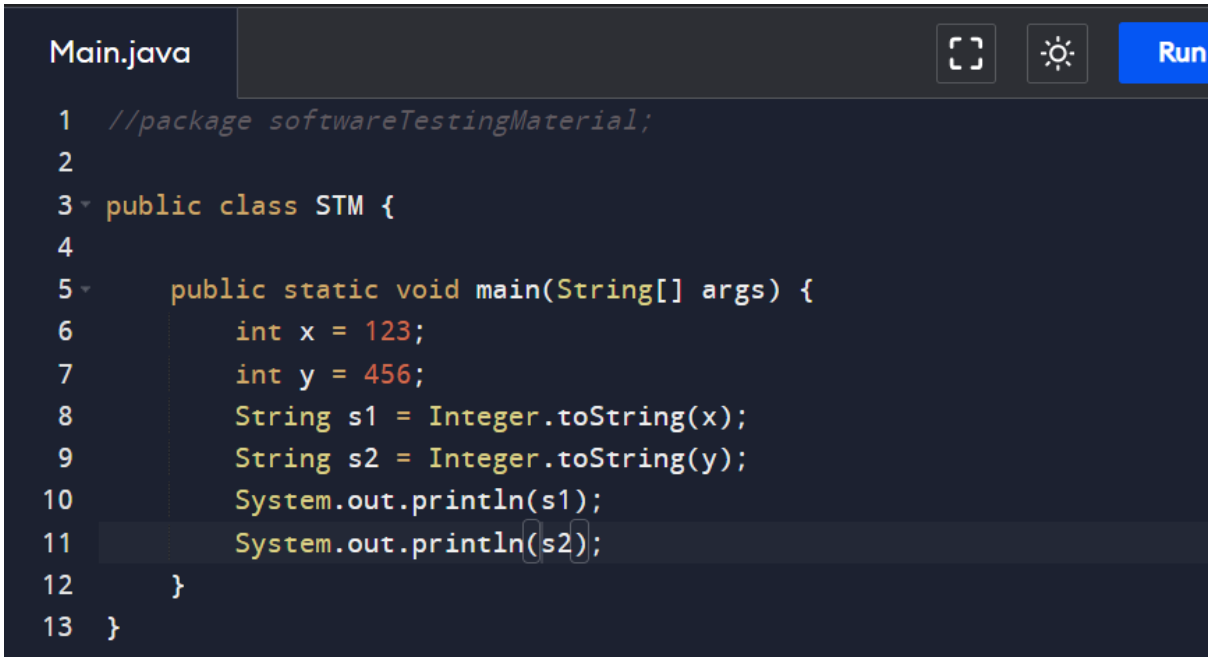
Java Program: questions and answers

1. How to convert Integer to String in Java?



2. How to convert String to Integer in Java?

3. How to convert Char to Integer in Java?

Main.java

Run

```java
//package softwareTestingMaterial;

public class STM {

    public static void main(String[] args) {
            // Initializing a character(ch)
        char c = '9';
            // Converting the character to an interger value
            int number = Integer.parseInt(String.valueOf(c));
            System.out.println(number);
    }
}

```

4. Write a program to print the pattern given below

1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
Here is the program to print the pattern mentioned above

Main.java

Run

```java
public class NumberPattern {

    public static void main(String[] args) {
        for (int x = 1; x <= 5; x++) {
            for (int y = 1; y <= x; y++) {
            System.out.print(y+" ");
            }
        System.out.println();
        }
    }
}

```

5. Write a program to print the pattern given below (Left Triangle Star Pattern)

```
*
* *
* * *
* * * *
* * * * *
```

Here is the program to print the pattern mentioned above

Main.java

Run

```java
public class star{
public static void main(String args[]) {
    int x, y, row=5;
    for(x=0; x<=row; x++) {
        for(y=0; y<=x; y++) {
            System.out.print("* ");
        }
    //Cursor goes to the new line after printing each line.
    System.out.println();
    }
}
}
```

6. Write a program to print the pattern given below (Right Triangle Star Pattern)

```
    *
   * *
  * * *
 * * * *
* * * * *
```

Here is the program to print the pattern mentioned above

Main.java

```java
public class star{
public static void main(String args[]) {
//x for rows,y for columns, and row denotes the number of rowstoprint
        int x, y, row=5;
        //outer loop for number of rows
        for(x=0; x<row; x++) { //inner loop for columns
            for(y=2*(row-x); y>=0; y--) {
         //To prints spaces
             System.out.print(" ");
            }
          //Inner loop for columns
         for(y=0; y<=x; y++){
            //To prints stars
            System.out.print("* ");
          }
          //Cursor goes to the new line after printing each line
         System.out.println();
       }
    }
}

```

7. Write a program to print the pattern given below (Pyramid Star Pattern)

```
    *
   * *
  * * *
 * * * *
* * * * *
```

Here is the program to print the pattern mentioned above

Main.java

```java
public class pyramid{
public static void main(String args[]) {
//x for rows, y for columns, and row denotes the number of rows to print
        int x, y, row = 5;
        //Outer loop for rows
        for (x=0; x<row; x++) { //inner loop for space
                    for (y=row-x; y>1; y--) { //To print space between two stars
                        System.out.print(" ");
                    }
                //inner loop for columns
        for (y=0; y<=x; y++ ) {
            //To print star
            System.out.print("* ");
        } //Cursor goes to the new line after printing each line.
       System.out.println();
    }
  }
}
```

8. How to reverse a String in Java?

Main.java

```java
public class ReverseString {
    public static void main(String[] args) {
    // Using StringBuffer class
    StringBuffer a = new StringBuffer("Software Testing Material");
    // use reverse() method to reverse string
    System.out.println(a.reverse());
    }
}
```

Another method:

Main.java Run

```java
public class ReverseString {
    public static void main(String[] args) {
        String input="Software Testing Material";
        StringBuilder input1 = new StringBuilder();
        input1.append(input);
        input1=input1.reverse();
        for (int i=0;i<input1.length();i++)
        System.out.print(input1.charAt(i));
    }
}
```

9. How To Find The Largest Value From The Given Array.

Main.java Run

```java
public class LargestValue {
    public static void main(String[] args){
        int[] arr={28,3,15,9,17,4,23,2};
        int val=arr[0];
        for(int i=0; i<arr.length; i++){
            if(arr[i] > val){
                val=arr[i];
            }
        }
        System.out.println("Largest value in the Given Array is "+ val);
    }
}
```

10. How to display all the prime numbers between 1 and 100

The number which is only divisible by 1 and itself is known as a prime number. For example 2, 3, 5, 7, 11… are prime numbers.

Main.java

Run

```java
public class PrimeNumbersOneToHundred {
    public static void main (String[] args){
        int i =0;
        int num =0;
        String  primeNumbers = "";
        for (i = 1; i <= 100; i++){
           int counter=0;
           for(num =i; num>=1; num--){
              if(i%num==0){
                  counter = counter + 1;
              }
           }
           if (counter ==2){
              primeNumbers = primeNumbers + i + " ";
           }
        }
        System.out.println("Prime numbers from 1 to 100 are :");
        System.out.println(primeNumbers);
    }
}
```

11. How to display all the prime numbers between 1 and n (n is the number, get the input from user)

Main.java

```java
import java.util.Scanner;
public class PrimeNumbersOneToN {
    public static void main (String[] args){
        Scanner scanner = new Scanner(System.in);
        int i =0;
        int num =0;
        String  primeNumbers = "";
        System.out.println("Enter the value of n :");
        int n = scanner.nextInt();
        scanner.close();
        for (i = 1; i <= n; i++)
        {
            int counter=0;
            for(num =i; num>=1; num--)
            {
          if(i%num==0)
          {
          counter = counter + 1;
          }
        }
        if (counter ==2)
        {
           primeNumbers = primeNumbers + i + " ";
        }
         }
        System.out.println("Prime numbers from 1 to n are :");
        System.out.println(primeNumbers);
    }
}

```

12. How to find the given number is a prime number or not by getting input from the user

Main.java

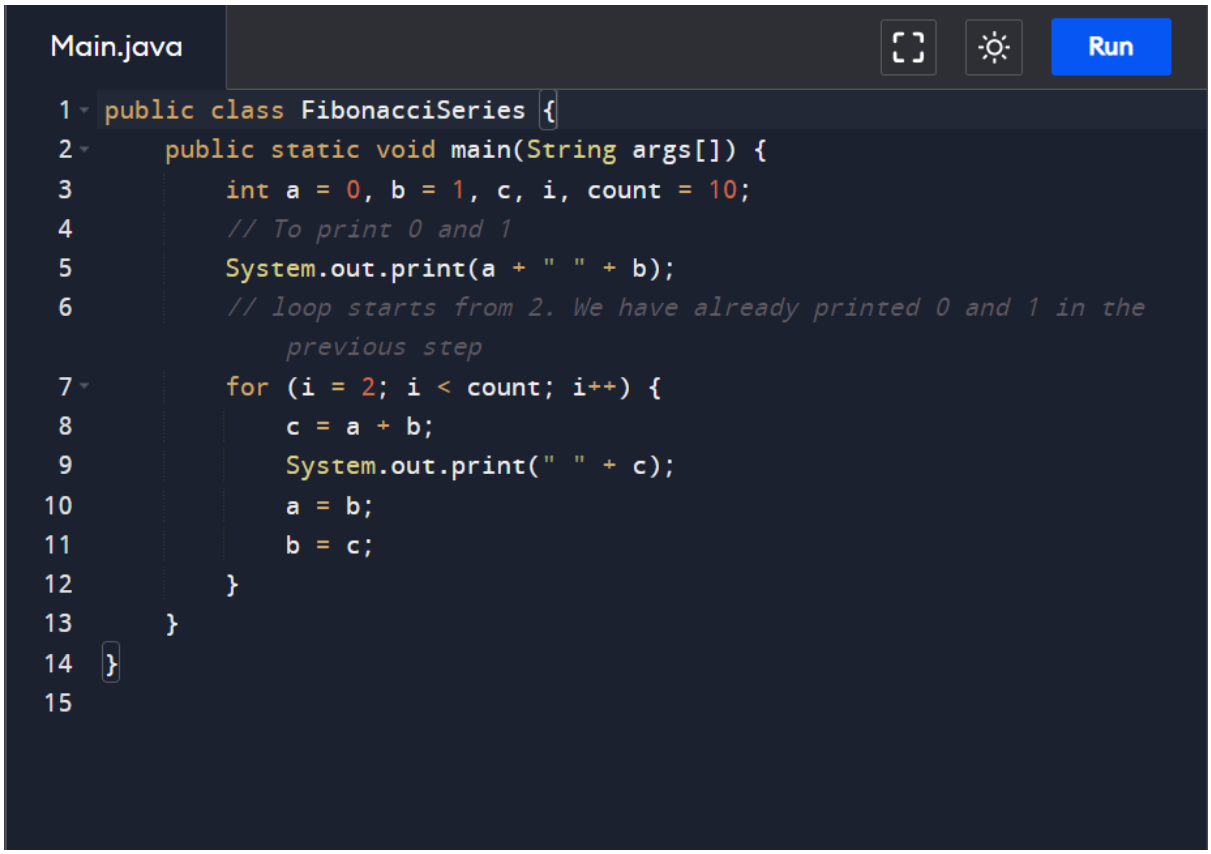
Run

```java
import java.util.Scanner;
public class PrimeNumberVerification {
    public static void main(String args[])
    {
        int i, j, flag = 0;
        System.out.print("Enter any number which you want to verify whether it is a prime number or not :");
        Scanner s = new Scanner(System.in);
        j = s.nextInt();
        for( i = 2; i < j; i++){
            if(j % i == 0){
                flag = 0;
                break;
            }
            else
            {
                flag = 1;
            }
        }
        if(flag == 1){
            System.out.println(j+" is a prime number.");
        }
        else{
            System.out.println(+j+" is not a prime number.");
        }
    }
}

```

13. Write a program to print Fibonacci Series

Method 1:



Method 2:

Main.java

Run

```java
import java.util.Scanner;
public class FibonacciSeriesOne {
    public static void main(String[] args){
    System.out.println("Enter Iteration to print Fibonacci Series");
    FibonacciCheck.checkFibonacci(new Scanner(System.in).nextInt());
    }
}
class FibonacciCheck {
    public static void checkFibonacci(int number){
        int first=0,second=1;
        int third=0;
        int i=1;
        System.out.print("Fibonacci Series upto: "+number+" is ");
        System.out.print(first+","+second+",");
        while(i<=number){
            third=first+second;
            System.out.print(third+",");
            first=second;
            second=third;
            ++i;
        }
    }
}
```

14. How to read a file line by line in Java?

We can read a file line by line in Java in two ways.

1.BufferedReaderClass
2. Scanner Class

Using BufferedReader Class:

BufferedReader Class belongs to java.io package and it provides readLine() method to read a file line by line in Java.

Main.java

Run

```java
import java.io.BufferedReader;
import java.io.FileReader;
import java.io.IOException;
public class ReadLineByProgram {
    public static void main(String[] args) {
        BufferedReader reader;
        try {
            reader = new BufferedReader(new FileReader("/Users/Rajkumar
                /Downloads/STM.txt"));
            String line = reader.readLine();
            while (line != null) {
                System.out.println(line);
                // read next line
                line = reader.readLine();
            }
            reader.close();
        }
        catch (IOException e) {
            e.printStackTrace();
            }
        }
}

```

Using Scanner Class:

Java Scanner class provides the nextLine() method to facilitates line by line of file's content.

Main.java

Run

```java
import java.io.File;
import java.io.FileNotFoundException;
import java.util.Scanner;
public class ReadLineByProgram {
public static void main(String[] args) {
    try {
        Scanner scanner = new Scanner(new File("/Users/Rajkumar
            /Downloads/STM.txt"));
        while (scanner.hasNextLine()) {
            System.out.println(scanner.nextLine());
                }
        scanner.close();
        }
    catch (FileNotFoundException e) {
        e.printStackTrace();
    }
}
}
```

15. Swap string without 3rd variable?

16.Duplicates in a String?

Main.java

Run

```java
public class FindDuplicates {
    public static void main(String[] args) {
        String str = "geeksforgeeks";
        char[] chars = str.toCharArray();

        System.out.println("Duplicate characters:");
        for (int i = 0; i < chars.length; i++) {
            for (int j = i + 1; j < chars.length; j++) {
                if (chars[i] == chars[j]) {
                    System.out.print(chars[j] + " ");
                }
            }
        }
    }
}
```

17. How to find the length of the string without using length?

Main.java

Run

```java
import java.util.Scanner;
public class Main {
    public static void main(String[] args) {
        int length=0;
        String s = "prepinsta";
        for (char c1 : s.toCharArray())
        length++;
        System.out.println("Length of String is : "+length);
    }
}
```

18.Largest number in an Array

Main.java

Run

```java
import java.util.Scanner;
public class Main
{
  public static void main(String args[])
  {
     int arr[] = {12, 13, 1, 10, 34, 10};
     int max = arr[0];
     for(int i=0; i<arr.length; i++)
     {
       if(max < arr[i])
       {
          max = arr[i];
       }
     }
    System.out.print(max);
  }
}
```

19.Reverse a string without using reverse function

Main.java

Run

```java
// java program to reverse a word
import java.io.*;
import java.util.Scanner;
class GFG {
    public static void main (String[] args) {
        String str= "Geeks", nstr="";
        char ch;
      System.out.print("Original word: ");
      System.out.println("Geeks"); //Example word
      for (int i=0; i<str.length(); i++)
      {
        ch= str.charAt(i); //extracts each character
        nstr= ch+nstr; //adds each character in front of the existing string
      }
      System.out.println("Reversed word: "+ nstr);
    }
}
```

20. Write code to print only the even numbers from an array.

Main.java Run

```java
public class EvenNumbersFromArray {
    public static void main(String[] args) {
        int[] numbers = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 };

        System.out.println("Even numbers:");
        for (int num : numbers) {
            if (num % 2 == 0) {
                System.out.println(num);
            }
        }
    }
}
```

21. Write code to find special character, number, capital and small letter in a given string.

Main.java Run

```java
import java.util.regex.Pattern;
import java.util.regex.Matcher;
public class CharacterChecker {
    public static void main(String[] args) {
        String str = "#GeeksForGeeks123@";
        // Create a regular expression to match the criteria
        String regex = "^(?=.*[a-z])(?=.*[A-Z])(?=.*\\d)(?=.*[-+_!@#$%^&
            *.,?]).+$";
        // Check if the string matches the regular expression
        if (str.matches(regex)) {
            System.out.println("Yes, the string contains uppercase letters,
                lowercase letters, special characters, and numeric values."
                );
        } else {
            System.out.println("No, the string does not meet all the
                criteria.");
        }
    }
}
```

21. Write code to check if a number is palindrome?

Main.java

Run

```java
class Main {
  public static void main(String[] args) {
    String str = "Radar", reverseStr = "";
    int strLength = str.length();
    for (int i = (strLength - 1); i >=0; --i) {
      reverseStr = reverseStr + str.charAt(i);
    }
    if (str.toLowerCase().equals(reverseStr.toLowerCase())) {
      System.out.println(str + " is a Palindrome String.");
    }
    else {
      System.out.println(str + " is not a Palindrome String.");
    }
  }
}
```

22. Write a Java code to identify, if the pair of strings are an Anagram or not

Main.java

```java
import java.util.Arrays;
public class AnagramChecker {
    public static void main(String[] args) {
        String string1 = "listen";
        String string2 = "silent";

        if (isAnagramSort(string1, string2)) {
            System.out.println("The strings are anagrams.");
        } else {
            System.out.println("The strings are not anagrams.");
        }
    }
    static boolean isAnagramSort(String s1, String s2) {
        if (s1.length() != s2.length()) {
            return false;
        }
        char[] a1 = s1.toCharArray();
        char[] a2 = s2.toCharArray();
        Arrays.sort(a1);
        Arrays.sort(a2);
        return Arrays.equals(a1, a2);
    }
}
```

23. Write a java code for factorial of a Number using recursive method ?

Java

```java
public class Factorial {
    public static void main(String[] args) {
        int number = 5; // Change this to calculate factorial of different numbers
        int factorial = calculateFactorial(number);
        System.out.println("Factorial of " + number + " = " + factorial);
    }

    /**
     * Recursive function to calculate factorial
     * @param n number to calculate factorial
     * @return factorial of n
     */
    public static int calculateFactorial(int n) {
        if (n == 0 || n == 1) {
            // base case: factorial of 0 or 1 is 1
            return 1;
        } else {
            // recursive case: n! = n * (n-1)!
            return n * calculateFactorial(n - 1);
        }
    }
}
```

24. Write Java code to print all the array elements that appear at least 2 times (meaning 2 or greater than two).

```java
import java.util.HashMap;
import java.util.Map;

public class DuplicateElements {
    public static void findDuplicates(int[] arr) {
        Map<Integer, Integer> elementCounts = new HashMap<>();

        // Count occurrences of each element
        for (int num : arr) {
            elementCounts.put(num, elementCounts.getOrDefault(num, 0) + 1);
        }

        System.out.println("Duplicate elements:");
        // Print elements that appear more than once
        for (Map.Entry<Integer, Integer> entry : elementCounts.entrySet()) {
            if (entry.getValue() >= 2) {
                System.out.print(entry.getKey() + " ");
            }
        }
        System.out.println(); // New line for better formatting
    }

    public static void main(String[] args) {
        int[] array = {1, 2, 3, 2, 4, 1, 5, 2, 6, 3};
        findDuplicates(array); // Output: Duplicate elements: 1 2 3
        int[] array2 = {1,1,1,1,1,1,1,1};
        findDuplicates(array2); // Output: Duplicate elements: 1
        int[] array3 = {1,2,3,4,5,6,7,8};
        findDuplicates(array3); // Output: Duplicate elements:
    }
}
```

2. Using Nested Loops (Less Efficient, but simpler to understand):

Java

```java
public class DuplicateElementsNestedLoop {
    public static void findDuplicates(int[] arr) {
        System.out.println("Duplicate elements:");
        for (int i = 0; i < arr.length; i++) {
            for (int j = i + 1; j < arr.length; j++) {
                if (arr[i] == arr[j]) {
                    System.out.print(arr[i] + " ");
                    // To avoid printing the same duplicate multiple times
                    // you could use a Set to store already printed duplicates or
                    break; // break is added to print each duplicate only once.
                }
            }
        }
        System.out.println();
    }
        public static void main(String[] args) {
        int[] array = {1, 2, 3, 2, 4, 1, 5, 2, 6, 3};
        findDuplicates(array); // Output: Duplicate elements: 1 2 3
        int[] array2 = {1,1,1,1,1,1,1,1};
        findDuplicates(array2); // Output: Duplicate elements: 1
        int[] array3 = {1,2,3,4,5,6,7,8};
        findDuplicates(array3); // Output: Duplicate elements:
    }
}
```