

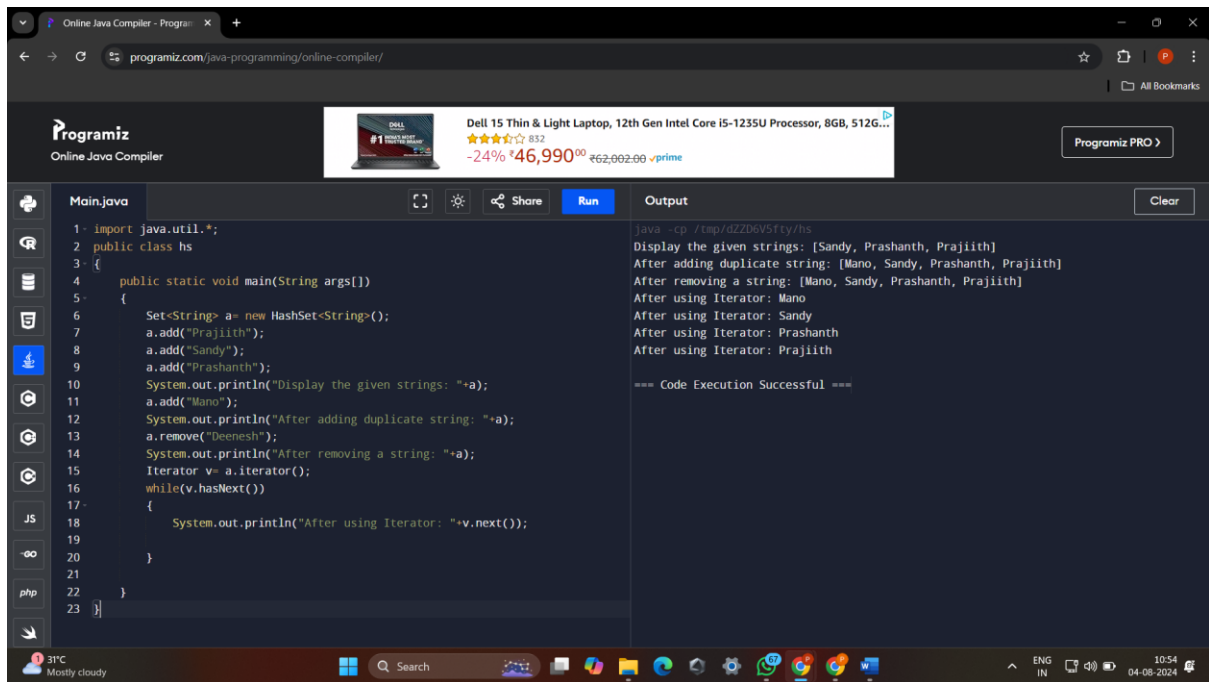
1. Using Hashed Set

```
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())
```

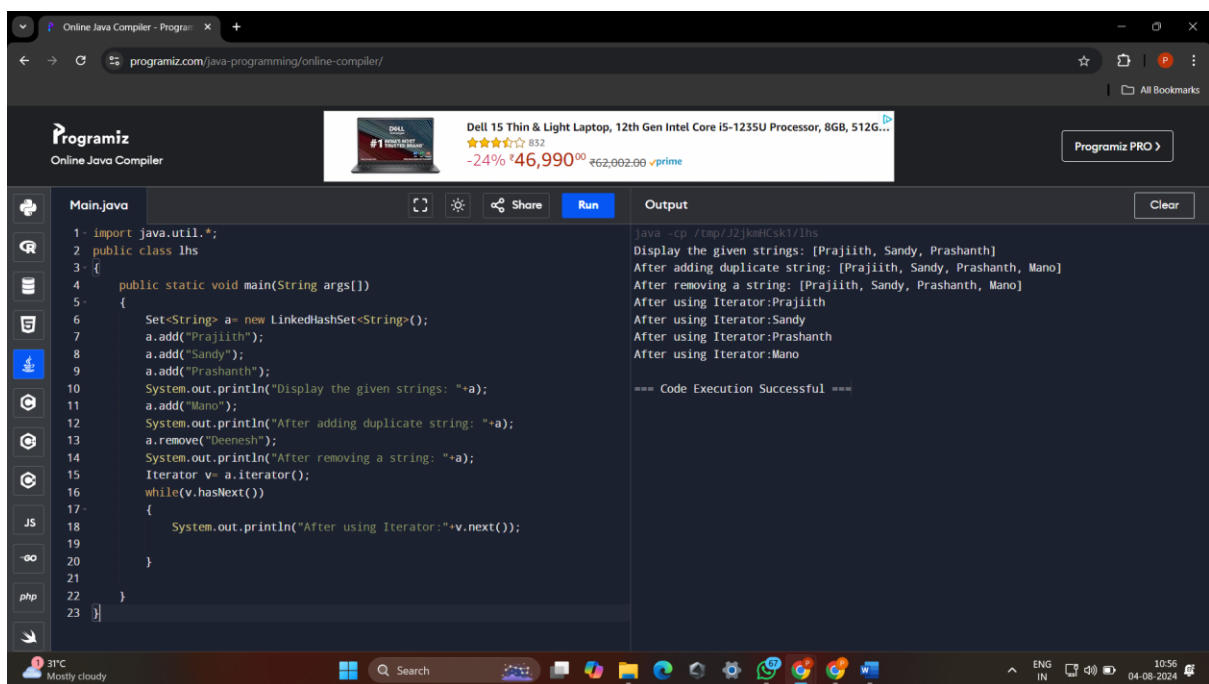
```

{
    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());

}

}

}

```

The screenshot shows a web browser window with the URL `programiz.com/java-programming/online-compiler/`. The page features a dark-themed editor with a file named `Main.java`. The code in the editor is as follows:

```

1- import java.util.*;
2- public class ts
3- {
4-     public static void main(String args[])
5-     {
6-         Set<String> a= new TreeSet<String>();
7-         a.add("Prajiith");
8-         a.add("Sandy");
9-         a.add("Prashanth");
10-        System.out.println("Display the given strings: "+a);
11-        a.add("Mano");
12-        System.out.println("After adding duplicate string: "+a);
13-        a.remove("Deenesh");
14-        System.out.println("After removing a string: "+a);
15-        Iterator v= a.iterator();
16-        while(v.hasNext())
17-        {
18-            System.out.println("After using Iterator:"+v.next());
19-        }
20-    }
21- }
22- }
23- }

```

On the right side of the editor, the **Output** panel displays the following text:

```

java -cp /tmp/Zj1GzuZYky/ts
Display the given strings: [Prajiith, Prashanth, Sandy]
After adding duplicate string: [Mano, Prajiith, Prashanth, Sandy]
After removing a string: [Mano, Prajiith, Prashanth, Sandy]
After using Iterator:Mano
After using Iterator:Prajiith
After using Iterator:Prashanth
After using Iterator:Sandy

=== Code Execution Successful ===

```

The browser's address bar shows the URL, and the top navigation bar includes the Programiz logo and a "Programiz PRO" button. A banner for a Dell laptop is visible above the editor. The bottom of the browser window shows a Windows taskbar with various application icons and a system tray indicating the temperature is 31°C and the date is 04-08-2024.