Java Programming
(Assignment –I)

Submitted in partial fulfilment of the requirements for the degree of

Post Graduate Diploma in Information Technology

by

Vijayananda D Mohire

(Registration No.200508208)

Information Technology Department
Symbiosis Bhavan,
1065 B, Gokhale Cross
Road, Model Colony, Pune – 411016,
Maharashtra, India
(2007)

Java Programming

Table of Contents

Java Programming

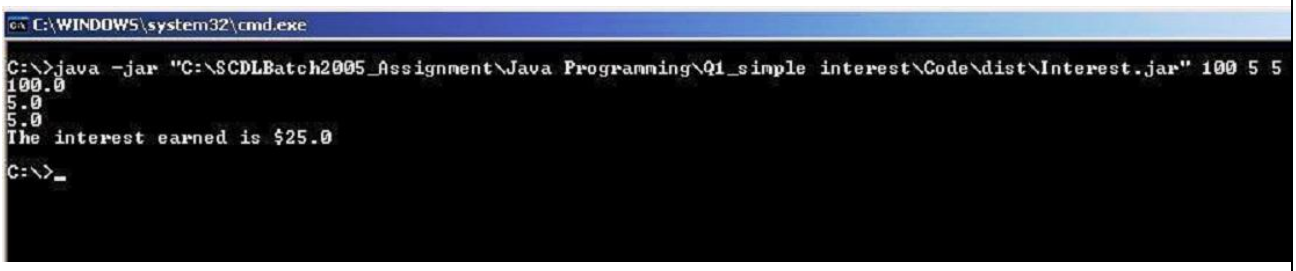
Question 1 Write a Java program that calculates and prints the simple interest using the formula :
Simple Interest = PNR / 100
Input values P,N,R should be accepted as command line input as below.
e.g. java SimpleInterest 5 10 15

Answer 1
Code:

```
package Interest_Pkg;

/**
 * Define Interest class
 * @author Vijayananda D Mohire
 *
 */
public class Interest {

    /** Creates a new instance of Interest, default constructor */
    public Interest() {
    }

    /**
     * Main function for entry point of the project
     * @param args the command line arguments
     *
     */
    public static void main(String[] args) {
        // Declare and initialize the P,N,R and SI variables
         double P=0;
         double N=0;
         double R =0;
         double SI = 0;

         // Check for the presence of Console input args
if (args.length > 0) {
    try {
        // Read the Console inputs and convert them to double data type
        P = Integer.parseInt(args[0]);
```

```
        N = Integer.parseInt(args[1]);
        R = Integer.parseInt(args[2]);
          }
    // wrap the above code in a try catch block
    catch (NumberFormatException e)
      {
        // For any Number Format exception caught, throw an error
        System.err.println("Argument must be an integer");

        // Quit application with status code 1
        System.exit(1);
        }
}
    // Print the values of Principal, Tenure and Rate of Interest
    System.out.println(P);
    System.out.println(N);
    System.out.println(R);

    // Calculate the Simple Interest using the formula
    SI = (P*N*R)/100;

    // Print the Simple Interest earned
    System.out.print("The interest earned is $");
    System.out.println(SI);

        }

}
```

Results:



Evaluator's Comments if any:

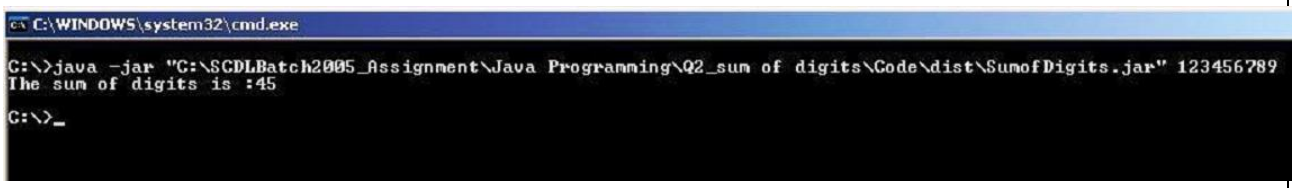
Question 2 Write a program to compute sum of digits of a given number. (Hint: Separate digits one by one from the number and then add)

Answer 2

```
package SumofDigits_Pkg;

/**
 * Define SumofDigits class
 * @author Vijayananda D Mohire
 */
public class SumofDigits {

    /** Creates a new instance of SumofDigits, default constructor */
    public SumofDigits() {
    }

    /**
     * Main function for entry point of the project
     * @param args the command line arguments
     */
    public static void main(String[] args) {

        // Declare and initialize the variables
        int Number = 0;
        int N =0;
        int sum = 0;
        int digit =0;

        // Check for the presence of Console input args
        if (args.length > 0) {
```

```
    try {
        // Read the Console inputs and convert them to integer data type
        Number = Integer.parseInt(args[0]);

     // wrap the above code in a try catch block
    } catch (NumberFormatException e) {

     // For any Number Format exception caught, throw an error
        System.err.println("Argument must be an integer");

        // Quit application with status code 1
        System.exit(1);
    }
}
        // If entered number is negative, negate the symbol to make it positive
        N = Number;
          if (N<0) N = -N;
          sum=0;

         // loop and sum the digits, use modulos operator to isolate each digit
          while (N!=0) {
              digit = N % 10;
              sum = sum + digit;
              N = N / 10;
    }
   // Print the value of Sum of the digits obtained above
        System.out.print("The sum of digits is :");
        System.out.println(sum);
    }
}
```

Results:



Evaluator's Comments if any:

Question 3 The numbers in the following sequence are called the fibonacci numbers . 0 , 1 , 1 , 2, 3 , 5 , 8 , 13 , ……..

Write a program using do…..while loop to calculate and print the first m Fibonacci numbers.

(Hint : After the first two numbers in the series, each number is the sum of preceding two numbers.)

Answer 3

Code:

```
package Fibonacci_Pkg;
 /**
  * Define Fibonacci class
  * @author Vijayananda D Mohire
  */

 /**
  *
  * Creates a new instance of Fibonacci, default constructor
  */
 public class fibonacci {

     /** Creates a new instance of fibonacci */
     public fibonacci() {
     }

 /**
  * Main function for entry point of the project
  * @param args the command line arguments
  */
 public static void main(String[] args) {

   // Declare and initialize the variables
    int n0 = 0, n1 = 1 , n2 = 0;
    int M = 0;
```

```
  // Check for the presence of Console input args
    if (args.length > 0) {
try {

   // Read the Console inputs and convert them to integer data type
    M = Integer.parseInt(args[0]);

// wrap the above code in a try catch block
} catch (NumberFormatException e) {

    // For any Number Format exception caught, throw an error
    System.err.println("Argument must be an integer");

    // Quit application with status code 1
    System.exit(1);
}
     }
   // Print the first two default values of the series
    System.out.print(n0 + "," + n1 + ",");

    // Loop and add the two numbers to get next number
    for ( int i = 0; i < M-2; i++) {
    n2 = n1 + n0;

    // Print next number in the series
   System.out.print(n2 + ",");

   // Swap the previous and next numbers
    n0= n1;
    n1= n2;
    }

    System.out.println();
       }
}
```

Results:

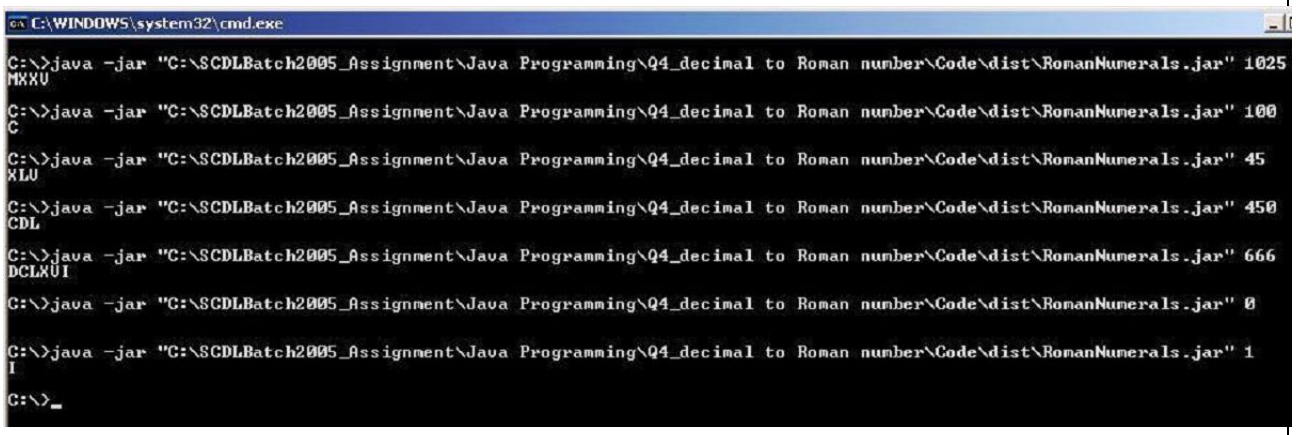
Question 4 Write a program that converts a decimal number to Roman number. Decimal Number is accepted as command line input at the time of execution.

Answer 4
Code:

```
package RomanNumerals_Pkg;
/**
 * Define RomanNumerals class
 * @author Vijayananda D Mohire
 */

public class RomanNumerals {

    /** Creates a new instance of RomanNumerals */
    public RomanNumerals() {
    }

    /**
    * Main function for entry point of the project
    * @param args the command line arguments
     */
    public static void main(String[] args) {

        // Declare and initialize the ROMAN variables
        char numerals[] = {'I', 'V', 'X', 'L', 'C', 'D', 'M'};

        StringBuffer s = new StringBuffer();
        char c1, c2, c3;
        int j;

     // Check for the presence of Console input args
     char[] decimal =   args[0].toCharArray();

    //  Loop until length of the input Number
    for(int i = 0; i < decimal.length; i++) {
      j  = (decimal.length - i - 1) * 2;
```

```
    // Assign Roman values based on the Units, Tenth and Hunderedth postion  of the
Number
      c1 = numerals[j];
      c2 = j + 1 < numerals.length ? numerals[j + 1] : ' ';
      c3 = j + 2 < numerals.length ? numerals[j + 2] : ' ';

      // Built the Roman number by appending it with equivalent Roman digit like I, V
etc..
      switch(decimal[i]) {
        case '1' : s = s.append(c1); break;
        case '2' : s = s.append(c1).append(c1); break;
        case '3' : s = s.append(c1).append(c1).append(c1); break;
        case '4' : s = s.append(c1).append(c2); break;
        case '5' : s = s.append(c2); break;
        case '6' : s = s.append(c2).append(c1); break;
        case '7' : s = s.append(c2).append(c1).append(c1); break;
        case '8' : s = s.append(c2).append(c1).append(c1).append(c1); break;
        case '9' : s = s.append(c1).append(c3);
      }
    }

      // Print the completly built number as above
    System.out.println(s);

       }

   }
```

Results:



Question 5 Write a program that prints prime numbers between 1 to n. Number n should be accepted as command line input.

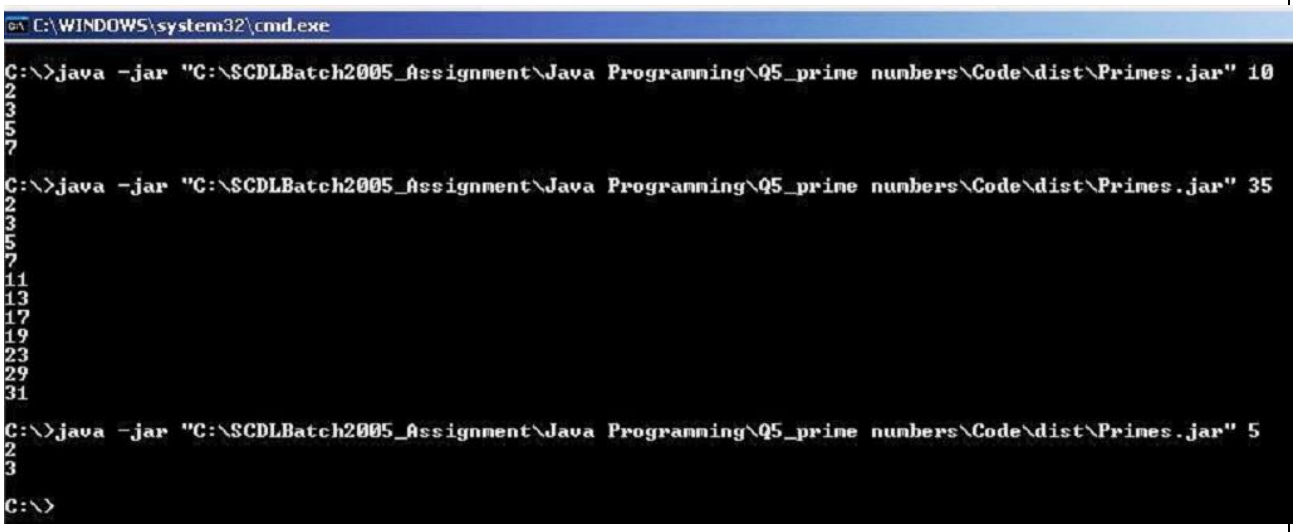
Answer 5
Code:

```
package PrimeNum_Pkg;
/**
 * Define Primes class
 * @author Vijayananda D Mohire
 */

public class Primes {

   /** The default stopping point for primes */
  public static final long DEFAULT_STOP = 4294967295L;
  /** The first prime number */
  public static final int FP = 2;

   /**
    * Set the Maximim Size limit
    */
  static int MAX = 10000;

   /** Creates a new instance of Primes */
   public Primes() {
   }

   /**
    * * Main function for entry point of the project
    * @param args the command line arguments
    */
   public static void main(String[] args) {

      // Declare and initialize the variables
   long[] prime = new long[MAX];
   long stop = DEFAULT_STOP;
```

```
 // Check for the presence of Console input args
if (args.length == 1) {
  stop = Long.parseLong(args[0]);
}

prime[1] = FP;       // P1 (ignore prime[0])
long n = FP+1;       // odd candidates
int j = 1;        // numberFound

boolean isPrime = true;  // for 3

do {

  if (isPrime) {
    if (j == MAX-1) {
      // Grow array dynamically if needed
      long[] np = new long[MAX * 2];
      System.arraycopy(prime, 0, np, 0, MAX);
      MAX *= 2;
      prime = np;
    }
    prime[++j] = n;  // P2
    isPrime = false;
  }
  n += 2;        // P4

  for (int k = 2; k <= j && k < MAX; k++) {  // P5, P6, P8
    long q = n / prime[k];
    long r = n % prime[k];
    if (r == 0) {
      break;
    }
    if (q <= prime[k]) {    // P7
      isPrime = true;
      break;
    }
  }

} while (n < stop);        // P3

// Print the completly Prime number list
```

```
    for (int i=1; i<=j; i++)
      System.out.println(prime[i]);
  }
    }
```

Results:



Question 6 Write a program to print the following output using the for loop.

1
2 2
3 3 3
4 4 4 4
5 5 5 5 5

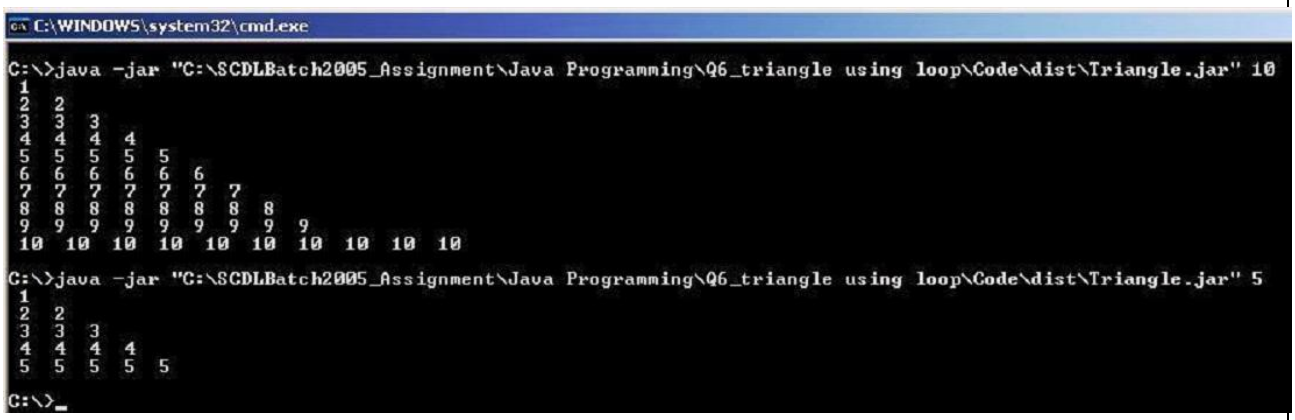
Answer 6

Code:

```
  package Triangle_Pkg;
/**
 * Define Triangle class
 * @author Vijayananda D Mohire
```

```
 */

public class Triangle {

    /** Creates a new instance of Triangle */
    public Triangle() {
    }

    /**
     * Main function for entry point of the project
     * @param args the command line arguments
     */
    public static void main(String[] args) {

      // Declare and initialize the variables
        int num =0;

        if (args.length == 1)
        num = Integer.parseInt(args[0]);
        else
        System.out.println("Please enter a Number");

          // Outer loop for new line increment
         for(int i=1;i<=num;i++){

             // Inner loop for number of times to print on each line
            for(int j=1;j<=i;j++){

             // Print the incremented Number
             System.out.print(" "+i+" ");

            }
              // Print new line
            System.out.print("\n");

          }
    }

}
```

Results:



Question 7 Write a program to find the Factorial of a number using Recursion. Factorial can be defined as Factorial(n) = 1 * 2 * 3 ….* (n-1) * n.

Answer 7

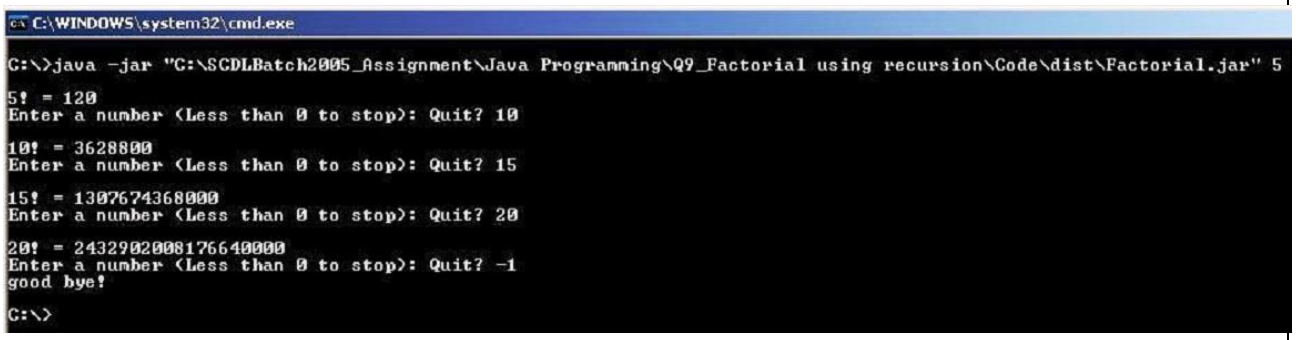
Code:

```
package Factorial_Pkg;
/**
 * Define Triangle class
 * @author Vijayananda D Mohire
 */

import java.io.Console;

/**
 * Define Factorial class
 * @author Vijayananda D Mohire
 */
public class Factorial {
    /** Creates a new instance of Factorial */
    public Factorial() {
    }

    /**
```

```
 * Main function for entry point of the project
 * @param args the command line arguments
 */

public static void main(String[] args) {

    // Declare and initialize the variables
    int num =0;
    String input;

    // Check for Input
    if (args.length > 0) {
try {
    num = Integer.parseInt(args[0]);

}
// Catch error and display the error
catch (NumberFormatException e) {
    System.err.println("Argument must be an integer");
    System.exit(1);
}
     }

    // Loop until Number is greater than 0
    while (num >= 0)
 {

 //Call the Recursive function  and print the return value
  System.out.print("\n" + num + "! = " + factorial (num));
   System.out.print("\nEnter a number (Less than 0 to stop): ");

   // Prompt for user to confirm to continue or Quit
    try {
    Console console = System.console();
    input  = console.readLine("Quit? ");
    num = Integer.parseInt( input );
     }
   catch (Exception e) {
       System.err.println("Argument must be an integer");
   }
   }
```

```
    // Come put of loop if user Quits by entering negative number
      System.out.println("good bye!");
    } // main method


    /**
     * Recursive factorial function
     * @param n long  Recursive factorial function input
     * @return long number
     */
      public static long factorial (long n)
    {
     // Return default value for number less than 2
     if (n < 2)
         return 1;
     // self call and return value after multiplying n and n-1
     return n * factorial(n - 1);
    }

    }
```

Results:



Question 8 Define an exception called "NoMatchException" that is thrown when a string is not equal to "Symbiosis". Write a Java program that uses this exception.

Answer 8

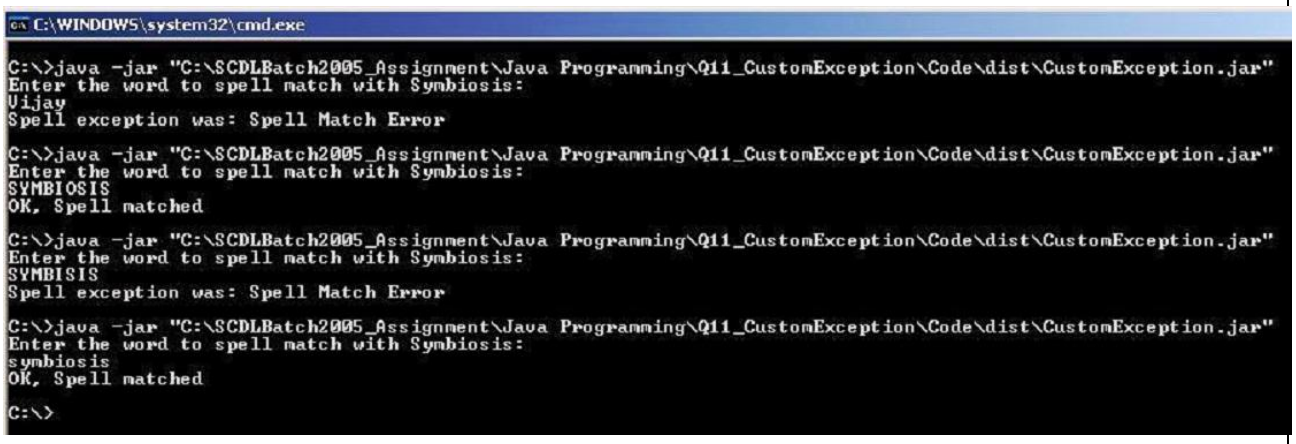
Code:

```
 package CustomException_Pkg;
import java.io.*;
/**
 * Define CustomException class
 * @author Vijayananda D Mohire
 */
public class CustomException {
  /**
 * Define variable
 * */
    static String s = "";

   /**
    * Main function for entry point of the project
    * @param args the command line arguments
    */

  public static void main (String args[])
  {
      // Declare and initialize the variables for reading the input
   InputStreamReader is = new InputStreamReader(System.in);
   BufferedReader buf = new BufferedReader(is);

   System.out.println("Enter the word to spell match with Symbiosis: ");

// try - catch to read the input line
   try
   {
     s = buf.readLine();
   }
   catch (IOException e)
   {
     System.out.println("IOException was " + e.getMessage());
   }

   // try - catch uses NoMatchException
   try
```

```
    {
      checkSpelling();   // this method uses NoMatchException
    }
    catch (NoMatchException se)   // but it is caught here
    {
      System.out.println("Spell exception was: " + se.getError());
    }
  } // end main

    /**
     * Check spelling of typed in word. Throw exception if wrong.
     * @throws CustomException_Pkg.NoMatchException Throws Exception if the
Synbol doesnot match string "Symbiosis"
     */

  private static void checkSpelling() throws NoMatchException
  {
    if (s.equalsIgnoreCase("Symbiosis"))
      System.out.println("OK, Spell matched");
    else
      throw new NoMatchException("Spell Match Error");
   }
}  // end main class

/**
  * Custom exception class that descends from Java's Exception class.
  */
class NoMatchException extends Exception
{
   /**
 * Define string variable
 * */
    String mistake;

/**
 * Default constructor - initializes instance variable to unknown
 * */
  public NoMatchException()
  {
    super();             // call superclass constructor
    mistake = "unknown";
```

```
 }

/**
    * Constructor receives some kind of message that is saved in an instance variable.
    * @param err Custom Error message
    */
  public NoMatchException(String err)
  {
    super(err);     // call super class constructor
    mistake = err;  // save message
  }

  /**
    * public method, callable by exception catcher. It returns the error message.
    * @return Returns error string
    */
  public String getError()
  {
    return mistake;
  }
}
```

Results:



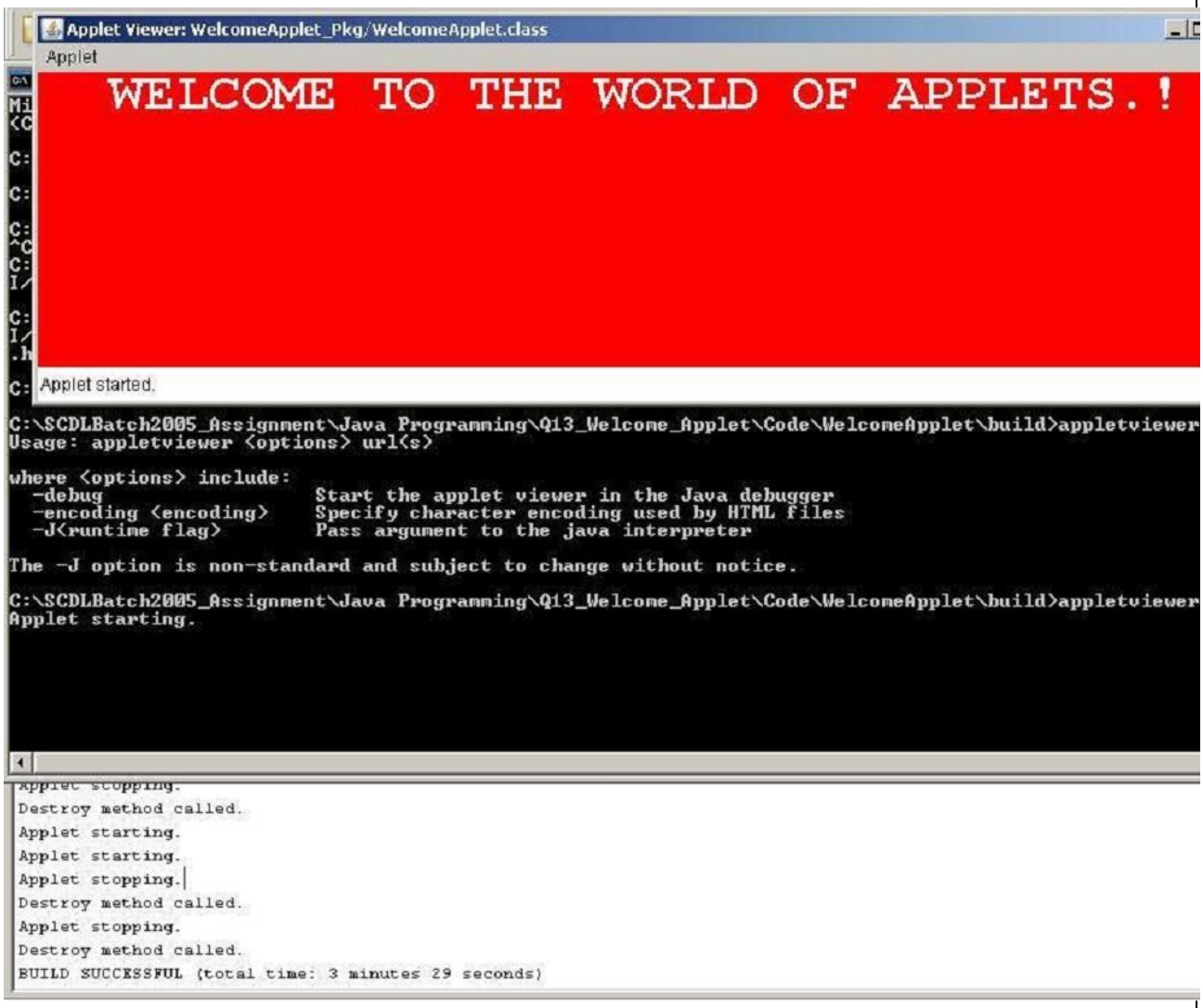
Question 9 Write a program to create an applet with red back ground and display the Message WELCOME TO THE WORLD OF APPLETS. Write its HTML document also. Specify width and height to the Applet.

Answer 9
Code:

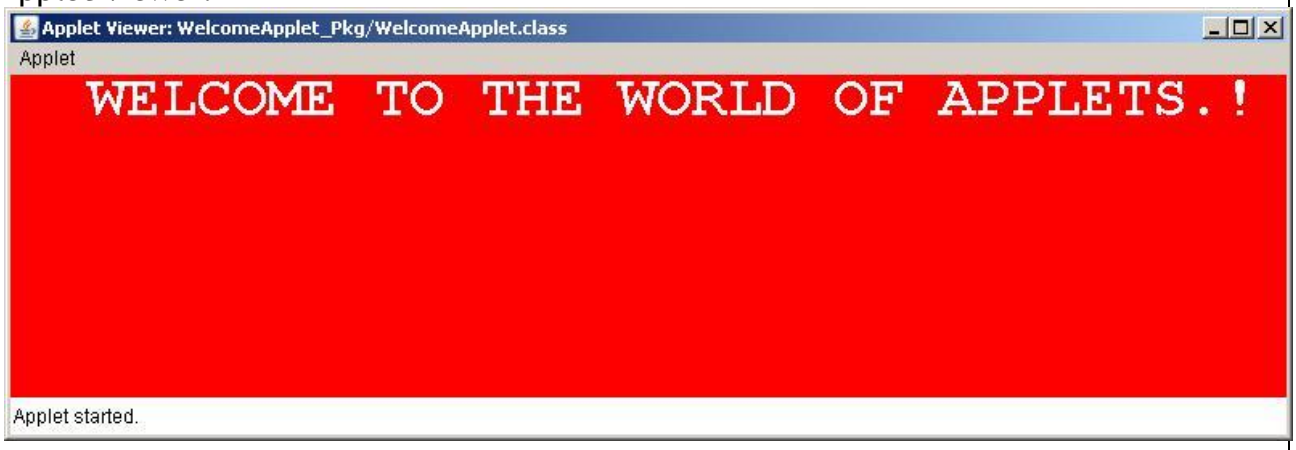
```
package WelcomeApplet_Pkg;

import java.applet.Applet;
import java.awt.*;

/**
 * Define WelcomeApplet class
 * @author Vijayananda D Mohire
 */
public class WelcomeApplet extends java.applet.Applet {

   /** Initialization method that will be called after the applet is loaded
    *  into the browser.
    */
   public void init() {

        setBackground( Color.RED ); // change here the applet window color
   }

   /** Start method that will be called at the begining.
    */
    public void start(){
    System.out.println("Applet starting.");
 }

    /** Stop method that will be called at the end.
    */
 public void stop(){
    System.out.println("Applet stopping.");
 }

 /** Destroy method that will be called at the last.
    */
 public void destroy(){
    System.out.println("Destroy method called.");
 }

 /**
```

```
    * Paint method that use Graphics to initiaize the required Font and Color
    * @param g Graphics variable
    */
  public void paint(Graphics g) {
  Font font1 = new Font( "Courier", Font.BOLD, 36 );
  g.setFont( font1 );
  g.setColor( Color.WHITE );
  g.drawString( "WELCOME TO THE WORLD OF APPLETS.!", 50, 25 );

  }
     }
----------WelcomeApplet_Html----------
<HTML>
<HEAD>
   <TITLE>Applet HTML Page</TITLE>
</HEAD>
<BODY>
<H3><HR WIDTH="100%">Applet HTML Page<HR WIDTH="100%"></H3>

<P>
<APPLET codebase="classes" code="WelcomeApplet_Pkg/WelcomeApplet.class"
width=500 height=200></APPLET>
</P>

<HR WIDTH="100%"><FONT SIZE=-1><I>Generated by NetBeans IDE</I></FONT>
</BODY>
</HTML>
```

Results:
Cmd line



Applet Viewer:



Question 10 Write a program that detects successive repeated occurrence of a letter in a word. For example, in the word “explanation” letter ‘a’ and ‘n’ occurs twice.

Answer 10

Code:

```
package repeatedletters;
import java.util.*;
import java.io.*;

/**
 * Define RepLetter class
 * @author Vijayananda D Mohire
 */
public class RepLetter {

   /**
    * Main function for entry point of the project
    * @param args the command line arguments
    * @throws java.io.IOException Throws IO Exception
    */
   public static void main(String[] args) throws IOException
{
// Use Buffered read
BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
System.out.print("Please Enter aString ");
System.out.println();
String str=br.readLine();

// Convert input String to Char array
char[]buffer =str.toCharArray();

//Loop for buffer length
for(int counter =0;counter<buffer.length;counter++)
{ char letter= buffer[counter];
```

```
int flage=0;
for ( int i=0; i<buffer.length; i++)
{
if (letter==buffer[i])

    // if letter repeation is found increment flag
flage++;
}
if(counter==0)
{
   // Type the occurance
System.out.println("The letter ="+letter+" is repeated "+flage+" No of Times ");
continue;
}
boolean flag=false;
for(int j=counter-1;j>=0;j--)
{
if(letter==buffer[j])
flag=true;
}
if(!flag)
{
System.out.println("The letter ="+letter+" is repeated "+flage+" No of Times ");
}} } }
```

Results: