

Review

Congrats on finishing! You can now use and define methods with output.

Here's what else you've learned in this lesson:

Methods return values with the `return` keyword.

Every method has a return type, designated in its method signature. That type must match the type of the value actually returned.

If a method returns no type, its return type is `void`.

`out` parameters can be used to return multiple values from a method.

You can always review this material on the [Microsoft documentation](#).

☒ Instructions

1.
Make sure you know how to apply all of these concepts before moving on!

Call `DecoratePlanet()` with the argument `destination`.

Store the returned value in the variable `welcomeMessage`.

Hint



In this example, the value returned by `Math.Floor()` is stored in the variable `smaller`:

```
int smaller;  
smaller = Math.Floor(5, 16);
```

2.

Call `Int32.TryParse()` , using `galaxyString` and `galaxyInt` as arguments.

Store the returned value in the variable `outcome` .

Hint



Here's the method declaration for `Int32.TryParse()` :

```
public static bool TryParse (string s, out int result);
```

The first parameter is of type `string`

The second parameter is an `out` parameter of type `int`

The method returns a `bool`