What are objects?

This lesson introduces objects and explains in detail what object properties are by using an example.

Introduction

In order to understand OOP in JavaScript, it's crucial to get familiarized with the concept of *objects*.

Objects are a primary part of JavaScript, as almost everything in it is an object. For example, *functions*, *arrays*, *regular expressions*, *dates*, and even data types like *boolean* and *strings*, if declared with the keyword `new`, can be considered an object.

What Is an Object?

In real life, *objects* are found everywhere, so these real-life scenarios can also be mapped into object-oriented code.

Example

Let's discuss this by looking at an example. Assume you have three shapes: *square*, *rectangle* and *circle*. Now, you want to calculate the areas of all three shapes.

If you had to write a piece of code that would calculate the areas of these three shapes, what would you do? A common approach would be to write functions for all three shapes to compute the respective areas.

Another approach would be to convert our code to the object-oriented paradigm by creating *objects* for each shape: square, rectangle, and circle. Here, each object has its own set of properties which include:

- **data values**
- **functions**

In our example, what values would we need to calculate the areas of the three shapes?

Properties of the objects
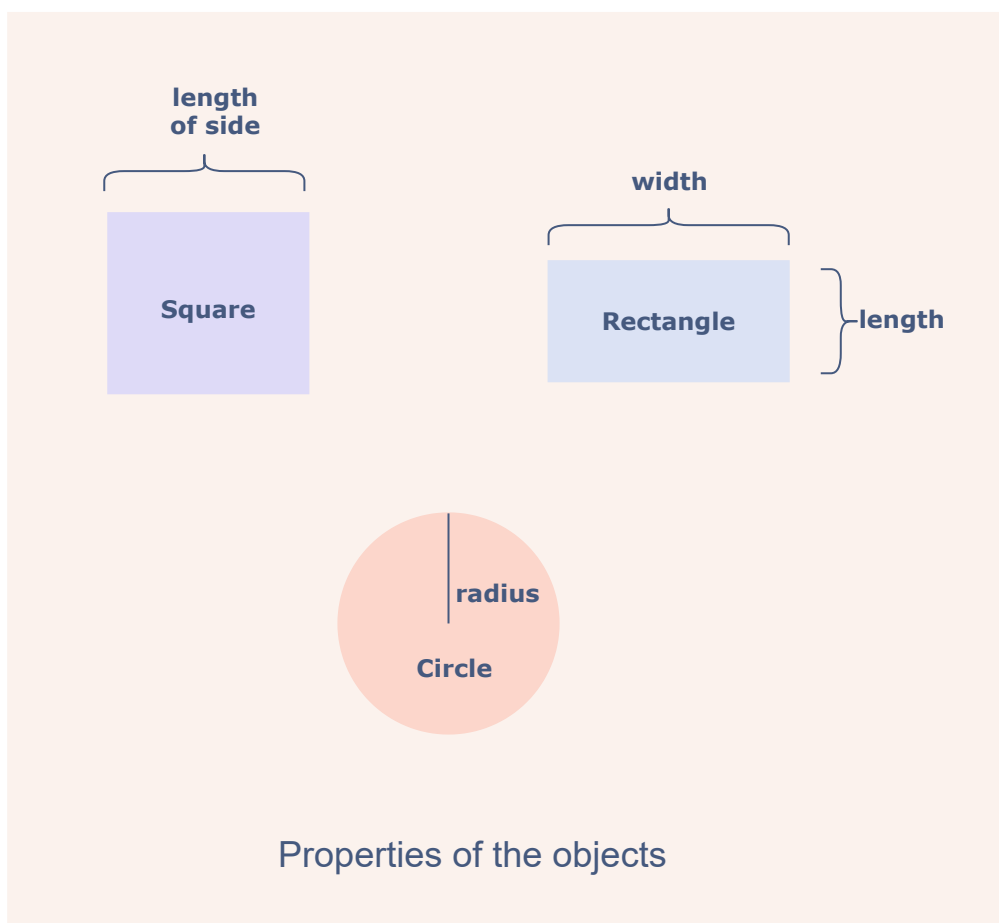
We need the *length*, *width*, and *radius*. These values will be encapsulated in

the *object* of that particular shape.

Similarly, we also require a *function* to calculate the *area*. This will be encapsulated in the object as a part of its *properties*.

Rectangle

width
length
area()

Object
encapsulating
its properties
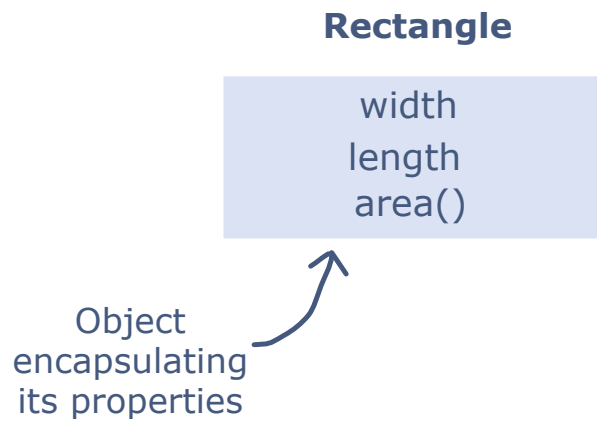
Now that you know what objects are, in the next lesson let's discuss what object literals are and how to create them.