DAY 8 MORNING ASSIGNMENT

(BY G V S S SRI LASYA)

1)Declare and initialize a list with 8 values.write for loop, foreach loop, lambda, linq query to print even numbers

CODE

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

namespace D8Project1
{

    /*********************************************************************
     Author : G V S S SRI LASYA
     Purpose : Declare and initialize a list with 8 values.
     write for loop, foreach loop, lambda, linq query to print even numbers

    *********************************************************************/


    internal class Program
    {
        static void Main(string[] args)
        {
            List<int> numbers = new List<int>() { 10, 11, 12, 13, 14, 15, 16, 17 };

            //printing even numbers using for loop
            for(int i = 0; i < numbers.Count; i++)
            {
                if (numbers[i] % 2 == 0)
                    Console.Write($"\n{numbers[i]}");
            }


            Console.WriteLine("\n");
            //printing even numbers using foreach
            foreach(var number in numbers)
            {
                if(number%2 == 0)
                    Console.Write($"\n{number}");
            }

            Console.WriteLine("\n");
            //printing even numbers using lambda expression
            numbers.Where(number => number % 2 == 0).ToList().ForEach(number =>
Console.Write($"\n{number}"));

            Console.WriteLine("\n");
            //printing even numbers using linq
            var result = from number in numbers
                         where number % 2 == 0
                         select number;
            result.ToList().ForEach(number => Console.Write($"\n{number}"));


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

OUTPUT

```
10
12
14
16


10
12
14
16


10
12
14
16


10
12
14
16
```

2) Create a class Employee with three variables as discussed in the class and create a list of Employees.Variables should be:
public int id;
public string name;
public int salary;
Print the output using:
a)for loop
b)foreach loop
c)lambda expression
d)linq query

CODE

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

namespace Day8Project2
{
    /*********************************************************************
    Author : G V S S SRI LASYA
    Purpose :  Create a class Employee with three variables and
    create a list of Employees.write  for loop,foreach loop,lambda expression
    linq query
    *********************************************************************/
    class Employee
    {
        public int id;
        public string name;
        public int salary;
    }
    internal class Program
    {
        static void Main(string[] args)
        {
            List<Employee> employees = new List<Employee>()
            {
            new Employee() { id = 1, name = "Karthika", salary = 200000 },
            new Employee() {id = 2, name = "Sravani", salary = 200000 },
            new Employee() {id = 3, name = "Pranavi", salary = 150000 },
            new Employee() {id = 4, name = "Tara", salary = 100000 },
            new Employee() {id = 3, name = "Manaswi", salary = 100000 }
            };



            //printing employees with salary>100000 using for loop
            for (int i = 0; i < employees.Count; i++)
            {
                if (employees[i].salary > 100000)
                    Console.Write($"\n{employees[i].name} with salary of
{employees[i].salary}");
            }


            Console.WriteLine("\n\n");
            //printing employees with salary>100000 using foreach
            foreach (var employee in employees)
            {
                if (employee.salary > 100000)
                    Console.Write($"\n{employee.name} with salary of {employee.salary}");
            }


            Console.WriteLine("\n\n");
            //printing employees with salary>100000 using lambda expression
            employees.Where(employee => employee.salary > 100000).ToList().ForEach(employee
=> Console.Write($"\n{employee.name} with salary of {employee.salary}"));
```

```
            Console.WriteLine("\n\n");
            //printing employees with salary>100000 using LINQ
            var result = from employee in employees
                         where employee.salary > 100000
                         select employee;
            result.ToList().ForEach(employee => Console.Write($"\n{employee.name} with salary
of {employee.salary}"));


            Console.ReadLine();

        }
    } }
```

OUTPUT

```
Karthika with salary of 200000
Sravani with salary of 200000
Pranavi with salary of 150000



Karthika with salary of 200000
Sravani with salary of 200000
Pranavi with salary of 150000



Karthika with salary of 200000
Sravani with salary of 200000
Pranavi with salary of 150000



Karthika with salary of 200000
Sravani with salary of 200000
Pranavi with salary of 150000
```

3) Create a class Product and add variables : id, name, price, brand
Print product (name and brand) whose price is more than 500 using:
a)for
b)foreach loop
c)lambda
d)linq query

CODE

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


/*******************************************************
Author : G V S S SRI LASYA
Purpose : Create a class Product and add variables
id, name, price, brand and print product (name and brand)
whose price is more than 500 using
for
foreach loop
lambda
linq query
*******************************************************/


namespace Day8Project3
{
    class Product
    {

        public int id;
        public string name;
        public int price;
        public string brand;

    }

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

            //creating array of objects for Product class
            Product[] products = new Product[]
            {
                new Product(){id = 1, name = "laptop", price = 40000, brand = "asus" },
                new Product() { id = 2, name = "bag", price = 3000, brand = "wildcraft"},
                new Product() { id = 3, name = "fan", price = 1000, brand = "bajaj"},
                new Product() { id = 4, name = "shoes", price = 3000, brand = "adidas" },
                new Product() { id = 5, name = "phone", price = 10000, brand = "samsung"}
            };


            //printing name and brand of products whose price is >5000 using for loop
            for (int i = 0; i < products.Length; i++)
            {
                if (products[i].price > 5000)
                    Console.Write($"\nproduct : {products[i].name}\t\t\tbrand :
{products[i].brand}");

            }


            Console.WriteLine("\n\n");
            //printing name and brand of products whose price is >5000 using foreach
            foreach (var product in products)
            {
                if (product.price > 5000)
                    Console.Write($"\nproduct : {product.name}\t\t\tbrand :
{product.brand}");
```

```
            }


            Console.WriteLine("\n\n");
            //printing name and brand of products whose price is >5000 using lambda
expression
            products.ToList().Where(product => product.price > 5000).ToList().ForEach(product
=> Console.Write($"\nproduct : {product.name}\t\t\tbrand : {product.brand}"));


            Console.WriteLine("\n\n");
            //printing name and brand of products whose price is >5000 using LINQ
            var result = from product in products
                         where product.price > 5000
                         select product;

            result.ToList().ForEach(product => Console.Write($"\nproduct :
{product.name}\t\t\tbrand : {product.brand}"));

            Console.ReadLine();

        }

    }

}
```

OUTPUT

```
product : laptop                        brand : asus
product : phone                 brand : samsung



product : laptop                        brand : asus
product : phone                 brand : samsung



product : laptop                        brand : asus
product : phone                 brand : samsung



product : laptop                        brand : asus
product : phone                 brand : samsung
```

4) Create a Department class and add variables : id,name,empcount
Write code to print id,name of departments whose empcount is greater than 50 using
a)for
b)foreach
c)lambda
d)linq query

CODE

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

/*************************************************************************
Author : G V S S SRI LASYA
Purpose : Create a Department class and add variables id,name,empcount
write code to print id,name of departments whose empcount is greater than 50
using
for
foreach
lambda
linq query
*************************************************************************/



namespace Day8Project4
{
    class Department
    {

        public int id;
        public string name;
        public int empCount;

    }

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

            //creating a Department list,initialising it
            List<Department> departments = new List<Department>()
            {
                new Department() { id = 1, name = "Department1", empCount = 20 },
                new Department() { id = 2, name = "Department2", empCount = 30 },
                new Department() { id = 3, name = "Department3", empCount = 70},
                new Department() { id = 4, name = "Department4", empCount = 60},
                new Department() { id = 5, name = "Department5", empCount = 100}
            };


            //printing name and id of departments whose empCount>50 using for loop
            for (int i = 0; i < departments.Count; i++)
            {
                if (departments[i].empCount > 50)
                    Console.Write($"\ndepartment : {departments[i].name}\t\tid :
{departments[i].id}");

            }


            Console.WriteLine("\n\n");
            //printing name and id of departments whose empCount>50 using foreach
            foreach (var department in departments)
            {
                if (department.empCount > 50)
                    Console.Write($"\ndepartment : {department.name}\t\tid:
{department.id}");
```

```
            }


            Console.WriteLine("\n\n");
            //printing name and id of departments whose empCount>50 using lambda expression
            departments.ToList().Where(department => department.empCount >
50).ToList().ForEach(department => Console.Write($"\ndepartment : {department.name}\t\tid :
{department.id}"));


            Console.WriteLine("\n\n");
            //printing name and id of departments whose empCount>50 using LINQ
            var result = from department in departments
                         where department.empCount > 50
                         select department;

            result.ToList().ForEach(department => Console.Write($"\ndepartment :
{department.name}\t\tid : {department.id}"));

            Console.ReadLine();

        }

    }

}
```

OUTPUT

```
department : Department3                id : 3
department : Department4                id : 4
department : Department5                id : 5



department : Department3                id: 3
department : Department4                id: 4
department : Department5                id: 5



department : Department3                id : 3
department : Department4                id : 4
department : Department5                id : 5


department : Department3                id : 3
department : Department4                id : 4
department : Department5                id : 5
```

5) Create your own class and variables and initialize with some values and print using:
a)for
b)foreach
c)lambda
d)linq query

CODE

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

/*********************************************************************
Author : G V S S SRI LASYA
Purpose :  Create your own class and variables and initialize with
some values.Print them using
for
foreach
lambda
linq query
*********************************************************************/


namespace Day8Project5
{
    class PoliceOfficer
    {

        public string name;
        public string designation;
        public int id;

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

                //creating a PoliceOfficer list,initialising it
                List<PoliceOfficer> policeOfficers = new List<PoliceOfficer>()
            {
                new  PoliceOfficer() { name = "Srinivas", designation = "DGP" , id  = 1},
                new  PoliceOfficer() { name = "Sravani", designation = "IG" ,id = 2},
                new  PoliceOfficer() { name = "Chandan", designation = "DIG", id = 3},
                new  PoliceOfficer() { name = "Varsha", designation = "SP", id = 4},
                new  PoliceOfficer(){  name = "Harshini",designation = "SP", id = 5}
            };


                //printing name and designation of officers whose id<=3 using for loop
                for (int i = 0; i < policeOfficers.Count; i++)
                {
                    if (policeOfficers[i].id <= 3)
                        Console.Write($"\nname : {policeOfficers[i].name}\t\tdesignation :
{policeOfficers[i].designation}");

                }


                Console.WriteLine("\n\n");
            //printing name and designation of officers whose id<=3 using foreach
            foreach (var policeOfficer in policeOfficers)
                {
                    if (policeOfficer.id <= 3)
                        Console.Write($"\name : {policeOfficer.name}\t\tdesignation:
{policeOfficer.designation}");
                }


                Console.WriteLine("\n\n");
            //printing name and designation of officers whose id<=3  using lambda expression
```

```
            policeOfficers.ToList().Where(policeOfficer => policeOfficer.id <=
3).ToList().ForEach(policeOfficer => Console.Write($"\nname :
{policeOfficer.name}\t\tdesignation : {policeOfficer.designation}"));


                Console.WriteLine("\n\n");
            //printing name and designation of officers whose id<=3  using LINQ
            var result = from policeOfficer in policeOfficers
                         where policeOfficer.id <= 3
                         select policeOfficer;

                result.ToList().ForEach(policeOfficer => Console.Write($"\nname :
{policeOfficer.name}\t\tdesignation : {policeOfficer.designation}"));

                Console.ReadLine();

            }

        }

    }
```

OUTPUT