**Statements could be of 3 types**

1. **Sequential**
2. **Conditional**
3. **Repetitive**

**Sequential Program > Where all the statements will run in a sequence one after other**

using System;

namespace FirstProject

{

    class Program

    {

        static void Main(string[] args)

        {

           // How to take input from user

             // Console.ReadLine();

             int num1, num2;

             Console.WriteLine("Enter Num1");

             // 0 -255

             num1 = Byte.Parse(Console.ReadLine());

             Console.WriteLine("Enter Num2");

             num2 = Convert.ToByte(Console.ReadLine());

             Console.WriteLine("Sum of {0} and {1} is {2}", num1, num2, (num1+num2));

             Console.WriteLine("Difference of {0} and {1} is {2}", num1, num2, (num1-num2));

             Console("Product of {0} and {1} is {2}", num1, num2, (num1\*num2));

             Console.WriteLine.WriteLine("Quotient of {0} and {1} is {2}", num1, num2, (num1/num2));

             Console.WriteLine("Remainder of {0} and {1} is {2}", num1, num2, (num1%num2));

        }

    }

}

**Conditional Program > Where all the statements will run depending upon some condition**

**By using**

1. **If**
2. **If – else**
3. **If elseif elseif else**
4. **switch**

**How to take input from user**

**Console.ReadLine()**

 num1 = Byte.Parse(Console.ReadLine());

 num1 = Int32.Parse(Console.ReadLine());

 num1 = Convert.ToByte(Console.ReadLine());

**Console.Read()**

1. **Syntax of if**

**If(condition) {statements}**

1. **Syntax of if – else**

**If(condition) {statements}**

**Else {statements}**

1. **Syntax of if – elseif elseif else**

**If(condition1) {statements}**

**Else If(condition2) {statements}**

**Else If(condition3) {statements}**

**Else {statements}**

1. **Switch**

**Switch(condition/expression)**

**{**

**Case value :{}**

**Case value :{}**

**Case value :{}**

**Case value :{}**

**Default : {}**

**}**

**Debugging > We want to see programs execution Flow**

**To use it , first add breakpoints.**

**Breakpoint is a point in your program which is when it is hit, after that you can see your programs flow**

**F10**

**F11**

**F12**

**What is difference in if else if & switch**

1. **In if else, it will check all conditions before it comes at the right condition**

**In switch, it will directly come at the right case (it takes less time to execute)**

1. **In switch , the only operator that can be used is equality**

using System;

namespace FirstProject

{

class Program

{

static void Main(string[] args)

{

// How to take input from user

// Console.ReadLine();

int num1, num2;

int choice;

Console.WriteLine("Enter Num1");

// 0 -255

num1 = Byte.Parse(Console.ReadLine());

Console.WriteLine("Enter Num2");

num2 = Convert.ToByte(Console.ReadLine());

Console.WriteLine("Enter your choice");

choice = Convert.ToByte(Console.ReadLine());

if (choice == 1)

Console.WriteLine("Sum of {0} and {1} is {2}", num1, num2, (num1 + num2));

else if (choice == 2)

Console.WriteLine("Difference of {0} and {1} is {2}", num1, num2, (num1 - num2));

else if (choice == 3)

Console.WriteLine("Product of {0} and {1} is {2}", num1, num2, (num1 \* num2));

else if (choice == 4)

Console.WriteLine("Quotient of {0} and {1} is {2}", num1, num2, (num1 / num2));

else if (choice == 5)

Console.WriteLine("Remainder of {0} and {1} is {2}", num1, num2, (num1 % num2));

else

Console.WriteLine("Invalid Choice")

; }

}

}

Repetitive Statements

By using Loops

Do – while

While

For

Foreach

How any statements in any loop

1. Starting Point Or initialization
2. Condition or Termination Point
3. Inre / Decre Point

**Do – while**

Syntax :

Initialization;

Do

{

Statements;

Inc / dec

} while(condition);

**while**

Syntax :

Initialization;

While(condition)

{

Statements;

Inc / dec

}

**For loop**

**For(initialization point;condition; icr/dec)**

**{**

Statements

**}**

**Foreach loop : it is used to work with Arrays / Collections**

**Foreach(type rangevariable in Array/Collection)**

**{**

**statements;**

**}**

foreach(int temp in num)

{

 Console.WriteLine(temp);

}

**Differences in for & foreach loop**

**For loop is faster, foreach loop uses a temporary variable**

**Foreach loop does not allow to modify contents of collection or Array**