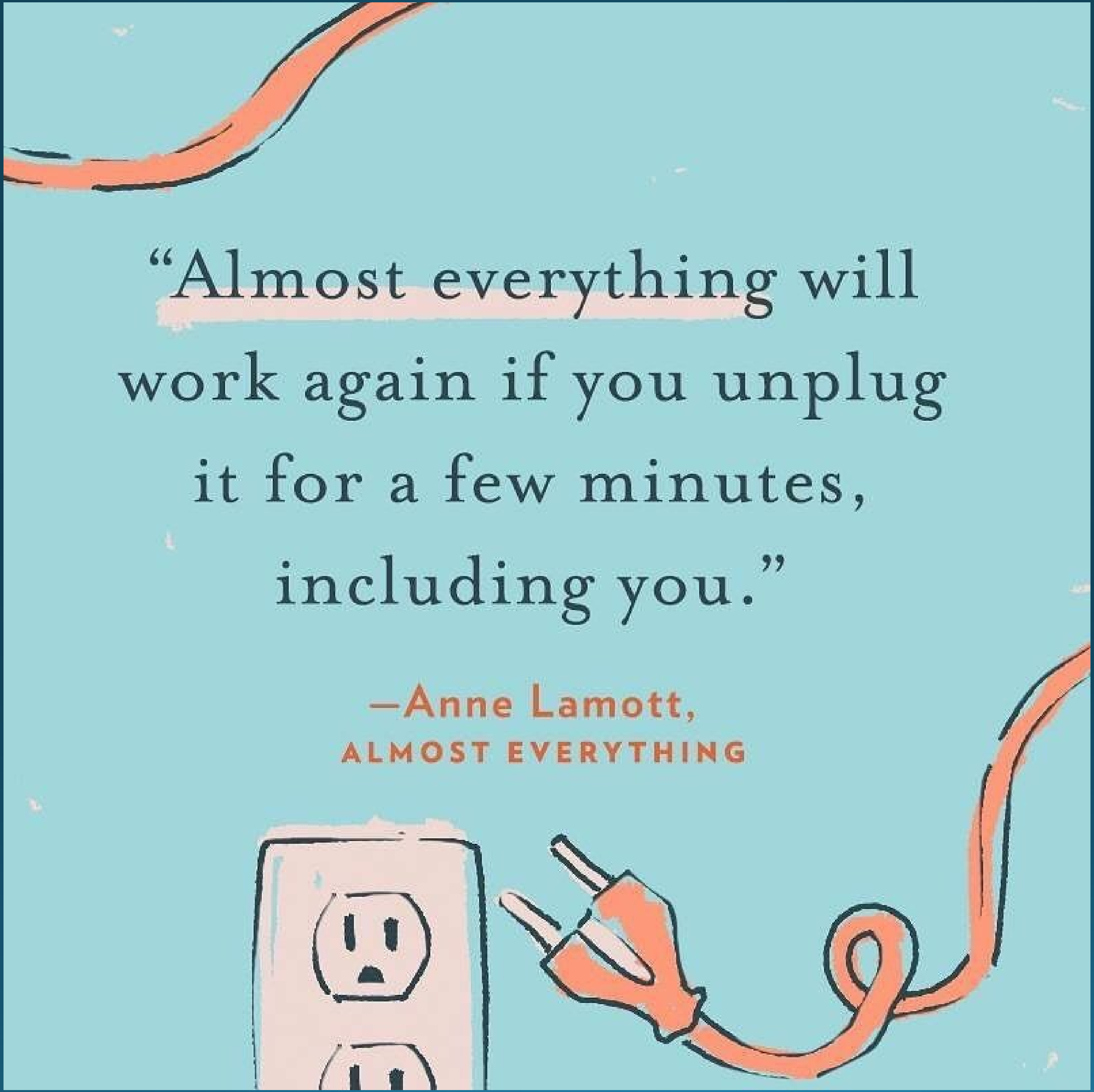
Functions

HTTP 5122

Front-End Web Development



"Almost everything will work again if you unplug it for a few minutes, including you."

—Anne Lamott, ALMOST EVERYTHING

Functions

Key Concepts

- A function is a block of code that waits until it is called to execute.
- Functions can have variables (called ‘parameters’) to use only within the function.
- Functions can return one value at the end of their logical operations.
- Functions can be used anywhere in our logic, and can be called from inside other functions.
- Variables declared inside a function cannot be "seen" outside that function. Where a variable can be seen is referred to as scope.

Blocks of code.

```
if( thisThing === true ) {
    //Run this code if condition is
    true.
}
```

Blocks of code.

```
if( thisThing === true ) {
    //Run this code if condition is
    true.
} else {
    //Run this code if not true
}
```

Blocks of code.

```
switch( thingToCompare ) {
    case “test”:
        //Code to run for a match
        break;
}
```

Blocks of code.

```
for( var i = 0; i < 5; i++ ) {
    //Run this code with each
    iteration of the loop.
}
```

Blocks of code.

```
while( thisThingIsTrue === true ){
   //Run this code until the
   running condition is not true.
}
```

A function is a block of code that waits until it is called to execute.

```
function doThis() {
    Code to run when this function
    is called.
}
```

Calling a function…

```
doThis();
doThis();
```

Functions can have parameters that act like variables within the function.

```
function doThis(one, two) {
    var z = one + two;
    alert(z);
}
```

Calling a function with parameters…

```
doThis(num1, num2);
```

Functions can return one value at the end of their logical operations.

```
function doThis(one, two) {
    var z = one + two;
    return z;
}
```

Calling a function with parameters & handling the return value…

```
var result = doThis(num1, num2);
```

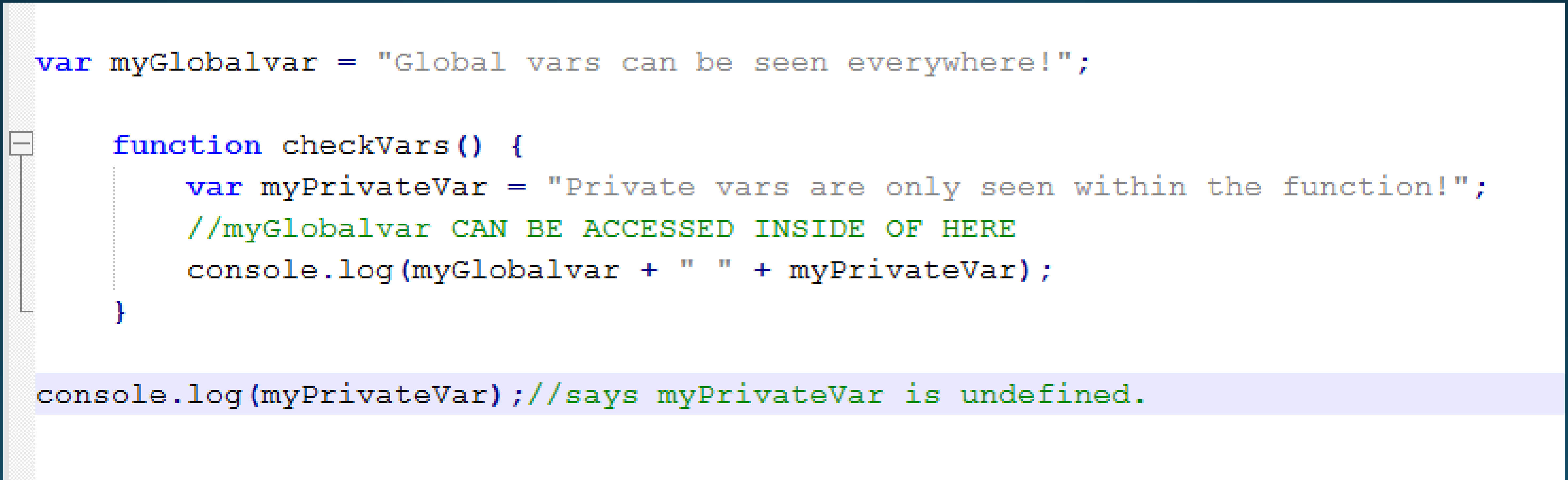
Functions can be used anywhere in our logic, and can be called from inside other functions.

```
if ( doThis(one, two) > 0 ) {



function getTotal(subtotal){
    var total = addTax(subtotal);
```

Variables declared inside a function cannot be "seen" outside of that function. Where a variable can be seen is referred to as scope.



```javascript
var myGlobalvar = "Global vars can be seen everywhere!";

    function checkVars() {
        var myPrivateVar = "Private vars are only seen within the function!";
        //myGlobalvar CAN BE ACCESSED INSIDE OF HERE
        console.log(myGlobalvar + " " + myPrivateVar);
    }

console.log(myPrivateVar);//says myPrivateVar is undefined.
```

Functions

HTTP 5122

Front-End Web Development