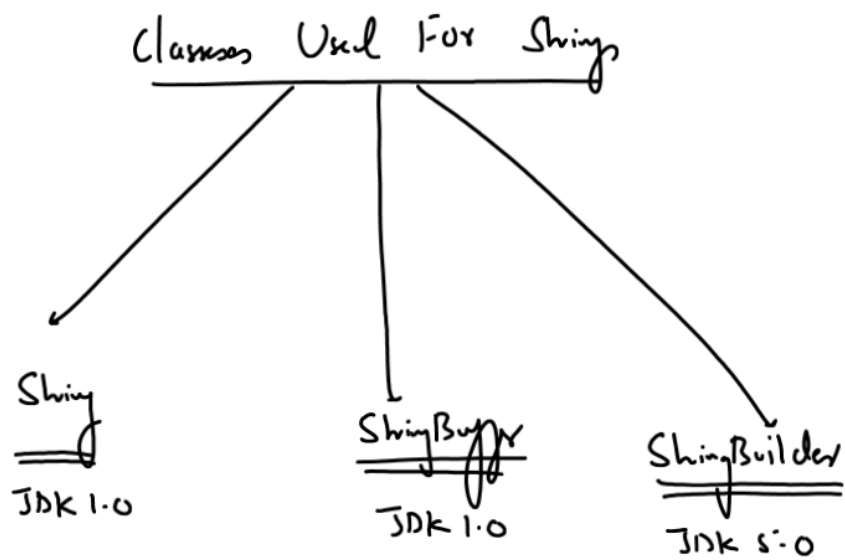


- ① "Hello"
- ② "Hello Use"
- ③ "Bhupat"
- ④ "s"
- ⑤ ""



String objects are "IMMUTABLE"

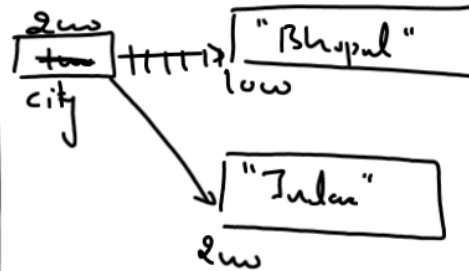
while

Not changeable

StringBuffer objects are "Mutable"

changeable

```
String city = "Bhopal";  
SOP(city); // Bhopal  
city = "Indore";  
SOP(city); // Indore
```

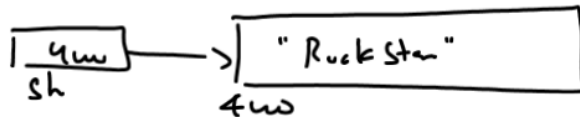
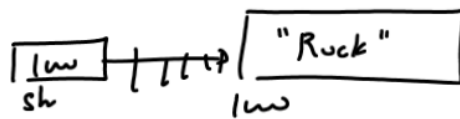


```
String sh = "Rock";
```

```
SOP(sh); Rock
```

```
sh = sh + "Star";
```

```
SOP(sh); RockStar
```

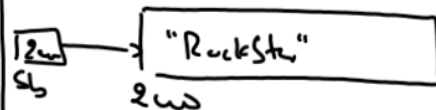


```
StringBuffer sb = new StringBuffer("Rock");
```

```
SOP(sb); Rock
```

```
sb.append("Star");
```

```
SOP(sb); RockStar
```



```

import java.util.*;
class InvalidNumeratorException extends Exception
{
    public InvalidNumeratorException(String message)
    {
        super(message);
    }
}

```

```

class DivideAndSum

```

```

{
    public static void main(String[] args)
    {
        Scanner kb=new Scanner(System.in);
        System.out.println("Enter 2 int:");
        int a=kb.nextInt();
        int b=kb.nextInt();

```

```

        try
        {
            if(a<=0)
            {
                InvalidNumeratorException obj=new InvalidNumeratorException("Numerator
must be positive");
                throw obj;
            }
            int c=a/b;
            System.out.println("Div is "+c);
        }
        catch(InvalidNumeratorException | ArithmeticException obj)
        {
            System.out.println("Exception is:"+obj);
        }
        finally
        {
            System.out.println("Have a good day!");
        }
    }
}

```