รหัสวิชา 21910-2011

บทเรียนที่ 8

คำสั่ง while loop

เนื้อหาสาระ (Content)

เนื้อหาในหน่วยนี้ ในบางโปรแกรมเราจำเป็นต้องเขียนโปรแกรมวนซ้ำเพื่อให้กลับไปทำงานได้หลายรอบตามที่ต้องการ Visual C* ได้เตรียมคำสั่ง while loop เพื่อใช้ในการจัดการให้โปรแกรมทำงานวนซ้ำได้หลายรอบ

8.1 คำสั่ง while loop

โครงสร้าง while loop เป็นโครงสร้างแบบง่ายที่สุดที่สามารถนำมาใช้เขียนโปรแกรมเพื่อวนทำคำสั่ง (หรือกลุ่มของคำสั่ง) ซ้ำหลาย ๆ รอบ รูปแบบ การใช้งาน while loop เป็นดังนี้

```
while (condition)
        statement;
```

เช่นเดียวกับโครงสร้าง If เราสามารถกำหนดการวนซ้ำให้กับกลุ่มของคำสั่งได้โดยใช้วงเล็บปีกกา ({...})

```
while (condition) {
        statement1;
        statement2;
        :
 statementN;
}
```

8.2 ตัวอย่างโปรแกรม

ตัวอย่างที่ 1 จงเขียนโปรแกรมเพื่อพิมพ์ค่าตั้งแต่ 1 ถึง N โดยรับค่า N จากผู้ใช้

```
using System;

using System. Collections. Generic;

using System.Linq;

using System. Text;
```

```
{
    class Program
    {
        static void Main(string args)
                int i, N;
                Console. Write("Please Enter N: ");
                N = int.Parse(Console.ReadLine());
                i = 1;
                while (i <= N)
                {

                 Console. WriteLine(i);
                 i++;
                }
                 Console.ReadKey();
        }


    }
}
```

อธิบายโปรแกรม

กำหนดตัวแปร i,N มีชนิดเป็น int (เลขจำนวนเต็ม)
ใช้คำสั่ง Console.Write เพื่อพิมพ์ข้อความอธิบาย Please Enter N
ใช้คำสั่ง Console.ReadLine()); เพื่ออ่านค่า N
กำหนดให้ตัวแปร i มีค่าเท่ากับ 1
ใช้คำสั่ง while เช็คว่าค่า i น้อยกว่าหรือเท่ากับค่า N ถ้าน้อยกว่าก็จะพิมพ์ค่า I ออกมา ในคำสั่ง Console.WriteLine(i); และบวกค่า i เพิ่มทีละ 1 ในคำสั่ง i++; และวนกลับไปทำซ้ำที่สั่ง while(i<=N)
เช่น ถ้าเราป้อนเลข 5 ให้กับค่า N ดังผลลัพธ์ เมื่อค่า i มีค่าเท่ากับ 5 แล้วก็จะออกจากการวนลูป While ทันที ดังตัวอย่างผลลัพธ์ของโปรแกรมข้างล่างนี้

ผลลัพธ์ของโปรแกรม

```
Please Enter N: 5
1
2
3
4
5
```

หากลองมาพิจารณาการทำงานของโปรแกรมที่ละขั้นตอน เมื่อโปรแกรมเริ่มทำงาน ผู้ใช้จะป้อนตัวเลขเพื่อใช้เป็นค่าของ N และโปรแกรมจะกำหนดให้ค่า 1 เป็นค่าเริ่มต้นของ i ที่บรรทัดถัดมาจากนั้นโครงสร้าง While จะมีผลทำให้โปรแกรมทำคำสั่งที่ Console.WriteLine(i); i++; จนกระทั่งนิพจน์ i <=N มีค่าเป็นเท็จในการวนซ้ำแต่ละครั้งโปรแกรมจะพิมพ์ค่า i และเพิ่มค่า : ขึ้นทีละหนึ่ง ผลการทำงานของโปรแกรมจึงเป็นการ พิมพ์ตัวเลขตั้งแต่ 1 (ซึ่งเป็นค่าเริ่มต้นของ i) จนถึงค่าของ N (ซึ่งเป็นค่าสุดท้ายของ : ที่ยังทำให้นิพจน์ i <= N เป็นจริง)

ตัวอย่างที่ 2 เราสามารถเขียนโปรแกรมเพื่อคำนวณหาผลรวมทั้งแต่ 1 ถึง N ใด ๆ โดยแก้ไขโปรแกรมในตัวอย่างที่ 1 เพียงเล็กน้อยเท่านั้นดังต่อไปนี้

```
using System;
using System. Collections. Generic;
using System.Linq;
using System.Text;
namespace ConsoleApplication15
{
        class Program
        {
            static void Main(string args)
            {
                int i, N,sum;
                Console. Write("Please Enter N: ");
                N= int.Parse(Console.ReadLine();
                i = 1; sum = 0;
                while (i <= N)
                {
                        Console. WriteLine(i);
                        sum = sum + i; it+;
                }
                        Console. WriteLine("Sum from 1 to {0) = (1) ", N, sum);
                        Console.ReadKey();
            }
        }
    }
```

ผลลัพธ์ของโปรแกรม บวกเลข 1 ถึง 4 แล้วหาผลบวกไว้ในตัวแปร sum ผลบวก คือ 10

Please Enter N: 4

1

2

3

4

Sum from 1 to 4 = 10

ในตัวอย่างที่ผ่านมาจะมีการกำหนดเงื่อนไขการทำงานของลูปโดยการกำหนดตัวแปรตัวหนึ่งขึ้นมาใช้เป็นตัวนับ (counter) ซึ่งมักจะถูกตั้งค่าเริ่มต้นให้เป็นศูนย์ก่อนเริ่มเข้าลูป และเพิ่มค่าขึ้นทีละหนึ่งหลังจากคำสั่งภายในลูปถูกทำงานเสร็จสิ้นหนึ่งรอบ โปรแกรมจะทำซ้ำไปเรื่อย ๆ จนกระทั่งตัวนับมีค่าเกินกว่าค่าที่กำหนดการวนลูปในลักษณะนี้เรียกว่าลูปวนนับ (counting loop)

อย่างไรก็ตาม เนื่องจากลูปวนนับไม่สามารถนำไปใช้ในสถานการณ์ที่เราไม่สามารถคาดการณ์ได้ว่าลูปจะต้องถูกเรียกทำงานกี่รอบจึงจะจบการทำงาน เช่น การวนรับค่าจากผู้ใช้เรื่อย ๆ จนกว่าผู้ใช้จะป้อนค่าพิเศษ ค่าหนึ่ง ในกรณีนี้เราจะไม่อาศัยตัวแปรที่ทำหน้าที่เป็นตัวนับ แต่จะใช้วิธีกำหนดเงื่อนไขของลูปให้มีการเฝ้าดูค่าในตัวแปร (ซึ่งมักใช้รับค่าจากผู้ใช้) ว่ามีค่าเท่ากับค่าพิเศษที่กำหนดให้สิ้นสุดการวนลูปหรือไม่การวนลูปในลักษณะนี้เรียกว่าลูปเฝ้ายาม (sentinel loop)

ตัวอย่างที่ 3 จงเขียนโปรแกรมเพื่อวนรับตัวเลขจากผู้ใช้จนกว่าผู้ใช้จะป้อนค่าติดลบจึงจบการทำงาน

```
using System;
class While4 {
        static void Main() {
        int N = 0;
          while (N>= 0) {// stop when N is negative
            Console. Write("Please input N: ");
            N = int.Parse(Console.ReadLine());
        }
Console. WriteLine('Bye Bye!!!");
      }
    }
```

ผลลัพธ์ของโปรแกรม

Please input N: 3

Please input N: 2

Please input N: 3000

Please input N: 9999

Please input N: -50

Bye Bye!!!

สรุปสาระสำคัญ

คำสั่ง while loop เป็นคำสั่งแบบง่ายที่สุดที่สามารถนำมาใช้เขียนโปรแกรมเพื่อทำงานวนทำคำสั่ง หรือ กลุ่มของคำสั่งวนซ้ำหลาย ๆ รอบ รูปแบบการใช้งาน while loop ดังนี้

ชยวกับคำสั่ง if เราสามารถกำหนดการวนซ้ำให้กับกลุ่มของคำสั่งได้โดยใช้วงเล็บปีกกา (L...

```
while (condition)
        statement;
```

เช่นเดียวกับคำสั่ง if เราสามารถกำหนดการวนซ้ำให้กับกลุ่มของคำสั่งได้โดยใช้วงเล็บปีกกา ({...})

```
while (condition) {
        statement1;
        statement2;
        :
        statementN;
}
```

ชยวกับคำสั่ง while (condition) {

```
    statement1; statement2;
```