04 Numbers and Integer Math

May 12, 2022

1 Numbers and Integer Math

1.1 Integer Math

You have a few `integers` defined below. An `integer` is a positive or negative whole number. > Before you run the code, what should c be?

1.2 Addition

[2]:
```
int a = 18;
int b = 6;
int c = a + b;
Console.WriteLine(c);
```

```
24
```

1.3 Subtraction

[3]:
```
int c = a - b;
Console.WriteLine(c);
```

```
12
```

1.4 Multiplication

[4]:
```
int c = a * b;
Console.WriteLine(c);
```

```
108
```

1.5 Division

[5]:
```
int c = a / b;
Console.WriteLine(c);
```

```
3
```

2 Order of operations

C# follows the order of operation when it comes to math. That is, it does multiplication and division first, then addition and subtraction. > What would the math be if C# didn't follow the order of operation, and instead just did math left to right?

[6]:
```
int a = 5;
int b = 4;
int c = 2;
int d = a + b * c;
Console.WriteLine(d);
```

```
13
```

2.1 Using parenthesis

You can also force different orders by putting parentheses around whatever you want done first

[7]:
```
int d = (a + b) * c;
Console.WriteLine(d);
```

```
18
```

You can make math as long and complicated as you want. > Can you make this line even more complicated?

[8]:
```
int d = (a + b) - 6 * c + (12 * 4) / 3 + 12;
Console.WriteLine(d);
```

```
25
```

2.2 Integers: Whole numbers no matter what

Integer math will always produce integers. What that means is that even when math should result in a decimal or fraction, the answer will be truncated to a whole number. > Check it out. WHat should the answer truly be?

[9]:
```
int a = 7;
int b = 4;
int c = 3;
int d = (a + b) / c;
Console.WriteLine(d);
```

```
3
```

[ ]:
```

```