**Name:**

**Enrolment Number:**

|  |  |
| --- | --- |
| **Task\_1** | Write the missing code in the given file “BinToDecimal.cs” code to generate the following output. |
| **ANS** | //C# program to convert a binary number into a decimal number.  using System;  class BinToDecimal  {  static void Main(string[] args)  {  int binNum = 0;  int decNum = 0;  int i = 0;  int rem = 0;    Console.Write("Enter a binary number: ");  binNum = int.Parse(Console.ReadLine());  /\*  Missing code  \*/  while (binNum > 0)  {  rem = binNum % 10;  decNum += rem \* (int)Math.Pow(2, i);  i++;  binNum /= 10;  }  Console.WriteLine("\nDecimal number: " + decNum);  }  } |
|  |  |
|  |  |
| **Task\_2** | Rearrange the given code given in “ReverseArray.cs” file to correct the program. The resultant program will be to enter 5 elements into an array and store and print the array in reverse order. |
| **ANS** | using System;  class ReverseArray  {  static void Main()  {  int[] arr1 = new int[5];  int[] arr2 = new int[5];  //Read numbers into array  int i = 0;  int j = 0;  Console.WriteLine("Enter numbers : ");  for (i = 0; i < 5; i++)  {  Console.Write("Element[" + (i + 1) + "]: ");  arr1[i] = int.Parse(Console.ReadLine());  }  //Assign elements of arr1 from last to first element to arr2  for (i = 0,j=arr1.Length-1; i < arr1.Length; i++)  {  arr2[i] = arr1[j--];  }  //Reverse array elements in arr2  Console.WriteLine("Reverse elements : ");  for (i = 0; i < 5; i++)  {  Console.WriteLine("Element[" + (i + 1) + "]: "+ arr2[i]);  }          }  } |
|  |  |
|  |  |
| **Task\_3** | Write Square( ) method into “OutDemo.cs" file and also call into main method to get following output. |
| **ANS** | using System;  class Program  {  static void Main(string[] args)  {  int x=10;  square(ref x);  Console.WriteLine("Sqaure of x: {0}", x);  square(ref x);  Console.WriteLine("Sqaure of x: {0}", x);  }  // Define Square method with out parameter  public static void square(ref int a)  {  a \*= a;  }  } |
|  |  |
|  |  |
| **Task\_4** | Complete the code in “MultiLeveInheritance.cs” file to implement multilevel inheritance by following the below figure. Accept and display data for one student. |
| **ANS** | using System;  namespace MockTest  {  class Student  {  public int Roll\_No { get; set; }  public string Name { get; set; }  }  class Test : Student  {    public int Mark\_1 { get; set; }  public int Mark\_2 { get; set; }  }  class Result : Test  {  public int Total()  {  return base.Mark\_1 + base.Mark\_2;  }  }  class Demo  {  public static void Main(string[] args)  {  Result student = new Result();  student.Name = "Sanjay";  student.Roll\_No = 41;  student.Mark\_1 = 90;  student.Mark\_2 = 80;  Console.WriteLine(student.Total());  }  }  } |
|  |  |
|  |  |
| **Task\_5** | Design a form like below. Implement following functionality to the application.  (ListBox tasks) |
| **ANS** | using System;  using System.Collections;  using System.Collections.Generic;  using System.ComponentModel;  using System.Data;  using System.Drawing;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  using System.Windows.Forms;  namespace EventApp  {  public partial class Form1 : Form  {  ArrayList items = new ArrayList() {"zebra","Education","Food For All","Freedom of Speech","Good Books","Good Movies" };  public Form1()  {  InitializeComponent();  }  private void buttonFill\_Click(object sender, EventArgs e)  {  listBox1.Items.Clear();  foreach (var item in items)  {  listBox1.Items.Add(item);  }  }  private void buttonShort\_Click(object sender, EventArgs e)  {  items.Sort();  listBox1.Items.Clear();  foreach (var item in items)  {  listBox1.Items.Add(item);  }  }  private void buttonCount\_Click(object sender, EventArgs e)  {  labelTotalItem.Text = listBox1.Items.Count.ToString();  }  private void buttonClear\_Click(object sender, EventArgs e)  {  listBox1.Items.Clear();  }  private void buttonRemoveItem\_Click(object sender, EventArgs e)  {  String selectedItem = listBox1.SelectedItem.ToString();  listBox1.Items.Remove(selectedItem);  }  }  } |
|  |  |
|  |  |
| **Task\_6** | Design and write code for the following. (Calculation) |
| **ANS** | using System;  using System.Collections.Generic;  using System.ComponentModel;  using System.Data;  using System.Drawing;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  using System.Windows.Forms;  namespace EventApp  {  public partial class Calculator : Form  {  public Calculator()  {  InitializeComponent();  }  private void buttonSum\_Click(object sender, EventArgs e)  {  double num1 = Convert.ToDouble(textBox1.Text);  double num2 = Convert.ToDouble(textBox2.Text);  MessageBox.Show(String.Format("{0} + {1} = {2}", num1, num2, num1 + num2));  }  private void buttonSub\_Click(object sender, EventArgs e)  {  double num1 = Convert.ToDouble(textBox1.Text);  double num2 = Convert.ToDouble(textBox2.Text);  MessageBox.Show(String.Format("{0} - {1} = {2}", num1, num2, num1 - num2));  }  private void buttonMul\_Click(object sender, EventArgs e)  {  double num1 = Convert.ToDouble(textBox1.Text);  double num2 = Convert.ToDouble(textBox2.Text);  MessageBox.Show(String.Format("{0} \* {1} = {2}", num1, num2, num1 \* num2));  }  private void buttonDiv\_Click(object sender, EventArgs e)  {  double num1 = Convert.ToDouble(textBox1.Text);  double num2 = Convert.ToDouble(textBox2.Text);  MessageBox.Show((num2 != 0) ?String.Format("{0} / {1} = {2}", num1, num2, num1 / num2):"Can't Divide By Zero");  }  }  } |
|  |  |
|  |  |
| **Task\_7** | Create a Database table for the Task\_8 form with the required columns. |
| **ANS** |  |
|  |  |
|  |  |
| **Task\_8** | Design a form as per following. Write code for Register button to insert record into a table designed in above task.(Event Registration Form) |
| **ANS** |  |
|  |  |
|  | using System;  using System.Collections.Generic;  using System.ComponentModel;  using System.Data;  using System.Data.SqlClient;  using System.Drawing;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  using System.Windows.Forms;  namespace EventApp  {  public partial class Event : Form  {  public Event()  {  InitializeComponent();  }  private void button1\_Click(object sender, EventArgs e)  {  string isGoing, firstName, lastName, email, phoneNumber, SmsNumber, numberOfGuest;  if (radioButtonYes.Checked)  isGoing = "Yes";  else  isGoing = "No";  firstName = textBoxFirstName.Text;  lastName = textBoxLastName.Text;  email = textBoxEmail.Text;  phoneNumber = textBoxPhoneNumber.Text;  SmsNumber = textBoxSmsNumber.Text;  numberOfGuest = comboBoxGuest.SelectedItem.ToString();  string conString = "Data Source=(LocalDB)\\MSSQLLocalDB;AttachDbFilename=\"C:\\Users\\Sanjay Sah\\Documents\\EventsCIE1.mdf\";Integrated Security=True;Connect Timeout=30";  string query = "insert into Participants(first\_name, last\_name, email, phone\_number, sms\_number, no\_of\_guest, is\_attending) Values(@firstName, @lastName, @email, @phoneNumber, @smsNumber, @noOfGuests, @isAttending)";  SqlConnection con = new SqlConnection(conString);  con.Open();  SqlCommand cmd = new SqlCommand(query, con);  cmd.Parameters.AddWithValue("@firstName", firstName);  cmd.Parameters.AddWithValue("@lastName", lastName);  cmd.Parameters.AddWithValue("@email", email);  cmd.Parameters.AddWithValue("@phoneNumber", phoneNumber);  cmd.Parameters.AddWithValue("@smsNumber", SmsNumber);  cmd.Parameters.AddWithValue("@noOfGuests", numberOfGuest);  cmd.Parameters.AddWithValue("@isAttending", isGoing);  cmd.ExecuteNonQuery();  con.Close();  MessageBox.Show("Insert Successfull");    }  }  } |