Day 7 Assignment
By
Triveni Anumolu
01-02-2022

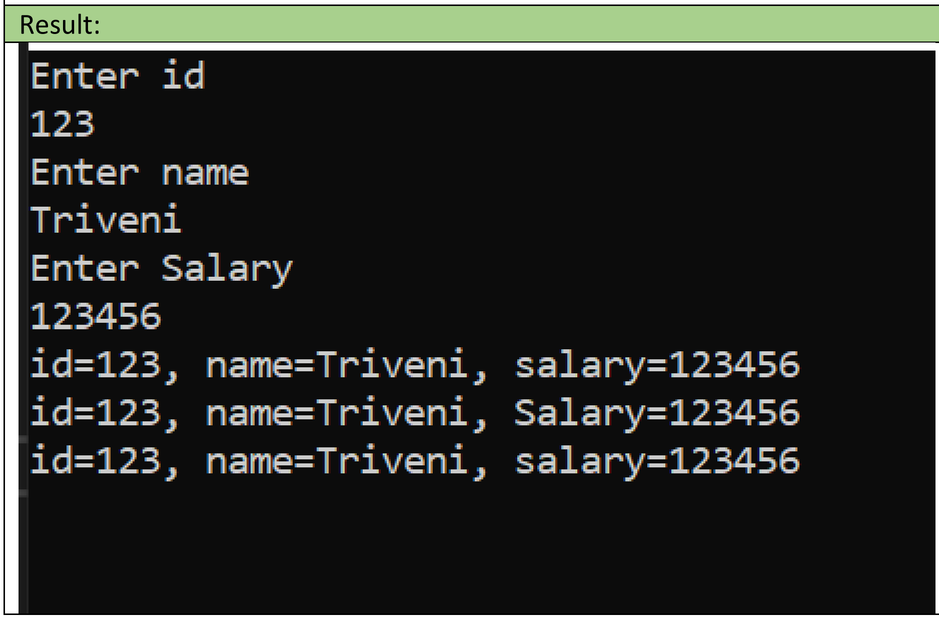
1. Create Employee class with three variables and two methods ReadEmployee and PrintEmployee and create an object and call methods.

Code:

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

namespace Day7Project1
{
    class Employee
    {
        /************************************************************
          Author: Triveni Anumolu
           Purpose: Creating class and creating object to call the method
         ************************************************************/

        public int id;
        public string name;
        public int salary;

        public void ReadEmployee()
        {
            Console.WriteLine("Enter id");
            id = Convert.ToInt32(Console.ReadLine());

            Console.WriteLine("Enter name");
            name = Console.ReadLine();
           Console.WriteLine("Enter Salary");
            salary = Convert.ToInt32(Console.ReadLine());
        }
        public void PrintEmployee()
        {
            //printing by using string interpolation
            Console.WriteLine($"id={id}, name={name}, salary={salary}");
            Console.WriteLine("id={0}, name={1}, Salary={2}", id, name, salary);
            Console.WriteLine("id=" + id + ",name=" + name + ",salary=" + salary);
```

```
      }
   }

   internal class Program
   {
      static void Main(string[] Args)
      {
         Employee emp = new Employee();
         emp.ReadEmployee();
         emp.PrintEmployee();
        Console.ReadLine();
      }
}
```

Result:



```
Enter id
123
Enter name
Triveni
Enter Salary
123456
id=123, name=Triveni, salary=123456
id=123, name=Triveni, Salary=123456
id=123, name=Triveni, salary=123456
```

2.Write the three definitions of class and four points about object discussed in the class.

Definitions of class:

1.A class is a group of variables and methods.
2.A class is like a design or blueprint to create objects.
3.A class consists of state and behaviour. State talks about variable and behaviour talks about methods.

Points about Object:

1.An object is an instance of the class.
2.We can create any number of objects.
3.Objects occupy memory.
4.objects are reference type.

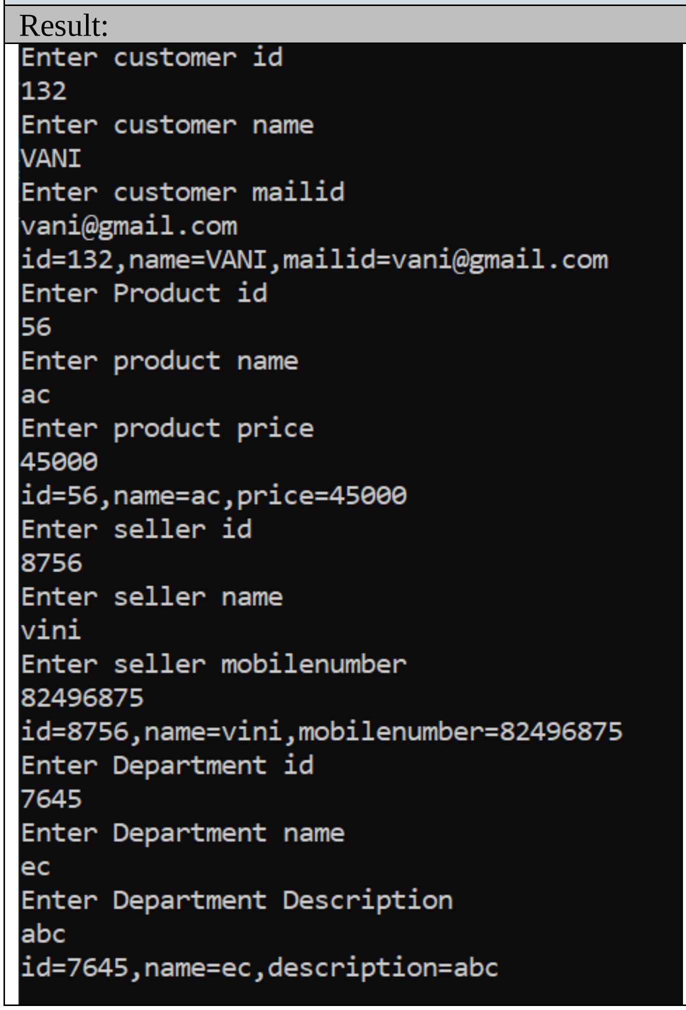
4.Create below classes:

- Customer
- Product
- Seller
- department

Code:

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

namespace ConsoleApp1
{
    /******************************************************************
     Author: Triveni Anumolu
     Purpose: Creating classes and objects
    *******************************************************************/
    class Customer
    {

        public int id;
        public string name;
        public string mailid;

        public void ReadCustomer()
        {
            Console.WriteLine("Enter customer id");
            id = Convert.ToInt32(Console.ReadLine());

            Console.WriteLine("Enter customer name");
            name = Console.ReadLine();

            Console.WriteLine("Enter customer mailid");
            mailid = (Console.ReadLine());
        }
        public void PrintCustomer()
        {
            Console.WriteLine($"id={id},name={name},mailid={mailid}");
        }
    }
    class Product
    {
        public int id;
        public string name;
        public int price;

        public void ReadProduct()
        {
            Console.WriteLine("Enter Product id");
            id = Convert.ToInt32(Console.ReadLine());

            Console.WriteLine("Enter product name");
            name = Console.ReadLine();

            Console.WriteLine("Enter product price");
            price = Convert.ToInt32(Console.ReadLine());
        }
        public void PrintProduct()
        {
            Console.WriteLine($"id={id},name={name},price={price}");
        }
    }
    class Seller
    {

        public int id;
        public string name;
        public string mobilenumber;

        public void ReadSeller()
        {
            Console.WriteLine("Enter seller id");
            id = Convert.ToInt32(Console.ReadLine());

            Console.WriteLine("Enter seller name");
            name = Console.ReadLine();

            Console.WriteLine("Enter seller mobilenumber");
            mobilenumber = Console.ReadLine();
        }
        public void PrintSeller()
        {
            Console.WriteLine($"id={id},name={name},mobilenumber={mobilenumber}");
        }
    }
    class Department
    {

        public int id;
        public string name;
        public string description;

        public void ReadDepartment()
        {
            Console.WriteLine("Enter Department id");
            id = Convert.ToInt32(Console.ReadLine());

            Console.WriteLine("Enter Department name");
            name = Console.ReadLine();

            Console.WriteLine("Enter Department Description");
            description = Console.ReadLine();
        }
        public void PrintDepartment()
        {
            Console.WriteLine($"id={id},name={name},description={description}");
        }


        internal class Program
        {
            static void Main(string[] Args)
            {

                Customer c1 = new Customer();
                c1.ReadCustomer();
                c1.PrintCustomer();
                Product p1 = new Product();
                p1.ReadProduct();
                p1.PrintProduct();
                Seller s1 = new Seller();
                s1.ReadSeller();
                s1.PrintSeller();
                Department d1 = new Department();
                d1.ReadDepartment();
                d1.PrintDepartment();

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

Result:



```
Enter customer id
132
Enter customer name
VANI
Enter customer mailid
vani@gmail.com
id=132,name=VANI,mailid=vani@gmail.com
Enter Product id
56
Enter product name
ac
Enter product price
45000
id=56,name=ac,price=45000
Enter seller id
8756
Enter seller name
vini
Enter seller mobilenumber
82496875
id=8756,name=vini,mobilenumber=82496875
Enter Department id
7645
Enter Department name
ec
Enter Department Description
abc
id=7645,name=ec,description=abc
```

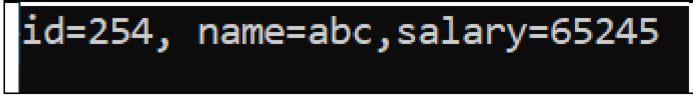
5. Create Employee class with 3 public variables. Create Employee object and initialize with values while creating object and print the values.

Code:

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

```
namespace Day7proj5
{

/************************************************************************************
    Author: Triveni Anumolu
    Purpose:Creating class and initializing variables while creating object

************************************************************************************/
   class Employee
   {
      public int id;
      public string name;
      public int salary;

   }
   class Program
   {
      static void Main(string[] args)
      {
         Employee emp = new Employee() { id = 254, name = "abc", salary = 65245 };
         Console.WriteLine($"id={emp.id}, name={emp.name},salary={emp.salary}");
         Console.ReadLine();

      }
   }
}
```

Result:



```
id=254, name=abc,salary=65245
```

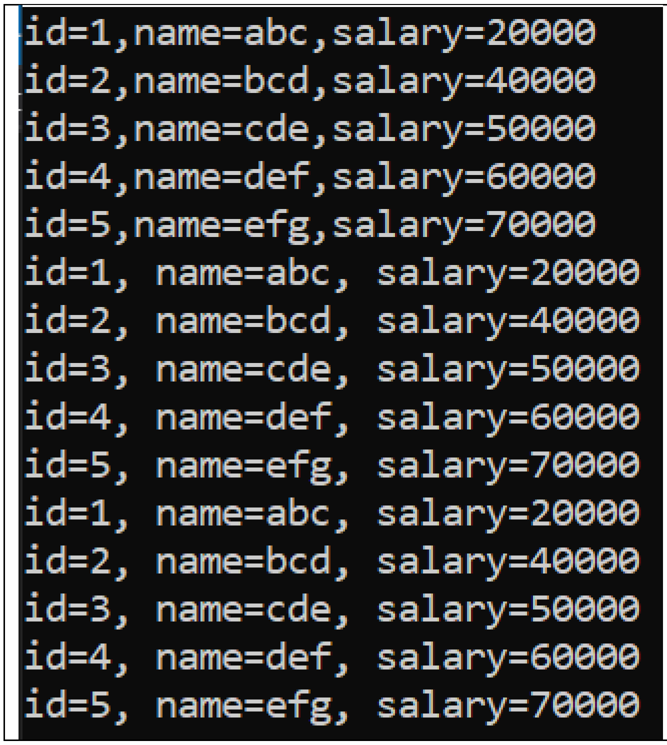
6.Create Employee class as shown below: class Employee { public int id; public string name; public int salary; } now create employees array object and initialize with 5 employees write code using a. for loop b. foreach loop c. lambda expression.

Code:

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

namespace ConsoleApp2
{
```

```
/**************************************************************************
     Author: Triveni Anumolu
    Purpose:Creating employee array object and initializing using loops

**************************************************************************/
    class Employee
    {
        public int id;
        public string name;
        public int salary;
    }
    class Program
    {
        static void Main(string[] args)
        {
            Employee[] emp = new Employee[]
            {
                new Employee(){id=1, name="abc",salary=20000 },
                new Employee(){id=2, name="bcd",salary=40000 },
                new Employee(){id=3, name="cde",salary=50000 },
                new Employee(){id=4, name="def",salary=60000 },
                new Employee(){id=5, name="efg",salary=70000 },
            };
            //for loop
            for(int i=0;i<emp.Length;i++)
            {

Console.WriteLine($"id={emp[i].id},name={emp[i].name},salary={emp[i].salary}");
            }
            //foreach loop
            foreach(var e in emp)
            {
                Console.WriteLine($"id={e.id}, name={e.name}, salary={e.salary}");
            }
            //lambda expression
            emp.ToList().ForEach(e =>Console.WriteLine($"id={e.id}, name={e.name},
salary={e.salary}"));

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

Result:



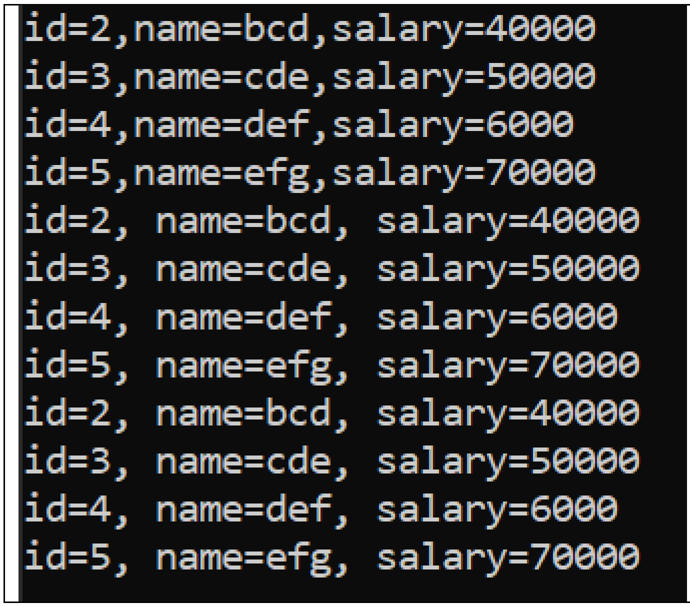
```
id=1,name=abc,salary=20000
id=2,name=bcd,salary=40000
id=3,name=cde,salary=50000
id=4,name=def,salary=60000
id=5,name=efg,salary=70000
id=1, name=abc, salary=20000
id=2, name=bcd, salary=40000
id=3, name=cde, salary=50000
id=4, name=def, salary=60000
id=5, name=efg, salary=70000
id=1, name=abc, salary=20000
id=2, name=bcd, salary=40000
id=3, name=cde, salary=50000
id=4, name=def, salary=60000
id=5, name=efg, salary=70000
```

7. For the above project,   write code to print employees who is getting salary >=5000 using for loop   foreach loop   lambda expression

Code:

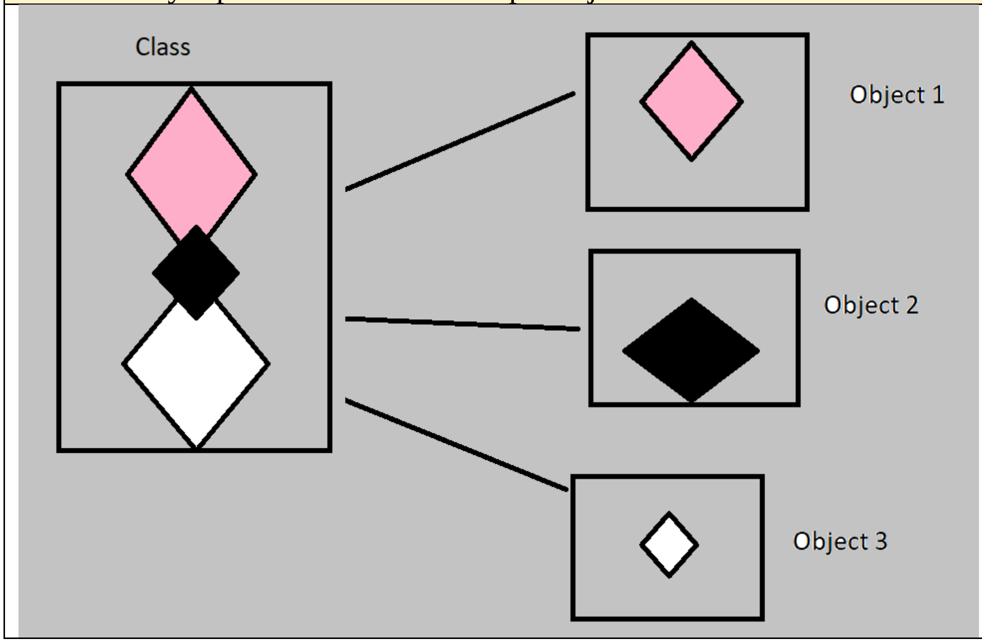
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace ConsoleApp2
{

/****************************************************************************
     Author: Triveni Anumolu
    Purpose:Creating employee array object and initializing using loops
    ***************************************************************************/
    class Employee
    {
        public int id;
        public string name;
        public int salary;
    }
    class Program
    {
        static void Main(string[] args)
        {
            Employee[] emp = new Employee[]
            {
                new Employee(){id=1, name="abc",salary=2000 },
                new Employee(){id=2, name="bcd",salary=40000 },
                new Employee(){id=3, name="cde",salary=50000 },
                new Employee(){id=4, name="def",salary=6000 },
                new Employee(){id=5, name="efg",salary=70000 },
            };
            //for loop
            for(int i=0;i<emp.Length;i++)
            {
                if(emp[i].salary>5000)

Console.WriteLine($"id={emp[i].id},name={emp[i].name},salary={emp[i].salary}");
            }
            //foreach loop
            foreach(var e in emp)
            {
                if(e.salary>5000)
                Console.WriteLine($"id={e.id}, name={e.name}, salary={e.salary}");
            }
            //lambda expression
            emp.ToList().Where(e=>e.salary>=5000).ToList().ForEach(e
=>Console.WriteLine($"id={e.id}, name={e.name}, salary={e.salary}"));

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

Result:



```
id=2,name=bcd,salary=40000
id=3,name=cde,salary=50000
id=4,name=def,salary=6000
id=5,name=efg,salary=70000
id=2, name=bcd, salary=40000
id=3, name=cde, salary=50000
id=4, name=def, salary=6000
id=5, name=efg, salary=70000
id=2, name=bcd, salary=40000
id=3, name=cde, salary=50000
id=4, name=def, salary=6000
id=5, name=efg, salary=70000
```

3. Pictorially represent class and multiple objects


Class
Object 1
Object 2
Object 3