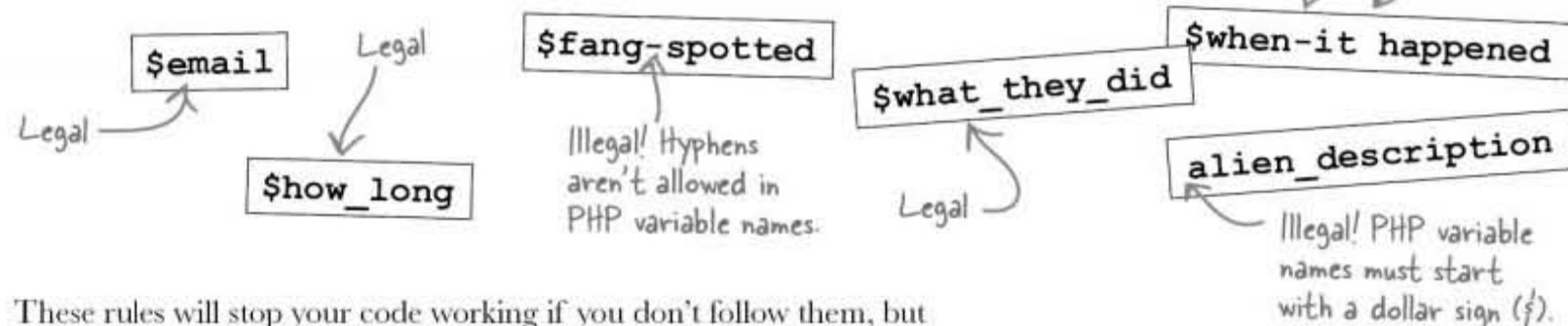


## Finding the perfect variable name

In addition to starting with a \$, PHP variable names are also case-sensitive. But that's not all—there are other important rules governing how you name variables. Some of these rules are syntax rules, meaning your code will break if you ignore them, while other rules are just good ideas passed down from wise old PHP coders.

Let's start with the official rules that will absolutely cause problems if you ignore them when naming variables. Follow these rules to create **legal** variable names.

- ✓ The first character must be a dollar sign (\$). *Got it!*
- ✓ A variable name must be at least one character in length. *Not counting the \$ character, which is required of every variable name.*
- ✓ The first character after the dollar sign can be a letter or an underscore (\_), and characters after that can be a letter, an underscore, or a number.
- ✓ Spaces and special characters other than \_ and \$ are not allowed in any part of a variable name.



These rules will stop your code working if you don't follow them, but there are a couple more rules that are good to follow as more of a coding convention. These rules help make PHP code a little more consistent and easier to read.

- ✓ Use all lowercase for variable names.
- ✓ Separate words in a multi-word variable name with underscores.

These last two rules won't break your code if you ignore them, and you'll certainly run across PHP code that doesn't adhere to them yet works just fine. This is because they are just a stylistic convention—but they will serve you well as you begin creating and naming variables of your own.

A variable is a container that you can store data in, and every variable has a unique name.

PHP variable names must begin with a dollar sign, and cannot contain spaces.