5th day of practice

Q.1 What is the difference between Abstract Class & Interface?

ABSTRACT CLASS

- **Abstract are 0 to 100% incomplete.**
- **To develop an incomplete method & it is mandatory to use abstract keyword.**
- **Can be consist of non- static/static complete method.**
- **Can not consist of static method in complete.**
- **Can consist of main method.**
- **Reference variable of abstract class can be created but object is not created.**
- **Abstract class does not support multiple inheritance.**

Ex.

```
Package app_java_3;
    Abstract public class A {
            Abstract public void test1 ();
    }
Package app_java_3;
    Abstract public class B extends A {
          Abstract public void test2 ();
   }
Package app_java_3;
   Public class C extends B {
       @Override
          Public void test2 () {
          }
        @Override
          Public void test1 () {
          }
   }
```

INTERFACE:

- **Interface are declared by using interface keyword.**
- **Interface are 100% abstract.**
- **Interface defines a collection of abstract method & constrants.**
- **By interface we can support the functionality of multiple inheritance.**

Ex.

```
Package app_java_2;
      Public interface A {
         Public void test1 ();
```

```
    }
Package app_ java _2;
    Public class B implements A {
      @Override
       Public void test1 () {
           SOP (1);
       }
       PSVM () {
          B b1 = new B ();
           B1.test ();
       }
   }
```

Q.2 Difference between final variable, finally block & finalize method?

Final variable:

1. If you make a variable final then its value cannot be changed.

2 .If You make static/non-static variable final then initialization is mandatory.

3. If you make a method final then overriding is not alowed.

4. If you make a class final then inheritance is not allowed.

Ex.

```
public class A{
      PSVM(){
        final int x = 10;
           x = 10; //error
      }
  }
```

Finally block:

1.Finally block is an extension of (try-catch) block.

2.Any code that we write inside finally block will run 100%.

3.Commonly used for cleanup operations (e.g., closing resources).

4.Guarantees execution after try or catch blocks, even if an exception occurs.

Ex.

```
public class A{
        psvm(){
          try{
          }catch (Exception e){
                e.printStackTrace();
          }finally{
          }
```

```
        }
    }
```

Finalize Method:

1. Finalize is a method present inside object class.

2. Garbage Collection logic is implemented in finalize Method.

Ex.

```
public class A extends Object{
    protected void finalize(){
      System.out.println(1000);
    }
      public static void main(String [] args){
         A a1 = new A();
         a1=null;
           System.gc();
      }
}
```

Q.3 Write the program to remove duplicate element from an array?

```
import java.util.ArrayList;
import java.util.HashSet;
public class A {
   public static void main(String args[]) {
      ArrayList<Integer> data = new ArrayList<Integer>();
      data.add(10);
      data.add(5);
      data.add(100);
      data.add(5);

      HashSet<Integer> arr = new HashSet<Integer>();

        for (int i = 0; i<data.size(); i++){
           arr.add(data.get(i));
        }
        System.out.println(arr);
    }
}
```

Q.4 Write java code to short an array?

Ex.

```
import java.util.ArrayList;
import java.util.Collections;
public class A {
   public static void main(String args[]) {
      ArrayList<String> data = new ArrayList<String>();
      data.add("mike");
      data.add("stalin");
      data.add("mike");
      data.add("adam");

     Collections.sort(data);
     System.out.println(data);
   }
}
```

Q.5 Write a program to find max. salary in ann Array?

Ex.

```
import java.util.ArrayList;
import java.util.Collections;
public class A {
   public static void main(String args[]) {
      ArrayList<Integer> data = new ArrayList<Integer>();
      data.add(1000);
      data.add(200);
      data.add(20);
      data.add(5000);

      Integer maxSalary = Collections.max(data);
      System.out.println(maxSalary);
    }
}
```

Q.6 Write a java program for palindrome?

Ex.

```
import java.util.Scanner;
public class Palindrome {
   public static void main(String[] args) {
      Scanner scanner = new Scanner(System.in);
```

```
        System.out.print("Enter a word or number: ");
        String input = scanner.nextLine();

        if (isPalindrome(input)) {
            System.out.println(input + " is a palindrome.");
        } else {
            System.out.println(input + " is not a palindrome.");
        }
    }
     public static boolean isPalindrome(String str) {
        int left = 0;
        int right = str.length() - 1;

        while (left < right) {
            if (str.charAt(left) != str.charAt(right)) {
                return false;
            }
            left++;
            right--;
        }
        return true;
    }
}
```