Bhanu Prakash Reddy Chilukuri

NBHealthtech  03-02-2022

Day9 morning assignment

|  |
| --- |
| 1. Write a C# program to read input from user and print  a. factorial of a number  b. factors of a number  c. check if it prime or not |

Code:

class Mathsoperations

{

private int input;

public void ReadInput()

{

Console.WriteLine("Enter Number");

input = Convert.ToInt32(Console.ReadLine());

}

/// <summary>

/// factorial of a number

/// </summary>

public void Factorial()

{

int fact = 1;

for(int i=1; i <= input; i++)

{

fact \*= i;

}

Console.WriteLine(fact);

}

/// <summary>

/// factors of a number

/// </summary>

public void PrintFactors()

{

for(int i=1;i<=input;i++)

{

if(input%i==0)

Console.WriteLine(i);

}

}

/// <summary>

/// prime number or not

/// </summary>

/// <returns>Is prime</returns>

public bool IsPrime()

{

int count = 0;

for(int i=1;i <= input;i++)

{

if (input % i == 0)

count++;

}

if(count == 2)

return true;

else

return false;

}

}

internal class Program

{

static void Main(string[] args)

{

Mathsoperations obj = new Mathsoperations();

obj.ReadInput();

obj.Factorial();

obj.PrintFactors();

if(obj.IsPrime())

Console.WriteLine("Input is prime number");

else

Console.WriteLine("input is not prme number");

Console.ReadLine();

}

}

Output:



|  |
| --- |
| 2. Write C# program to read two numbers from use and print  a. sum of two numbers  b. difference of two numbers  c. product of two numbers  d. division of two numbers. |

Code:

class MathsTask

{

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());

}

/// <summary>

/// Addition of two numbers

/// </summary>

/// <returns>Sum</returns>

public int AddNumbers()

{

return a + b;

}

/// <summary>

/// Subtraction of two numbers

/// </summary>

/// <returns>Sub</returns>

public int SubNumbers()

{

return a - b;

}

/// <summary>

/// Product of two numbers

/// </summary>

/// <returns>Product</returns>

public int MultiplyNumbers()

{

return a \* b;

}

/// <summary>

/// Divison of two numbers

/// </summary>

/// <returns>Division</returns>

public int Dividenumbers()

{

return a / b;

}

}

internal class Program

{

static void Main(string[] args)

{

MathsTask mt = new MathsTask();

mt.ReadInput();

Console.WriteLine(mt.AddNumbers());

Console.WriteLine(mt.SubNumbers());

Console.WriteLine(mt.MultiplyNumbers());

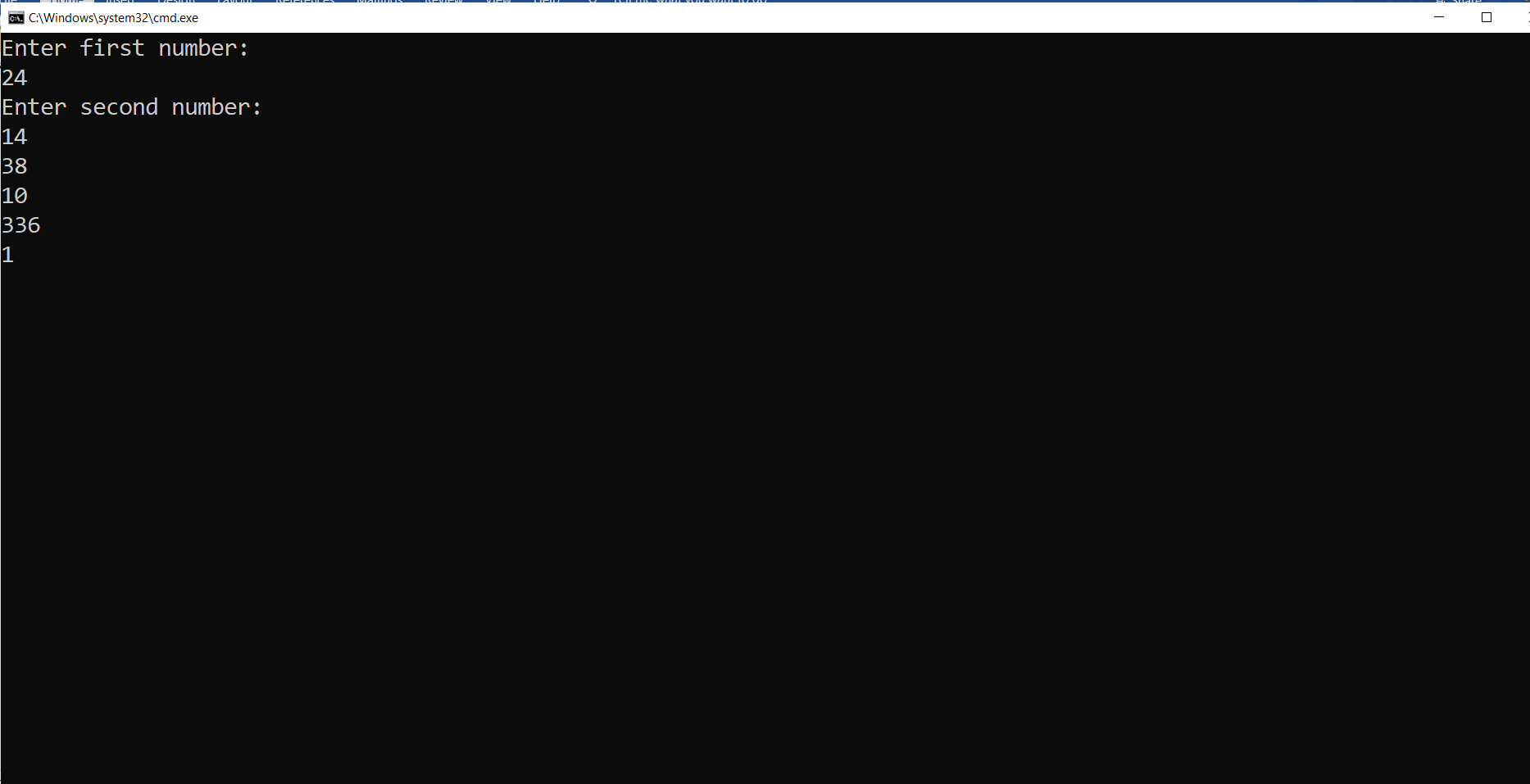
Console.WriteLine(mt.Dividenumbers());

Console.ReadLine();

}

}

Output:



|  |
| --- |
| 3. Create an employee class with below variables  id, name, salary, company  write methods to read data and print data. |

Code:

class Employee

{

public int id;

public string name;

public int salary;

public string company;

/// <summary>

/// Read employee data

/// </summary>

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 = "NatinonsBenefits";

}

/// <summary>

/// Print Employee data

/// </summary>

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();

emp.ReadData();

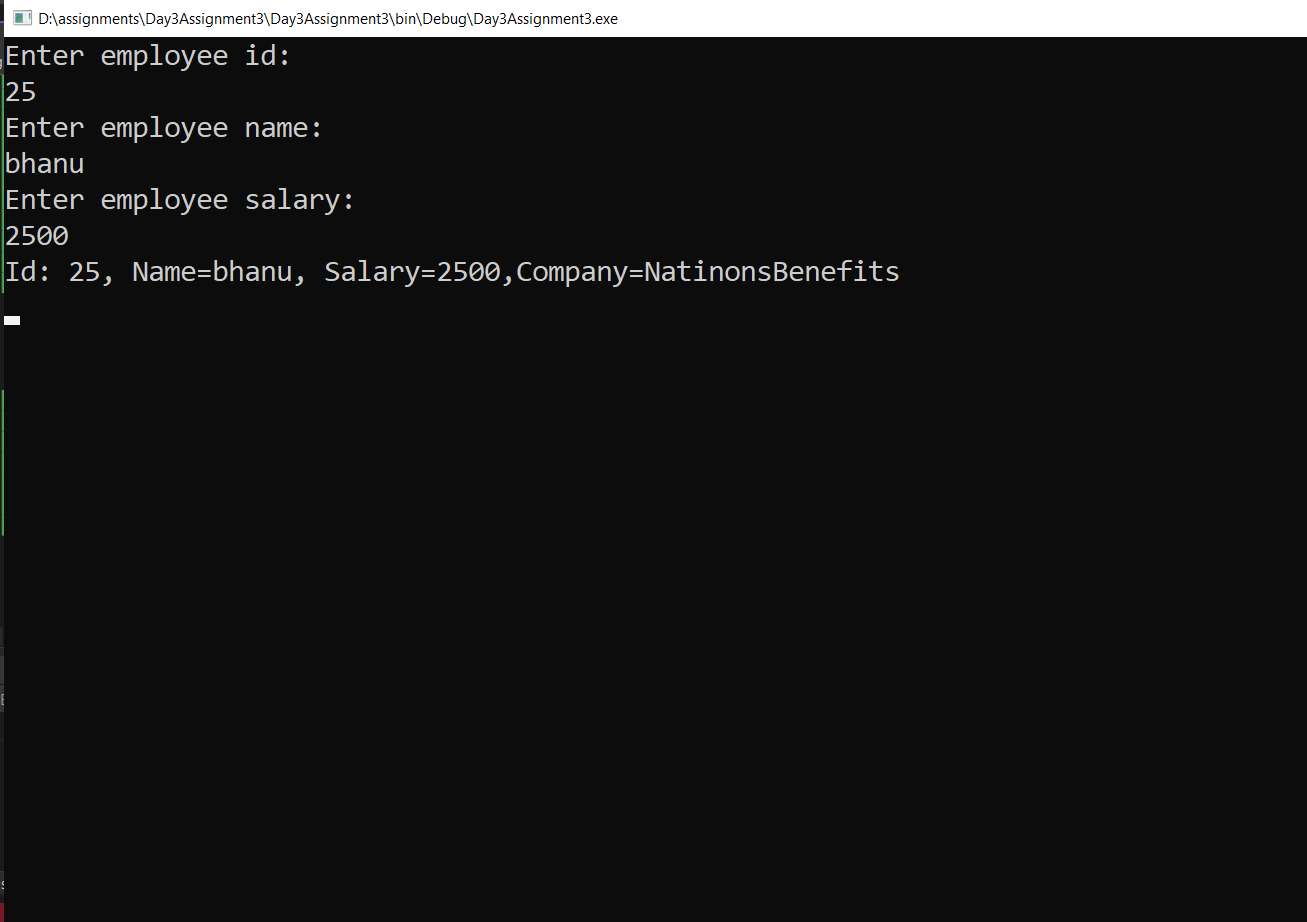
emp.PrintData();

Console.ReadLine();

}

}

Output:



|  |
| --- |
| 4. Research and find the difference between normal variable and static variable. |

DIFFEERENCE BETWEEN NORMAL VARIBALE AND STATIC VARIABLE

|  |  |
| --- | --- |
| NORMAL VARIABLE | STATIC VARIABLE |
| Instance variables are declared in a class, but outside a method, constructor or any block. | Static variables are also known as class variables. It declared with the static keyword in a class, but outside a method, constructor or a block. |
| Instance variables are created when an object is created with the use of the keyword ‘new’ and destroyed when the object is destroyed. | Static variables are created when the program stars and destroyed when the program stops. |
| We can access instance variables through object references. | Static variables can be accessed directly using class name. |
| Instance variables are initialized for 0 times if no instance is created and n times if n instances are created. | A static variable can be initialized for only time. |
| Syntax:  Class Abc  {  int a;  } | Syntax:  Class Abc  {  static int a;  } |
|  |  |

|  |
| --- |
| 5. Write 5 points discussed about constructor |

**Constructor**

* A Constructor is used to initialize class variables while creating an object.
* By default, we will have default constructor which will initialize to default values.
* The moment the programmer create user define constructor the default constructor will be gone.
* If you need a default constructor with the user define constructor, create your own default constructor.
* Constructor name should be same as your class name.
* If your using same variables in constructor variables as that of the class variables, use this. to differentiate with the class variables.

EX: this.id = id

|  |
| --- |
| 6. Create Employee class with two constructors as discussed in the class. |

Code:

internal class Program

{

class Employee

{

private int id;

private string name;

private int salary;

public static string company = "NationsBenefits";

/// <summary>

/// Default Constructor

/// </summary>

public Employee()

{

this.id = 0;

this.name = null;

this.salary = 0;

}

/// <summary>

/// User Define Constructor

/// </summary>

public Employee(int id, string name, int salary)

{

this.id = id;

this.name = name;

this.salary = salary;

}

/// <summary>

/// Get input employee details

/// </summary>

public void ReadData()

{

Console.Write("Enter Emloyee Id: ");

id = Convert.ToInt32(Console.ReadLine());

Console.Write("Enter Emloyee Name: ");

name = Console.ReadLine();

Console.Write("Enter Emloyee Salary: ");

salary = Convert.ToInt32(Console.ReadLine());

}

/// <summary>

/// Print the employee details

/// </summary>

public void Printdata()

{

Console.WriteLine($"Employee Id: {id}, Employee Name: {name}, Employee Salary: {salary}, Company name: {company}.");

}

}

static void Main(string[] args)

{

Employee emp = new Employee();

emp.ReadData();

emp.Printdata();

Employee emp1 = new Employee(22, "Bhanu", 5000);

emp1.Printdata();

Console.ReadLine();

}

}

Output:

