

Q1. Explain StringBuffer class in Java?

Ans:

String(Immutable):

StringBuffer(Mutable): If our content are not fixed and we want to perform some modification but thread safety is required then we should go for String Buffer class

StringBuilder(Mutable): If our content are not fixed and we want to perform some modification but thread safety is not required then we should go for String Builder class

Constructor:

1. StringBuffer();

//default initial capacity is 16

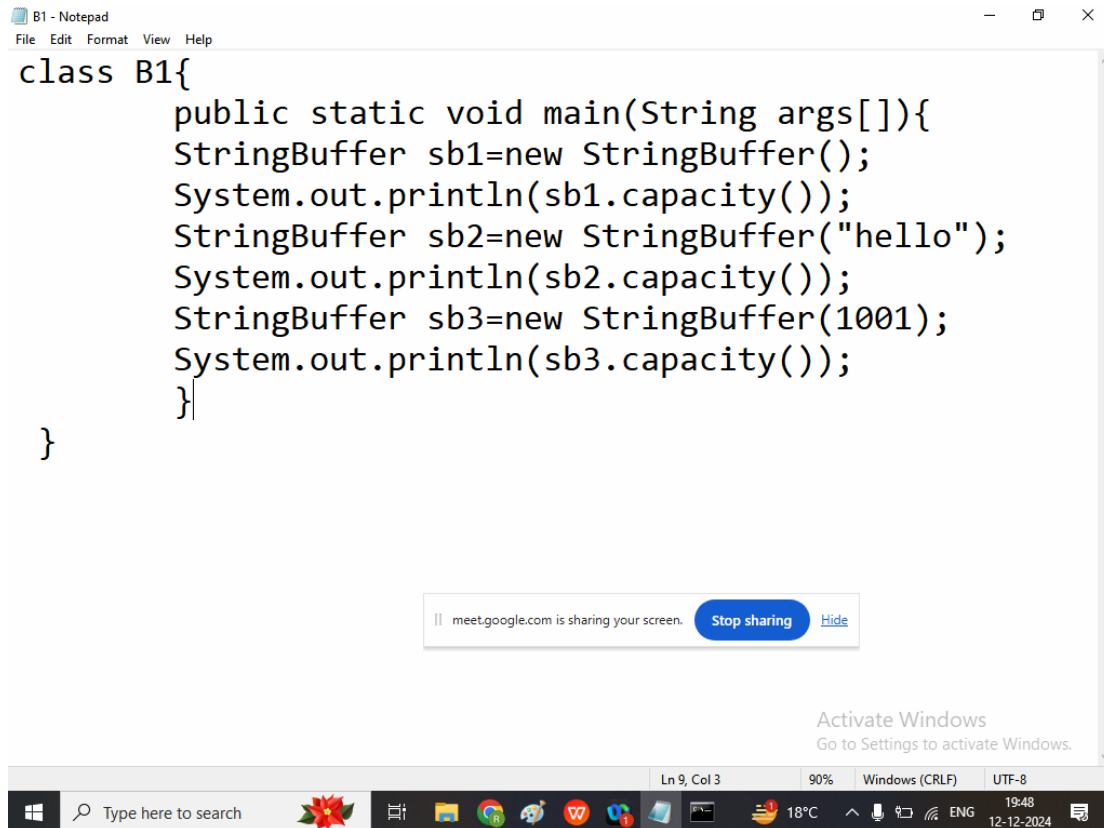
New capacity =(current capacity +1)\*2

2. StringBuffer(String str):

//default capacity=default capacity

SB+length of the String

3. StringBuffer(int initialCapacity)



```
class B1{
    public static void main(String args[]){
        StringBuffer sb1=new StringBuffer();
        System.out.println(sb1.capacity());
        StringBuffer sb2=new StringBuffer("hello");
        System.out.println(sb2.capacity());
        StringBuffer sb3=new StringBuffer(1001);
        System.out.println(sb3.capacity());
    }
}
```

meet.google.com is sharing your screen. [Stop sharing](#) [Hide](#)

Activate Windows  
Go to Settings to activate Windows.

Ln 9, Col 3 90% Windows (CRLF) UTF-8

Type here to search 18°C 19:48 12-12-2024

## Methods of StringBuffer and StringBuilder

1. `int capacity()`: it will return current capacity of the SB Object

2. `append()`: It is used to insert data at the end of the StringBuffer object

`append(String)`

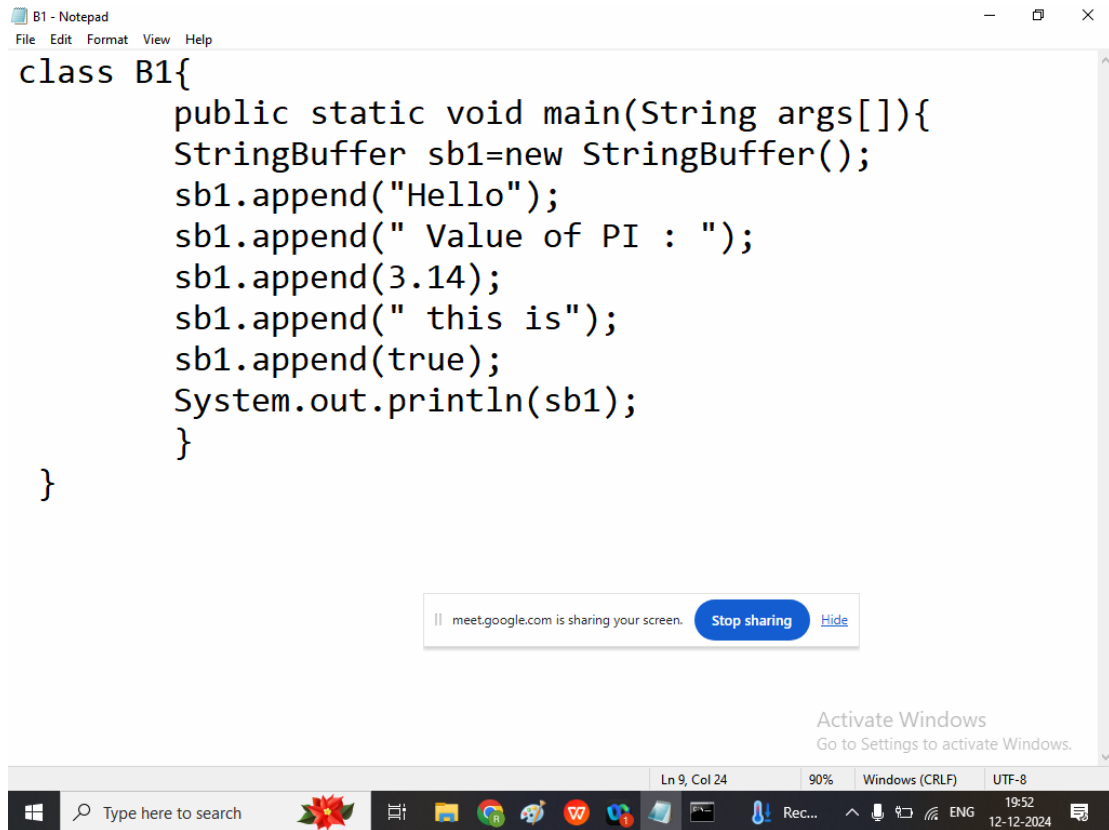
`append(int)`

`append(float)`

`append(String)`

`append(char)`

`append(boolean)`



The screenshot shows a Windows 10 desktop environment. A Notepad window titled 'B1 - Notepad' is open, displaying the following Java code:

```
class B1{  
    public static void main(String args[]){  
        StringBuffer sb1=new StringBuffer();  
        sb1.append("Hello");  
        sb1.append(" Value of PI : ");  
        sb1.append(3.14);  
        sb1.append(" this is");  
        sb1.append(true);  
        System.out.println(sb1);  
    }  
}
```

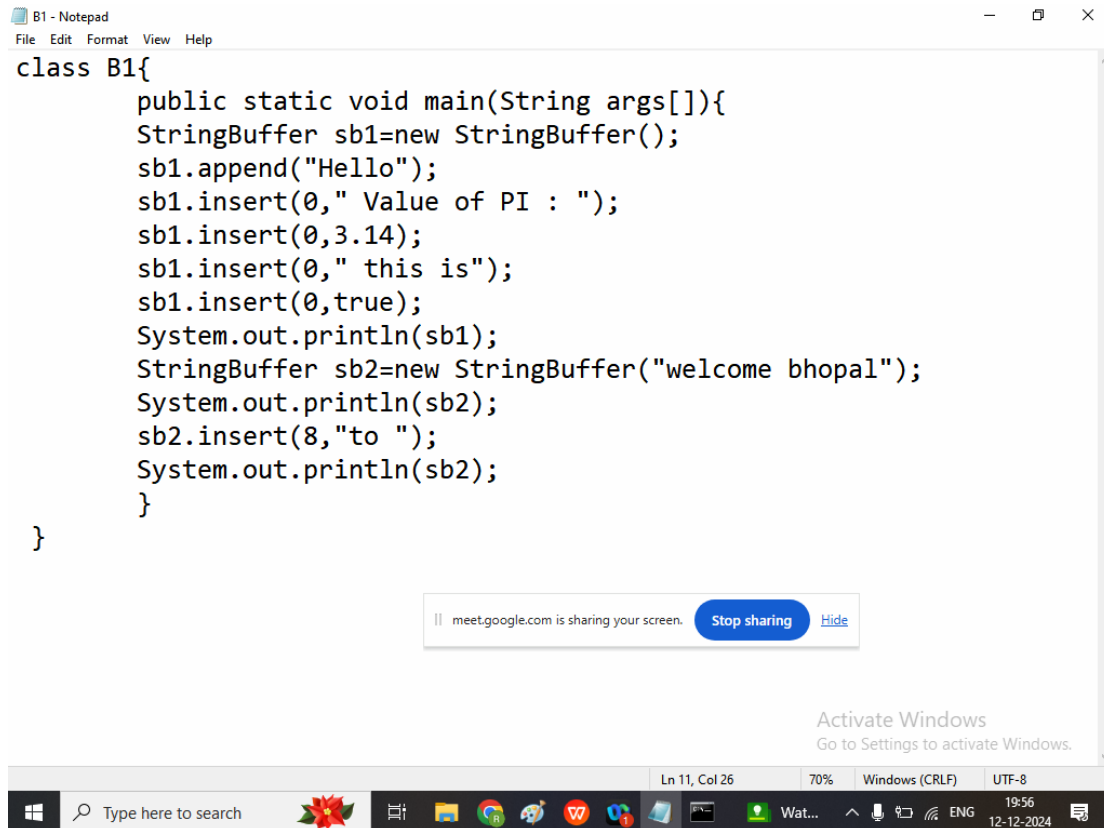
Below the code, a notification bar indicates that 'meet.google.com' is sharing the screen, with buttons for 'Stop sharing' and 'Hide'. At the bottom right of the Notepad window, there is a watermark that says 'Activate Windows Go to Settings to activate Windows.'.

The Windows taskbar is visible at the bottom, showing the Start button, a search bar, and several pinned application icons including File Explorer, Google Chrome, and Microsoft Word. The system tray on the right shows the date and time as '19:52 12-12-2024'.

3. Insert(): It is used to add data into SB object in any position

Syntax:

insert(position,data);



```
class B1{
    public static void main(String args[]){
        StringBuffer sb1=new StringBuffer();
        sb1.append("Hello");
        sb1.insert(0," Value of PI : ");
        sb1.insert(0,3.14);
        sb1.insert(0," this is");
        sb1.insert(0,true);
        System.out.println(sb1);
        StringBuffer sb2=new StringBuffer("welcome bhopal");
        System.out.println(sb2);
        sb2.insert(8,"to ");
        System.out.println(sb2);
    }
}
```

Q1.Explain StringBuffer class  
deleteCharAt() method?

Ans: It is used to delete character by  
character from the given SB object?

Example:

Write a java program to delete first and  
middle and last character of String

Enter Any String: welcome

Result After Delete: elom

```
class B1{
    public static void main(String args[]){
        StringBuffer sb1=new
StringBuffer("welcome");
        System.out.println(sb1);
        sb1.deleteCharAt(sb1.length()/2);
        sb1.deleteCharAt(0);
        sb1.deleteCharAt(sb1.length()-1);
        System.out.println(sb1.toString());

    }
}
```

---

Q1. Explain delete() method of StringBuffer class?

Ans: It is used to delete more than one character at a time then we should go for delete method

Syntax:

```
delete(int start,int end);
```

```
class A1{
```

```
    public static void main(String args[]){
```

```
        StringBuffer s=new
```

```
StringBuffer("welcome to bhopal");
```

```
        System.out.println(s);
```

```
        s.delete(3,6);
```

```
        System.out.println(s);
```

```
}  
}
```

---

Q2. Explain StringBuffer class  
setCharAt()?

Ans: It is used to update String Buffer  
Object

Syntax:

```
setCharAt(int position,char);
```

```
class A1{
```

```
    public static void main(String args[]){
```

```
StringBuffer s=new  
StringBuffer("AMAN");  
  
System.out.println(s);  
  
s.setCharAt(0,'N');  
  
System.out.println(s);
```

```
}
```

```
}
```

---



Q3. Write a java program to update all vowel by Z in the Given String?

Enter Any String: **welcome**

**wZlcZmZ**

```
class A1{  
    public static void main(String  
args[]){  
        StringBuffer s=new  
StringBuffer("welcome");  
        System.out.println(s);  
        for(int i=0;i<s.length();i++){  
            char c=s.charAt(i);  
            if(c=='a' || c=='i' || c=='o' || c=='u' || c=='e'  
)
```

```
s.setCharAt(i,'Z');
```

```
}
```

```
}
```

```
System.out.println(s);
```

```
}
```

```
}
```

---

**Q3. Explain StringBuffer class reverse()?**

**Ans:** It is used to reverse the StringBuffer Object

```
class A1{  
    public static void main(String args[]){  
        StringBuffer s=new  
StringBuffer("ABCDE");  
  
        System.out.println(s);  
  
        s.reverse();  
  
        System.out.println(s);  
    }  
}
```

}

}

---

Homework:

Q1. Write a java program to check given String is Palindrome or Not?

**Example1:**

Enter Any String: NAMAN

This is Palindrome String

**Example2:**

Enter Any String: MADAM

This is Palindrome String

### **Example1:**

Enter Any String: CAR

This is Not a Palindrome String

Q2. Write a java program to check given string is panagram or not?

Ans:

### **Example1:**

Enter Any String:

abcdefghijklmnopqrstuvwxyz

True

### **Example2:**

Enter Any String:

abcdefghijklmnopnorstuvwxyz

false

---

---

Q2. Write a java program to check given String is anagram or not?

Example1:

Enter String 1: listen

Enter String 2: silent

This is anagram string

---

---