Object Members

`Object` has a few useful members and they're accessible by every type. Here are some important ones:

- `Equals(Object)` — returns `true` if the current instance and the argument are equal (using value equality for value types and referential equality for reference types)

- `GetType()` — returns the type of the object

- `ToString()` — returns a string describing the object

You can see each method in action here:

```
Object o1 = new Object();
// t is System.Object
Type t = o1.GetType();

string s = o1.ToString();
// Prints "System.Object"
Console.WriteLine(s);

Object o2 = o1;
// Equals true
bool b = o1.Equals(o2);
```

Remember that we can access inherited members from a derived class. In this case, every type inherits from `Object`, so every type can access these members!

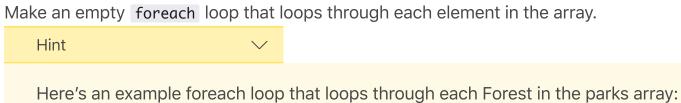For the full list of `Object` members read the Microsoft documentation.

☑Instructions

1.
Create an array of type `Object[]` that contains `b`, `d`, `r`, and `i`.

Hint ⌄

Here's an example array declaration. This array is of type `int[]`:

```
int[] playerScores = { 7, 20, 22, 15 };
```

2.

Make an empty `foreach` loop that loops through each element in the array.

Here's an example foreach loop that loops through each Forest in the parks array:

```
foreach (Forest f in parks)
{

}
```

3.

In the body of the loop, call the `GetType()` method of each element and print out the result.

You can call `GetType()` and `Console.WriteLine()` in the same line:

```
Console.WriteLine(element.GetType());
```