



การคำนวณทางคอมพิวเตอร์



ชนิดข้อมูลพื้นฐาน

ชนิดข้อมูลพื้นฐาน มีดังนี้

- ชนิดตัวเลข (numeric type)
- ชนิดตัวอักษร (charactertype)
- ชนิดตรรกะ (boolean type)
- ชนิดวันที่ (date type)
- ชนิดวัตถุ (object type)

ขนาดและช่วงข้อมูล ของตัวแปรชนิดตัวเลข

ชนิดของตัวแปร	ขนาดความจุ	ช่วงข้อมูล
Byte	8-bit	มีค่าระหว่าง 0 ถึง 255
Short	16-bit	มีค่าระหว่าง -32,768 ถึง 32,767
Integer	32-bit	มีค่าระหว่าง -2,147,483,648 ถึง 2,147,483,647
Long	64-bit	มีค่าระหว่าง -9,223,372,036,854,775,808 ถึง 9,223,372,036,854,775,807
float	7 digit	มีค่าระหว่าง $-3.4028235E+38$ ถึง $-1.401298E-45$ สำหรับค่าลบ; $1.401298E-45$ ถึง $3.4028235E+38$ สำหรับค่าบวก
Double	64-bit	มีค่าระหว่าง $1.79769313486231570E+308$ ถึง $-4.94065645841246544E-324$ สำหรับค่าลบ; $4.94065645841246544E-324$ ถึง $1.79769313486231570E+308$ สำหรับค่าบวก
Decimal	128-bit	มีค่าระหว่าง 0 through +/- 79,228,162,514,264,337,593,543,950,335 เมื่อไม่มีจุดตำแหน่งทศนิยม; 0 through +/- 7.9228162514264337593543950335 เมื่อมีทศนิยม 28 ตำแหน่ง



ตัวอย่าง การประกาศตัวแปร ข้อมูลชนิดตัวเลข

```
int i, j, k;
```

```
int myInt = 5;
```

```
float f, salary;
```

```
double d;
```



ตัวอย่าง การประกาศตัวแปร ข้อมูลชนิดตัวอักษร

```
char c, ch;
```

```
char namr;
```

```
char id;
```



ตัวอย่าง การประกาศตัวแปร ข้อมูลชนิดตรรกะ

```
bool b=true;
```

```
bool b=false;
```



ตัวอย่าง การประกาศตัวแปร ข้อมูลชนิดวันที่

```
DateTime date = DateTime.Now;
```



ตัวอย่าง การประกาศตัวแปร ข้อมูลชนิดวัตถุ (object type)

```
object obj;  
obj = 100; // this is boxing
```




ตัวดำเนินการพื้นฐาน (Operator)

โปรแกรมเกมคอมพิวเตอร์ มีตัวดำเนินการพื้นฐานที่ใช้ในการคำนวณ มีดังต่อไปนี้

- 1) ตัวดำเนินการเท่ากับ ใช้เครื่องหมาย =
- 2) ตัวดำเนินการคูณ ใช้เครื่องหมาย *
- 3) ตัวดำเนินการหาร ใช้เครื่องหมาย /
- 4) ตัวดำเนินการบวก ใช้เครื่องหมาย +
- 5) ตัวดำเนินการลบ ใช้เครื่องหมาย -
- 6) ตัวดำเนินการหารคิดเฉพาะเศษ ใช้เครื่องหมาย %



ตัวดำเนินการพื้นฐาน (Operator)

การลดรูปของเครื่องหมาย (short and assignment operators) เป็นวิธีการทำให้การเขียนตัวดำเนินการมีขนาดสั้นลง ตัวอย่าง เช่น ถ้าต้องการเพิ่มค่าอีก 1 ให้กับตัวแปร สามารถเขียนคำสั่งได้คือ `Count = Count+1` หรือสามารถทำการลดรูปเป็น `Count+=1` หรือสามารถทำการเพิ่มค่าทีละ 1 โดยใช้เครื่องหมาย `Count++` ถ้าต้องการลดค่าอีก 1 ให้กับตัวแปร สามารถเขียนคำสั่งได้คือ `Count = Count-1` หรือสามารถทำการลดรูปเป็น `Count-=1` หรือสามารถทำการลดค่าทีละ 1 โดยใช้เครื่องหมาย `Count--`



ตัวอย่าง โปรแกรมเกมคอมพิวเตอร์ในการทำงานตัวดำเนินการ

```
using System;
class Binary
{
    public static void Main()
    {
        int x, y, result;
        float floatresult;
        x = 10;
        y = 2;
        result = x+y; // 10 + 2 = 12
        Console.WriteLine("x+y: {0}", result);
    }
}
```

ตัวอย่าง โปรแกรมเกมคอมพิวเตอร์ในการทำงานตัวดำเนินการ (ต่อ)

```
result = x-y; // 10 - 2 = 8
Console.WriteLine("x-y: {0}", result);
    result = x*y; // 10 * 2 = 20
Console.WriteLine("x*y: {0}", result);
    result = x/y; // 10 / 2 = 5
Console.WriteLine("x/y: {0}", result);
floatresult = (float)x/(float)y; // 10 / 2 = 5.0
Console.WriteLine("x/y: {0}", floatresult);
    result = x%y; // 10 % 2 = 0 เป็นการหารแล้วเอาเศษ
Console.WriteLine("x%y: {0}", result);
}
}
```

ตัวดำเนินการเปรียบเทียบทางคณิตศาสตร์

สัญลักษณ์	คำอธิบาย
<	สัญลักษณ์ใช้ในการเปรียบเทียบข้อมูล 2 ค่า หาค่าน้อยกว่า
<=	สัญลักษณ์ใช้ในการเปรียบเทียบข้อมูล 2 ค่า หาค่าน้อยกว่าหรือเท่ากับ
>	สัญลักษณ์ใช้ในการเปรียบเทียบข้อมูล 2 ค่า หาค่ามากกว่า
>=	สัญลักษณ์ใช้ในการเปรียบเทียบข้อมูล 2 ค่า หาค่ามากกว่าหรือเท่ากับ
==	สัญลักษณ์ใช้ในการเปรียบเทียบข้อมูล 2 ค่า หาค่ามากกว่าหรือเท่ากับ
!=	สัญลักษณ์ใช้ในการเปรียบเทียบข้อมูล 2 ค่า ค่าเท่ากัน
	สัญลักษณ์ใช้ในการเปรียบเทียบข้อมูล 2 ค่า ค่าไม่เท่ากัน