

**NB Healthcare Technologies Pvt Ltd**

**Day 9 Morning Assignment (3 – Feb- 2022)**

**By**

**Vamsi Krishna Mandapati**

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:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Day9Project1
{
    class MathsOperations
    {
        private int input;

        public void ReadInput()
        {
            Console.WriteLine("Enter Number:");
            input = Convert.ToInt32(Console.ReadLine());
        }

        public void Factorial()
        {
            int fact = 1;
            for(int i =1; i <= input; i++)
            {
                fact = fact * i;
            }
            Console.WriteLine(fact);
        }

        public void PrintFactors()
```

```

    {
        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 % 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 numner");
        else
            Console.WriteLine("input is not a prime
number");

        Console.ReadLine();
    }
}

```

```
}  
}
```

Output:

D:\NB HealthCare Training\DotNet Projects\Day 9 Morning Assignments\Day9Project1\Day9Project1\bin\Debug\Day9Project1.exe

```
Enter Number:  
5  
120  
1  
5  
input is prime numner  
_
```

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:

```
using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Text;  
using System.Threading.Tasks;  
  
namespace Day9Project2  
{  
    class MathTask  
    {  
        public int a;  
        public 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 SubtractNumbers()
    {
        return a - b;
    }

    public int MultiplyNumbers()
    {
        return a * b;
    }

    public float DivideNumbers()
    {
        return a / b;
    }
}

internal class Program
{
    static void Main(string[] args)
    {
        MathTask mt = new MathTask();
        mt.ReadInput();
        Console.WriteLine(mt.AddNumbers());
        Console.WriteLine(mt.SubtractNumbers());
        Console.WriteLine(mt.MultiplyNumbers());
        Console.WriteLine(mt.DivideNumbers());

        Console.ReadLine();
    }
}

```

Output:

D:\NB HealthCare Training\DotNet Projects\Day 9 Morning Assignments\Day9Project2\Day9Project2\bin\Debug\Day9Project2.exe

```
Enter first number
7
Enter second number
2
9
5
14
3
```

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

Code:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Day9Project3
{
    class Employee
    {
        public int id;
        public string name;
        public int salary;
        public static string company = "NationsBenefits";

        public Employee(int eid, string ename, int esalary)
        {
            id = eid;
            name = ename;
            salary = esalary;
        }

        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(01, "Vamsi", 9000);  
        emp.PrintData();  
  
        Console.ReadLine();  
    }  
}
```

Output:

D:\NB HealthCare Training\DotNet Projects\Day 9 Morning Assignments\Day9Project3\Day9Project3\bin\Debug\Day9Project3.exe

id = 1, name = Vamsi, salary = 9000, company = NationsBenefits

#### 4. Research and find the difference between normal variable and static variable.

Normal variable	Static variable
1. Normal Variable is different for different objects	1. Static Variable is same for all objects
2.No Keyword is required for defining/initialising the normal variable.	2. static keyword is required for defining/initialising the static variable.
3. Normal variables stores in each objects, so it consume more memory.	3. Static variable stores in class at once , so it saves memory.
<p>4. If a variable is declared inside the class and outside the methods and outside the constructors without static keyword, then it is called as instance variable or non static variable.</p> <p>The memory for the instance variable will be allocated multiple times i.e. one time for every object that is created. If we do not want to allocate the memory for a variable multiple times, then declare the variable with static keyword.</p>	<p>4. If a variable is declared inside the class, outside the methods and outside the constructors with static keyword, then it is called as static variable.</p> <p><b>Syntax for static variable:</b> static datatype variableName;</p> <ul style="list-style-type: none"> <li>• The memory for the static variable will be allocated during the class loading time.</li> <li>• The memory for static variable will be allocated in method area.</li> <li>• The memory for static variable will be allocated one time for entire class.</li> <li>• All the objects will share the same copy of the static variable.</li> <li>• static keyword can be applied to both variables and methods.</li> <li>• If a static variable is declared and not initialized, then it will be initialized automatically with</li> </ul>

	default value. If we don't want static variable to contain default value, then we can initialize the static variable with our own value at the time of declaration.
5. The instance members(instance variables and instance methods) can be accessed only by using reference(object). The class can contain both instance members(instance variables, instance methods) and static members(static variables, static methods).	5. static members(static variables and static methods) can be accessed either by using a class name or by using a reference(object). It is recommended to access the static members by using class name, because we cannot guarantee the existence of the object.

### 5. Write 5 points discussed about constructor

- 1.A constructor is used to initialize class variables while creating an object.
- 2.By Default, C# will have **default constructor** which initialize to default values.(even if we didn't see it , which will initialize to default values).  
Employee emp = new Employee();
- 3.The moment when we write our own constructor, default constructor will be gone.(still if we need default constructor we can create default constructor again exclusively,with default values).
- 4.A Constructor name should be same as that class name.
- 5.We can create any no.of constructors for a class.
6. if constructor variable names are same as class variable names then we have to use this(this.) keyword at(before) class variables.



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

Code:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

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;
            this.salary = 0;
        }

        public Employee(int eid, string ename, int esalary)
        {
            this.id = eid;
            this.name = ename;
            this.salary = esalary;
        }

        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(01, "Vamsi", 9000);  
        emp1.PrintData();
```

```
        Employee emp2 = new Employee();  
        emp2.PrintData();
```

```
        Console.ReadLine();
```

```
    }
```

```
}
```

```
}
```

Output:

System;

D:\NB HealthCare Training\DotNet Projects\Day 9 Morning Assignments\Day9Project4\Day9Project4\bin\Debug\Day9Project4.exe

```
id = 1, name = Vamsi, salary = 9000, company = NationsBenefits
```

```
id = 0, name = , salary = 0, company = NationsBenefits
```