Question 62: Placing too many lines of code inside Main() leads to code that can be difficult to read. How can this be addressed?

Answer 62: C# allows creating methods. Methods are usually created to perform a specific task, and allow breaking a huge amount of code into small, understanble pieces.

Do these steps only once. Press this arrow repeatedly to step through the code.

1. Debug 2. Step Into 3. 

```
using static System.Console;
class Program
{
    static void Main()
    {
        double input = 100; //1. Declares and sets input
        PrintIncreasedValue(input); //2. Calls method with input
    }
    static void PrintIncreasedValue ( double x )//3. Method header
    {
        WriteLine($"{x} increased by 10% is {x * 1.1}.");//4. Body of method
    }
}
```

This is the "calling code"

```
100 increased by 10% is 110.
```

1. "static" means method can be called directly by typing its name.
2. "void" means this method does not send a value back to the calling code
3. PrintIncreasedValue is the name
4. "double" is the data type of the parameter
5. x is the parameter

```
static void PrintIncreasedValue ( double x )
{
    WriteLine($"{x} increased by 10% is {x * 1.1}.");
}
```

```
static void Main()
{
    double input = 100;1.
    PrintIncreasedValue(input);
}
static void PrintIncreasedValue ( double x )3.
{
    WriteLine($"{x} increased by 10% is {x * 1.1}.");4.
}
```

5.

2. To call a method means to write its name and, in this case, pass in an argument. The argument is a variable name like "input.

3. Argument goes into method through parameter.

x exists in this method as a local variable.