Practical 9: Introduction to collection framework

Name: Sutariya Savankumar

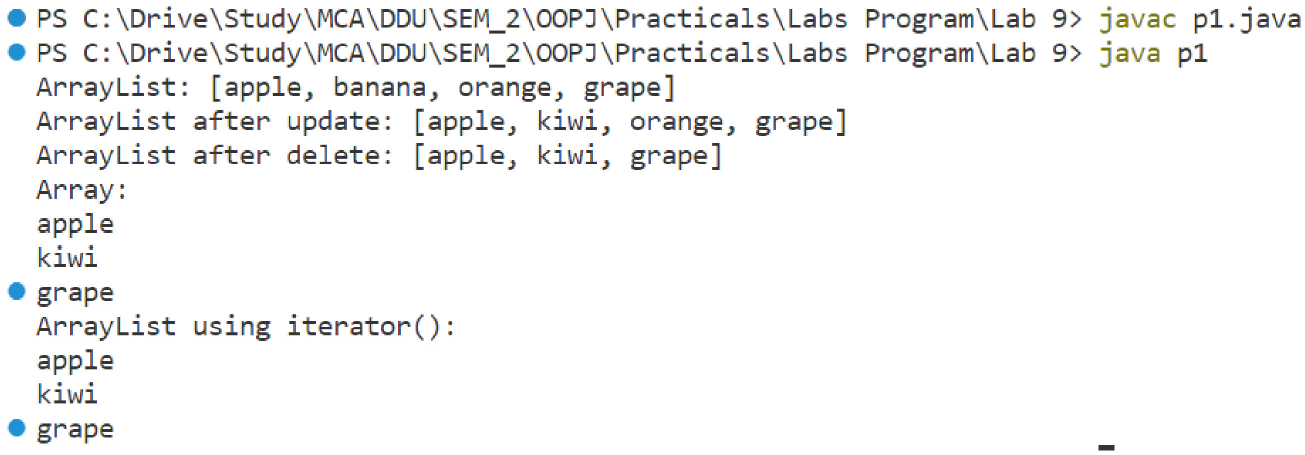
Roll no: MA065

1. **Write Create an ArrayList and perform following operations for that:**
    a. **Add**
    b. **Update**
    c. **Delete**
    d. **Convert in to array**
    e. **Display using iterator()**

Code

```
import java.util.ArrayList;
import java.util.Iterator;

public class p1 {
    public static void main(String[] args) {
        // Create an ArrayList of strings
        ArrayList<String> fruits = new ArrayList<String>();
        // Add elements to the ArrayList
        fruits.add("apple");
        fruits.add("banana");
        fruits.add("orange");
        fruits.add("grape");
        // Print the ArrayList
        System.out.println("ArrayList: " + fruits);
        // Update an element in the ArrayList
        fruits.set(1, "kiwi");
        // Print the ArrayList after update
        System.out.println("ArrayList after update: " + fruits);
        // Delete an element from the ArrayList
        fruits.remove(2);
        // Print the ArrayList after delete
        System.out.println("ArrayList after delete: " + fruits);
        // Convert the ArrayList to an array
        String[] fruitsArray = fruits.toArray(new String[0]);
        // Print the array
        System.out.println("Array: ");
        for (String fruit : fruitsArray) {
            System.out.println(fruit);
        }
        // Display the ArrayList using iterator()
        System.out.println("ArrayList using iterator(): ");
        Iterator<String> iterator = fruits.iterator();
        while (iterator.hasNext()) {
            System.out.println(iterator.next());
        }
    }
}
```

Output



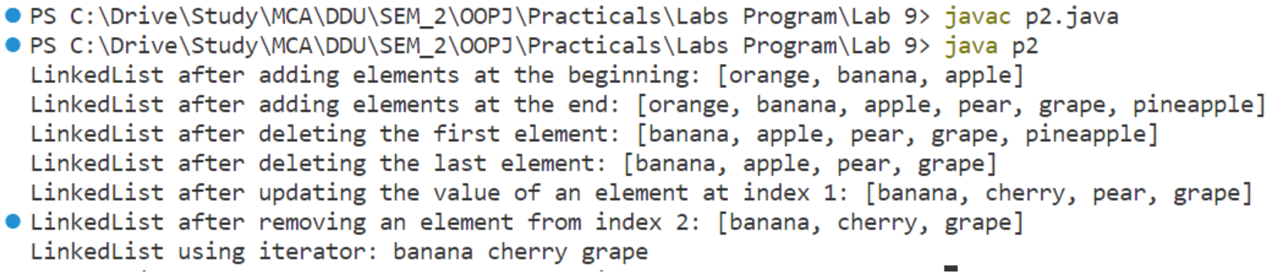
```
● PS C:\Drive\Study\MCA\DDU\SEM_2\OOPJ\Practicals\Labs Program\Lab 9> javac p1.java
● PS C:\Drive\Study\MCA\DDU\SEM_2\OOPJ\Practicals\Labs Program\Lab 9> java p1
  ArrayList: [apple, banana, orange, grape]
  ArrayList after update: [apple, kiwi, orange, grape]
  ArrayList after delete: [apple, kiwi, grape]
  Array:
  apple
  kiwi
● grape
  ArrayList using iterator():
  apple
  kiwi
● grape
```

2. **Create a LinkedList and perform following operations for that:**
    a. **Insert at first**
    b. **Insert at last**
    c. **Delete from first**
    d. **Delete from last**
    e. **Update specific index node value**
    f. **Remove from specific index**
    g. **Display using iterator()**

Code

```
import java.util.LinkedList;
import java.util.ListIterator;

public class p2 {
    public static void main(String[] args) {
        // Create a new LinkedList
        LinkedList<String> linkedList = new LinkedList<String>();

        // Insert elements at the beginning of the LinkedList
        linkedList.addFirst("apple");
        linkedList.addFirst("banana");
        linkedList.addFirst("orange");
        System.out.println("LinkedList after adding elements at the beginning: " +
linkedList);

        // Insert elements at the end of the LinkedList
        linkedList.addLast("pear");
        linkedList.addLast("grape");
        linkedList.addLast("pineapple");
        System.out.println("LinkedList after adding elements at the end: " +
linkedList);

        // Delete the first element of the LinkedList
        linkedList.removeFirst();
        System.out.println("LinkedList after deleting the first element: " +
linkedList);

        // Delete the last element of the LinkedList
        linkedList.removeLast();
        System.out.println("LinkedList after deleting the last element: " +
linkedList);

        // Update the value of an element at a specific index
        linkedList.set(1, "cherry");
        System.out.println("LinkedList after updating the value of an element at index
1: " + linkedList);

        // Remove an element from a specific index
        linkedList.remove(2);
        System.out.println("LinkedList after removing an element from index 2: " +
linkedList);

        // Display the LinkedList using an iterator
        ListIterator<String> iterator = linkedList.listIterator();
        System.out.print("LinkedList using iterator: ");
        while (iterator.hasNext()) {
            System.out.print(iterator.next() + " ");
        }
    }
}
```

Output



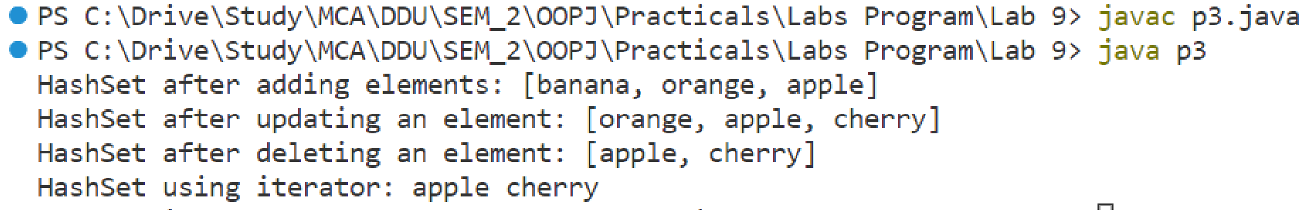
```
● PS C:\Drive\Study\MCA\DDU\SEM_2\OOPJ\Practicals\Labs Program\Lab 9> javac p2.java
● PS C:\Drive\Study\MCA\DDU\SEM_2\OOPJ\Practicals\Labs Program\Lab 9> java p2
  LinkedList after adding elements at the beginning: [orange, banana, apple]
  LinkedList after adding elements at the end: [orange, banana, apple, pear, grape, pineapple]
  LinkedList after deleting the first element: [banana, apple, pear, grape, pineapple]
  LinkedList after deleting the last element: [banana, apple, pear, grape]
  LinkedList after updating the value of an element at index 1: [banana, cherry, pear, grape]
● LinkedList after removing an element from index 2: [banana, cherry, grape]
  LinkedList using iterator: banana cherry grape
```

3. **Create a HashSet and perform following operations for that:**
    a. **Insert**
    b. **Update**
    c. **Delete**
    d. **Display using iterator()**

Code

```
import java.util.HashSet;
import java.util.Iterator;

public class p3 {

    public static void main(String[] args) {
        // Create a new HashSet
        HashSet<String> hashSet = new HashSet<String>();

        // Insert elements into the HashSet
        hashSet.add("apple");
        hashSet.add("banana");
        hashSet.add("orange");
        System.out.println("HashSet after adding elements: " + hashSet);

        // Update an element in the HashSet
        hashSet.remove("banana");
        hashSet.add("cherry");
        System.out.println("HashSet after updating an element: " + hashSet);

        // Delete an element from the HashSet
        hashSet.remove("orange");
        System.out.println("HashSet after deleting an element: " + hashSet);

        // Display the HashSet using an iterator
        Iterator<String> iterator = hashSet.iterator();
        System.out.print("HashSet using iterator: ");
        while (iterator.hasNext()) {
            System.out.print(iterator.next() + " ");
        }
    }
}
```

Output



```
● PS C:\Drive\Study\MCA\DDU\SEM_2\OOPJ\Practicals\Labs Program\Lab 9> javac p3.java
● PS C:\Drive\Study\MCA\DDU\SEM_2\OOPJ\Practicals\Labs Program\Lab 9> java p3
  HashSet after adding elements: [banana, orange, apple]
  HashSet after updating an element: [orange, apple, cherry]
  HashSet after deleting an element: [apple, cherry]
  HashSet using iterator: apple cherry
```

4. Create a HashMap and perform following operations for that:

a. Insert

b. Update

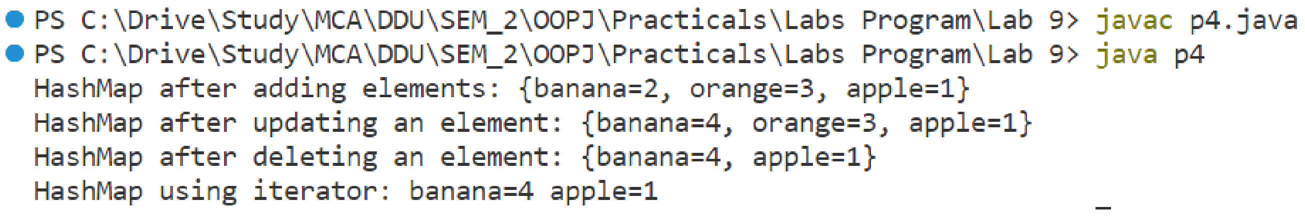
c. Delete

d. Display using iterator()

Code

```
import java.util.HashMap;
import java.util.Iterator;
import java.util.Map;

public class p4 {

    public static void main(String[] args) {
        // Create a new HashMap
        HashMap<String, Integer> hashMap = new HashMap<String, Integer>();

        // Insert elements into the HashMap
        hashMap.put("apple", 1);
        hashMap.put("banana", 2);
        hashMap.put("orange", 3);
        System.out.println("HashMap after adding elements: " + hashMap);

        // Update an element in the HashMap
        hashMap.put("banana", 4);
        System.out.println("HashMap after updating an element: " + hashMap);

        // Delete an element from the HashMap
        hashMap.remove("orange");
        System.out.println("HashMap after deleting an element: " + hashMap);

        // Display the HashMap using an iterator
        Iterator<Map.Entry<String, Integer>> iterator = hashMap.entrySet().iterator();
        System.out.print("HashMap using iterator: ");
        while (iterator.hasNext()) {
            Map.Entry<String, Integer> entry = iterator.next();
            System.out.print(entry.getKey() + "=" + entry.getValue() + " ");
        }
    }
}
```

Output



```
PS C:\Drive\Study\MCA\DDU\SEM_2\OOPJ\Practicals\Labs Program\Lab 9> javac p4.java
PS C:\Drive\Study\MCA\DDU\SEM_2\OOPJ\Practicals\Labs Program\Lab 9> java p4
HashMap after adding elements: {banana=2, orange=3, apple=1}
HashMap after updating an element: {banana=4, orange=3, apple=1}
HashMap after deleting an element: {banana=4, apple=1}
HashMap using iterator: banana=4 apple=1
```