|  |
| --- |
| Day9 Assignment  By  Paluru Mounika  03-02-2022 |

1.c# program to print

a)factorial of number

b)factors of a number

c)check if it is prime or not

|  |
| --- |
| Program: |
| Code: |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Day9Project1  {  class Operations  {  private int input;  public void ReadInput()  {  Console.WriteLine("Enter number: ");  input = Convert.ToInt32(Console.ReadLine());  }  public int Factorial()  {  int fact = 1;  for (int i = 1; i <= input; i++)  {  fact = fact \* i;  }  return fact;  }  public void Factors()  {  for (int i = 1; i <= input; i++)  {  if (input % i == 0)  Console.WriteLine(i);  }  }  public bool IsPrime()  {  int count = 0;  for (int i = 1; i < input; i++)  {  if (input % 1 == 0)  count++;  }  if (count == 2)  return true;  else  return false;  }  }  internal class Program  {  static void Main(string[] args)  {  Operations ob = new Operations();  ob.ReadInput();  Console.WriteLine(ob.Factorial());  ob.Factors();  if (ob.IsPrime())  Console.WriteLine("Input is PRIME Number");  else  Console.WriteLine("Not a Prime Number");  Console.ReadLine();  }  }  } |
| Output: |
|  |

2.c# program to readtwo numbers from user and print

a)sum of two numbers

b)difference of two numbers

c)product of two numbers

d)division of two numbers

|  |
| --- |
| Program: |
| Code: |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  // \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  // Author : paluru mounika  // Purpose : Arithmetic Operations  //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  namespace Day9Project2  {  class ArithmeticOperations  {  private int a;  private int b;  public void ReadInput()  {  Console.WriteLine("Enter First Number: ");  a = Convert.ToInt32(Console.ReadLine());  Console.WriteLine("Enter Second Number: ");  b = Convert.ToInt32(Console.ReadLine());  }  public int AddNumbers()  {  return a + b;  }  public int Difference()  {  return a - b;  }  public int Product()  {  return a \* b;  }  public int Division()  {  return a % b;  }  }  internal class Program  {  static void Main(string[] args)  {  ArithmeticOperations ar = new ArithmeticOperations();  ar.ReadInput();  Console.WriteLine(ar.AddNumbers());  Console.WriteLine(ar.Difference());  Console.WriteLine(ar.Product());  Console.WriteLine(ar.Division());  Console.ReadLine();  }  }  } |
| Output: |
|  |

3.creat an employee class with variables of id,name,salary,company

Write methodes to read data and print data.

|  |
| --- |
| Program: employee class with 4 variables |
| Code: |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  // Author : paluru mounika  // Purpose: Display Employee Details  // \*\*\*\*\*\*\*\*\*\*\*\*\*\*  namespace Day9Project3  {  class Employee  {  public int id;  public string name;  public int salary;  public string company;  public void ReadData()  {  Console.WriteLine("Enter Employee ID: ");  id = Convert.ToInt32(Console.ReadLine());  Console.WriteLine("Enter Employee Name: ");  name = Console.ReadLine();  Console.WriteLine("Enter Employee Salary: ");  salary = Convert.ToInt32(Console.ReadLine());  company = "NationsBenifts";  }  public void PrintData()  {  Console.WriteLine($"Id:{id}, Name:{name}, Salary:{salary}, Company={company}");  }  }  internal class Program  {  static void Main(string[] args)  {  Employee emp1 = new Employee();  emp1.ReadData();  emp1.PrintData();  Employee emp2 = new Employee();  emp2.ReadData();  emp2.PrintData();  Console.ReadLine();  }  }  } |
| Output: |
|  |

4.difference between normal variable and statinc variable.

Static Variables:

- A static variable is associated with the class has only one copy per class but not for each object. An instance of a class does not have static variables.

- Static variables can be accessed by static or instance methods

- Memory is allocated when the class is loaded in context area at run time.

Non-Static Variables:

- Non-static variables will have one copy each per object. Each instance of a class will have one copy of non-static variables.

- Instance variables can be accessed only by the instance methods.

- Instance variables are allocated at compile time.

**5.write 5 points about constructor**

**1. A constructors is used to initialize class variables.**

**2. By default, C# has one constructor i.e., Default constructor to initialize class variables.**

**3. If user create user-defined constructor the default constructor will disappear.**

**4. Constructor name should be same as class name. If we use same variables as class variable use this. Keyword to differentiate class variable.**

**5. For a constructor, there should not be any return type not even void.**

**Eg : Public Employee(int id, string name)**

**6.creat employee class with 2 constructors**

|  |
| --- |
| **Program:**employee class with 2 constructor |
| **Code:** |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  // Author : paluru mounika  // Purpose : Employee Class Constructor  //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  namespace Day9Project4  {  class Employee  {  public int id;  public string name;  public int salary;  public static string company = "NationsBenefits";  public Employee()  {  this.id = 0;  this.name = null;  }  public Employee(int eid, string ename, int esalary)  {  id = eid;  name = ename;  salary = esalary;  }  public void ReadData()  {  Console.WriteLine("Enter Employee ID: ");  id = Convert.ToInt32(Console.ReadLine());  Console.WriteLine("Enter Employee Name: ");  name = Console.ReadLine();  Console.WriteLine("Enter Employee Salary: ");  salary = Convert.ToInt32(Console.ReadLine());  company = "NationsBenifts";  }  public void PrintData()  {  Console.WriteLine($"Id:{id}, Name:{name}, Salary:{salary}, Company={company}");  }  }  internal class Program  {  static void Main(string[] args)  {  Employee emp = new Employee(1, "mounika", 40000);  emp.PrintData();  Console.ReadLine();  }  }  } |
| **Output:** |
|  |