

# Java String,StringBuffer and StringBuilder Methods and Examples

---

-K.L.MADHAVI

# Java String class methods

```
String Str=new String("Hello")
```

Method name	Syntax to use	Description
1.length()	Str.length()	Returns length of String.
2.indexOf()	indexOf(Str)	Returns the index value of 1 <sup>st</sup> character in given string.
3.charAt()	Str.charAt(index)	Returns the character at given index.
4.replace()	Str.replace(old String,new String)	It replaces old string with new string.
5.toLowerCase()	Str.toLowerCase()	Everything will be converted into lowercase.
6.toUpperCase()	Str.toUpperCase()	Everything will be converted into uppercase.
7.compareTo()	Str.compareTo(str)	Compares 2 strings

# Continue...

Method name	Syntax to use	Description
8.trim()	Str.trim()	Remove all the white spaces which appear before and after the string.
9.concat()	Str2=Str.concat(“string”)	It is used to concatenate 2 strings.

# Example Programs

---

## 1.length() Method:

```
public class Stringoperation1
{
    public static void main(String args[])
    {
        String s="Madhu";
        System.out.println(s.length());//5
    }
}
```

## 2.indexOf() Method:

```
public class Stringoperation2
{
    public static void main(String args[])
    {
        String s="Madhu";
        System.out.println(s.indexOf(a));//1
    }
}
```

## 3.charAt() Method:

```
public class Stringoperation3
{
    public static void main(String args[])
    {
        String s="Madhu";
        System.out.println(s.charAt(2));//d
    }
}
```

## 4.replace() Method:

```
public class Stringoperation4
{
    public static void main(String args[])
    {
        String s=new String("santosh is friend of satya");
        System.out.println(s.replace("santosh","madhu"));//madhu is
        friend of satya
    }
}
```

## 5.toLowerCase() Method:

```
public class Stringoperation5
{
    public static void main(String args[])
    {
        String s="Madhu";
        System.out.println(s.toLowerCase());//madhu
    }
}
```

# Continue...

---

## **6.toUpperCase() Method:**

```
public class Stringoperation6
{
    public static void main(String args[])
    {
        String s="Madhu";
        System.out.println(s.toUpperCase());//MADHU
    }
}
```

## **8.trim() Method:**

```
public class Stringoperation8
{
    public static void main(String args[])
    {
        String s=" Madhu ";
        System.out.println(s.trim());//Madhu
    }
}
```

## **7.compareTo() Method:**

```
public class Stringoperation7
{
    public static void main(String args[])
    {
        String s="Madhu";
        System.out.println(s.compareTo(madhu));//-32
    }
}
```

## **9.concat() Method:**

```
public class Stringoperation9
{
    public static void main(String args[])
    {
        String s=new String("Madhu");
        String s1=s.concat("Satya");
        System.out.println(s1);//MadhuSatya
    }
}
```

# Program for String class methods

---

Class StringDemo

```
{
    public static void main(String args[])
    {
        String Str=new String(" Hello");
        System.out.println(Str);
        System.out.println(Str.length());
        System.out.println(Str.indexOf("H");
        System.out.println(Str.toUpperCase());
        System.out.println(Str.toLowerCase());
        System.out.println(Str.trim());
        System.out.println("Character at="+Str.charAt(4));
        System.out.println(Str.compareTo("hello");
        Str1=(Str.concat(" welcome"));
        System.out.println(Str1);
    }
}
```

**Output:**

```
Hello
7
2
HELLO
hello
Hello
Character at=1
-72
Hello welcome
```

StringBuffer Str=new StringBuffer(“welcome”)

# Java StringBuffer class methods

Method name	Syntax to use	Description
append()	Str.append(String)	append() method concatenates the given argument with this String.
insert()	Str.insert(index,String)	insert() method inserts the given String with this string at the given position.
delete()	Str.delete(Start_index,End_index)	delete() method deletes the String from the specified beginIndex to endIndex.
replace()	Str.replace(Start_index,End_index,String)	replace() method replaces the given String from the specified beginIndex and endIndex.
reverse()	Str.reverse()	reverse() method reverses the current String.
length()	Str.length()	Returns length of string.

# Continue...

---

Method name	Syntax to use	Description
subString()	Str.subString(index)	Returns the string from that given index num to end of string.
subSequence()	Str.subSequence(Start_index,End_index)	Returns the String between the given start and end index.
charAt()	Str.charAt(index)	Returns the character at given index.
indexOf()	indexOf(Str)	Returns the index value of 1 <sup>st</sup> character in given string.



# Program for StringBuffer methods

---

Class StringBufferDemo

```
{
    public static void main(String args[])
    {
        StringBuffer Str=new StringBuffer("welcome");
        System.out.println(Str);
        System.out.println(Str.append(" to java"));
        System.out.println(Str.insert(4,"How"));
        System.out.println(Str.delete(4,7));
        System.out.println(Str.reverse());
        System.out.println(Str.length());
        System.out.println(Str.indexOf("welcome"));
        System.out.println(Str.reverse());
        System.out.println(Str.indexOf("welcome"));
        System.out.println(Str.charAt(8));
        System.out.println(Str.substring(8));
        System.out.println(Str);
        System.out.println(Str.subSequence(11,15));
        System.out.println(Str);
    }
}
```

## Output:

```
welcome
welcome to java
welcHowome to java
welcome to java
avaj ot emoclew
15
-1
welcome to java
0
t
to java
welcome to java
java
welcome to java
```

```
StringBuilder Str=new StringBuilder("welcome")
```

# Java StringBuilder class methods

Method name	Syntax to use	Description
append()	Str.append(String)	append() method concatenates the given argument with this String.
insert()	Str.insert(index,String)	insert() method inserts the given String with this string at the given position.
replace()	Str.replace(Start_index,End_index,String)	replace() method replaces the given String from the specified beginIndex and endIndex.
delete()	Str.delete(Start_index,End_index)	delete() method deletes the String from the specified beginIndex to endIndex.
reverse()	Str.reverse()	reverse() method reverses the current String.
capacity()	Str.capacity()	Returns the current capacity

# Program for StringBuilder methods

---

```
Class StringBuilderDemo
{
    public static void main(String args[])
    {
        StringBuilder Str=new StringBuilder("hello");
        System.out.println(Str);
        System.out.println(Str.append(" madhu"));
        System.out.println(Str.delete(0,5));
        System.out.println(Str.insert(0,"how r u"));
        System.out.println(Str.reverse());
        System.out.println(Str.reverse());
        System.out.println(Str.replace(0,7,"hi"));
        System.out.println(Str.capacity());

    }
}
```

## **Output:**

```
hello
hello madhu
madhu
how r u madhu
uhdam u r who
how r u madhu
hi madhu
21
```