CLASS: StringBuffer

The `StringBuffer` class is a **predefined** class in the `java.lang` package. It was introduced in Java to overcome the **multiple object creation issue** associated with the `String` class during content modifications.

- If frequent operations involve **content changes**, it is recommended to use `StringBuffer` instead of `String`.
- Modifications in `StringBuffer` are applied **directly to the same object**, whereas with `String`, the JVM creates a **new object** for each change.

Why Use `StringBuffer`?

- When frequent content changes are needed, `StringBuffer` performs better because the **JVM does not create a new object** each time — it modifies the same instance.
- This makes it ideal for **content-changing operations**.

CLASS: StringBuilder

- Every method in `StringBuffer` is **synchronized**, allowing only one thread to operate at a time.
- This synchronization results in **increased waiting time**, affecting system performance.
- To address this issue, `StringBuilder` was introduced.

`StringBuilder`

- It is **functionally identical** to `StringBuffer` (including constructors and methods), except:
  - Methods in `StringBuilder` are **not synchronized**.
  - Multiple threads can operate on it **simultaneously**.
  - It is **not thread-safe**, but offers **higher performance**.
- Threads are **not required to wait**, resulting in **better performance** when thread-safety is not a concern.

Constructors of StringBuffer

- `StringBuffer();`
- `StringBuffer(int);`
- `StringBuffer(String);`

StringBuffer sb = new StringBuffer();

- Creates an **empty** `StringBuffer` with a **default capacity** of **16**.
- If exceeded, capacity grows as per:

  ```
  newCapacity = (oldCapacity + 1) * 2
  ```

```java
class SB1 {
    public static void main(String[] args) {
        StringBuffer sb = new StringBuffer();
        System.out.println(sb.capacity()); // 16
        System.out.println(sb.length());   // 0
        sb.append("123456789");
        System.out.println(sb.capacity()); // 16
        System.out.println(sb.length());   // 9
        sb.append("012345678901234567890123456789");
        System.out.println(sb.capacity()); // 39
        System.out.println(sb.length());   // 39
    }
}
```

StringBuffer sb = new StringBuffer(int);

- Creates an **empty** buffer with specified capacity.

```java
class SB2 {
    public static void main(String[] args) {
        StringBuffer sb = new StringBuffer(100);
        System.out.println(sb.capacity()); // 100
    }
}
```

StringBuffer sb = new StringBuffer(String);

- Creates a buffer initialized with a given `String`.
- `initialCapacity = string.length() + 16`

```java
class SB3 {
    public static void main(String[] args) {
        StringBuffer sb = new StringBuffer("Sambit");
        System.out.println(sb.capacity()); // 22
    }
}
```

Methods

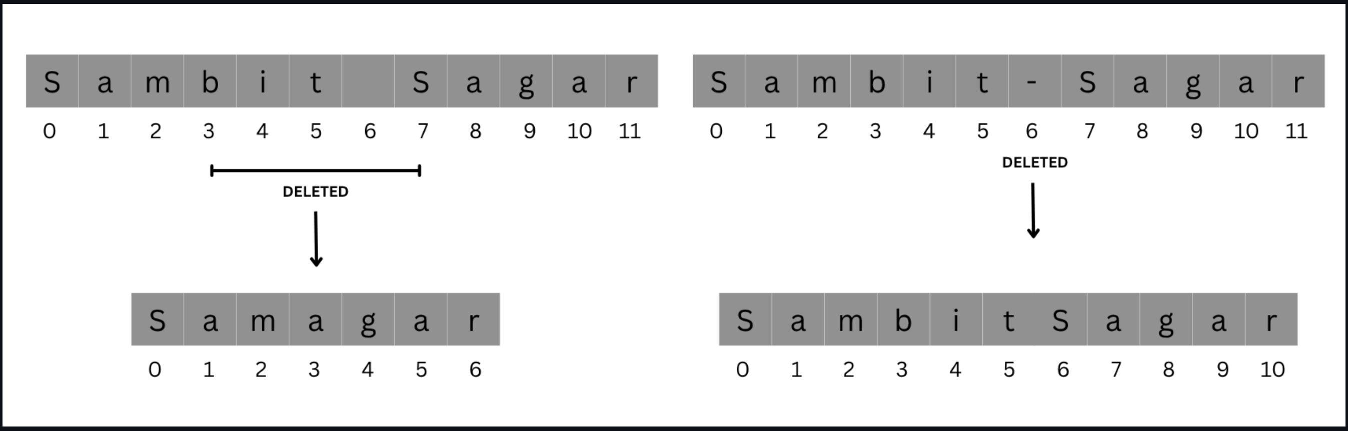
Return Type	Method
`StringBuffer`	`.append(String);`
`int`	`.length();`
`int`	`.capacity();`
`char`	`.charAt(int);`
`void`	`.setCharAt(int, char);`
`StringBuffer`	`.delete(int, int);`
`StringBuffer`	`.deleteCharAt(int);`
`StringBuffer`	`.reverse();`
`void`	`.setLength(int);`
`void`	`.trimToSize();`
`void`	`.ensureCapacity(int);`

`.append(String);`, `.length();`, `.capacity();`, `.charAt(int);`, `.setCharAt(int, char);`

```java
class StringBufferMethods1 {
    public static void main(String[] args) {
        StringBuffer sb = new StringBuffer();
        sb.append("123456789");
        System.out.println(sb);             // 123456789
        System.out.println(sb.capacity());  // 16
        System.out.println(sb.length());    // 9
        System.out.println(sb.charAt(4));   // 5
        sb.setCharAt(0, 'E');
        System.out.println(sb);             // E23456789
    }
}
```

`.delete(int, int);`, `.deleteCharAt(int);`

```java
class StringBufferMethods2 {
    public static void main(String[] args) {
        StringBuffer sb = new StringBuffer();
        sb.append("Sambit Sagar");
        System.out.println(sb);                     // Sambit Sagar
        System.out.println(sb.delete(3, 8));        // Samagar
        sb = new StringBuffer("Sambit-Sagar");
        System.out.println(sb);                     // Sambit-Sagar
        System.out.println(sb.deleteCharAt(6));     // SambitSagar
    }
}
```



.reverse();

```java
StringBuffer sb = new StringBuffer("12345");
System.out.println(sb.reverse()); // 54321
```

.setLength(int);

```java
StringBuffer sb = new StringBuffer("Sambit Sagar");
sb.setLength(10);
System.out.println(sb); // Sambit Sag
```

.trimToSize();

```java
class StringBufferMethods3 {
    public static void main(String[] args) {
        StringBuffer sb = new StringBuffer("Sambit");
        System.out.println(sb.capacity()); // 22
        System.out.println(sb.length());   // 6
        sb.trimToSize();
        System.out.println(sb.capacity()); // 6
        System.out.println(sb.length());   // 6
    }
}
```

.ensureCapacity(int);

```java
class StringBufferMethods4 {
    public static void main(String[] args) {
        StringBuffer sb = new StringBuffer();
        System.out.println(sb.capacity()); // 16
        sb.ensureCapacity(20);
        System.out.println(sb.capacity()); // 34
    }
}
```