DAY 8 : Morning Assignment
By
Vihar D.

Assignment 1

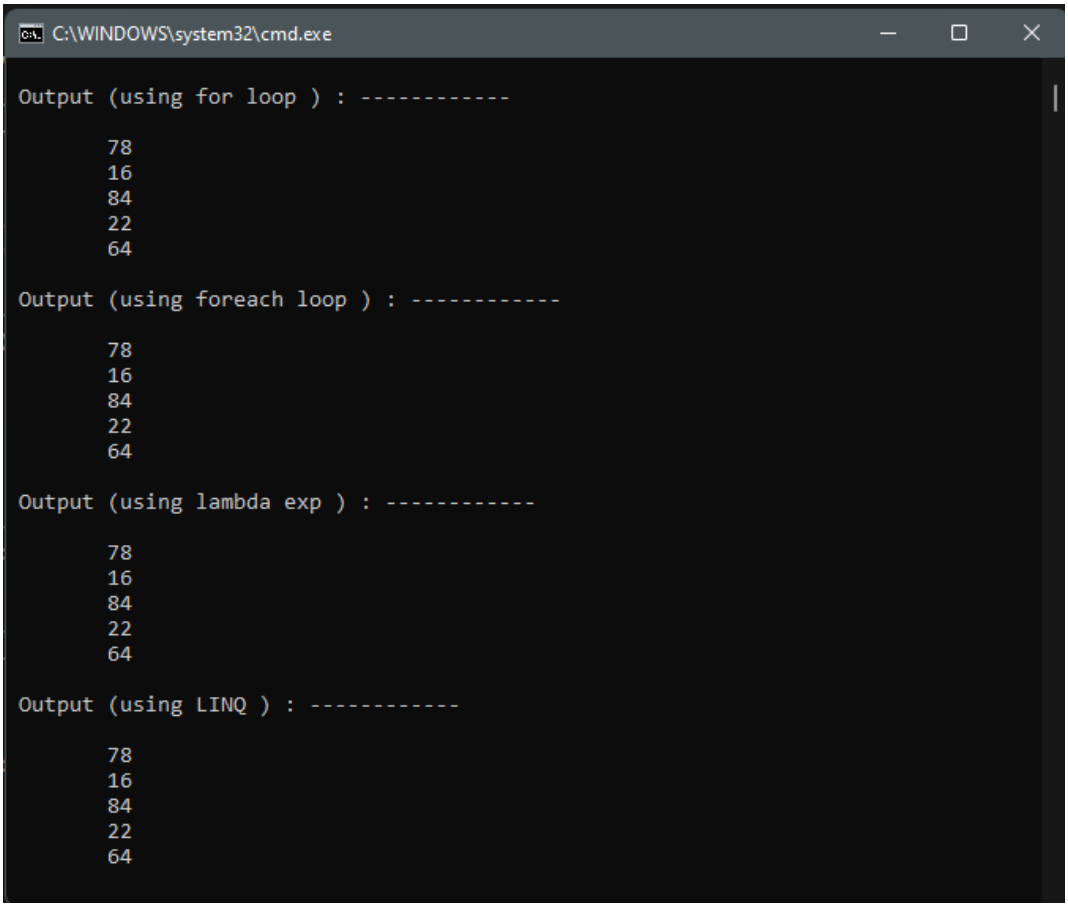
Create a list with 8 values & find even numbers from the list using for, foreach, lambda and LINQ loop types.

Answer :

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

namespace values8_even_4looptypes
{
    internal class Program
    {
        static void Main(string[] args)
        {

            List<int> data = new List<int>() { 65,89,41,78,16,84,39,67,22,64 };

            //using for loop-----------------------------------------------------
            Console.WriteLine("\n Output (using for loop ) : ----------\n");
            for (int i = 0; i < data.Count; i++)
            {
                if (data[i] % 2 == 0)
                    Console.WriteLine("\t{0}", data[i]);
            }

            //using foreach loop-------------------------------------------------
            Console.WriteLine("\n Output (using foreach loop ) : ----------\n");
            foreach (var d in data)
            {
                if (d % 2 == 0)
                    Console.WriteLine("\t{0}", d);
            }
```

```
        //using lambda expression-----------------------------------------------------------
        Console.WriteLine("\n Output (using lambda exp ) : ----------\n");
        data.Where(x => x % 2 == 0).ToList().ForEach(x => Console.WriteLine("\t{0}", x));

        //using LINQ query------------------------------------------------------------------
        Console.WriteLine("\n Output (using LINQ ) : ----------\n");
        var output = from d in data
                     where d % 2 == 0
                     select d;
        output.ToList().ForEach(x => Console.WriteLine("\t{0}", x));
        Console.ReadLine();
    }
  }
}
```

Output :



Assignment 2

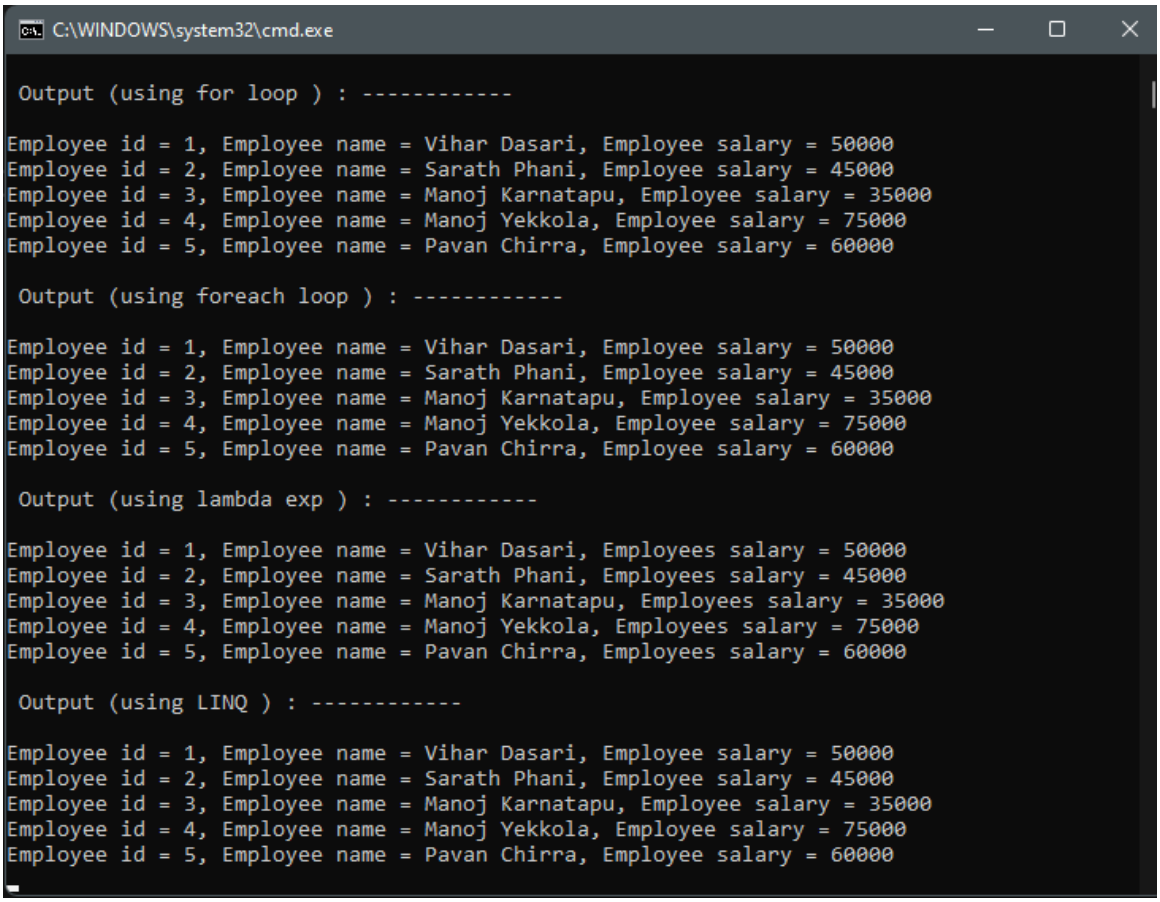
Create a class of list employees and print using for, foreach, lambda and LINQ loop types.

Answer :

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

namespace class_emp_4looptypes
{
   class Employee
   {
      public int emp_id;
      public string emp_name;
      public int emp_salary;
   }
   internal class Program
   {
      static void Main(string[] args)
      {
         List<Employee> emp = new List<Employee>()
         {
            new Employee(){emp_id = 1, emp_name = "Vihar Dasari", emp_salary = 50000},
            new Employee(){emp_id = 2, emp_name = "Sarath Phani", emp_salary = 45000},
            new Employee(){emp_id = 3, emp_name = "Manoj Karnatapu", emp_salary = 35000},
            new Employee(){emp_id = 4, emp_name = "Manoj Yekkola", emp_salary = 75000},
            new Employee(){emp_id = 5, emp_name = "Pavan Chirra", emp_salary = 60000},
         };

         //using for Loop-------------------------------------------------------
         Console.WriteLine("\n Output (using for loop ) : ----------\n");
         for (int i = 0; i < emp.Count; i++)
         {
            Console.WriteLine($"Employee id = {emp[i].emp_id}, " +
                              $"Employee name = {emp[i].emp_name}, " +
                              $"Employee salary = {emp[i].emp_salary}");
         }
```

```
        //using foreach Loop----------------------------------------------------------
        Console.WriteLine("\n Output (using foreach loop ) : ----------\n");
        foreach (var e in emp)
        {
            Console.WriteLine($"Employee id = {e.emp_id}, " +
                              $"Employee name = {e.emp_name}, " +
                              $"Employee salary = {e.emp_salary}");
        }

        //using lambda expression-----------------------------------------------------
        Console.WriteLine("\n Output (using lambda exp ) : ----------\n");
        emp.ForEach(e => Console.WriteLine($"Employee id = {e.emp_id}, " +
                                           $"Employee name = {e.emp_name}, " +
                                           $"Employees salary = {e.emp_salary}"));

        //using LINQ query------------------------------------------------------------
        Console.WriteLine("\n Output (using LINQ ) : ----------\n");
        var output = from e in emp
                     select e;
        output.ToList().ForEach(e => Console.WriteLine($"Employee id = {e.emp_id}, " +
                                                       $"Employee name = {e.emp_name}, " +
                                                       $"Employee salary = {e.emp_salary}"));
        Console.ReadLine();
    }
  }
}
```

Output :


C:\WINDOWS\system32\cmd.exe
Output (using for loop) : ------------
Employee id = 1, Employee name = Vihar Dasari, Employee salary = 50000
Employee id = 2, Employee name = Sarath Phani, Employee salary = 45000
Employee id = 3, Employee name = Manoj Karnatapu, Employee salary = 35000
Employee id = 4, Employee name = Manoj Yekkola, Employee salary = 75000
Employee id = 5, Employee name = Pavan Chirra, Employee salary = 60000
Output (using foreach loop) : ------------
Employee id = 1, Employee name = Vihar Dasari, Employee salary = 50000
Employee id = 2, Employee name = Sarath Phani, Employee salary = 45000
Employee id = 3, Employee name = Manoj Karnatapu, Employee salary = 35000
Employee id = 4, Employee name = Manoj Yekkola, Employee salary = 75000
Employee id = 5, Employee name = Pavan Chirra, Employee salary = 60000
Output (using lambda exp) : ------------
Employee id = 1, Employee name = Vihar Dasari, Employees salary = 50000
Employee id = 2, Employee name = Sarath Phani, Employees salary = 45000
Employee id = 3, Employee name = Manoj Karnatapu, Employees salary = 35000
Employee id = 4, Employee name = Manoj Yekkola, Employees salary = 75000
Employee id = 5, Employee name = Pavan Chirra, Employees salary = 60000
Output (using LINQ) : ------------
Employee id = 1, Employee name = Vihar Dasari, Employee salary = 50000
Employee id = 2, Employee name = Sarath Phani, Employee salary = 45000
Employee id = 3, Employee name = Manoj Karnatapu, Employee salary = 35000
Employee id = 4, Employee name = Manoj Yekkola, Employee salary = 75000
Employee id = 5, Employee name = Pavan Chirra, Employee salary = 60000

Assignment 3

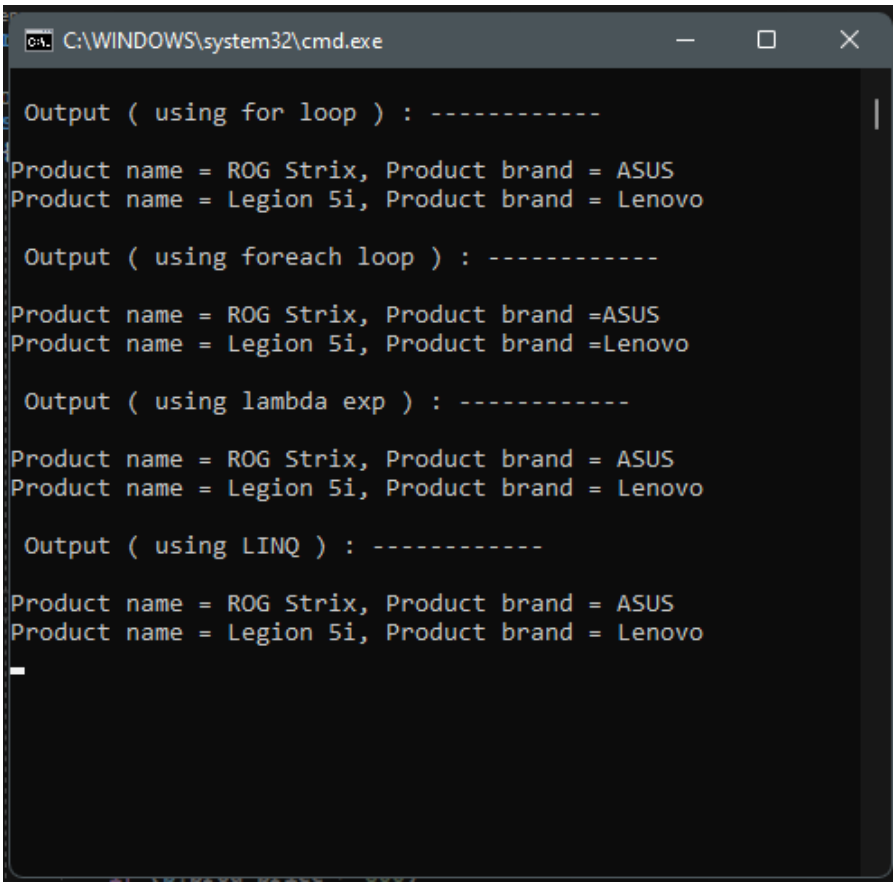
Create a class of list Product and add variables, then print Product (name and brand) whose price is > 500 using for, foreach, lambda and LINQ loop types.

Answer :

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

namespace class_prod_4looptypes
{
   class Product
   {
      public int prod_id;
      public string prod_name;
      public int prod_price;
      public string prod_brand;
   }
   internal class Program
   {
      static void Main(string[] args)
      {
         List<Product> prod = new List<Product>()
          {
          new Product() { prod_id = 1, prod_name = "ROG Strix", prod_price = 820, prod_brand =
"ASUS"},
          new Product() { prod_id = 2, prod_name = "Legion 5i", prod_price = 630, prod_brand =
"Lenovo"},
          new Product() { prod_id = 3, prod_name = "Pavilion", prod_price= 250, prod_brand ="HP"},
          new Product() { prod_id = 4, prod_name = "Nitro Predator", prod_price = 350, prod_brand =
"Acer"}
          };

         //using for
Loop----------------------------------------------------------------------------------------------------------
         Console.WriteLine("\n Output ( using for loop ) : ----------\n");
          for (int i = 0; i < prod.Count; i++)
         {
            if (prod[i].prod_price > 500)
```

```
            {
                Console.WriteLine($"Product name = {prod[i].prod_name}, " +
                                  $"Product brand = {prod[i].prod_brand}");
            }
        }

        //using foreach
Loop----------------------------------------------------------------------------------------------------------------------------
        Console.WriteLine("\n Output ( using foreach loop ) : ----------\n");
        foreach (var p in prod)
        {
            if (p.prod_price > 500)
                Console.WriteLine($"Product name = {p.prod_name}, " +
                                  $"Product brand ={p.prod_brand}");
        }

        //using lambda
expression----------------------------------------------------------------------------------------------------------------------
        Console.WriteLine("\n Output ( using lambda exp ) : ----------\n");
        prod.Where(p => p.prod_price > 500).ToList().ForEach(p => Console.WriteLine($"Product
name = {p.prod_name}, " +
                                                                $"Product brand = {p.prod_brand}"));

        //using LINQ
query---------------------------------------------------------------------------------------------------------------------------
        Console.WriteLine("\n Output ( using LINQ ) : ----------\n");
        var output = from p in prod
                     where p.prod_price > 500
                     select p;
        output.ToList().ForEach(p => Console.WriteLine($"Product name = {p.prod_name}, " +
                                            $"Product brand = {p.prod_brand}"));
        Console.ReadLine();
    }
  }
}
```

Output :


C:\WINDOWS\system32\cmd.exe
Output (using for loop) : ------------
Product name = ROG Strix, Product brand = ASUS
Product name = Legion 5i, Product brand = Lenovo
Output (using foreach loop) : ------------
Product name = ROG Strix, Product brand =ASUS
Product name = Legion 5i, Product brand =Lenovo
Output (using lambda exp) : ------------
Product name = ROG Strix, Product brand = ASUS
Product name = Legion 5i, Product brand = Lenovo
Output (using LINQ) : ------------
Product name = ROG Strix, Product brand = ASUS
Product name = Legion 5i, Product brand = Lenovo

Assignment 4

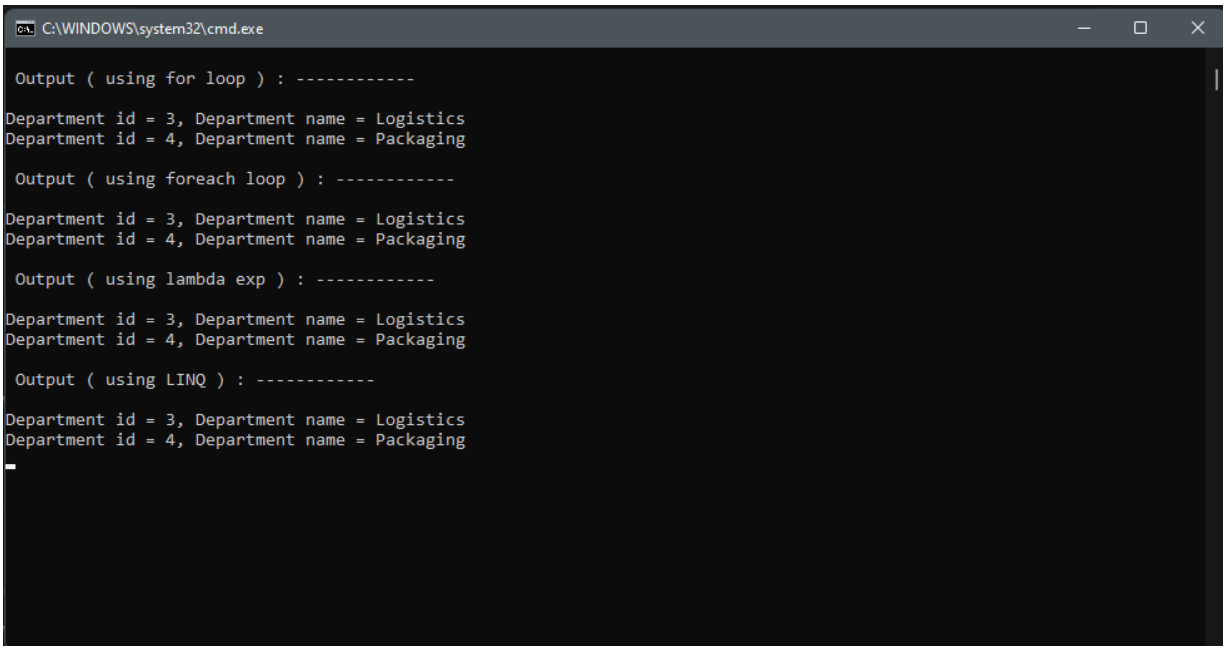
Create a Department class and add variables, then print id & name of departments whose employee count > 50 using for, foreach, lambda and LINQ loop types.

Answer :

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

namespace dept_4looptypes
{
    class Department
    {
        public int dept_id;
        public string dept_name;
        public int dept_empCount;
    }

    internal class Program
    {
        static void Main(string[] args)
        {
            List<Department> dept = new List<Department>()
            {
             new Department() {dept_id = 1, dept_name = "Manager", dept_empCount = 10},
             new Department() {dept_id = 2, dept_name = "Administration", dept_empCount = 50},
             new Department() {dept_id = 3, dept_name = "Logistics", dept_empCount = 200},
             new Department() {dept_id = 4, dept_name = "Packaging", dept_empCount = 350}
            };

            //using for
Loop-----------------------------------------------------------------------------------------------------------------
            Console.WriteLine("\n Output ( using for loop ) : ----------\n");
            for (int i = 0; i < dept.Count; i++)
            {
                if (dept[i].dept_empCount > 50)
                {
                    Console.WriteLine($"Department id = {dept[i].dept_id}, " +
                                      $"Department name = {dept[i].dept_name}");
```

```
            }
        }

        //using foreach
Loop----------------------------------------------------------------------------------------------------------------
        Console.WriteLine("\n Output ( using foreach loop ) : ----------\n");
        foreach (var d in dept)
        {
            if (d.dept_empCount > 50)
            {
                Console.WriteLine($"Department id = {d.dept_id}, " +
                                  $"Department name = {d.dept_name}");
            }
        }

        //using lambda
expression----------------------------------------------------------------------------------------------------------
        Console.WriteLine("\n Output ( using lambda exp ) : ----------\n");
        dept.Where(d => d.dept_empCount > 50).ToList().ForEach(d =>
Console.WriteLine($"Department id = {d.dept_id}, " +
                                                        $"Department name = {d.dept_name}"));

        //using LINQ
query---------------------------------------------------------------------------------------------------------------
        Console.WriteLine("\n Output ( using LINQ ) : ----------\n");
        var output = from d in dept
                     where d.dept_empCount > 50
                     select d;
        output.ToList().ForEach(d => Console.WriteLine($"Department id = {d.dept_id}, " +
                                            $"Department name = {d.dept_name}"));
        Console.ReadLine();
    }
  }
}
```

Output :


C:\WINDOWS\system32\cmd.exe
Output (using for loop) : ------------
Department id = 3, Department name = Logistics
Department id = 4, Department name = Packaging
Output (using foreach loop) : ------------
Department id = 3, Department name = Logistics
Department id = 4, Department name = Packaging
Output (using lambda exp) : ------------
Department id = 3, Department name = Logistics
Department id = 4, Department name = Packaging
Output (using LINQ) : ------------
Department id = 3, Department name = Logistics
Department id = 4, Department name = Packaging

Assignment 5

Pictorially represent class and multiple objects

Answer :

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

namespace random_class_4looptypes
{
    class bikes
    {
        public string bike_model;
        public string bike_brand;
        public int bike_make_year;
        public int bike_powercc;
    }
    internal class Program
    {
        static void Main(string[] args)
        {
            List<bikes> bike = new List<bikes>()
            {
                new bikes(){ bike_model = "Svartpilen 401", bike_brand = "Husqvarna", bike_make_year
= 2015, bike_powercc = 375},
                new bikes(){ bike_model = "Duke 250", bike_brand = "KTM", bike_make_year = 2012,
bike_powercc = 250},
                new bikes(){ bike_model = "ThunderBird 350x", bike_brand = "Royal Enfield",
bike_make_year = 2018, bike_powercc = 350},
                new bikes(){ bike_model = "GS750r", bike_brand = "BMW", bike_make_year = 2016,
bike_powercc = 750},
                new bikes(){ bike_model = "Panigale 899", bike_brand = "Ducati", bike_make_year =
2019, bike_powercc = 900},
                new bikes(){ bike_model = "CB500x", bike_brand = "Honda", bike_make_year = 2021,
bike_powercc = 500}
            };

            //all bike models-------------------------------------------------------
            Console.WriteLine("\n---------- All Bike Models ( using for loop ) : ----------\n");
```

```
        for (int i = 0; i < bike.Count; i++)
        {
            Console.WriteLine($"Bike model = {bike[i].bike_model}, " +
                              $"Bike brand = {bike[i].bike_brand}, " +
                              $"Bike make year = {bike[i].bike_make_year}, " +
                              $"Bike CC = {bike[i].bike_powercc}");
        }

        //using for loop-------------------------------------------------------------
        Console.WriteLine("\n----------- Bike Models >= 500cc ( using for loop ) : -----------\n");
        for (int i = 0; i < bike.Count; i++)
        {
            if (bike[i].bike_powercc >= 500)
            {
                Console.WriteLine($"Bike model = {bike[i].bike_model}, " +
                                  $"Bike brand = {bike[i].bike_brand}, " +
                                  $"Bike make year = {bike[i].bike_make_year}, " +
                                  $"Bike CC = {bike[i].bike_powercc}");
            }

        }

        //using foreach loop---------------------------------------------------------
        Console.WriteLine("\n----------- Bike Models >= 500cc ( using foreach loop ) : -----------\n");
        foreach (var b in bike)
        {
            if(b.bike_powercc >= 500)
            {
                Console.WriteLine($"Bike model = {b.bike_model}, " +
                                  $"Bike brand = {b.bike_brand}, " +
                                  $"Bike make year = {b.bike_make_year}, " +
                                  $"Bike CC = {b.bike_powercc}");
            }
        }

        //using lambda expression----------------------------------------------------
        Console.WriteLine("\n----------- Bike Models >= 500cc ( using lambda exp ) : -----------\n");
        bike.Where(b => b.bike_powercc >= 500).ToList().ForEach(b => Console.WriteLine($"Bike
model = {b.bike_model}, " +
                                                                $"Bike brand = {b.bike_brand}, " +
                                                                $"Bike make year = {b.bike_make_year}, "
+
```

```
                                                    $"Bike CC = {b.bike_powercc}"));

        //using LINQ query----------------------------------------------------------
        Console.WriteLine("\n----------- Bike Models >= 500cc ( using LINQ ) : ----------\n");
        var output = from b in bike
                     where b.bike_powercc >= 500
                     select b;
        output.ToList().ForEach(b => Console.WriteLine($"Bike model = {b.bike_model}, " +
                                    $"Bike brand = {b.bike_brand}, " +
                                    $"Bike make year = {b.bike_make_year}, " +
                                    $"Bike CC = {b.bike_powercc}"));
        Console.ReadLine();

    }
  }
}
```

Output :

```
C:\WINDOWS\system32\cmd.exe

------------ All Bike Models ( using for loop ) : ------------

Bike model = Svartpilen 401, Bike brand = Husqvarna, Bike make year = 2015, Bike CC = 375
Bike model = Duke 250, Bike brand = KTM, Bike make year = 2012, Bike CC = 250
Bike model = ThunderBird 350x, Bike brand = Royal Enfield, Bike make year = 2018, Bike CC = 350
Bike model = GS750r, Bike brand = BMW, Bike make year = 2016, Bike CC = 750
Bike model = Panigale 899, Bike brand = Ducati, Bike make year = 2019, Bike CC = 900
Bike model = CB500x, Bike brand = Honda, Bike make year = 2021, Bike CC = 500

------------ Bike Models >= 500cc ( using for loop ) : ------------

Bike model = GS750r, Bike brand = BMW, Bike make year = 2016, Bike CC = 750
Bike model = Panigale 899, Bike brand = Ducati, Bike make year = 2019, Bike CC = 900
Bike model = CB500x, Bike brand = Honda, Bike make year = 2021, Bike CC = 500

------------ Bike Models >= 500cc ( using foreach loop ) : ------------

Bike model = GS750r, Bike brand = BMW, Bike make year = 2016, Bike CC = 750
Bike model = Panigale 899, Bike brand = Ducati, Bike make year = 2019, Bike CC = 900
Bike model = CB500x, Bike brand = Honda, Bike make year = 2021, Bike CC = 500

------------ Bike Models >= 500cc ( using lambda exp ) : ------------

Bike model = GS750r, Bike brand = BMW, Bike make year = 2016, Bike CC = 750
Bike model = Panigale 899, Bike brand = Ducati, Bike make year = 2019, Bike CC = 900
Bike model = CB500x, Bike brand = Honda, Bike make year = 2021, Bike CC = 500

------------ Bike Models >= 500cc ( using LINQ ) : ------------

Bike model = GS750r, Bike brand = BMW, Bike make year = 2016, Bike CC = 750
Bike model = Panigale 899, Bike brand = Ducati, Bike make year = 2019, Bike CC = 900
Bike model = CB500x, Bike brand = Honda, Bike make year = 2021, Bike CC = 500
```