Operator Shortcuts

Often we need to update a variable in our program. We can do so by modifying that variable using an arithmetic expression, then re-saving it to the same variable name:

```
int apple = 0;
apple = apple + 1;
Console.Write(apple); // prints 1
```

Programmers are always looking for shortcuts. For instance, we can condense the same program above using the shorthand `++`. The combined addition signs represent the idea of *incrementing by one*. We can do the same with the subtraction symbol `--`.

```
// a shorter way to do the same thing
int apple = 0;
apple++;
Console.Write(apple); // prints 1
```

If we want the amount to increment by another value, say 3, we would do the following:

```
int apple = 0;
apple += 3; // is the same as apple = apple + 3
Console.Write(apple); // prints 3
```

Again, if we want to decrement, you would do `-=3`.

Ultimately, programmers disagree on this, so everyone does it differently! On Codecademy, we prefer clarity over conciseness, so we'll use the long form (apple = apple + 3) except for incrementing by one (apple++).

☑Instructions

1.

Ever heard of the phrase, "two steps forward, one step back?" It means that you can make progress in some task, but also might suffer some setbacks. It's also a great way to

illustrate the concept of incrementing and decrementing!

Start by creating a variable named `steps`. Set it to 0.

Hint

Since we'll be moving in whole steps, we'll use an `int` data type.

2.
Two steps forward

Next, increment the value of steps by two and resave this new value to steps.

Hint

There are two ways of incrementing a variable:

```
bottle = bottle + 4;
```

```
bottle += 4;
```

3.
One step back.

Now, decrement the variable `steps` by 1.

Hint

There are three ways of decrementing a variable by 1:

```
bottle = bottle - 1;
```

```
bottle -= 1;
```

```
bottle--;
```

4.

Print the final value of `steps` to the console.

Hint

Use the following command to print something:

```
Console.WriteLine(variableName);
```