

```
1 package StringManipulations;
2
3 public class StringCompare {
4
5     public static void main(String[] args) {
6
7         //String literals
8         String s1 = "hello selenium";
9
10        String s2 = new String("hello selenium");
11
12
13        System.out.println(s1 == s2);
14    }
15 }
```

Console
<terminated> StringCompare (1) [Java]
false

```
14        System.out.println(s1.equals(s2));
```

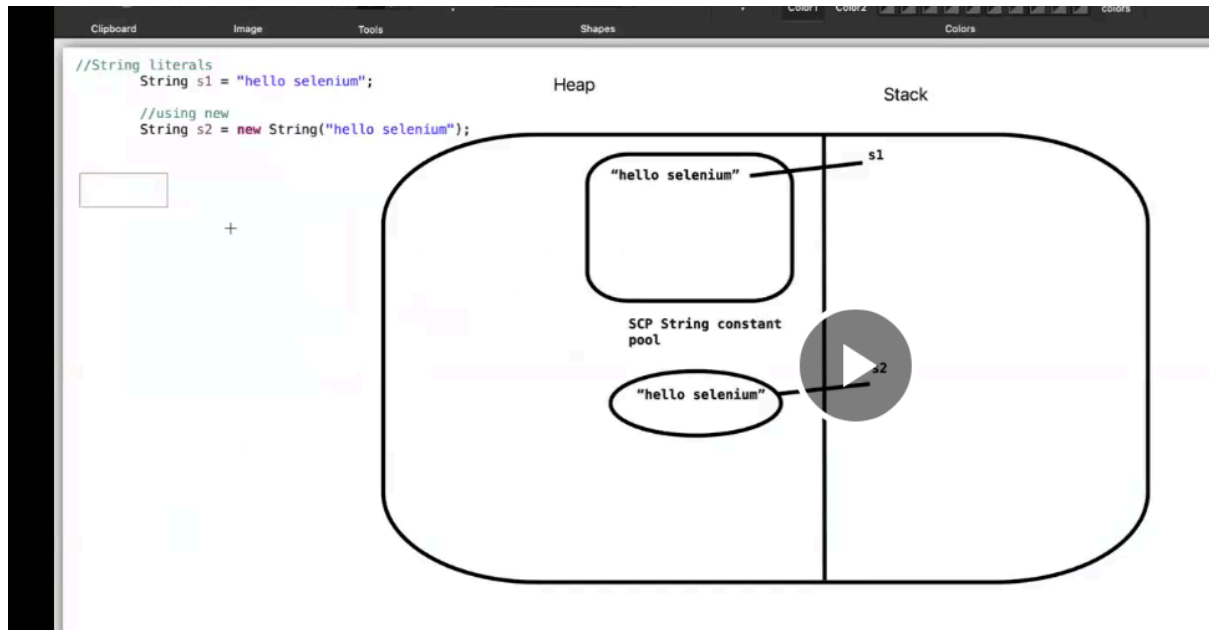
True

.equals –

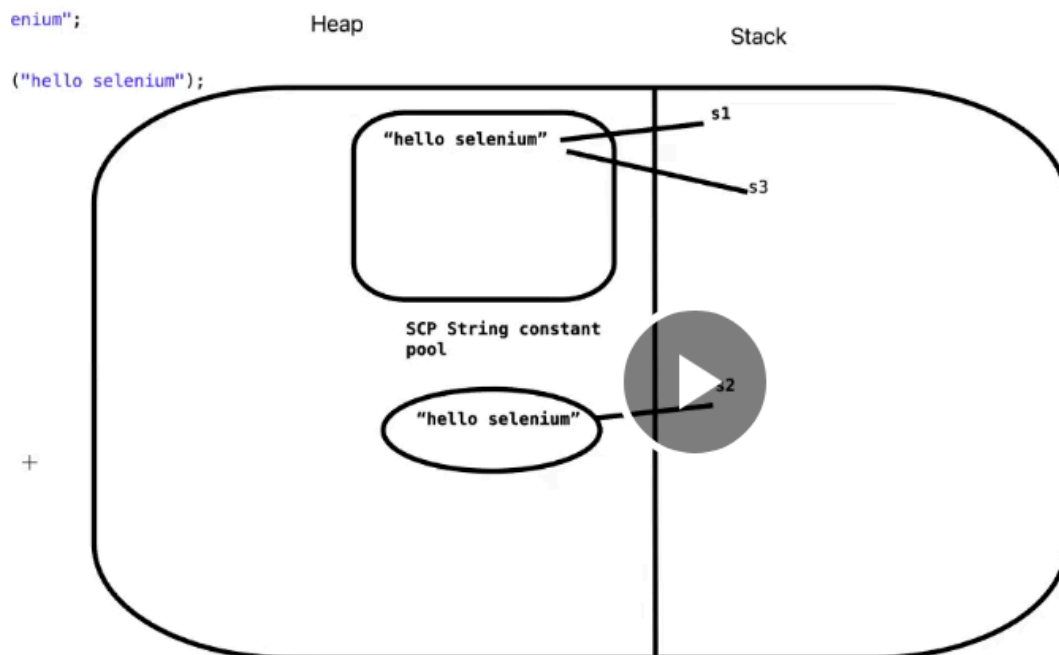
Compares the string content irrespective of how we create strings either using literals or objects. Best way to compare strings.

String literals always stored in scp (inside heap memory).

== will compare if the references point to same thing.



```
String s3 = "hello selenium";
```



```
17
18 String s3 = "hello selenium";
19
20 System.out.println(s1 == s3); //false
21 System.out.println(s1.equals(s3)); //true
```

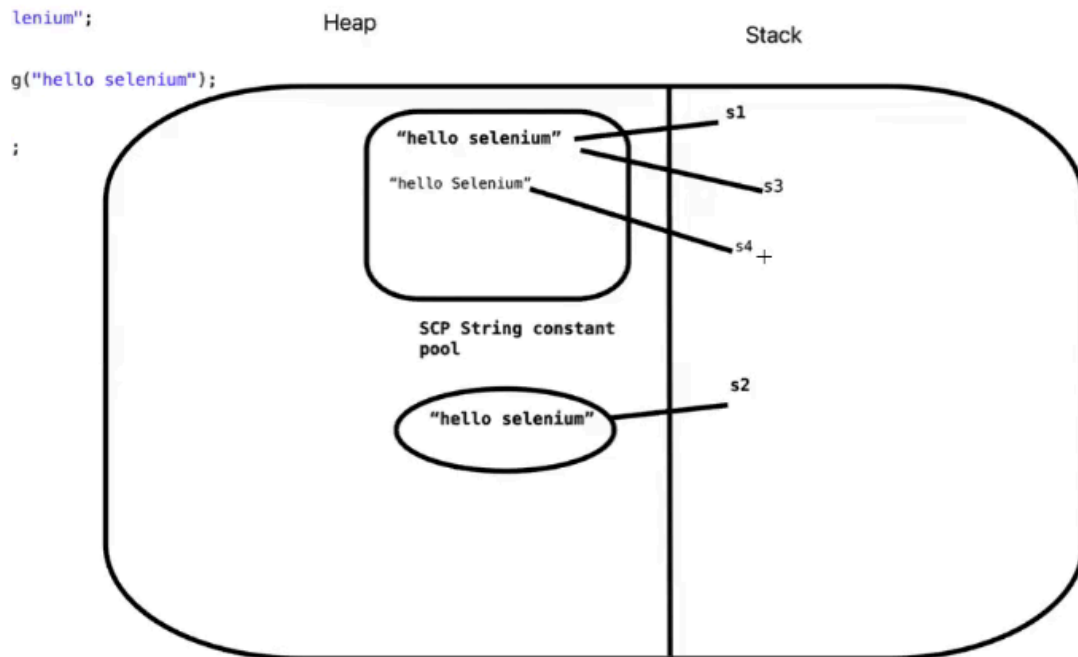
True

True

```
21      System.out.println(s1.equals(s3)); // true
22      System.out.println(s2 == s3); // false
```

False

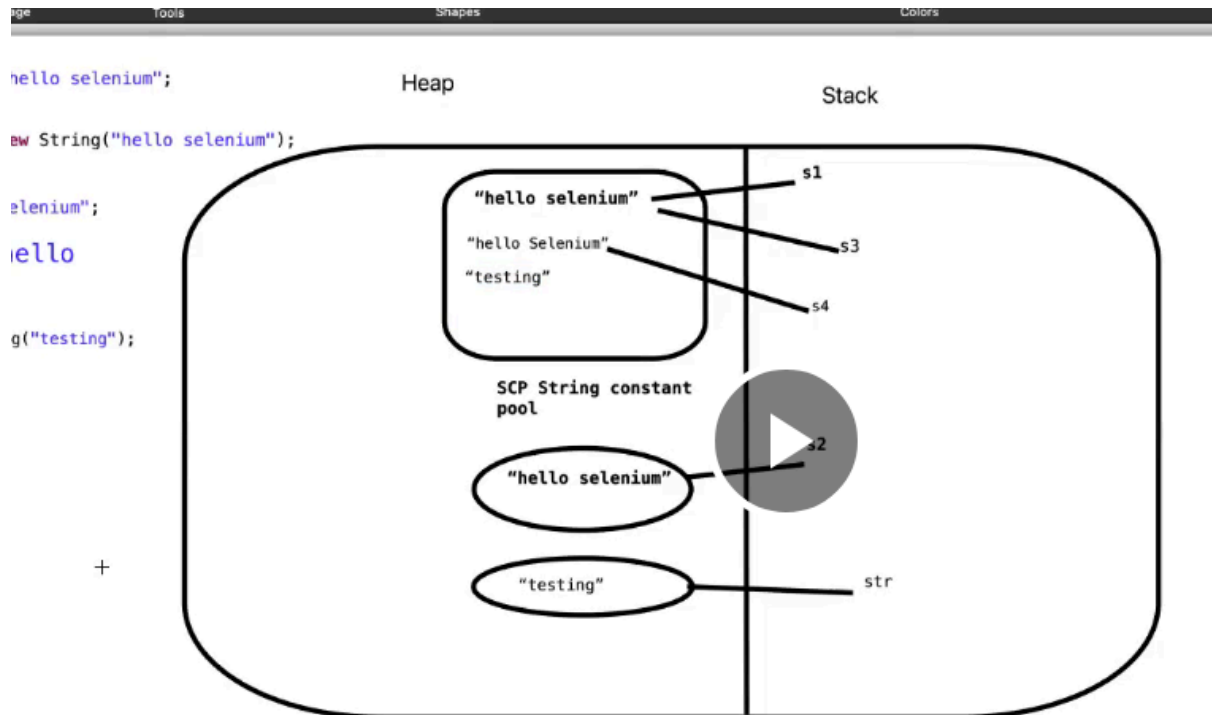
```
String s4 = "hello
Selenium";
```



Two values created in memory- (18.00)

When we write like this.

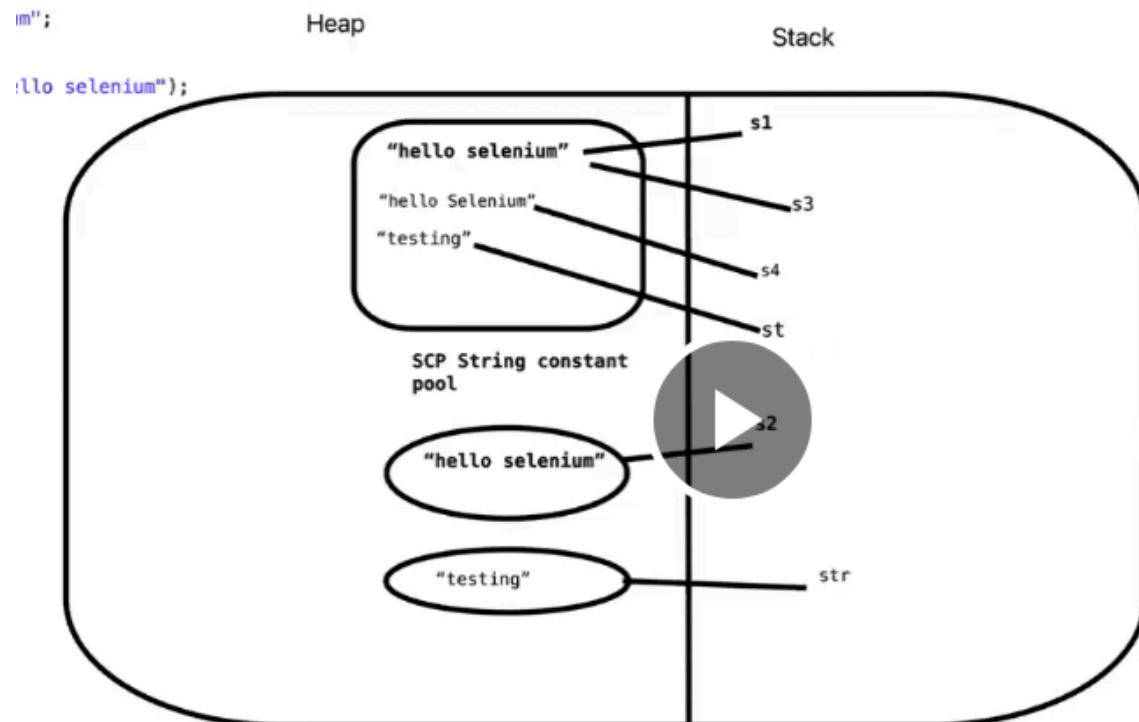
```
String str = new String("testing");
```



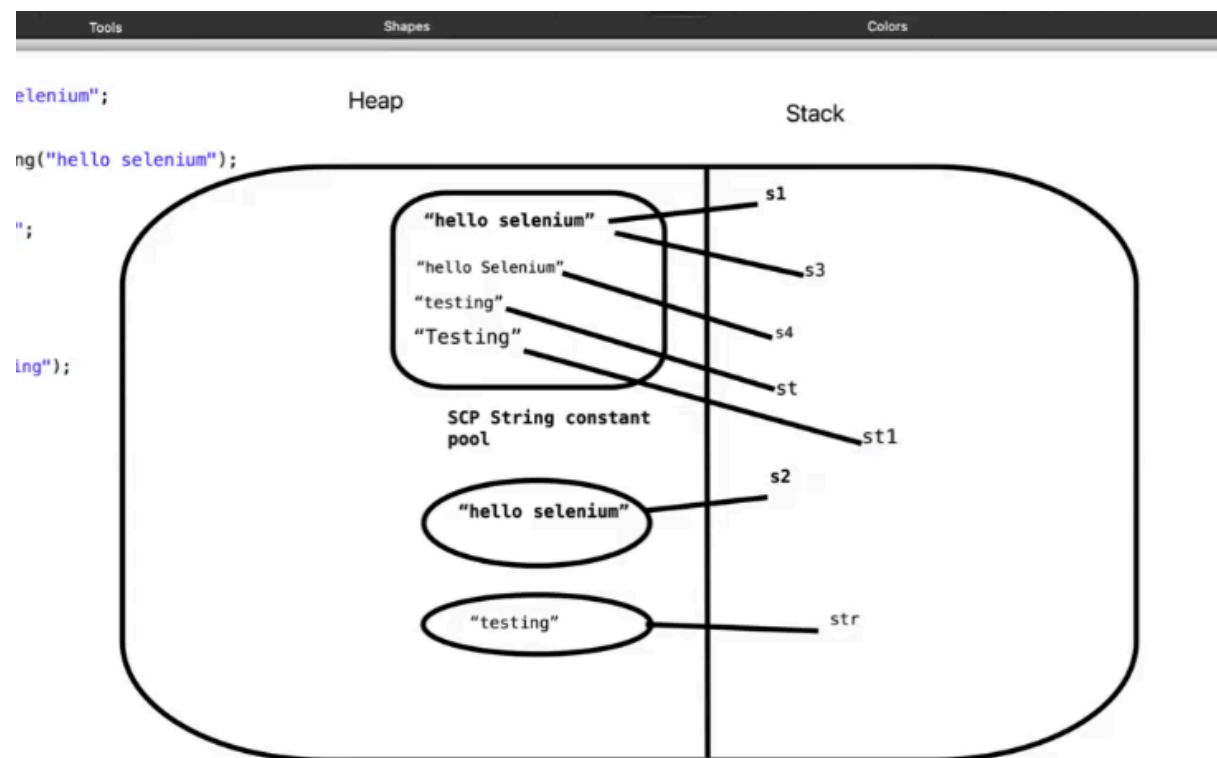
One will not have reference in scp, other will have.

```
String st = "testing";
```

Scp cannot hold duplicate values.



```
String st1 = "Testing";
```



```

26 String str = new String("testing");//2-->heap + SCP(testing)
27 String st = "testing";//SCP --> 0
28 String st1 = "Testing";

```

Why special treatment to strings-

Most commonly used data type.

Literals are best way of string creation to avoid duplication of value.

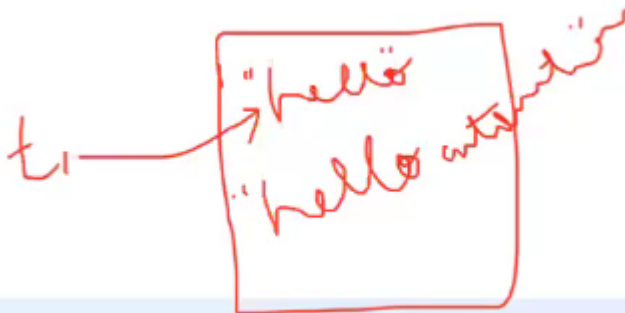
```

31 //
32 String t1 = "hello";
33 t1.concat("automation");
34 System.out.println(t1);

```

Hello

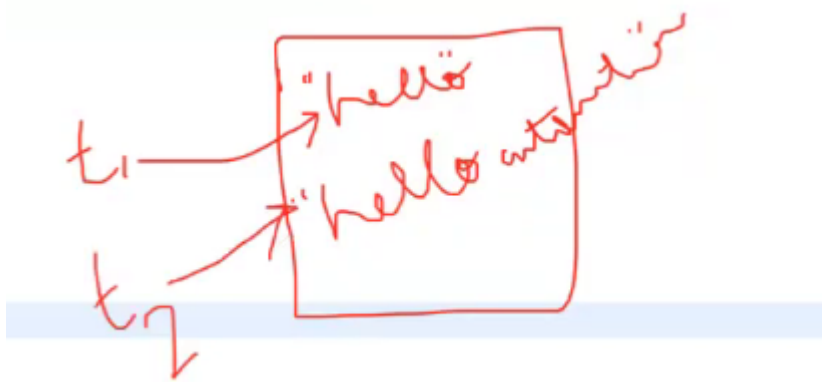
Strings are immutable, once created cannot update original value.



```

36 String t2 = "helloautomation";

```



```

31 //
32 String t1 = "hello";
33 t1 = t1.concat("automation");
34 System.out.println(t1); //hello
35
36 String t2 = "helloautomation";
37

```



Helloautomation

```

36 String t2 = "helloautomation";
37 System.out.println(t1.equals(t2));

```

True

```

31 //
32 String t1 = "hello";
33 t1.concat("automation");
34 System.out.println(t1); //hello
35
36 String t2 = "helloautomation";
37 System.out.println(t1.equals(t2));
38
39

```

False

Garbage collector applicable for literals also.

```

55 System.out.println("PASS");
56 }
57
58
59 //
60 String t1 = "hello world ";
61 System.out.println(t1.trim());
62 System.out.println(t1);
63

```

hello world
hello world
HELLO JAVA WORLD
hello java world
01/01/1990
helloworld
true
false
PASS

H.W
H.W

String builder and string buffer are mutable class.

Literals not allowed-

```

8   StringBuilder sb = new StringBuilder();
9
10  StringBuilder t1 = "hello";
11
12
13  //StringBuffer
14
15

```

Type mismatch: cannot convert from String to StringBuilder
1 quick fix available:
Change type of 't1' to 'String'

```

1 package StringManipulations;
2
3 public class StringBuilderAndStringBuffer {
4
5     public static void main(String[] args) {
6
7         //StringBuilder
8         StringBuilder sb = new StringBuilder("selenium");//1
9
10        sb.append("automation");
11        System.out.println(sb);
12
13    }
14 }

```

Console
seleniumautomation

Updates same entry.

Builder is from 1.4.


Buffer from 1.0.

Allowed in builder and buffer.

```

16
17 String st = "selenium";
18 //System.out.println(st.rever);
19
20 StringBuilder sb1 = new StringBuilder("selenium");//1
21
22 System.out.println(sb1.reverse());
23
24
25

```



Reverse of selenium printed.


Reverse normal string-

Pass it as parameter to builder or buffer and use reverse.

```

16
17 String st = "selenium";
18 //System.out.println(st.rever);
19
20 StringBuilder sb1 = new StringBuilder(st);//1
21 System.out.println(sb1.reverse());
22
23
24
25

```



Reverse of selenium printed.

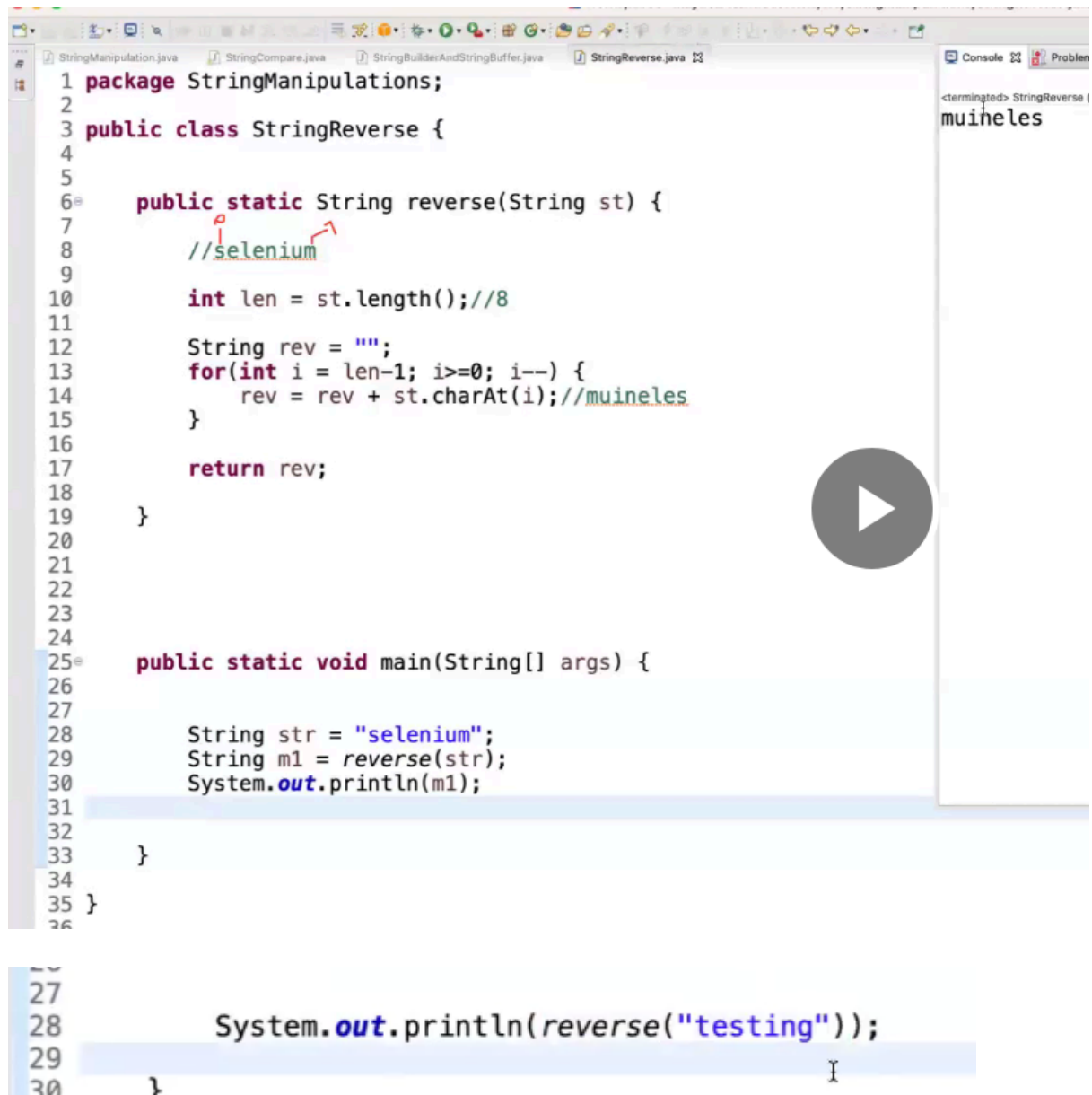
```

22
23 System.out.println(st);
24
25

```

Selenium.

Reverse string in coding way-



```

1 package StringManipulations;
2
3 public class StringReverse {
4
5
6     public static String reverse(String st) {
7         //selenium
8
9         int len = st.length(); //8
10
11         String rev = "";
12         for(int i = len-1; i>=0; i--) {
13             rev = rev + st.charAt(i); //muineles
14         }
15
16         return rev;
17     }
18
19 }
20
21
22
23
24
25     public static void main(String[] args) {
26
27         String str = "selenium";
28         String m1 = reverse(str);
29         System.out.println(m1);
30
31     }
32
33 }
34
35 }

```

```

27
28     System.out.println(reverse("testing"));
29
30 }

```

Reverse of testing printed.

```

29     System.out.println(reverse("T"));
30

```

T

Check for strings with length 1-

Optimised code, though earlier code works.

```

1 package StringManipulations;
2
3 public class StringReverse {
4
5     public static String reverse(String st) {
6
7         if (st.length() == 1) {
8             return st;
9         }
10
11         int len = st.length();
12
13         String rev = "";
14         for (int i = len - 1; i >= 0; i--) {
15             rev = rev + st.charAt(i); // muineles
16         }
17
18         return rev;
19     }
20
21     public static void main(String[] args) {
22
23         String str = "selenium";
24         String m1 = reverse(str);
25         System.out.println(m1);
26
27         System.out.println(reverse("testing"));
28         System.out.println(reverse("T"));
29
30
31     }
32 }
33
34 }
35

```

T

Pass null-

```

29     System.out.println(reverse("T"));
30     System.out.println(reverse(null));

```

```

Exception in thread "main" java.lang.NullPointerException: Cannot invoke "String.length()"
    at StringManipulations.StringReverse.reverse(StringReverse.java:7)
    at StringManipulations.StringReverse.main(StringReverse.java:30)

```

We did not handle null.

Null and length check-(1.18)

to handle using try catch add throw under try and see - 1.45.

```

5 public static String reverse(String st) {
6
7     //null check
8     if(st == null) {
9         throw new RuntimeException("VALUE CAN NOT BE NULL");
10    }
11
12    //length == 1
13    if (st.length() == 1) {
14        return st;
15    }

```

Exception in thread "main" java.lang.RuntimeException: VALUE CAN NOT BE NULL
 at StringManipulations.StringReverse.reverse(StringReverse.java:8)
 at StringManipulations.StringReverse.main(StringReverse.java:35)

paste reverstring5-

```

1 package com.day26;
2
3 import java.util.Iterator;
4
5 public class reversestring5 {
6
7     public static String reverseString(String name) {
8
9
10
11         //when string length is 1 then return as is.
12         if(name.length()==1) {
13             return name;
14         }
15
16         //if null check is not first then what happens.
17         //program terminates abruptly when null comes.
18         //we need to use try catch to solve this or test null values
19         //from main at last.
20         //Dead code
21         if(name==null) {
22             throw new RuntimeException("string cannot be null");
23         }
24
25         int len=name.length();
26
27         String rev="";

```

```

26
27     String rev="";
28
29     for(int i=len-1;i>=0;i--) {
30
31         rev=rev+name.charAt(i);
32     }
33
34     return rev;
35 }
36
37 public static void main(String[] args) {
38
39     String s31=null;
40     String s4=reverseString(s31);
41     System.out.println(s4);
42
43     String s1="selenium";
44     String s2=reverseString(s1);
45     System.out.println(s2);
46
47     String s21="k";
48     String s3=reverseString(s21);
49     System.out.println(s3);
50

```

```

48     String s3=reverseString(s21);
49     System.out.println(s3);
50
51
52
53 }
54
55 }
56
57 //Exception in thread "main" java.lang.NullPointerException:
58 //  Cannot invoke "String.length()" because "name" is null
59 //at com.day26.reversestring5.reverseString(reversestring5.java:12)
60 //at com.day26.reversestring5.main(reversestring5.java:36)
61
62

```

paste reverstring4-

```
reversestring4.java X
1 package com.day26;
2
3 import java.util.Iterator;
4
5 public class reversestring4 {
6
7     public static String reverseString(String name) {
8
9         //first always null check for string.
10        //null has to be compared with ==.
11        if(name==null) {
12            throw new RuntimeException("string cannot be null");
13        }
14
15        //when string length is 1 then return as is.
16        if(name.length()==1) {
17            return name;
18        }
19
20        int len=name.length();
21
22        String rev="";
23
24        for(int i=len-1;i>=0;i--) {
25
26            rev=rev+name.charAt(i);
27        }
28    }
```

```

25
26         rev=rev+name.charAt(i);
27     }
28
29     return rev;
30 }
31
32 public static void main(String[] args) {
33
34     String s1="selenium";
35     String s2=reverseString(s1);
36     System.out.println(s2);
37
38     String s21="k";
39     String s3=reverseString(s21);
40     System.out.println(s3);
41
42
43     String s31=null;
44     String s4=reverseString(s31);
45     System.out.println(s4);
46 }
47
48 }
49

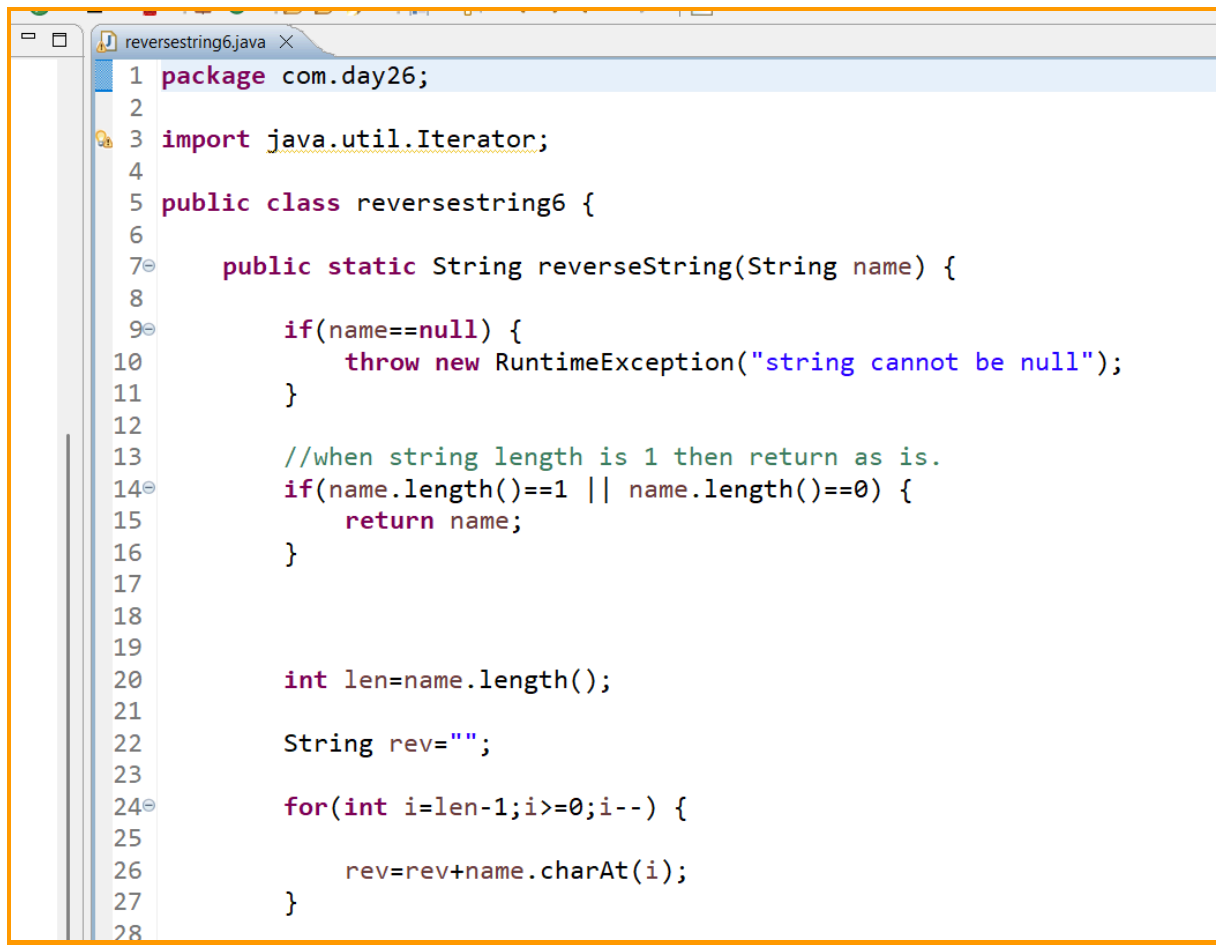
```

```

47
48 }
49
50 //muineles
51 //k
52 //Exception in thread "main" java.lang.RuntimeException: string cannot be null
53 //   at com.day26.reversestring4.reverseString(reversestring4.java:11)
54 //   at com.day26.reversestring4.main(reversestring4.java:43)
55
56

```

paste reversestring6-



```
1 package com.day26;
2
3 import java.util.Iterator;
4
5 public class reversestring6 {
6
7     public static String reverseString(String name) {
8
9         if(name==null) {
10             throw new RuntimeException("string cannot be null");
11         }
12
13         //when string length is 1 then return as is.
14         if(name.length()==1 || name.length()==0) {
15             return name;
16         }
17
18
19         int len=name.length();
20
21         String rev="";
22
23         for(int i=len-1;i>=0;i--) {
24
25             rev=rev+name.charAt(i);
26         }
27
28     }
```

```

26         rev=rev+name.charAt(i);
27     }
28
29     return rev;
30 }
31
32 public static void main(String[] args) {
33
34
35
36     String s1="selenium";
37     String s2=reverseString(s1);
38     System.out.println(s2);
39
40     String s21="k";
41     String s3=reverseString(s21);
42     System.out.println(s3);
43
44     String s5=" ";
45     String s6=reverseString(s5);
46     System.out.println(s6);

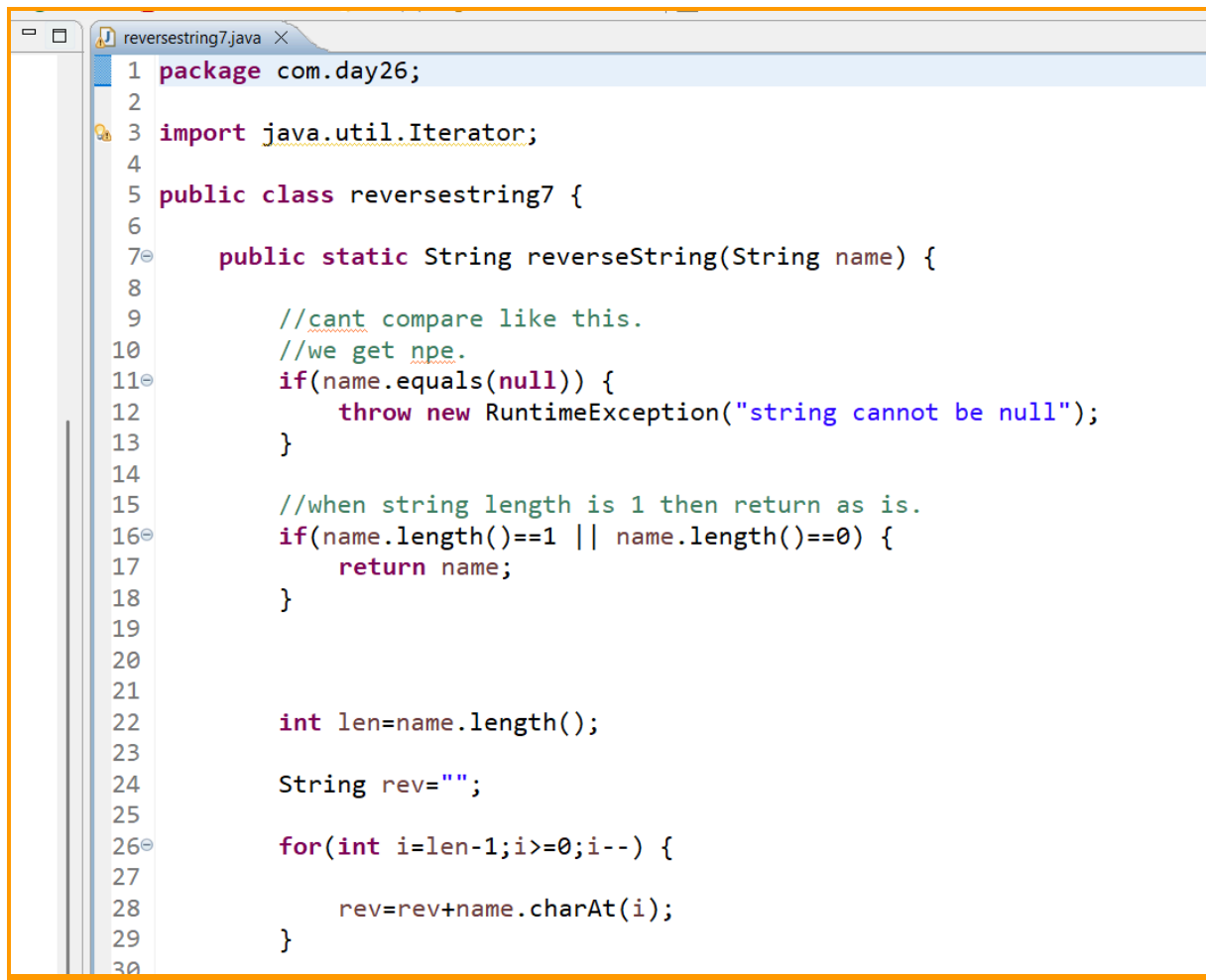
```

```

45     String s6=reverseString(s5);
46     System.out.println(s6);
47
48     String s61="";
49     String s7=reverseString(s61);
50     System.out.println(s7);
51
52     String s31=null;
53     String s4=reverseString(s31);
54     System.out.println(s4);
55
56
57
58 }
59
60 }
61
62 //muineles
63 //k
64 //
65 //
66 //Exception in thread "main" java.lang.RuntimeException: string cannot be null
67 //   at com.day26.reversestring6.reverseString(reversestring6.java:10)
68 //   at com.day26.reversestring6.main(reversestring6.java:53)
69

```

paste reverstring7-



```
1 package com.day26;
2
3 import java.util.Iterator;
4
5 public class reversestring7 {
6
7     public static String reverseString(String name) {
8
9         //cant compare like this.
10        //we get npe.
11        if(name.equals(null)) {
12            throw new RuntimeException("string cannot be null");
13        }
14
15        //when string length is 1 then return as is.
16        if(name.length()==1 || name.length()==0) {
17            return name;
18        }
19
20
21
22        int len=name.length();
23
24        String rev="";
25
26        for(int i=len-1;i>=0;i--) {
27
28            rev=rev+name.charAt(i);
29        }
30    }
```

```

28         rev=rev+name.charAt(i);
29     }
30
31     return rev;
32 }
33
34 public static void main(String[] args) {
35
36
37
38     String s1="selenium";
39     String s2=reverseString(s1);
40     System.out.println(s2);
41
42     String s21="k";
43     String s3=reverseString(s21);
44     System.out.println(s3);
45
46     String s5=" ";
47     String s6=reverseString(s5);
48     System.out.println(s6);

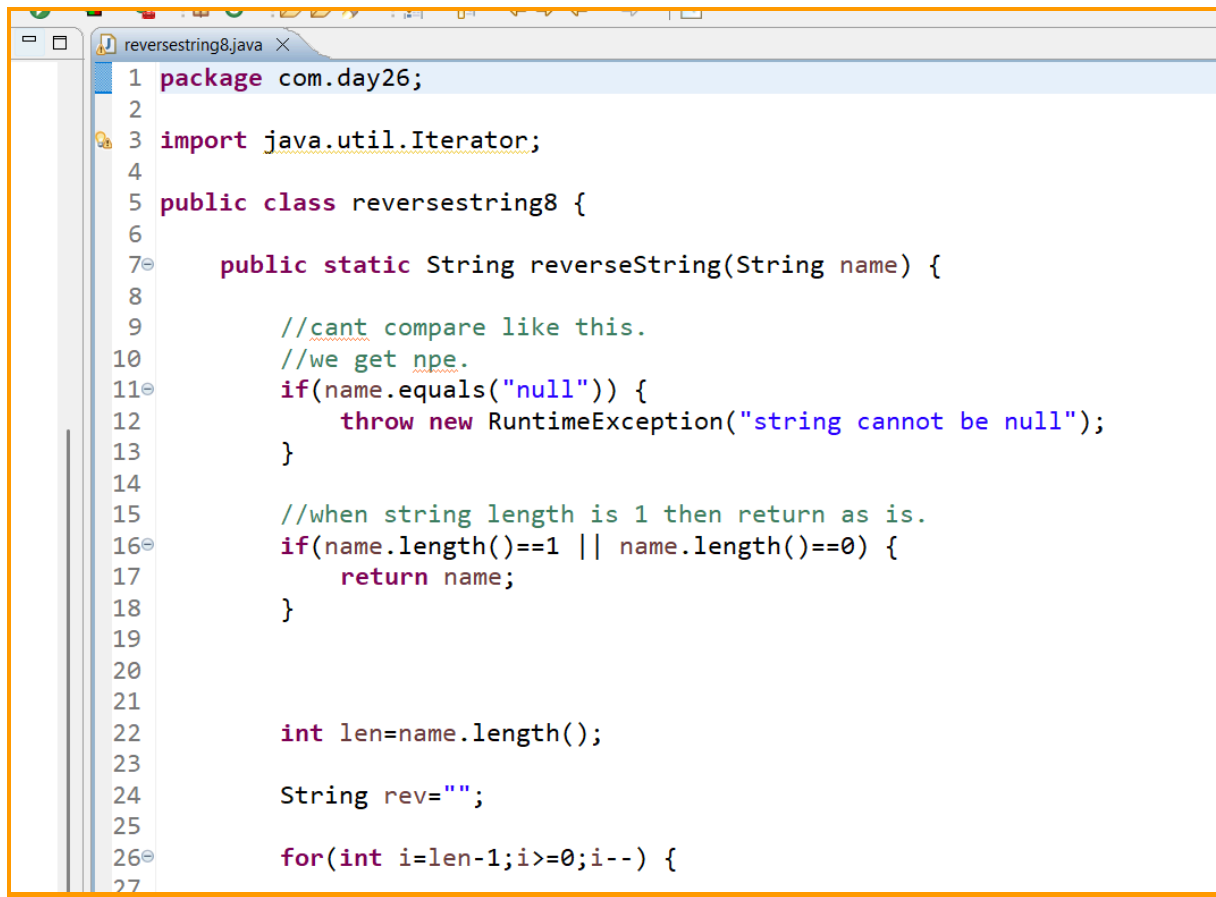
```

```

49
50     String s31="";
51     String s7=reverseString(s61);
52     System.out.println(s7);
53
54     String s31=null;
55     String s4=reverseString(s31);
56     System.out.println(s4);
57
58
59
60 }
61
62 }
63
64 //muineles
65 //k
66 //
67 //
68 //Exception in thread "main" java.lang.NullPointerException:
69 //  Cannot invoke "String.equals(Object)" because "name" is null
70 //  at com.day26.reversestring7.reverseString(reversestring7.java:9)
71 //  at com.day26.reversestring7.main(reversestring7.java:53)
72

```

paste reverstring8-



```
1 package com.day26;
2
3 import java.util.Iterator;
4
5 public class reversestring8 {
6
7     public static String reverseString(String name) {
8
9         //cant compare like this.
10        //we get npe.
11        if(name.equals("null")) {
12            throw new RuntimeException("string cannot be null");
13        }
14
15        //when string length is 1 then return as is.
16        if(name.length()==1 || name.length()==0) {
17            return name;
18        }
19
20
21
22        int len=name.length();
23
24        String rev="";
25
26        for(int i=len-1;i>=0;i--) {
27
```

```

25
26     for(int i=len-1;i>=0;i--) {
27
28         rev=rev+name.charAt(i);
29     }
30
31     return rev;
32 }
33
34 public static void main(String[] args) {
35
36
37
38     String s1="selenium";
39     String s2=reverseString(s1);
40     System.out.println(s2);
41
42     String s21="k";
43     String s3=reverseString(s21);
44     System.out.println(s3);
45
46     String s5=" ";
47     String s6=reverseString(s5);
48     System.out.println(s6);
49

```

```

47     String s6=reverseString(s5);
48     System.out.println(s6);
49
50     String s61="";
51     String s7=reverseString(s61);
52     System.out.println(s7);
53
54     String s31=null;
55     String s4=reverseString(s31);
56     System.out.println(s4);
57
58
59
60 }
61
62 }
63
64 //muineles
65 //k
66 //
67 //
68 //Exception in thread "main" java.lang.NullPointerException:
69 //  Cannot invoke "String.equals(Object)" because "name" is null
70 //  at com.day26.reversestring8.reverseString(reversestring8.java:11)
71 //  at com.day26.reversestring8.main(reversestring8.java:55)
72

```

paste reverstring9-

```
reversestring9.java x
1 package com.day26;
2
3 import java.util.Iterator;
4
5 public class reversestring9 {
6
7     public static String reverseString(String name) {
8
9         if(name==null) {
10             try {
11                 throw new RuntimeException("string cannot be null");
12             } catch (Exception e) {
13                 System.out.println(e);
14             }
15         }
16     }
17
18     //when string length is 1 then return as is.
19     if(name.length()==1 || name.length()==0) {
20         return name;
21     }
22
23
24
25     int len=name.length();
26 }
```

```
24
25     int len=name.length();
26
27     String rev="";
28
29     for(int i=len-1;i>=0;i--) {
30
31         rev=rev+name.charAt(i);
32     }
33
34     return rev;
35 }
36
37 public static void main(String[] args) {
38
39
40
41     String s1="selenium";
42     String s2=reverseString(s1);
43     System.out.println(s2);
44
45     String s21="k";
46     String s3=reverseString(s21);
47     System.out.println(s3);
48 }
```



```

45     String s21= "K" ;
46     String s3=reverseString(s21);
47     System.out.println(s3);
48
49     String s5=" ";
50     String s6=reverseString(s5);
51     System.out.println(s6);
52
53     String s61="";
54     String s7=reverseString(s61);
55     System.out.println(s7);
56
57     String s31=null;
58     String s4=reverseString(s31);
59     System.out.println(s4);
60
61
62
63 }
64
65 }

```

```

64
65 }
66
67 //muineles
68 //k
69 //
70 //
71 //java.lang.RuntimeException: string cannot be null
72 //Exception in thread "main" java.lang.NullPointerException:
73 //  Cannot invoke "String.length()" because "name" is null
74 //  at com.day26.reversestring9.reverseString(reversestring9.java:19)
75 //  at com.day26.reversestring9.main(reversestring9.java:58)
76
77

```

past reverstring10-

```

reversestring10.java x
1 package com.day26;
2
3 import java.util.Iterator;
4
5 public class reversestring10 {
6
7     public static String reverseString(String name) {
8
9         try {
10             if(name==null) {
11                 System.out.println("name is null");
12             }
13         } catch (Exception e) {
14             System.out.println(e);
15         }
16
17
18
19         //when string length is 1 then return as is.
20         if(name.length()==1 || name.length()==0) {
21             return name;
22         }
23

```

```

21         return name;
22     }
23
24
25
26     int len=name.length();
27
28     String rev="";
29
30     for(int i=len-1;i>=0;i--) {
31
32         rev=rev+name.charAt(i);
33     }
34
35     return rev;
36 }
37
38 public static void main(String[] args) {
39

```

```

37
38     public static void main(String[] args) {
39
40
41
42         String s1="selenium";
43         String s2=reverseString(s1);
44         System.out.println(s2);
45
46         String s21="k";
47         String s3=reverseString(s21);
48         System.out.println(s3);
49
50         String s5=" ";
51         String s6=reverseString(s5);
52         System.out.println(s6);
53
54         String s61="";
55         String s7=reverseString(s61);
56         System.out.println(s7);
57
58         String s31=null;
59         String s4=reverseString(s31);
60         System.out.println(s4);

```

```

59         String s4=reverseString(s31);
60         System.out.println(s4);
61
62
63
64     }
65
66 }
67
68 //muineles
69 //k
70 //
71 //
72 //name is null
73 //Exception in thread "main" java.lang.NullPointerException:
74 //     Cannot invoke "String.length()" because "name" is null
75 //     at com.day26.reversestring10.reverseString(reversestring10.java:20)
76 //     at com.day26.reversestring10.main(reversestring10.java:59)
77
78

```

Handle length as zero-

```

11
12 //length == 1 or 0
13 if (st.length() == 1 || st.length() == 0) {
14     return st;
15 }

```

```

37 System.out.println(reverse(""));
38

```

```

39
40

```

```

39 System.out.println(reverse(" "));
40

```

```

41

```

Datatype mismatch-

```

39 System.out.println(reverse(" "));
40 System.out.println(reverse(123));
41

```

123-

```

39 System.out.println(reverse(" "));
40 System.out.println(reverse("123"));
41

```

321

Null in double quote-

```

40 System.out.println(reverse("123"));
41 System.out.println(reverse("null"));

```

Llun 1.24.13

```

42 System.out.println(reverse("testing "));

```

```

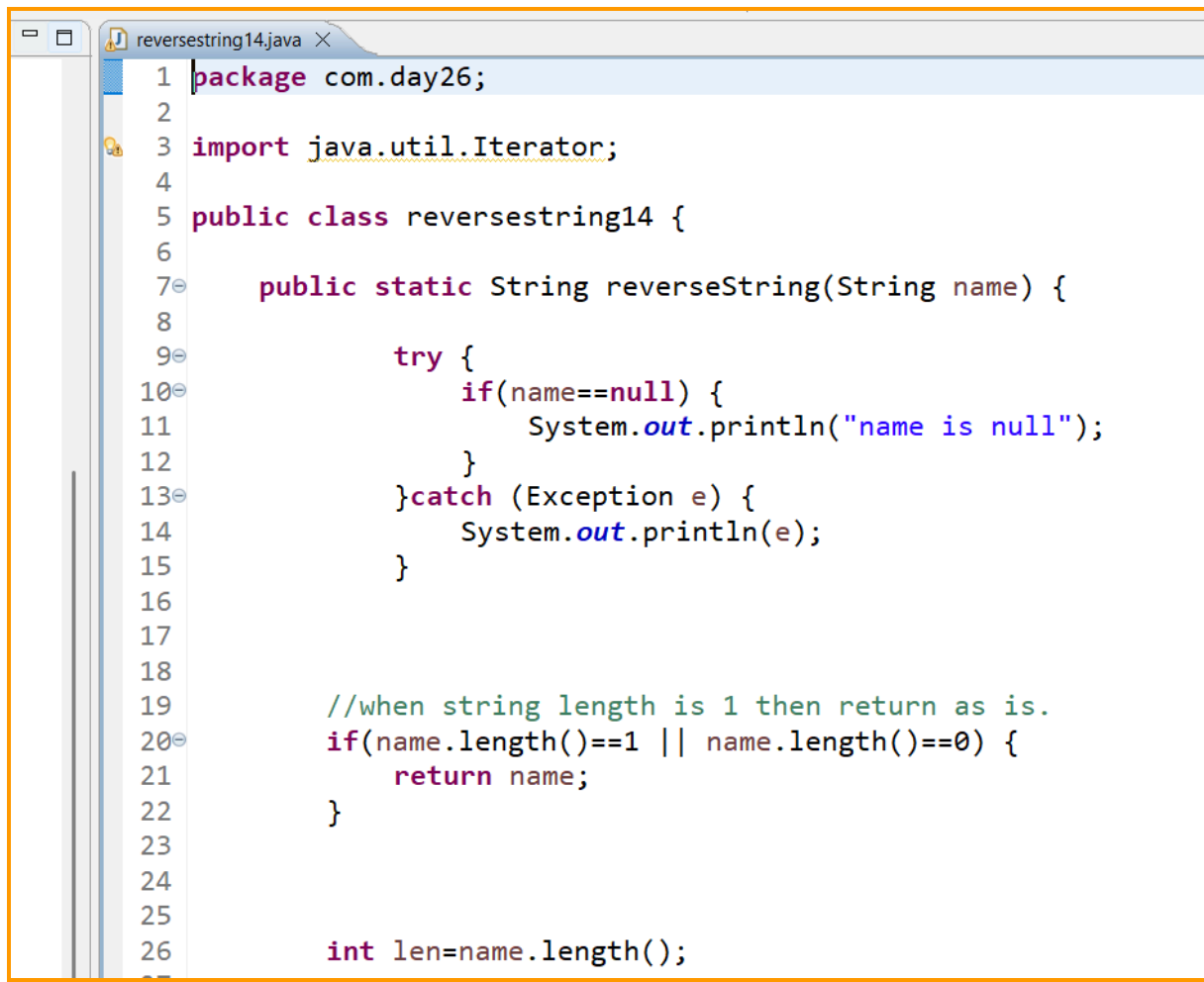
43
44

```

To ignore space-

```
StringManipulation.java StringCompare.java StringBulderAndStringBuffer.java StringReverse.java
1 package StringManipulations;
2
3 public class StringReverse {
4
5     public static String reverse(String st) {
6
7         //null check
8         if(st == null) {
9             throw new RuntimeException("VALUE CAN NOT BE NULL");
10        }
11
12        //length == 1 or 0
13        if (st.length() == 1 || st.length() == 0) {
14            return st;
15        }
16
17        int len = st.length();
18
19        String rev = "";
20        for (int i = len - 1; i >= 0; i--) {
21            rev = rev + st.charAt(i); // muineles
22        }
23
24        return rev.trim();
25
26    }
```

paste reversestring14-



```
1 package com.day26;
2
3 import java.util.Iterator;
4
5 public class reversestring14 {
6
7     public static String reverseString(String name) {
8
9         try {
10             if(name==null) {
11                 System.out.println("name is null");
12             }
13         } catch (Exception e) {
14             System.out.println(e);
15         }
16
17
18
19         //when string length is 1 then return as is.
20         if(name.length()==1 || name.length()==0) {
21             return name;
22         }
23
24
25
26         int len=name.length();
```

```
25
26     int len=name.length();
27
28     String rev="";
29
30     for(int i=len-1;i>=0;i--) {
31
32         rev=rev+name.charAt(i);
33     }
34
35     //to ignore spaces when returning value.
36     return rev.trim();
37 }
38
39 public static void main(String[] args) {
40
41
42
43     String s1="selenium";
44     String s2=reverseString(s1);
45     System.out.println(s2);
46
47     String s21="k";
48     String s3=reverseString(s21);
49     System.out.println(s3);
50 }
```

```
48      String s3=reverseString(s21);
49      System.out.println(s3);
50
51      String s5=" ";
52      String s6=reverseString(s5);
53      System.out.println(s6);
54
55      String s61="";
56      String s7=reverseString(s61);
57      System.out.println(s7);
58
59      String s41="123";
60      String s8=reverseString(s41);
61      System.out.println(s8);
62
63      String s411="null";
64      String s81=reverseString(s411);
65      System.out.println(s81);
```

```
64      String s81=reverseString(s411);
65      System.out.println(s81);
66
67      String s4111=" testing null ";
68      String s811=reverseString(s4111);
69      System.out.println(s811);
70
71      String s31=null;
72      String s4=reverseString(s31);
73      System.out.println(s4);
74
```



```
73         System.out.println(s4);
74
75
76
77
78
79     }
80
81 }
82
83 //muineles
84 //k
85 //
86 //
87 //321
88 //llun
89 //llun gnitset
90 //name is null
91 //Exception in thread "main" java.lang.NullPointerException:
92 //  Cannot invoke "String.length()" because "name" is null
93 //  at com.day26.reversestring14.reverseString(reversestring14.java:20)
94 //  at com.day26.reversestring14.main(reversestring14.java:72)
95
```



Homework – reverse this. Use split, because the position should not change, we don't want to do complete reversal. we want

```
System.out.println(reverse("testing "));
```

```
String test = "Java Python Ruby"; //avaJ
```

not reverse

character by character.

```
String test = "Java Python Ruby";
```

paste reverse string 15-

```

1 package com.day26;
2
3 import java.util.Arrays;
4 import java.util.Iterator;
5
6 public class reversestring15 {
7
8     public static String reverseString(String name) {
9
10        if(name==null) {
11            System.out.println("name is null");
12        }
13
14        //when string length is 1 then return as is.
15        if(name.length()==1 || name.length()==0) {
16            return name;
17        }
18
19        String s1[]=name.split(" ");
20        System.out.println(Arrays.toString(s1));//[java, ruby, python]
21
22        int len=s1.length;
23        System.out.println(len);//3
24
25        String individualones="";
26        String rev="";

```

```

25
26     String individualones="";
27     String rev="";
28
29     //when we have this reversed word outside.
30     //then every loop the previous word is also appended to current word.
31     String reversedwords="";
32     for(int i=0;i<=len-1;i++) {
33 //         String reversedwords="";
34
35         individualones=s1[i];
36         System.out.println(individualones);
37
38         int len1=s1[i].length();
39
40         for(int j=len1-1; j>=0; j--) {
41             reversedwords=reversedwords+individualones.charAt(j);
42         }
43         rev=rev+reversedwords+" ";
44     }
45     return rev + " ";
46 }

```

```
45         return rev + " ";
46     }
47
48     public static void main(String[] args) {
49
50
51
52         String test="java ruby python";
53         String s81=reverseString(test);
54         System.out.println(s81);
55
56
57
58
59
60     }
61 }
62
63
64
65 //[java, ruby, python]
66 //3
67 //java
68 //ruby
69 //python
70 //avaj avajybur avajburnohtyp
71
```

paste reversestring16-

```

reversestring16.java X
1 package com.day26;
2
3 import java.util.Arrays;
4 import java.util.Iterator;
5
6 public class reversestring16 {
7
8     public static String reverseString(String name) {
9
10        if(name==null) {
11            System.out.println("name is null");
12        }
13
14        //when string length is 1 then return as is.
15        if(name.length()==1 || name.length()==0) {
16            return name;
17        }
18
19        String s1[]=name.split(" ");
20        System.out.println(Arrays.toString(s1));//[java, ruby, python]
21
22        int len=s1.length;
23        System.out.println(len);//3
24
25        String individualones="";
26        String rev="";
27
28

```

```

27        String individualones="";
28        String rev="";
29
30        for(int i=0;i<=len-1;i++) {
31            //have it inside loop.
32            //so for every loop it will be fresh and blank.
33            String reversedwords="";
34
35            individualones=s1[i];
36            System.out.println(individualones);
37
38            int len1=s1[i].length();
39
40            for(int j=len1-1; j>=0; j--) {
41                reversedwords=reversedwords+individualones.charAt(j);
42            }
43            //once outside the loop concat with the blank rev and add space.
44            rev=rev+reversedwords+" ";
45        }
46        return rev + " ";
47    }
48

```

```

46         return rev + " ";
47     }
48
49     public static void main(String[] args) {
50
51
52
53         String test="java ruby python";
54         String s81=reverseString(test);
55         System.out.println(s81);
56
57
58
59
60
61     }
62 }
63
64
65 //[java, ruby, python]
66 //3
67 //java
68 //ruby
69 //python
70 //avaj ybur nohtyp
71

```

When we give wrong split option –
it won't give error, when the delimiter not present
then returns complete string.

```

126
127     String u = "java_python";
128     String pu[] = u.split(";");
129     System.out.println(Arrays.toString(pu));
130

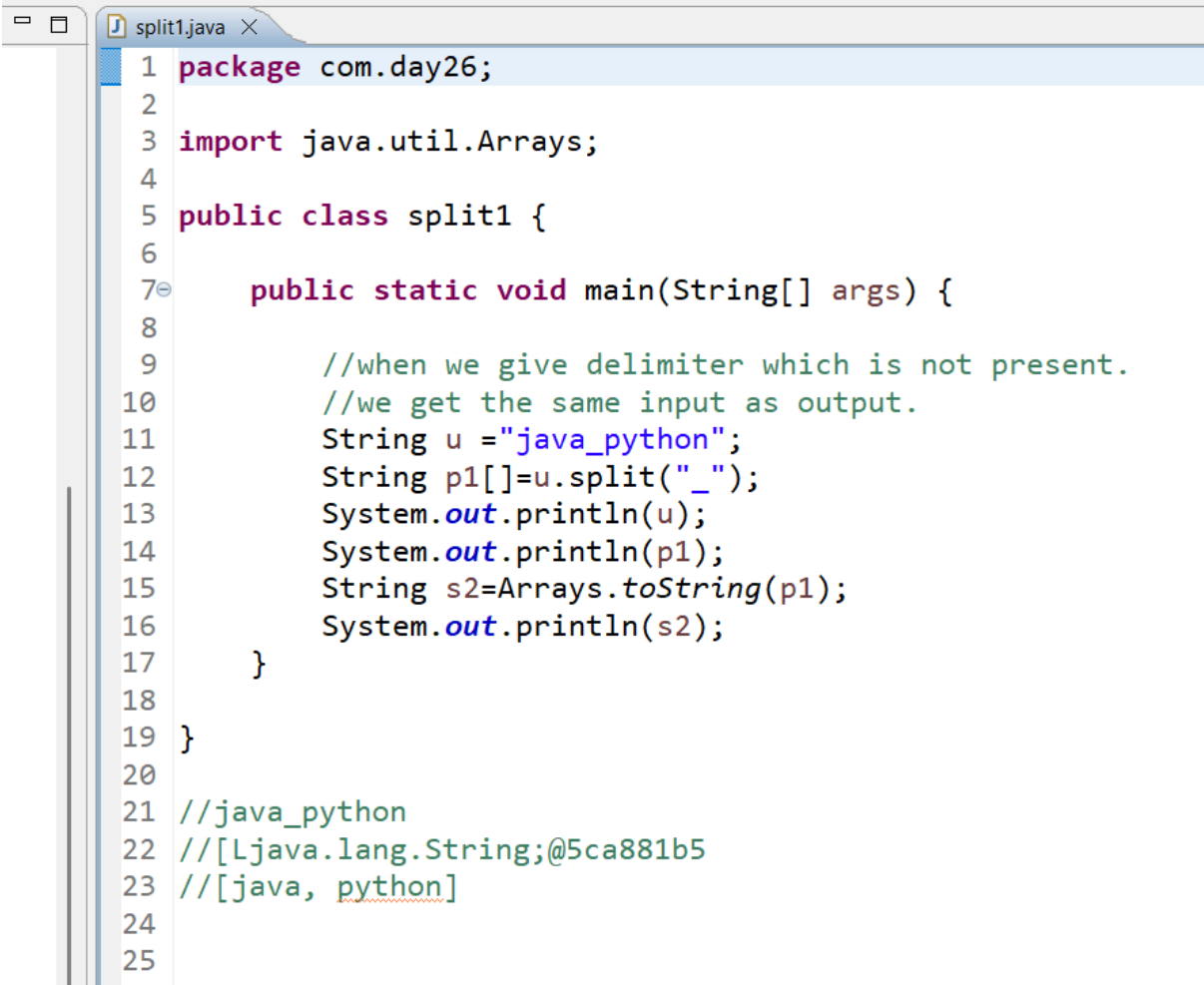
```

```

[java_python]

```

paste split1-

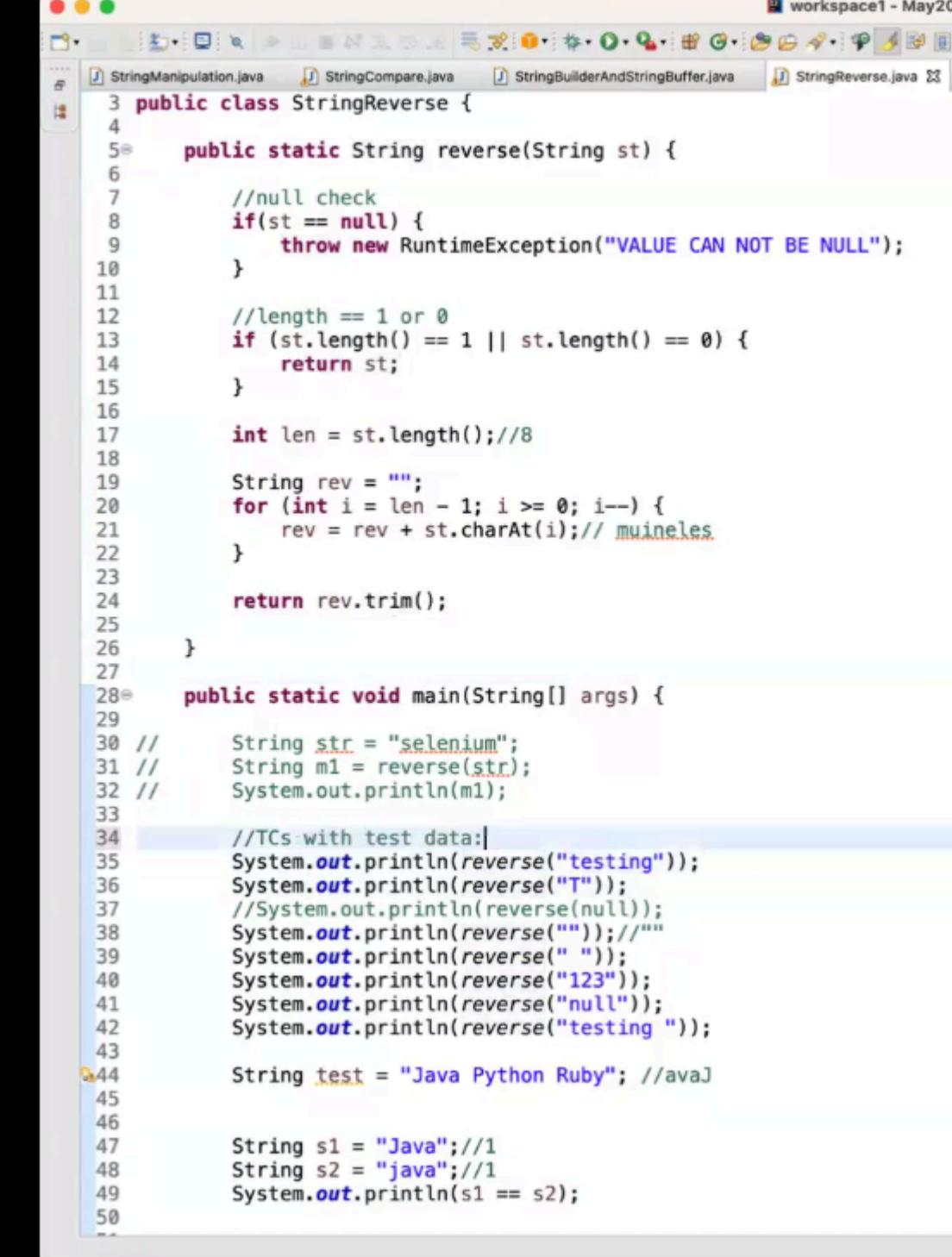


```
1 package com.day26;
2
3 import java.util.Arrays;
4
5 public class split1 {
6
7     public static void main(String[] args) {
8
9         //when we give delimiter which is not present.
10        //we get the same input as output.
11        String u ="java_python";
12        String p1[]=u.split("_");
13        System.out.println(u);
14        System.out.println(p1);
15        String s2=Arrays.toString(p1);
16        System.out.println(s2);
17    }
18 }
19
20
21 //java_python
22 //[Ljava.lang.String;@5ca881b5
23 //[java, python]
```

note-

always write null check for the strings first.

in interview create method and then call main method with different test data-



```

workspace1 - May20
StringManipulation.java StringCompare.java StringBuilderAndStringBuffer.java StringReverse.java
3 public class StringReverse {
4
5     public static String reverse(String st) {
6
7         //null check
8         if(st == null) {
9             throw new RuntimeException("VALUE CAN NOT BE NULL");
10        }
11
12        //length == 1 or 0
13        if (st.length() == 1 || st.length() == 0) {
14            return st;
15        }
16
17        int len = st.length();//8
18
19        String rev = "";
20        for (int i = len - 1; i >= 0; i--) {
21            rev = rev + st.charAt(i);// muineles
22        }
23
24        return rev.trim();
25    }
26
27    public static void main(String[] args) {
28
29        // String str = "selenium";
30        // String m1 = reverse(str);
31        // System.out.println(m1);
32
33        //TCs with test data:|
34        System.out.println(reverse("testing"));
35        System.out.println(reverse("T"));
36        //System.out.println(reverse(null));
37        System.out.println(reverse(""));
38        System.out.println(reverse(" "));
39        System.out.println(reverse("123"));
40        System.out.println(reverse("null"));
41        System.out.println(reverse("testing "));
42
43        String test = "Java Python Ruby"; //avaJ
44
45
46        String s1 = "Java";//1
47        String s2 = "java";//1
48        System.out.println(s1 == s2);
49
50

```