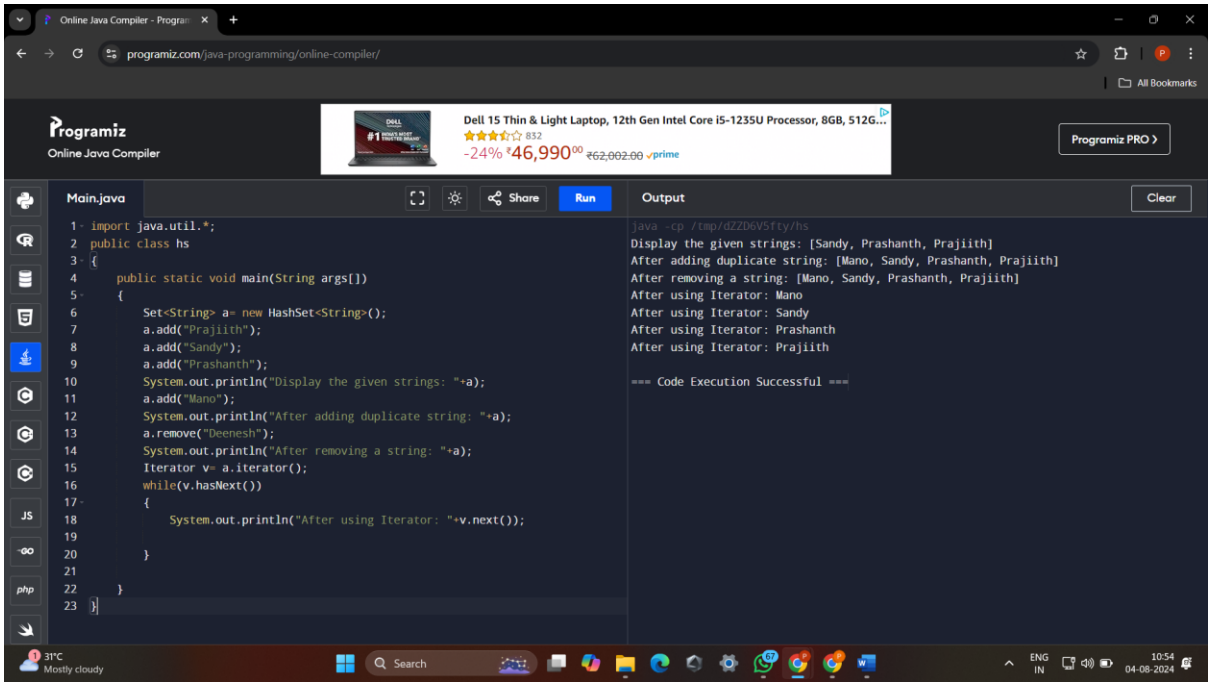
1. Using Hashed Set

```
import java.util.*;
public class hs
{
   public static void main(String args[])
   {
      Set<String> a= new HashSet<String>();
      a.add("Prajiith");
      a.add("Sandy");
      a.add("Prashanth");
      System.out.println("Display the given strings: "+a);
      a.add("Mano");
      System.out.println("After adding duplicate string: "+a);
      a.remove("Deenesh");
      System.out.println("After removing a string: "+a);
      Iterator v= a.iterator();
      while(v.hasNext())
      {
         System.out.println("After using Iterator: "+v.next());

      }

   }
}
```

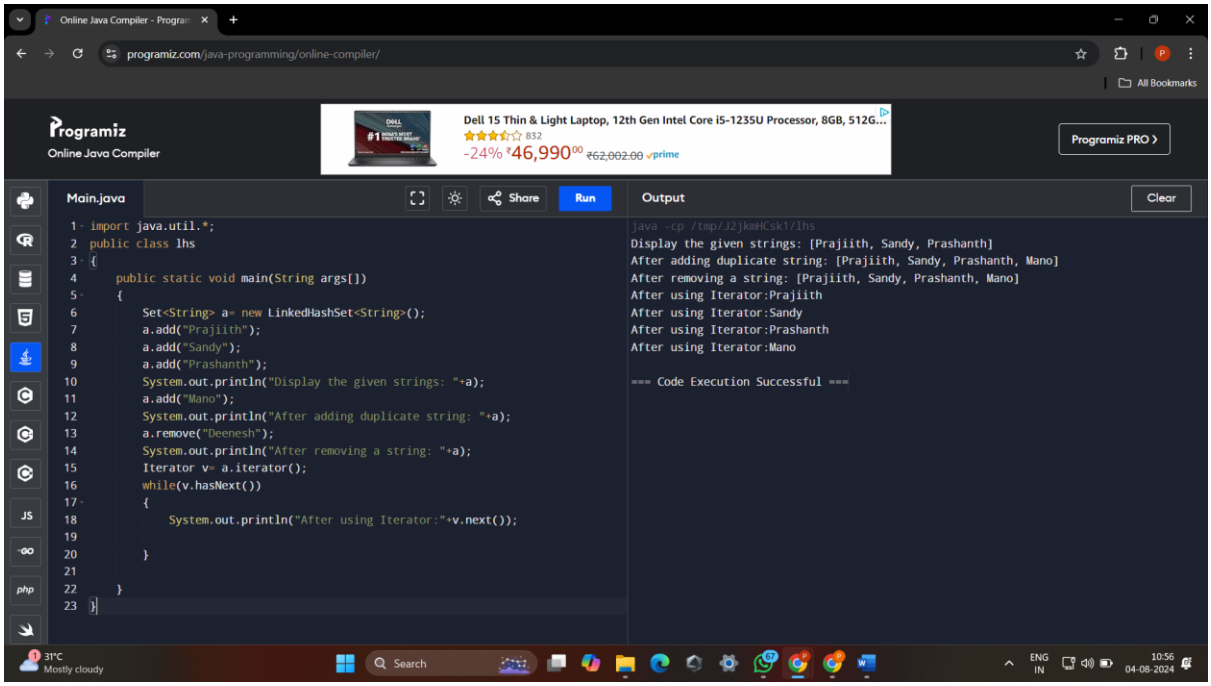


2. Using Linked Hahed Set

```
import java.util.*;
public class lhs
{
   public static void main(String args[])
   {
      Set<String> a= new LinkedHashSet<String>();
      a.add("Prajiith");
      a.add("Sandy");
      a.add("Prashanth");
      System.out.println("Display the given strings: "+a);
      a.add("Mano");
      System.out.println("After adding duplicate string: "+a);
      a.remove("Deenesh");
      System.out.println("After removing a string: "+a);
      Iterator v= a.iterator();
      while(v.hasNext())
```

```
        {
            System.out.println("After using Iterator:"+v.next());

        }
    }
}
```



3. Using Tree Set

```
import java.util.*;
public class ts
{
   public static void main(String args[])
   {
      Set<String> a= new TreeSet<String>();
      a.add("Prajiith");
```

```
        a.add("Sandy");
        a.add("Prashanth");
        System.out.println("Display the given strings: "+a);
        a.add("Mano");
        System.out.println("After adding duplicate string: "+a);
        a.remove("Deenesh");
        System.out.println("After removing a string: "+a);
        Iterator v= a.iterator();
        while(v.hasNext())
        {
           System.out.println("After using Iterator:"+v.next());

        }

    }
}
```