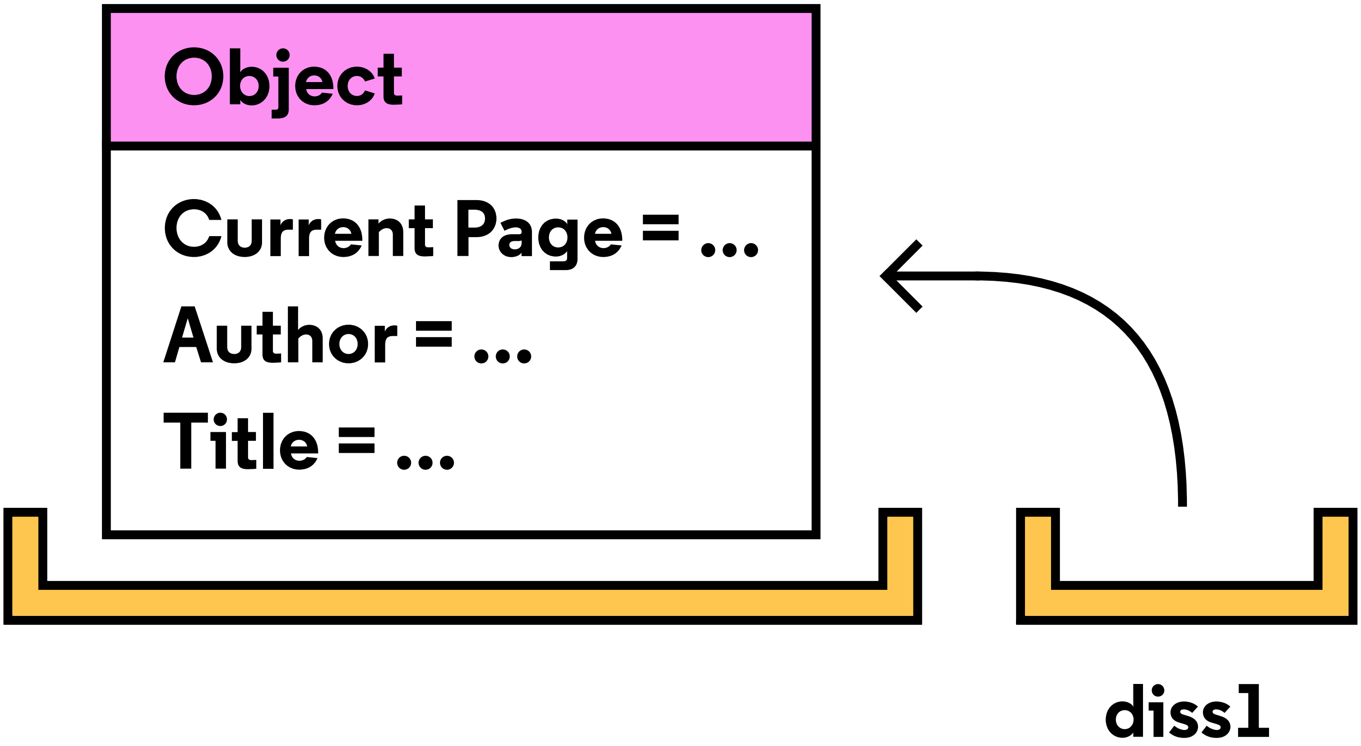
References of the Same Type

Classes are *reference types*. That means that when we create a new instance of a class and store it in a variable, the variable is a *reference* to the object.

Let’s see what’s happening behind the scenes. When this code is run:

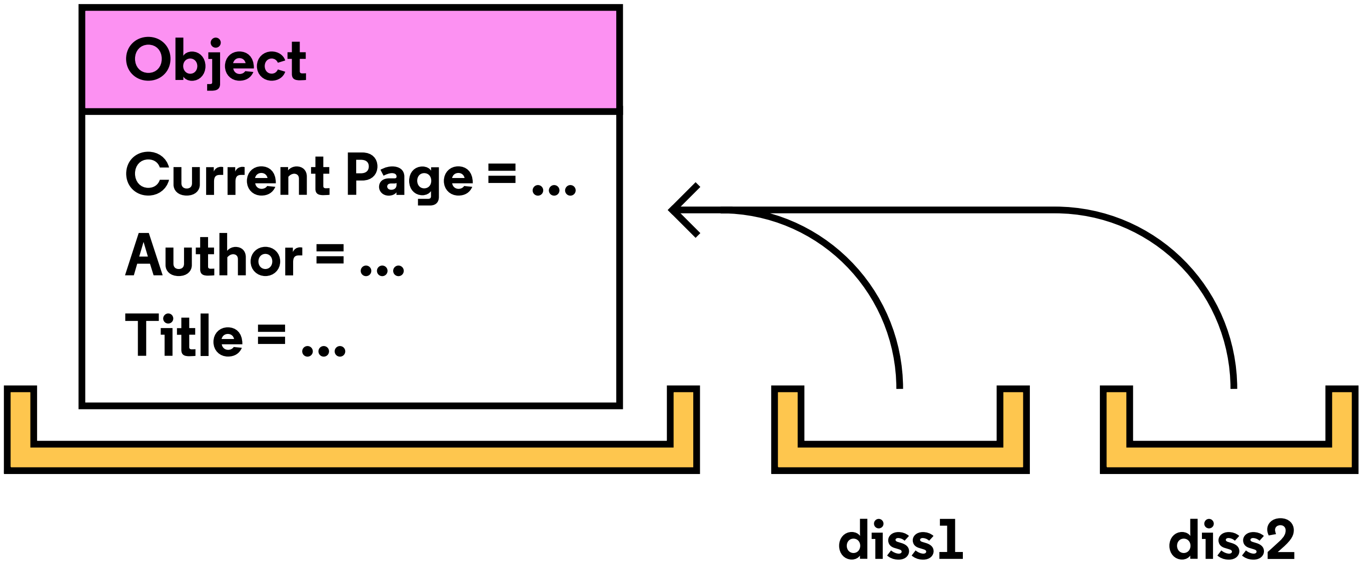
```
Dissertation diss1 = new Dissertation();
```

A new `Dissertation` instance is constructed and stored in the computer’s memory. You can imagine a slot in your computer holding the instance’s type, property values, etc. `diss1` is a reference to that location in memory.



`diss1` is not the actual object, it is a *reference* to the object. Thus an object can have multiple references:

```
Dissertation diss1 = new Dissertation();
Dissertation diss2 = diss1;
```



Now there are two references to the same location in memory: we can say that `diss1` and `diss2` refer to the same object. If changes are made to that object, then they will be reflected in both references to it:

```
Dissertation diss1 = new Dissertation();
Dissertation diss2 = diss1;
diss1.CurrentPage = 0;
diss2.CurrentPage = 16;
Console.WriteLine(diss1.CurrentPage);
Console.WriteLine(diss2.CurrentPage);
```

> The middle two lines of this code are setting the `CurrentPage` property of the same object (first setting it to `0`, then `16`)
>
> The last two lines will print the same value, `16`

You can imagine references like directions to a house: they tell you where to find the house, but they are not the house itself!

☑**Instructions**

1.

Create a new `Diary` object with a current page of `5` using the constructor `new Diary(5)`. Store a reference to that object and name it `dy1`. Its type should also be `Diary`.

Hint ⌄

Here’s an example of constructing a `Random` object:

```
Random rand = new Random();
```

2.
Create another reference to that object and name it `dy2`.

Hint

Remember that any variable representing an object is a *reference*to the object, not the object itself.

3.
Flip `dy2` by calling its `Flip()` method. This increases the `CurrentPage` property by 1.

4.
Print out the `CurrentPage` property for both `dy1` and `dy2`. They should be the same!

Hint

They should print the same value because they are two references to the same object.