

JAVA Imp Programs: (Youtube ref: Naveen Automation)

1.Reverse a String

```
ReverseString.java
1 package testcases;
2
3 public class ReverseString {
4
5     public static void main(String[] args) {
6
7         String s = "Selenium";
8
9         //1. using for loop
10        int len = s.length(); //8
11        String rev = "";
12
13        for(int i =len-1; i>=0; i--){
14            rev = rev + s.charAt(i); //muineleS
15        }
16
17        System.out.println(rev);
18
19        //2. using StringBuffer class:
20        StringBuffer sf = new StringBuffer(s);
21        System.out.println(sf.reverse());
22
23
24    }
25
26 }
27
```

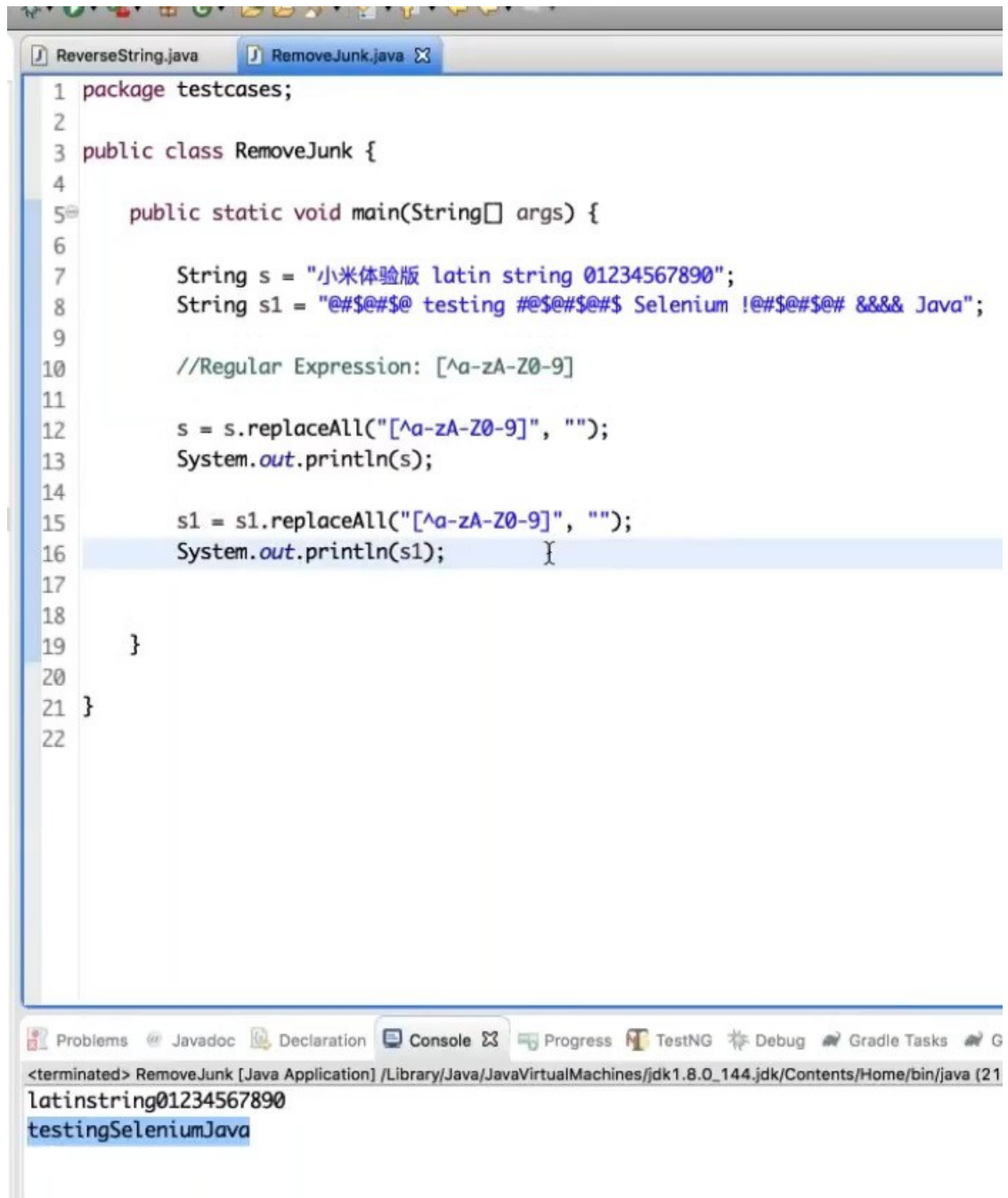
Problems Javadoc Declaration Console Progress Test

<terminated> ReverseString [Java Application] /Library/Java/JavaVirtualMachines/jdk

muineleS

muineleS

2. Remove or replace a junk string:



```
1 package testcases;
2
3 public class RemoveJunk {
4
5     public static void main(String[] args) {
6
7         String s = "小米体验版 latin string 01234567890";
8         String s1 = "@#$%$@ testing #@$%#$@$ Selenium !@$%#$@# &&& Java";
9
10        //Regular Expression: [^a-zA-Z0-9]
11
12        s = s.replaceAll("[^a-zA-Z0-9]", "");
13        System.out.println(s);
14
15        s1 = s1.replaceAll("[^a-zA-Z0-9]", "");
16        System.out.println(s1);
17
18    }
19 }
20
21 }
22 }
```

Problems Javadoc Declaration Console Progress TestNG Debug Gradle Tasks G

<terminated> RemoveJunk [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_144.jdk/Contents/Home/bin/java (21

latinstring01234567890

testingSeleniumJava

3.Reverse integer :

```
*ReverseInteger.java
1 package testcases;
2
3 public class ReverseInteger {
4
5     public static void main(String[] args) {
6
7         int num = 12345; // 54321
8         int rev = 0;
9
10        while(num != 0){
11            rev = rev * 10 + num % 10; //5432
12            num = num / 10; //12
13        }
14
15        System.out.println("Reverse num is::"+ rev);
16
17
18    }
19
20 }
21 }
22
```

Problems Javadoc Declaration Console Progress TestNG

terminated> ReverseInteger [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_144

Reverse num is::54321

4. Missing Number in an integer array:

```
2
3 public class MissingNumber {
4
5     public static void main(String[] args) {
6
7         //int a[] = {1,2,3,4,5,7,8,9,10,.....100};
8         //1+2+4+5 = 12
9         //1+2+3+4+5 = 15
10        //15-12 = 3
11
12        //22
13        //28 -22 = 6
14
15        int a[] = {1,2,3,4,5,7,8,9,10};
16        int sum = 0;
17        for(int i = 0; i<a.length; i++){
18            sum = sum + a[i]; //49
19        }
20        System.out.println(sum);
21
22        int sum1 = 0;
23        for(int j=1; j<=10; j++){
24            sum1 = sum1 + j; //55
25        }
26
27        System.out.println(sum1);
28
29        System.out.println("missing number is::"+ (sum1-sum));
30
31
32
33
```

Problems Javadoc Declaration Console Progress TestNG Debug

<terminated> MissingNumber [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_144.jdk/C

49

55

missing number is::6

5. Find largest & smallest values:



```
1 package testcases;
2
3 import java.util.Arrays;
4
5 public class SmallestAndLargestNumber {
6
7     public static void main(String[] args) {
8
9         int numbers[] = {-10, 24, 50, -88, 987656};
10
11         int largest = numbers[0];
12         int smallest = numbers[0];
13
14         for(int i = 1; i < numbers.length; i++){
15             if(numbers[i] > largest){
16                 largest = numbers[i]; //987656
17             }
18             else if(numbers[i] < smallest){
19                 smallest = numbers[i]; //-88
20             }
21         }
22
23         System.out.println("\n given array is:" + Arrays.toString(numbers));
24         System.out.println("Largest number is ::" + largest);
25         System.out.println("Smallest number is::" + smallest);
26
27     }
28 }
29
30 }
31
```

Problems Javadoc Declaration Console Progress TestNG Debug Gradle Tasks Gradle E

<terminated> SmallestAndLargestNumber [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_144.jdk/Contents/Hom

given array is: [-10, 24, 50, -88, 987656]
Largest number is :: 987656
Smallest number is :: -88

6. Finding the duplicate elements:



The screenshot shows an IDE window titled "DuplicateElements.java". The code defines a class `DuplicateElements` with a `main` method. It initializes a string array `names` with values: `["Java", "JavaScript", "Ruby", "C", "Python", "Java"]`. The first solution (commented as "worst solution") uses a nested loop to compare each element with every other element. The output in the console shows "duplicate element is::Java" and "duplicate element is::C".

```
1 package testcases;
2
3 public class DuplicateElements {
4
5     public static void main(String[] args) {
6
7         String names[] = {"Java", "JavaScript", "Ruby", "C", "Python", "Java"};
8
9         //1. compare each element: O(nxn) --- worst solution
10        for(int i = 0; i < names.length; i++){
11            for(int j = i+1; j < names.length; j++){
12
13                if(names[i].equals(names[j])){
14                    System.out.println("duplicate element is::" + names[i]);
15                }
16            }
17        }
18
19
20
21
22
23
24
25
26    }
27
28 }
29
```

Problems Javadoc Declaration Console Progress TestNG Debug Gradle Tasks Gradle Exec

<terminated> DuplicateElements [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_144.jdk/Contents/Home/bin/java (2

duplicate element is::Java
duplicate element is::C

```
//2. using HashSet: Java Collection: it stores unique values:
Set<String> store = new HashSet<String>();
for(String name : names){
    if(store.add(name) == false){
        System.out.println("duplicate element is::" + name);
    }
}
```

7. String Manipulation:

```
3 public class StringManipulation {
4
5     public static void main(String[] args) {
6
7         String str = "The rains have started here";
8         String str1 = "The rains Have started here";
9
10
11         System.out.println(str.length());
12
13         System.out.println(str.charAt(5));
14
15         System.out.println(str.indexOf('s')); //1st occurrence of s
16
17         System.out.println(str.indexOf('s', str.indexOf('s')+1)); //2nd occurrence of s
18
19         System.out.println(str.indexOf("have"));
20
21         System.out.println(str.indexOf("hello")); //-1
22
23         //string comparison:
24         System.out.println(str.equals(str1));
25
26         System.out.println(str.equalsIgnoreCase(str1));
27
28         //substring:
29         System.out.println(str.substring(0, 9));
30
31     }
32
33 }
```

Problems Javadoc Declaration Console Progress TestNG Debug Gradle Tasks Gradle Executions

<terminated> StringManipulation [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_144.jdk/Contents/Home/bin/java (25-Nov-2017)

```
27
a
8
15
10
-1
false
true
```



```

//substring:
System.out.println(str.substring(0, 9));

//trim:
String s = "    Hello World    ";
System.out.println(s.trim());

System.out.println(s.replace(" ", ""));

String date = "01-01-2017"; // 01/01/2017
System.out.println(date.replace("-", "/"));

//split:
String test = "Hello_World_Test_Selenium";
String testval[] = test.split("_");
for(int i=0 ; i< testval.length; i++){
    System.out.println(testval[i]);
}

String s2 = "cares";
System.out.println(s2.concat("s"));

String x = "Hello";
String y = "World";
int a = 100;
int b = 200;

System.out.println(x+y);
System.out.println(a+b);
System.out.println(x+y+a+b);
System.out.println(a+b+x+y);
System.out.println(x+y+(a+b));

```

```

The rains
Hello World
HelloWorld
01/01/2017
Hello
World
Test
Selenium
caress
HelloWorld
300
HelloWorld100200
300HelloWorld
HelloWorld300

```


8. Swaping two integer without using the third variable:

```
SwapWithoutThirdVar.java
1 package testcases;
2
3 public class SwapWithoutThirdVar {
4
5     public static void main(String[] args) {
6
7         int x = 5; //0101
8         int y = 10; //1010
9
10        //x = 10, y = 5
11
12        //1. with using third var : t
13        // int t;
14        // t = x; //5
15        // x = y; //10
16        // y = t; //5
17
18        //2. without using third var: using + operator
19        // x = x + y; //15
20        // y = x - y; //5
21        // x = x - y; // 10
22
23        //3. without using third var: using * operator
24        // x = x * y; //50
25        // y = x / y; //5
26        // x = x / y; //10
27
28        //4. using XOR: ^
29        x = x ^ y; //15 --> 1111
30        y = x ^ y; //10 --> 1010
31        x = x ^ y; // 5 --> 0101
32
33
34        System.out.println(x);
35        System.out.println(y);
36
37
38
39
40    }
41
42 }
43
```

Output:

10
5

9. Swaping two string without using the third variable:

```
1 package testcases;
2
3 public class StringSwap {
4
5     public static void main(String[] args) {
6
7         //WAP to swap strings without using temp/third variable:
8
9         String a = "Hello";
10        String b = "World";
11
12        System.out.println("before swapping: ");
13        System.out.println("the value of a is:"+ a);
14        System.out.println("the value of b is:"+ b);
15
16        //1. append a and b:
17        a = a+b; //HelloWorld
18
19        //2. Store initial string a in String b:
20        b = a.substring(0, a.length()-b.length());
21
22        //3. Store initial string b in String a:
23        a = a.substring(b.length());
24
25        System.out.println("the value of a and b after swapping");
26
27        System.out.println("the value of a is:"+ a);
28        System.out.println("the value of b is:"+ b);
29
30
31
32

```

Problems Javadoc Declaration Console Progress Results of running method

<terminated> StringSwap [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_144.jdk/Contents/
before swapping:
the value of a is:Hello
the value of b is:World
the value of a and b after swapping
the value of a is:World
the value of b is:Hello

10. Prime Number:

```
PrimeNumber.java
JavaQuestions > src > Questions > PrimeNumber > isPrimeNumber(int) : boolean
1 package Questions;
2
3 public class PrimeNumber {
4     //2 is the lowest prime number
5     //3
6     //num = 20
7     public static boolean isPrimeNumber(int num){
8
9         //edge/corner cases:
10        if(num<=1){
11            return false;
12        }
13
14        for(int i=2; i<num; i++){
15            if(num % i == 0){
16                return false;
17            }
18        }
19        return true;
20    }
21 }
22
```

PrimeNumber.java

JavaQuestions src Questions PrimeNumber getPrimeNumbers(int) : void

```
21     }
22
23
24     public static void getPrimeNumbers(int num){
25         for(int i=2; i<=num; i++){
26             if(isPrimeNumber(i))
27                 System.out.println(i + " ");
28         }
29     }
30
31
32
33
34     public static void main(String[] args) {
35
36         System.out.println("2 is prime number: " + isPrimeNumber(2));
37         System.out.println("3 is prime number: " + isPrimeNumber(3));
38         System.out.println("10 is prime number: " + isPrimeNumber(10));
39         System.out.println("17 is prime number: " + isPrimeNumber(17));
40
41         System.out.println("0 is prime number: " + isPrimeNumber(0));
42         System.out.println("1 is prime number: " + isPrimeNumber(1));
43
44         System.out.println("-3 is prime number: " + isPrimeNumber(-3));
45
46         getPrimeNumbers(7);
47         getPrimeNumbers(13);
48         getPrimeNumbers(20);
49     }
```

Problems Javadoc Declaration Console

<terminated> PrimeNumber (1) [Java Application] /Library/Java/

```
2 is prime number: true
3 is prime number: true
10 is prime number: false
17 is prime number: true
0 is prime number: false
1 is prime number: false
-3 is prime number: false
```

```
2
3
5
7
2
3
5
7
11
13
2
3
5
7
11
13
17
19
```

11. Factorial Number:

```
1 package Questions;
2
3 public class FactorialNumber {
4
5     //fact of 3 = 3 * 2 *1 = 6
6     //fact of 4 = 4 * 3 * 2 *1 = 24
7     //5 = 5*4*3*2*1 = 120
8     //10=10*9*8*7----*1
9     //1 = 1
10    //fact(0) = 1
11
12    //1. without recursive -- use for loop
13
14    public static int factorial(int num){
15        int fact=1;
16        if(num==0)
17            return 1;
18        for(int i=1; i<=num; i++){
19            fact=fact*i; //1*2*3*4
20        }
21        return fact;
22    }
23 }
```

13

14=

15

16

17

18

19

20

21

22

23

24

25

26=

27

28

29

30

31

32

33

34=

35

36

37

38

39

40

41

42

43

```
public static int factorial(int num){
```

```
    int fact=1;
```

```
    if(num==0)
```

```
        return 1;
```

```
    for(int i=1; i<=num; i++){
```

```
        fact=fact*i; //1*2*3*4
```

```
    }
```

```
    return fact;
```

```
}
```

```
//2. with Recursive function: a function is calli
```

```
public static int fact(int num){
```

```
    if(num==0)
```

```
        return 1;
```

```
    else
```

```
        return (num * fact(num-1));
```

```
}
```

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

```
    System.out.println(factorial(5));
```

```
    System.out.println(factorial(1));
```

```
    System.out.println(factorial(0));
```

```
    System.out.println(fact(4));
```

```
}
```

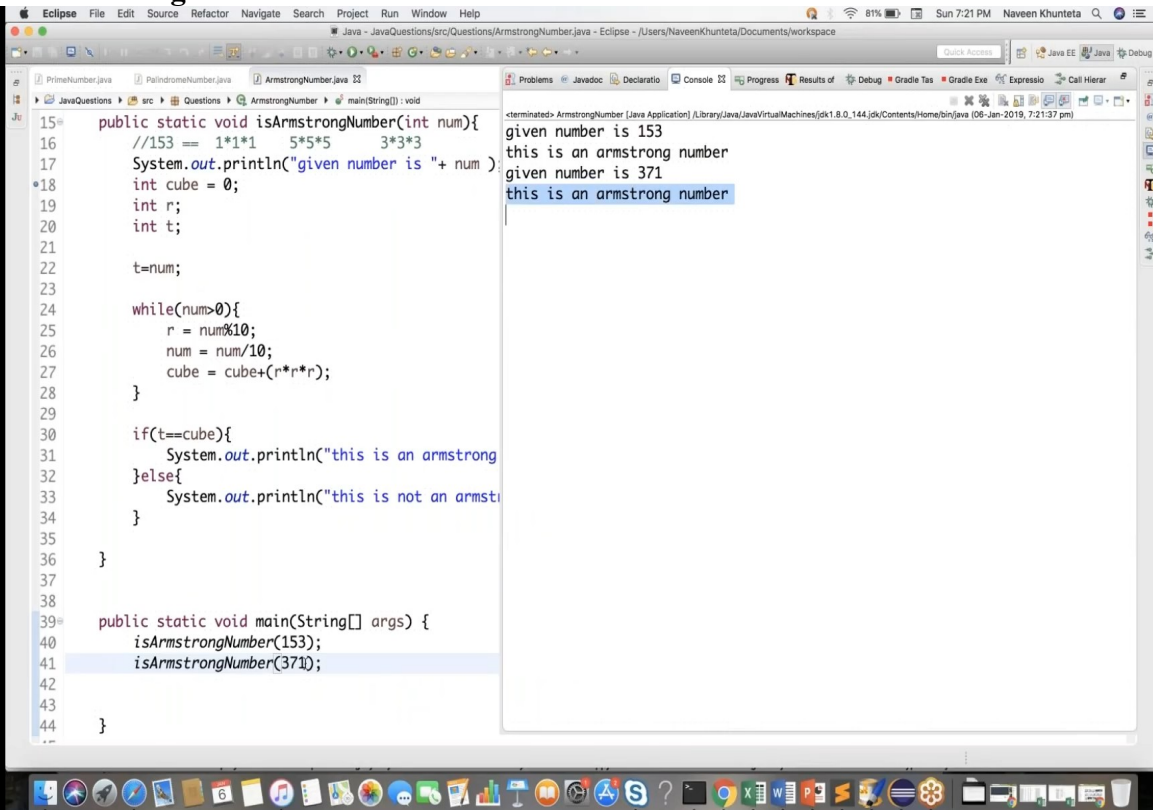
120

1

1

24

12. Armstrong Number:



```
15 public static void isArmstrongNumber(int num){
16     //153 == 1*1*1    5*5*5    3*3*3
17     System.out.println("given number is "+ num );
18     int cube = 0;
19     int r;
20     int t;
21
22     t=num;
23
24     while(num>0){
25         r = num%10;
26         num = num/10;
27         cube = cube+(r*r*r);
28     }
29
30     if(t==cube){
31         System.out.println("this is an armstrong
32     }else{
33         System.out.println("this is not an armst
34     }
35 }
36
37
38
39 public static void main(String[] args) {
40     isArmstrongNumber(153);
41     isArmstrongNumber(371);
42
43
44 }
```

terminated: ArmstrongNumber [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_144.jdk/Contents/Home/bin/java [08-Jan-2019, 7:21:37 pm]

given number is 153
this is an armstrong number
given number is 371
this is an armstrong number

13. Palindrome Number:

```
3 public class PalindromeNumber {
4
5     //151 454 34543 161 78987
6
7     public static void isPalindromeNumber(int num){
8
9         int r=0;
10        int sum=0;
11        int t;
12
13        t=num;
14
15        while(num>0){
16            r = num%10; //get the remainder
17            sum = (sum*10)+r;
18            num=num/10;
19        }
20    }
```

```

13
14     t=num;
15
16     while(num>0){
17         r = num%10; //get the remainder
18         sum = (sum*10)+r;
19         num=num/10;
20     }
21
22     if(t==sum){
23         System.out.println("palindrome number");
24     }else{
25         System.out.println("not palindrome number");
26     }
27
28
29 }
30
31
32 public static void main(String[] args) {
33
34     isPalindromeNumber(151);
35     isPalindromeNumber(152);
36     isPalindromeNumber(78987);
37
38
39
40 }
41
42 }
43

```

<terminated> PalindromeNumber [Java Application] /Lit

Given Number is :151

palindrome number

Given Number is :152

not palindrome number

Given Number is :78987

palindrome number