iterator li = hs.iterator();

iterator() is a method part of collection. And this method return type is Iterator interface reference. li.next() method check element is present or not. If present it return true and next() method get that elements.

We can’t retrieve value from Map using for each loop, iterator or listIterator.

Map provided one of the pre defined method as keyset(), it retrieve all keys from map as set.

Collection Framework with generics

CollectionClass<Type> object = new CollectionClass<Type>();

Here type can be Integer, Float, Character, Double, String, or user defined object.

List<Integer> ll = new ArrayList<Integer>();

Collection Framework with user defined objects.

Set<Integer> ss1 = new HashSet<Integer>();

Ss1.add(10);

Ss1.add(10);

System.out.println(ss1.size()); 1

Set<Account> ss = new HashSet<Account>();

Account acc1 = new Account(100,”Steven”,1000);

Account acc2 = new Account(100,”Lex”,1200);

ss.add(acc1);

ss.add(acc2);

System.out.println(ss.size()); 2