Define a Method

Up until now, you've been calling built-in methods: methods that are available whenever you use C#. Sometimes you need a custom method for your specific program. In that case, you'll need to define your own!

The basic structure of a method definition looks like this:

```
static void YourMethodName()
{
}
```

We'll skip over static and void for the moment.

In C#, it's convention to use PascalCase to name your method. The name starts with an uppercase letter and each word following begins with an uppercase as well. It's not required in C#, but it makes your code easier to read for other developers.

The body of your method goes between the curly braces: { } . Whenever the method is called, the code in the body is executed.

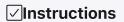
```
static void YourMethodName()
{
   Console WriteLine("Hi there!");
}
```

Just like any other method, we call it with parentheses:

```
YourMethodName ();
```

Look closely at the code in the editor and you'll see that you've been defining methods all along! Main() is a method. Every time you run the code, the Main() method is executed.

Since Main() is already a method, we'll define our own methods outside of Main().



1.
Define a method named VisitPlanets() outside of the Main() method and run the code.

VisitPlanets() can print anything you'd like to the console, but something like "You visited many new planets..." would be appropriate.



2. Why isn't your method executed? It's not called within Main(). Call it in Main() and run the code again.

```
Static void Main

{
    // Call your method here!
}
```