

JAVASCRIPT: MORE ON FUNCTIONS

PLAN FOR TODAY

- More on functions in JavaScript
- Example: Message passing
- Activity: Make similar extension groups
- Break, Hannah will make work groups
- Get into your groups
- Work time

FUNCTIONS

There are three ways to define a function:

- Function declaration
- Function expression
- Arrow function

Functions can also be *named* or *anonymous*.

FUNCTION DECLARATION

Function declarations are used to create named functions.

```
// Defining the function:  
function sum(num1, num2) {  
    return num1 + num2;  
}
```

```
// Calling the function:  
sum(3, 6); // 9
```

FUNCTION EXPRESSION

Function expressions create functions inside an expression

```
// Here, a function is assigned to a variable  
const dog = function () {  
  return "Woof!";  
};  
  
console.log(dog()); // Logs "woof"
```

ARROW FUNCTIONS

- Newer kind of function expression (introduced ES6)
- Super common due to their compact syntax
- Does not require the `function` keyword
- Uses a fat arrow `=>` to separate parameters from the body
- They look weird at first, but they are ubiquitous

EXAMPLE ARROW FUNCTIONS

```
// Arrow function with two parameters  
const sum = (firstParam, secondParam) => {  
  return firstParam + secondParam;  
};  
console.log(sum(2, 5)); // Prints: 7  
  
// Arrow function with no parameters  
const printHello = () => {  
  console.log("hello");  
};  
printHello(); // Prints: hello
```

SINGLE-PARAMETER ARROW FUNCTIONS

```
const checkWeight = weight => {  
  console.log(`Baggage weight : ${weight} kilograms.`);  
};  
  
checkWeight(25); // Prints: Baggage weight : 25 kilograms.
```

We don't require () around the parameter list with only one parameter

CONCISE ARROW FUNCTIONS

```
// Concise arrow functions  
const multiply = (a, b) => a * b;  
console.log(multiply(2, 30)); // Prints: 60
```

Arrow functions with a single expression can omit the
return keyword

ANONYMOUS FUNCTIONS

Anonymous functions are functions that are defined without a name

```
// Named function
function rocketToMars() {
  return "BOOM!";
}

// An anonymous function assigned to a variable
const rocketToMars = function () {
  return "BOOM!";
};

// An anonymous function that is immediately invoked after
  declaration
(function () {
  console.log('Immediately invoked function execution');
})();
```

ANONYMOUS FUNCTIONS AS FUNCTION ARGUMENTS

```
// This function accepts a function parameter, calls it, and logs  
// its return value  
function printResult(func) {  
  console.log(func())  
}  
  
// Here we pass in an anonymous function that returns a string  
printResult(() => {  
  return "hello world"  
})
```

You will see this syntax everywhere!

MESSAGE PASSING EXAMPLE: BACKGROUND COLORIZER

see [extension-toolkit](#)

EXTENSION WORK GROUPS

- In the [google sheet](#), create a row with your name and extension idea
- Hannah will make groups during break
- Get into your groups and share:
 - What have you tried so far?
 - What is working?
 - What isn't working?

WORK TIME!

