

Day 4: Arrows

JavaScript ES6 Arrows

Arrows are a great way to make your code expressive and easier on the eyes.
Here's some useful videos related to the topic:

Arrows

An *arrow function* expression (also known as a *fat arrow* function) has a shorter syntax compared to function expressions. Arrow functions are always [anonymous](#).

SAMPLE CODE

```
var digits = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9];
var even_digits = digits.filter( num => num % 2 === 0 );
console.log(even_digits); // [ 0, 2, 4, 6, 8 ]

var names = [ "Hacker", "Rank", "Website" ];
var names_length = names.map( name => name.length );
console.log(names_length); // [ 6, 4, 7 ]
```

OUTPUT

```
[ 0, 2, 4, 6, 8 ]
[ 6, 4, 7 ]
```

Arrows share the same lexical *this* as their surrounding code.

SAMPLE CODE

```
var employee = {
  _name : "Vineet",
  _task : ["test the beta.", "debug the code.", "create challenges."],
  _printTask() {
    this._task.forEach( t => console.log(this._name + " has to " + t));
  }
};

employee._printTask();
```

OUTPUT

Vineet has to test the beta.
Vineet has to debug the code.
Vineet has to create challenges.

Task

You are given a variable, `my_function`. Your task is to assign it with an arrow function.

The `my_function` should take an array as its parameter and return an array with all its *even* elements incremented by \$1\$, and *odd* elements decremented by \$1\$.

Note

- **DON'T** use `function` instead of an arrow function.
- **DON'T** print anything on the console.
- Replace the blank (`_____`) with an arrow function.
- The name of the array parameter can be anything. For example, `some_array`.