Methods

The third type of member in classes is *methods*. This lesson assumes that you are already familiar with methods, so the syntax should look familiar.

In the past you learned that methods are a useful way to organize chunks of code to perform a task. But most methods belong to a class (even the ones you have written!), so methods are also used to define how an instance of a class behaves. You can think of them as the “actions” that an object can perform.

This code defines a method `IncreaseArea()` that changes the value of the `Area` property:

```
class Forest {
  public int Area
  { /* property body omitted */  }
  public int IncreaseArea(int growth)
  {
    Area = Area + growth;
    return Area;
  }
}
```

You would call the method like so:

```
Forest f = new Forest();
int result = f.IncreaseArea(2);
Console.WriteLine(result); // Prints 2
```

☑Instructions

1.

In the `Forest` class, define a public method `Grow()`. It should:

take zero arguments

increase the `Trees` property by 30 and the `Age` property by 1

return the updated number of trees

Hint

Here’s the first line of the method definition:

```
public int Grow()
```

2.
Define a public method `Burn()`. It should:

take zero arguments

decrease the `Trees` property by 20 and increase the `Age` property by 1

return the updated number of trees

Hint

Here’s the first line of the method definition:

```
public int Burn()
```