* + 1. Write a console based application to dynamically load an assembly. Display the available types and their members. Prompt the user to invoke any method atruntime.

using System;

using System.Reflection;

namespace Reflectionexample

{

class Employee

{

public int empno

{

get;

set;

}

public string empName

{

get;

set;

}

public double salary

{

get;

set;

}

public Employee()

{

empno = 0;

empName = string.Empty;

salary = 0;

}

// Parameterised Constructor

public Employee(int eno, string n, double sal)

{

empno = eno;

empName = n;

salary = sal;

}

public void displayData()

{

Console.WriteLine(" Employee Number : {0}", empno);

Console.WriteLine("Eployee Name : {0}", empName);

Console.WriteLine("Employee Salary :{0}", salary);

}

}

class refl

{

static void Main(string[] args)

{

Assembly executing = Assembly.GetExecutingAssembly();

Type[] types = executing.GetTypes();

foreach (var item in types)

{

Console.WriteLine("Class : {0}", item.Name);

MethodInfo[] methods = item.GetMethods();

foreach (var method in methods)

{

Console.WriteLine("--> Method : {0}", method.Name);

ParameterInfo[] parameters = method.GetParameters();

foreach (var arg in parameters)

{

Console.WriteLine("----> Parameter : {0} Type : {1}",

arg.Name, arg.ParameterType);

}

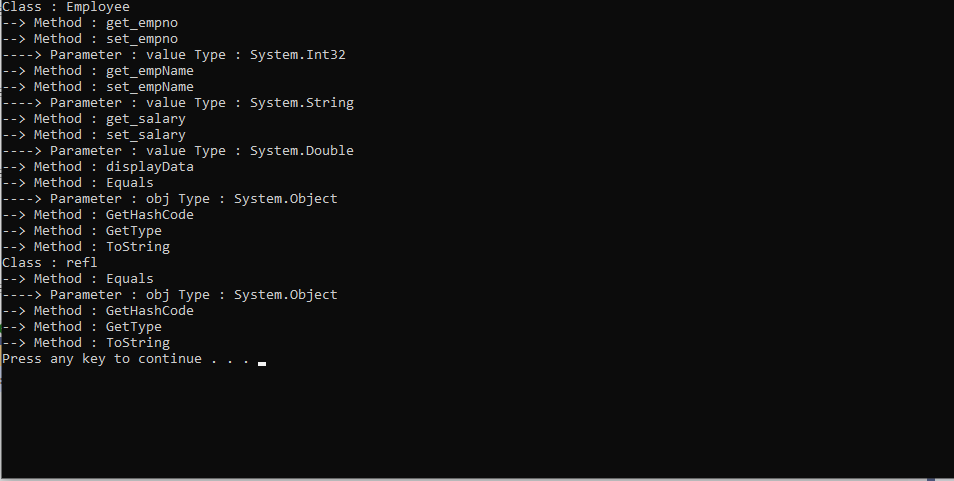
}

}

}

}

}



2. Define a SoftwareAttributewhich is having its base class as “Attribute” with following privatemembers:

* + - * StringProjectName.
      * StringDescription.
      * StringClientName.
      * StringStartedDate.
      * StringEndDate.

1. Write properties for all thesemembers.
2. Design a simple Console Application for testing refection concept of .NET Framework:
   * Define SoftwareAttributeas Custom Attribute
   * Write two classes HDFCAccount, ICICIAccountApply

SoftwareAttributeto these classes.

* + Write Testclass which will read attributes applied on each classes using reflectiontechnique.

using System;

using System.Reflection;

using System.Collections.Generic;

[AttributeUsage(AttributeTargets.All)]

public class SoftwareAttribute : Attribute

{

String ProjectName;

String Description;

String ClientName;

String StartedDate;

String EndDate;

public SoftwareAttribute(String p, String d, String c, String s, String e)

{

ProjectName = p;

Description = d;

ClientName = c;

StartedDate = s;

EndDate = e;

}

public static void AttributeDisplay(Type classType)

{

Console.WriteLine("Methods of class {0}", classType.Name);

MethodInfo[] methods = classType.GetMethods();

for (int i = 0; i < methods.GetLength(0); i++)

{

object[] attributesArray = methods[i].GetCustomAttributes(true);

foreach (Attribute item in attributesArray)

{

if (item is SoftwareAttribute)

{

SoftwareAttribute attributeObject = (SoftwareAttribute)item;

Console.WriteLine("{0} - {1}, {2}, {3} , {4} ,{5} ", methods[i].Name, attributeObject.ProjectName, attributeObject.Description, attributeObject.ClientName, attributeObject.EndDate, attributeObject.EndDate);

}

}

}

}

}

class ICICI

{

double AccountNumber;

string Name;

double Bankbalance;

public ICICI(double a, string n, double b)

{

AccountNumber = a;

Name = n;

Bankbalance = b;

}

[SoftwareAttribute("accessor", "gives the values of account number", "client name icici", "5th aug", "5th aug")]

public double getAccountNumber()

{

return AccountNumber;

}

[SoftwareAttribute("accessor", "gives the values of account holder name", "client name icici", "5th aug", "5th aug")]

public string getName()

{

return Name;

}

[SoftwareAttribute("accessor", "gives the values of bankbalance", "client name icici", "5th aug", "5th aug")]

public double getbankbalance()

{

return Bankbalance;

}

}

class HDFC

{

double AccountNumber;

string Name;

double Bankbalance;

public HDFC(double accountNumber, string name, double bankbalance)

{

AccountNumber = accountNumber;

Name = name;

Bankbalance = bankbalance;

}

[SoftwareAttribute("accessor", "gives the values of account number", "client name HDFC", "5th aug", "5th aug")]

public double getAccountNumber()

{

return AccountNumber;

}

[SoftwareAttribute("accessor", "gives the values of account holder name", "client name HDFC", "5th aug", "5th aug")]

public string getName()

{

return Name;

}

[SoftwareAttribute("accessor", "gives the values of account holder name", "client name HDFC", "5th aug", "5th aug")]

public double getbankbalance()

{

return Bankbalance;

}

}

class Program

{

static void Main(string[] args)

{

SoftwareAttribute.AttributeDisplay(typeof(ICICI));

Console.WriteLine();

SoftwareAttribute.AttributeDisplay(typeof(HDFC));

}

}

