

A Note on Parameters

One thing to watch for with parameters: they can only be used inside their method!

```
static void YourMethodName(string message)
{
    Console.WriteLine(message);
}
Console.WriteLine message); // causes an error!
```

You'll see an error like...

```
error CS0103: The name 'message' does not exist in the current context
```

When talking to other developers about this type of issue, you might hear the term *scope*. While the entire concept of scope won't be covered in this lesson, you should know how it applies here: a parameter's scope is within its method, which means that the name (`message` in this case) is only valid within its method. If the parameter name is used outside the method, it has no meaning, so it throws an error.

☒ Instructions

1.

Try causing an error by using the parameter in `VisitPlanets()` outside of the method body.

For example, if your parameter is named `numberOfPlanets`, you could try to print `numberOfPlanets` in `Main()`.

Hint



You can cause an error by referencing a variable name that is only defined in the `VisitPlanets()` method.

`numberOfPlanets` is only defined in `VisitPlanets()`. You can reference it in `Main()` by trying to change its value or trying to print it to the console.

