* **Var** Mutable variable, old way. Reason it is not used anymore is that its scope is whole file. This makes separation of code difficult and prone to bugs on a team environment sometimes.
* **let** same as var, but it only exists within the current scope rather than the whole file. This is the new and preferred way now
* **const** immutable variable, can’t change later once assigned. Good practice is to use const on every variable that is not supposed to change. Once we assign a value to const it can’t change anymore.
  + We must **initialize** a const, otherwise it will give us an error.
  + For data structures it’s a bit different though, because even though const are immutable, we can change for example a const map or list by adding values into it. This doesn’t actually change the const since the variable itself point to the start of the map/list. Basically, as long as reference to a data structure not changed, we cause edit values of a const data structure like array or object.

**Naming conventions:**

* Variable names can only contain numbers, letters \_ and $.
* Variable names can’t start with a number.
* Can’t use keywords as a variable name.

**Truthy and Falsy values:**



* There are 5 types of **falsy** values in JavaScript. When we attempt to convert them to a Boolean, these will be converted to false.
* Everything else is **truthy** values. If we try to convert them to Boolean, they will be converted to true.



