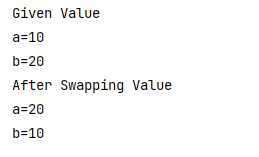
# **1**. Suppose, a=10 and b=20. Now swap the value using a temp variable. Output: a=20, b=10

public class SwapValueUsingTemp {  
 public static void main(String[] args){  
 int a = 10;  
 int b = 20;  
 System.*out*.println("Given Value");  
 System.*out*.println("a="+a);  
 System.*out*.println("b="+b);  
 int temp = a;  
 a = b;  
 b = temp;  
 System.*out*.println("After Swapping Value");  
 System.*out*.println("a="+a);  
 System.*out*.println("b="+b);  
 }  
}

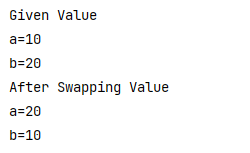
Output Screenshot:



# **2**. Suppose, a=10 and b=20. Now swap the value without a temp variable. Output: a=20, b=10

public class SwapValueUsingTemp {  
 public static void main(String[] args){  
 int a = 10;  
 int b = 20;  
 System.*out*.println("Given Value");  
 System.*out*.println("a="+a);  
 System.*out*.println("b="+b);  
 int temp = a;  
 a = b;  
 b = temp;  
 System.*out*.println("After Swapping Value");  
 System.*out*.println("a="+a);  
 System.*out*.println("b="+b);  
 }  
}

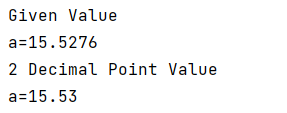
Output Screenshot:



# **3**. Suppose a=15.5276. print the value upto 2 decimal point. Output: a=15.52

public class ValueUpto2DecimalPoint {  
 public static void main(String[] args) {  
 double a = 15.5276;  
 System.*out*.println("Given Value");  
 System.*out*.println("a=" + a);  
 System.*out*.println("2 Decimal Point Value");  
 System.*out*.printf("a=%.2f%n", a);  
 }  
}

Output Screenshot:



# **4**. Write a program that will find your key is found in the given array using binary search method numbers=[1,6,9,3,5,4,7]

## key=5

import java.io.\*;  
import java.lang.\*;  
import java.util.\*;  
  
public class FindKeyUsingBinarySearch {  
 public static void main(String[] args) {  
 int[] numbers = {1, 6, 9, 3, 5, 4, 7};  
 int key = 5;  
 Arrays.*sort*(numbers);  
 boolean isKeyFound = *binarySearch*(numbers, key);  
 }  
 private static boolean binarySearch(int[] array, int key) {  
 int low = 0;  
 int high = array.length - 1;  
 while (low <= high) {  
 int mid = (low + high) / 2;  
 int midValue = array[mid];  
 if (midValue == key) {  
 System.*out*.println("Key=" +key+ " found at index " + midValue);  
 break;  
 } else if (midValue < key) {  
 low = mid + 1;  
 } else {  
 high = mid - 1;  
 }  
 if (midValue > key) {  
 System.*out*.println("Key is not found in array");  
 break;  
 }  
 }  
 return false;  
 }  
}

Output Screenshot:



# **5**. Write a program that will find your key is found in the given array using linear search method numbers=[1,6,9,3,5,4,7]

## key=5

import java.io.\*;  
import java.lang.\*;  
import java.util.\*;  
  
public class FindKeyUsingLinearSearch {  
 public static void main(String[] args) {  
 int[] numbers = {1,6,9,3,5,4,7};  
 int key = 5;  
 int result = *linearSearch*(numbers, key);  
 if(result == -1) {  
 System.*out*.println("Key is not found in array");  
 } else {  
 System.*out*.println("Key=" +key+ " found at index " +result);  
 }  
 }  
 public static int linearSearch(int[] array, int key) {  
 for(int i=0; i<array.length; i++) {  
 if(array[i] == key) {  
 return i;  
 }  
 }  
 return -1;  
 }  
}

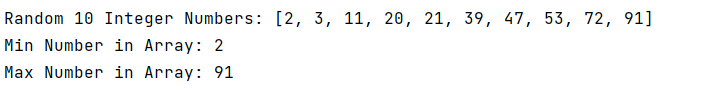
Output Screenshot:



# **6.** Generate random 10 integer numbers in an array and print out all the numbers from array and also print the max and min number from the array.

import java.io.\*;  
import java.lang.\*;  
import java.util.\*;  
  
public class RandomNumbersMaxMIn {  
 public static void main(String[] args) {  
 Random random = new Random();  
 int[] numbers = new int[10];  
 for (int i = 0; i < numbers.length; i++) {  
 numbers[i] = random.nextInt(100) + 1;  
 }  
 Arrays.*sort*(numbers);  
 System.*out*.println("Random 10 Integer Numbers: " + Arrays.*toString*(numbers));  
 System.*out*.println("Min Number in Array: " + numbers[0]);  
 System.*out*.println("Max Number in Array: " + numbers[numbers.length - 1]);  
 }  
}

Output Screenshot:



# **7**. Write a program that will breakdown the amount and count notes for any given amount. Here is the notes in the given array:

## notes=[1000,500,200,100,50,20,10,5,2,1]

Example 1: User input: 575

Output:

500 1

50 1

20 1

5 1

Example 2: User input: 2148

Output:

1000 2

100 1

20 2

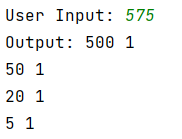
5 1

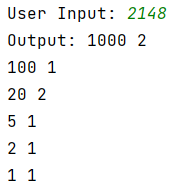
2 1

1 1

import java.io.\*;  
import java.lang.\*;  
import java.util.\*;  
  
public class NoteBreakdown {  
 public static void main(String[] args) {  
 int[] notes = {1000, 500, 200, 100, 50, 20, 10, 5, 2, 1};  
 Scanner scanner = new Scanner(System.*in*);  
 System.*out*.print("User Input: ");  
 int amount = scanner.nextInt();  
 System.*out*.print("Output: ");  
 for (int i = 0; i < notes.length; i++) {  
 if (amount >= notes[i]) {  
 int count = amount / notes[i];  
 System.*out*.println(notes[i] +" "+ count);  
 amount = amount % notes[i];  
 }  
 }  
 }  
}

Output Screenshot:





# **8**. Print the prime numbers of 2 to 100

public class PrimeNumbers2to100 {  
 public static void main(String[] args) {  
 for (int num = 2; num <= 100; num++) {  
 if (*isPrime*(num)) {  
 System.*out*.println(num);  
 }  
 }  
 }  
 public static boolean isPrime(int num) {  
 if (num <= 1) {  
 return false;  
 }  
 for (int i = 2; i <= Math.*sqrt*(num); i++) {  
 if (num % i == 0) {  
 return false;  
 }  
 }  
 return true;  
 }  
}

Output Screenshot:



# **9**. Write a program that will give following output:

1

12

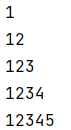
123

1234

12345

public class NumberPattern1 {  
 public static void main(String[] args) {  
 int n = 5;  
 for (int i = 1; i <= n; i++) {  
 for (int j = 1; j <= i; j++) {  
 System.*out*.print(j);  
 }  
 System.*out*.println();  
 }  
 }  
}

Output Screenshot:



# **10**. Write a program that will give following output:

12345

2345

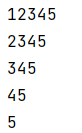
345

45

5

public class NumberPattern2 {  
 public static void main(String[] args) {  
 int n = 5;  
 for (int i = 1; i <= n; i++) {  
 for (int j = i; j <= n; j++) {  
 System.*out*.print(j);  
 }  
 System.*out*.println();  
 }  
 }  
}

Output Screenshot:



# **11**. Find out the second largest element of the given array numbers=[5,3,9,7,4,1,8]

public class SecondLargestElementInArray {  
 public static void main(String[] args) {  
 int[] numbers = {5, 3, 9, 7, 4, 1, 8};  
 int secondMax;  
 secondMax = *printSecondLargest*(numbers);  
 System.*out*.println("The second largest element of the given array is " + secondMax);  
 }  
 public static int printSecondLargest(int[] numbers) {  
 int max = numbers[0];  
 int secondMax = numbers[0];  
 for (int i = 1; i < numbers.length; i++) {  
 if (numbers[i] > max) {  
 secondMax = max;  
 max = numbers[i];  
 } else if (numbers[i] > secondMax && numbers[i] != max) {  
 secondMax = numbers[i];  
 }  
 }  
 return secondMax;  
 }  
}

Output Screenshot:



# **12**. Write a program that will shuffle (values will randomly change their position) from the given array numbers=[1,2,3,4,5,6,7,8,9,0]

random output: 3 5 0 1 7 9 6 4 2 8

import java.io.\*;  
import java.lang.\*;  
import java.util.\*;  
  
public class ShuffleArrayValuesPosition {  
 public static void main(String[] args) {  
 int[] numbers = {1, 2, 3, 4, 5, 6, 7, 8, 9, 0};  
 *shuffle*(numbers);  
 System.*out*.print("Random Output: ");  
 for (int num : numbers) {  
 System.*out*.print(num + " ");  
 }  
 }  
 public static void shuffle(int[] array) {  
 Random random = new Random();  
 for (int i = array.length - 1; i > 0; i--) {  
 int index = random.nextInt(i + 1);  
 int a = array[index];  
 array[index] = array[i];  
 array[i] = a;  
 }  
 }  
}

Output Screenshot:



# **13**. Take any number as input and print the reverse of the number.

input: 12345

output: 54321

import java.io.\*;  
import java.lang.\*;  
import java.util.\*;  
  
public class ReverseNumber {  
 public static void main(String[] args) {  
 Scanner scanner = new Scanner(System.*in*);  
 System.*out*.print("Input: ");  
 int numbers = scanner.nextInt();  
 int reversed = 0;  
 while (numbers != 0) {  
 int digit = numbers % 10;  
 reversed = reversed \* 10 + digit;  
 numbers = numbers / 10;  
 }  
 System.*out*.print("Output: " + reversed);  
 }  
}

Output Screenshot:



# **14**. Find out the duplicate numbers in the given array and in which position they are found. numbers=[1,2,3,1,2,4,5,6,4]

Output:

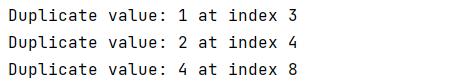
Duplicate value: 1 at index 3

Duplicate value: 2 at index 4

Duplicate value: 4 at index 8

import java.io.\*;  
import java.lang.\*;  
import java.util.\*;  
  
public class DuplicateNumberPositions {  
 public static void main(String[] args) {  
 int[] numbers = {1, 2, 3, 1, 2, 4, 5, 6, 4};  
 Map<Integer, Integer> duplicateCounts = new HashMap<>();  
 for (int i = 0; i < numbers.length; i++) {  
 if (duplicateCounts.containsKey(numbers[i])) {  
 System.*out*.println("Duplicate value: " + numbers[i] + " at index " + i);  
 } else {  
 duplicateCounts.put(numbers[i], i);  
 }  
 }  
 }  
}

Output Screenshot:



# **15**. Count number of words, number of characters without spaces, number of vowels and consonant from the given string:

"I live in Bangladesh"

Output:

Number of words: 4

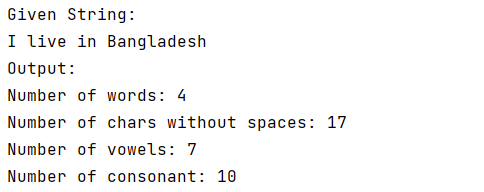
Number of chars without spaces: 17

Number of vowels: 7

Number of consonant: 10

public class StringAnalysis {  
 public static void main(String[] args) {  
 String string = "I live in Bangladesh";  
 System.*out*.println("Given String: ");  
 System.*out*.println(string);  
 int wordsCount = 0, charsCount = 0, vowelsCount = 0, consonantCount = 0;  
 String[] words = string.split("\\s+");  
 wordsCount = words.length;  
 for (String word : words) {  
 charsCount = charsCount + word.length();  
 for (char c : word.toCharArray()) {  
 if (Character.*isLetter*(c)) {  
 char lowerCase = Character.*toLowerCase*(c);  
 if (lowerCase == 'a' || lowerCase == 'e' || lowerCase == 'i' || lowerCase == 'o' || lowerCase == 'u') {  
 vowelsCount++;  
 } else {  
 consonantCount++;  
 }  
 }  
 }  
 }  
 System.*out*.println("Output:");  
 System.*out*.println("Number of words: " + wordsCount);  
 System.*out*.println("Number of chars without spaces: " + charsCount);  
 System.*out*.println("Number of vowels: " + vowelsCount);  
 System.*out*.println("Number of consonant: " + consonantCount);  
 }  
}

Output Screenshot:



# **16**. Encrypt word: ROADTOSDET [when A=F]

Output: WTFIYTXIJY

public class EncryptWord {  
 public static void main(String[] args) {  
 String originalWord = "ROADTOSDET";  
 String encryptedWord = *encrypt*(originalWord);  
 System.*out*.println("Given Word to Encrypt: " + originalWord);  
 System.*out*.println("Output: " + encryptedWord);  
 }  
 private static String encrypt(String word) {  
 StringBuilder encrypted = new StringBuilder();  
 for (int i = 0; i < word.length(); i++) {  
 char currentChar = word.charAt(i);  
 if (Character.*isUpperCase*(currentChar)) {  
 int newPosition = (currentChar - 'A' + 5) % 26;  
 char encryptedChar = (char) ('A' + newPosition);  
 encrypted.append(encryptedChar);  
 } else {  
 encrypted.append(currentChar);  
 }  
 }  
 return encrypted.toString();  
 }  
}

Output Screenshot:



# **17**. Check if the given string is palindrome or not. Palindrome means after reversing a string, it will be same.

Input: Civic

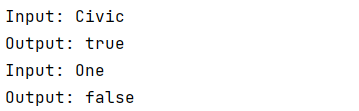
Output: true

Input: One

Output: false

import java.io.\*;  
import java.lang.\*;  
import java.util.\*;  
  
public class PalindromeChecker {  
 public static boolean isPalindrome(String input) {  
 input = input.toLowerCase();  
 String reversed = new StringBuilder(input).reverse().toString();  
 return input.equals(reversed);  
 }  
 public static void main(String args[]) {  
 String input1 = "Civic";  
 System.*out*.println("Input: " + input1);  
 System.*out*.println("Output: " + *isPalindrome*(input1));  
  
 String input2 = "One";  
 System.*out*.println("Input: " + input2);  
 System.*out*.println("Output: " + *isPalindrome*(input2));  
 }  
}

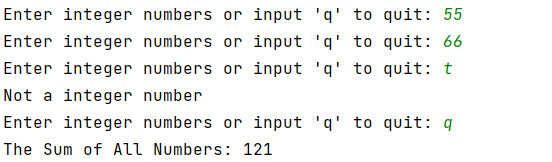
Output Screenshot:



# **18**. Write a program that will take integer numbers as user input continuously and print the sum of numbers until user input q from the keyboard. When user input q, program will be quit. If user inputs another character, then the program will ask to input the number again.

import java.io.\*;  
import java.lang.\*;  
import java.util.\*;  
  
public class ContinuousSum {  
 public static void main(String[] args) {  
 Scanner scanner = new Scanner(System.*in*);  
 int sum = 0;  
 String input;  
 boolean flag = true;  
 while (flag) {  
 System.*out*.print("Enter an integer number or input 'q' to quit: ");  
 input = scanner.nextLine();  
 if (input.equalsIgnoreCase("q")) {  
 flag = false;  
 } else {  
 try {  
 int number = Integer.*parseInt*(input);  
 sum += number;  
 } catch (NumberFormatException e) {  
 System.*out*.println("Not a integer number");  
 }  
 }  
 }  
 System.*out*.println("The Sum of All Numbers: " + sum);  
 }  
}

Output Screenshot:



# **19**. Extract the OTP from the SMS. "Your one-time password is 246148. Don't share this code with anyone \r\nBvwt3f8js2S"

import java.io.\*;  
import java.lang.\*;  
import java.util.\*;  
import java.util.regex.Matcher;  
import java.util.regex.Pattern;  
  
public class OTPExtractor {  
 private static String extractOTP(String SMS) {  
 String OTPpattern = "\\b\\d{6}\\b";  
 Pattern pattern = Pattern.*compile*(OTPpattern);  
 Matcher matcher = pattern.matcher(SMS);  
 if (matcher.find()) {  
 return matcher.group();  
 } else {  
 return null;  
 }  
 }  
 public static void main(String[] args) {  
 String SMS = "Your one-time password is 246148. Don't share this code with anyone \r\nBvwt3f8js2S";  
 System.*out*.println("Extracted OTP: " + *extractOTP*(SMS));  
 }  
}

Output Screenshot:



# **20**. A core i7 laptop price is 85000 tk and a gaming mouse price is 2500 tk. If I buy the laptop and 1 piece mouse, what will be my total cost after giving 15% discount? [Extract the digits from the paragraph and calculate the price]

public class PriceExtractor {  
 public static void main(String[] args) {  
 String paragraph = "A core i7 laptop price is 85000 tk and a gaming mouse price is 2500 tk";  
 String[] words = paragraph.split("\\s+");  
 int laptopPrice = 0;  
 int mousePrice = 0;  
 for (String word : words) {  
 if (word.matches("\\d+")) {  
 int number = Integer.*parseInt*(word);  
 if (laptopPrice == 0) {  
 laptopPrice = number;  
 } else if (mousePrice == 0){  
 mousePrice = number;  
 }  
 }  
 }  
 double totalCost = laptopPrice + mousePrice;  
 double discount = (15.0 / 100.0);  
 totalCost = totalCost - (totalCost \* discount);  
 System.*out*.println("My total cost after giving 15% discount is " + totalCost + " tk");  
 }  
}

Output Screenshot:

