JAVA PROGRAMMING LAB

**LAB 2**

**Submitted by:** Bharat kumar

**Enrollment no:** 12814802719

**Topic covered:** Java Control statements, Command line arguments

**LAB 2.1**

**Aim:** WAP to check Vowel or Consonant using Switch Case

**Theory:**

The alphabets A, E, I, O and U (smallcase and uppercase) are known as Vowels and rest of the alphabets are known as consonants. Here we will write a java program that checks whether the input character is vowel or Consonant using [**Switch Case in Java**](https://beginnersbook.com/2017/08/java-switch-case/).

## **Example: Program to check Vowel or Consonant using Switch Case**

In this program we are not using [break statement](https://beginnersbook.com/2017/08/java-break-statement/) with cases intentionally, so that if user enters any vowel, the program continues to execute all the subsequent cases until Case 'U' is reached and thats where we are setting up the value of a boolean variable to true. This way we can identify that the alphabet entered by user is vowel or not.

**Algorithm:**

1. Enter and Store User Input.
2. Duplicate the input and store it.
3. Change the Case of the duplicate for lesser comparison and uniformity.
4. Pass the input through Switch-Case construct.
5. Check for Vowel.
6. Print the Result accordingly.

**Source code:**

import java.util.Scanner;

public class prog1 {

public static void main(String[] args) {

String ch;

Scanner s = new Scanner(System.in);

System.out.println("Enter an alphabet");

ch = s.next();

switch (ch) {

case "a":

case "e":

case "i":

case "o":

case "u":

System.out.println(ch + " is vowel");

break;

default:

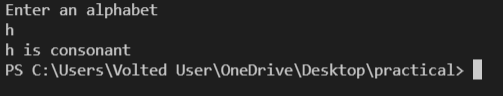
System.out.println(ch + " is consonant");

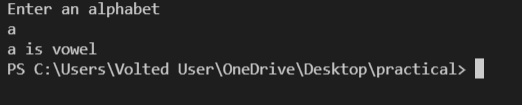
}

}

}

**Output**





**LAB 2.2**

**Aim:** WAP to display first n prime numbers.

**Theory:**

#### **Logic**

We have taken a while loop which acts like a counter means as soon as we get a prime number the i value will be incremented.Then we have taken a for loop for taking every number.the if condition is done so as to make sure that if the no is prime or not.Flag variable is used which signals us that if number taken is prime or not.

**Source code:**

import java.util.Scanner;

public class prog2 {

static void print\_primes\_till\_N(int N)

{

// Declare the variables

int i, j, flag;

// Print display message

System.out.println("Prime numbers between 1 and "+ N + " are:");

// Traverse each number from 1 to N

// with the help of for loop

for (i = 1; i <= N; i++)

{

// Skip 0 and 1 as they are

// neither prime nor composite

if (i == 1 || i == 0)

continue;

// flag variable to tell

// if i is prime or not

flag = 1;

for (j = 2; j \*j <=i; ++j)

{

if (i % j == 0)

{

flag = 0;

break;

}

}

// flag = 1 means i is prime

// and flag = 0 means i is not prime

if (flag == 1)

System.out.print(i + " ");

}

}

// Driver code

public static void main (String[] args)

{

int n;

Scanner s = new Scanner(System.in);

System.out.println("Enter the value of n:");

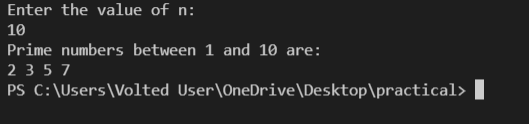
n = s.nextInt();

print\_primes\_till\_N(n);

}

}

**Output:**



**LAB 2.3**

**Aim:** WAP to check whether the input year is leap or not

**Theory:**

This is a Java Program to Find if a Given Year is a Leap Year.

Enter any year as an input. We first whether the given year is divisible by 400 or not. If it is divisible then it is a leap year else we check for further conditions. Now if it is divisible by 100 then it is not a leap year or else we further divide it by 4. If it is divisible then it is a leap year else its not.

Here is the source code of the Java Program to Find if a Given Year is a Leap Year. The Java program is successfully compiled and run on a Windows system. The program output is also shown below.

**Source code:**

import java.util.Scanner;

public class prog3 {

static boolean checkYear(int year)

{

// If a year is multiple of 400,

// then it is a leap year

if (year % 400 == 0)

return true;

// Else If a year is multiple of 100,

// then it is not a leap year

if (year % 100 == 0)

return false;

// Else If a year is multiple of 4,

// then it is a leap year

if (year % 4 == 0)

return true;

return false;

}

// Driver method

public static void main(String[] args)

{

int n;

Scanner s = new Scanner(System.in);

System.out.println("Enter any year:");

n = s.nextInt();

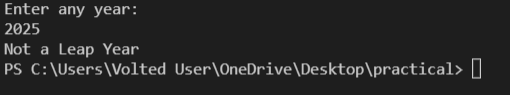
System.out.println( checkYear(n)? "Leap Year" :

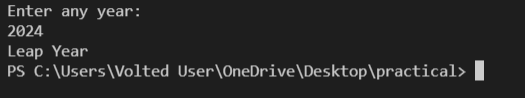
"Not a Leap Year" );

}

}

**Output:**





**LAB 2.4**

**Aim:** . Write an application that accepts two doubles as its command line arguments, multiple these together and display the product

**Theory:**

## **What Are Command Line Arguments?**

The command-line arguments are passed to the program at run-time. Passing command-line arguments in a Java program is quite easy. They are stored as strings in the [String array](https://www.edureka.co/blog/string-array-in-java/) passed to the args parameter of main() method in Java.

**Table Of Contents # 1 :**

## Command Line Arguments – Def, Syntax, Examples

In fact command line arguments are using just for a subject purpose and may be it is used in advance java core system in order to build the applications, just an expectation. Since , there may be a lot of ways to execute the program.

But , it is necessary that you need to know as a newbie about command line arguments.What are command line arguments?

The console is an interface between the user and program.

When a user enters the inputs on the console using commands, we sending the input as an argument to the main method in java that’s why in public static void main() we creating a string array to store values which work at executing time.

**Source code:**

public class prog4 {

public static void main(String[] args) {

int product;

int a = Integer.parseInt(args[0]);

int b = Integer.parseInt(args[1]);

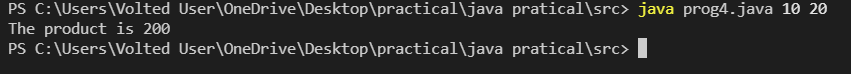
product = a \* b;

System.out.println("The product is " + product);

}

}

**Output:**



**LAB 2.5**

**Aim:** Write an application that accepts one command line argument, display the line of reporting if number is even or odd.

**Theory:**

In this program, you'll learn to check if a number entered by an user is even or odd. This will be done using if...else statement and ternary operator in Java.

To understand this example, you should have the knowledge of the following [Java programming](https://www.programiz.com/java-programming) topics:

* [Java if...else Statement](https://www.programiz.com/java-programming/if-else-statement)
* [Java Scanner Class](https://www.programiz.com/java-programming/scanner)

**Source code:**

public class prog5 {

// Function to the check Even or Odd

public static int isEvenOrOdd(int num)

{

return (num % 2);

}

// Driver code

public static void main(String[] args)

{

// Check if length of args array is

// greater than 0

if (args.length > 0) {

// Get the command line argument and

// Convert it from string type to integer type

int num = Integer.parseInt(args[0]);

// Get the command line argument

// and check if it is even or odd

int res = isEvenOrOdd(num);

// Check if res is 0 or 1

if (res == 0)

// Print Even

System.out.println("Even\n");

else

// Print Odd

System.out.println("Odd\n");

}

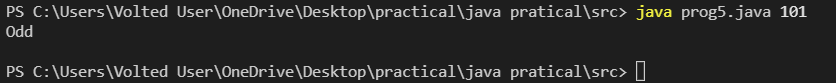
else

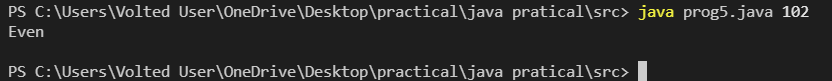
System.out.println("No command line " + "arguments found.");

}

}

**Output:**





**LAB 2.6**

**Aim:** Write an application that accepts radius of a circle as its command line argument display the area.

**Theory:**

In geometry, the area enclosed by a circle of radius r is πr2. Here the Greek letter π represents a constant, approximately equal to 3.14159, which is equal to the ratio of the circumference of any circle to its diameter.

The circumference of a circle is the linear distance around its edge.

**Source code:**

public class prog6 {

public static void main(String[] args) {

if (args.length > 0) {

int r = Integer.parseInt(args[0]);

double area = 3.14 \* r \* r;

System.out.println("Area of the circle is: " + area);

}

else {

System.out.println("No command line "

+ "arguments found.");

}

}

}

**Output:**



VIVA VOCE QUESTION

1. What is a pointer and does Java support pointers?

Ans Pointer is a reference handle to a memory location. Improper handling of pointers leads to memory leaks and reliability issues hence Java doesn't support the usage of pointers.

2. What is the base class of all classes?

Ans java.lang.Object

3. Does Java support multiple inheritance?

Ans Java doesn't support multiple inheritance.

4. Is Java a pure object-oriented language?

Ans Java uses primitive data types and hence is not a pure object-oriented language.

5. Are arrays primitive data types?

Ans In Java, Arrays are objects.