EAD HOMEWORK\_1

Q1: What is IL and how IL code is converted into machine executable code?

Answer: IL stands for Intermediate Language. It is non-executable code, containing metadata(data about data e.g.: datatypes ,variables ,sizes etc.) of your compiled code.

Intermediate Language (IL) convert code to native code using Core CLR(common Language Runtime) and Ryu JIT(Just In Time)

Note: Ryu JIT only execute part of code which we wants.

Q2: Differentiate between .NET Core and .NET Frameworks. (Minimum 4 Points)?

Answer:

|  |  |
| --- | --- |
| .NET Core | .NET Frameworks |
| **NET Core** is used to create server applications that run on Windows, Linux and Mac | **NET framework** to create Windows desktop applications and server based applications |
| **.Net Core** is an open source. | Not fully open source  Certain components of the **.Net Framework** are open source. |
| .NET Core is packaged and installed independently of the underlying operating system as it is cross-platform. | .NET Framework is installed as a single package for Windows operating system. |
| .NET Core offers high performance and scalability. | .Net Framework is less effective in comparison to .Net Core in terms of performance and scalability of applications. |
| .NET Core is compatible with various operating systems — Windows, Linux, and Mac OS. | .NET Framework is compatible only with the Windows operating system. |

Q3 Why Decimal data type variable takes more space and store less data than double type variable? Please explain it with example.

Answer: Decimal data type variable takes more space and store less data than double type variable because the decimal type is accurate because it stores the number as a large integer and shifts the decimal point. For example, 0.1 is stored as 1, with a note to shift the decimal point one place to the left. 12.75 is stored as 1275, with a note to shift the decimal point two places to the left.

For example:

In Double 0.1 + 0.2 does NOT equal 0.3 , The double type is not guaranteed to be accurate because some numbers literally cannot be represented as floating-point values.

In Decimal 0.1 + 0.2 equals 0.3, because for accuracy purpose, decimal is used , that’s why we should use decimal whenever we want to compare floating values

Q4: What are the default values of

Answers

1. Int 0

2. uint 0

3. Double 0

4. Float 0

5. Decimal 0

6. Object

Q5:Please find out the errors if they have in these snippets and rewrite the code.

REWRITE:

**(1)**

static void Main(string[] args)

{

string String = null;

int numbers = 85;

String = $"Ahmad got {numbers} marks/n";

object obj = String;

obj = String.ToLower();

Console.WriteLine(obj);

}

**(2)**

static void Main(string[] args)

{

int age = int.Parse(Console.ReadLine());

Console.WriteLine(age);

}

Q6: Differentiate between var and dynamic with an example.

Answer: ‘dynamic’ type checks the data at runtime and execute according to data type where as ‘var’ type hold every type of data in compile time.

e.g.

|  |  |
| --- | --- |
| var name=”salman”;  name=name.toAscii();  // it gives syntax error because there is not ‘toAscii()’’ function in string class,  Error knows at compile time | dynamic name=”salman”;  name=name.toAscii();  // it gives error at runtime because dynamic checks meta data at runtime |

Q7: Write a program that displays the ascii value of a key when pressed by user.

static void Main(string[] args)

{

Console.WriteLine("Enter Key:");

ConsoleKeyInfo ascii = Console.ReadKey();

Console.WriteLine();

Console.WriteLine(

format: "Asci Value of Entered key: {0}",

arg0: (int)Convert.ToChar(ascii.KeyChar)

); ;

}

Q8: Write a program that displays the table of a number X to the length N. Your program must get these values as arguments in the main function. You need to use formatted output in your program to get full marks.

static void Main(string[] args)

{

int a = Convert.ToInt32(args[0]);

int b = Convert.ToInt32(args[1]);

for(int i=1;i<=b;i++)

{

Console.WriteLine(

format: "{0,-2} x {1,2} = {2,4}",

arg0: a,

arg1: i,

arg2: a \* i

);

}

}

Q9: Write a program that takes 5 Fruits names along with their prices in command line arguments and display them in proper output formatting.

static void Main(string[] args)

{

for(int i=0;i<=9;i=i+2)

{

Console.WriteLine(

format: "{0,-12}{1,12}",

arg0: args[i],

arg1: args[i+1]

);

}

}