JAVA PROGRAMMING LAB

**LAB 1**

**Submitted by:** Bharat Kumar

**Enrollment no:** 12814802719

**Topic covered:** Basic concept of Java programming, Compilation and Execution Process, Data Types, operators, Reading user input, Strings

**LAB 1.1**

**Aim:** Write a program to print “Hello World” on the screen.

**Theory:**

The process of Java programming can be simplified in three steps:

* Create the program by typing it into a text editor and saving it to a file – HelloWorld.java.
* Compile it by typing “javac HelloWorld.java” in the terminal window.
* Execute (or run) it by typing “java HelloWorld” in the terminal window.

**Source code:**

class Simple{

public static void main(String args[]){

System.out.println("Hello World");

}

}

**Output**



**LAB 1.2**

**Aim:** WAP that convert string to character using toString() and valueOf() .

**Theory:**

Given a [string](https://www.geeksforgeeks.org/strings-in-java/), the task is to convert this [string](https://www.geeksforgeeks.org/strings-in-java/) into a character array in Java.

**Examples:**

**Input:** Hello World

**Output:** [H, e, l, l, o,, W, o, r, l, d]

**Method : Naive Approach**

* + Step 1: Get the string.
  + Step 2: Create a character array of the same length as of string.
  + Step 3: Traverse over the string to copy character at the i’th index of string to i’th index in the array.
  + Step 4: Return or perform the operation on the character array.

**Source code:**

public class StringToChar {

public static void main(String[] args) {

String s = "Mountains";

char[] charString = s.toCharArray();

for (int i = 0; i < charString.length; i++) {

System.out.print("'"+charString[i]+"'"+",");

}

}

}

**Output:**



**LAB 1.3**

**Aim:** WAP that convert Char to string

**Theory:**

**Method :** Using valueOf() method of String class

Another way to convert a character array to a string is to use the **valueOf()** method present in the String class. This method inherently converts the character array to a format where the entire value of the characters present in the array is displayed. This method generally converts int, float, double, char, boolean, and even object to a string. Here we will achieve the goal by converting our character array to string.

**Source code:**

class CharToString{

public static String toString(char[] a)

{

String string = String.valueOf(a);

return string;

}

public static void main(String args[])

{

char s[] = { 'G', 'o', 'k', 'u',};

System.out.println(toString(s));

}

}

**Output:**



**LAB 1.4**

**Aim:** Program to find ASCII code of a character

**Theory:**

ASCII acronym for American Standard Code for Information Interchange. It is a 7-bit character set contains 128 (0 to 127) characters. It represents the numerical value of a character. For example, the ASCII value of A is 65.

In this section, we will learn how to print ASCII value or code through a Java program.

There are two ways to print ASCII value in [Java](https://www.javatpoint.com/java-tutorial):

* Assigning a Variable to the int Variable
* Using Type-Casting

## Assigning a Variable to the int Variable

To print the [ASCII](https://www.javatpoint.com/ascii) value of a character, we need not use any method or class. Java internally converts the character value to an ASCII value.

Let's find the [ASCII](https://www.javatpoint.com/ascii-full-form) value of a character through a [Java program](https://www.javatpoint.com/java-programs).

In the following program, we have assigned two characters a and b in the ch1 and ch2 variables, respectively. To find the ASCII value of a and b, we have assigned ch1 and ch2 variables to the integer variables asciivalue1 and asciivalue2, respectively. Finally, we have printed the variable asciivalue1 and asciivalue2 in which ASCII values of the characters are stored.

**Source code:**

import java.util.Scanner;

public class ASCII

{

public static void main(String args[])

{

System.out.print("Enter a character: ");

Scanner sc = new Scanner(System.in);

char chr = sc.next().charAt(0);

int asciiValue = chr;

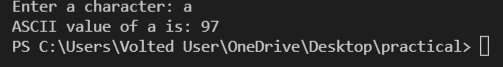
System.out.println("ASCII value of " +chr+ " is: "+asciiValue);

}

}

**Output:**





**LAB 1.5**

**Aim:** Swapping two numbers using bitwise operators.

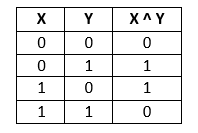
**Theory:**

In Java, there are many ways to swap two numbers. Generally, we use either swap() method of the Math class or use a third (temporary) variable to swap two numbers. Except these two ways, we can also swap two numbers using the bitwise operator (XOR) and using division and multiplication.

In this section, we will focus on creating a [Java program](https://www.javatpoint.com/java-programs) to swap two numbers using bitwise operator (^).

## Using Bitwise Operator

Bitwise Operator: Bitwise XOR operator is used to swap two numbers. It is represented by the symbol (^). It compares bits of two operands and returns false or 0 if they are equal and returns true or 1 if they are not equal. The truth table of XOR operator is as follows:



We can use the bitwise XOR operator to swap two numbers without using the swap() method and third variable. We must follow the steps given below:

* Find the binary equivalent of given variables, say X and Y.
* Find X^Y and store it in x, i.e. X = X ^ Y.
* Again, find X^Y and store it in Y, i.e. Y = X ^ Y.
* Find X^Y and store it in X, i.e. X = X ^ Y.
* The numbers are swapped.

Now implement the above steps in an example and understand the swapping.

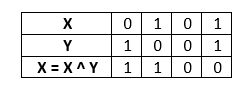
Example: Swap the variables X = 5 and Y = 9 using the bitwise operator.

Solution:

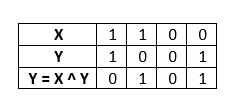
Step 1: Binary equivalent of the variables X and Y are:

X = 5 = 0101 and Y = 9 = 1001

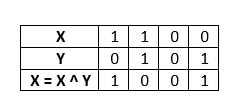
Step 2: Find X = X ^ Y.



Step 2: Find Y = X ^ Y.



Step 3: Find X = X ^ Y.



We see that the variable X contains 1001 which is equivalent to 9 and Y contains 0101 which is equivalent to 5. Therefore, the variables X and Y are swapped.

X = 9 and Y = 5

Let's implements the above logic in a Java program.

**Source code:**

import java.util.Scanner;

public class bitwise {

public static void main(String[] args){

int a,b;

Scanner sc=new Scanner(System.in);

System.out.print("Enter first no. ");

a=sc.nextInt();

System.out.print("Enter second no. ");

b=sc.nextInt();

a=a^b;

b=a^b;

a=a^b;

System.out.print("First no: "+a);

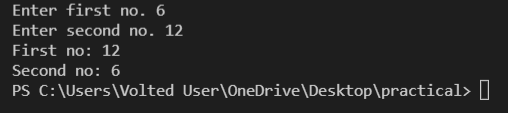
System.out.print("\nSecond no: "+b);

sc.close();

}

}

**Output:**



**Viva Voce Question**

1. What is the most important feature of Java?

Ans Java is a platform independent language.

2. What do you mean by platform independence?

Ans Platform independence means that we can write and compile the java code in one platform (e.g. Windows) and can execute the class in any other supported platform e.g. (Linux, Solaris, etc.).

3. What is a JVM?

Ans JVM is Java Virtual Machine which is a run time environment for the compiled java class files.

4. Are JVM's platform independent?

JVM's are not platform independent. JVM's are platform specific run time implementation provided by the vendor.

5. What is the difference between a JDK and a JVM?

JDK is Java Development Kit which is for development purpose and it includes execution environment also. But JVM is purely a run time environment and hence you will not be able to compile your source files using a JVM