Functions

A function is a block of JavaScript code that is defined once but may be executed, or invoked, any number of times.

Function declaration and invocation

```
function print(value) {
    console.log(value);
}

function isEmail(email) {
    // Do validation
    return true/false;
}
Print("hello");
Var isValidEmail = isEmail("a");
```

Local and outer variables

```
var name = "poghos";//outer
function showInfo() {
    var message = "hello, " + name;// local
    console.log(message);
}
```

Default parameters

- es5
  - function showMessage(from, text) {
    ```
    text = text || "No text given";
    console.log(from + ": " + text);
    ```
    }
- Es6
  - function showMessage(from, text = "No text given") {
    ```
    console.log(from + ": " + text);
    ```
    }

Naming function

- "get…" – return a value,
- "calc…" – calculate something,
- "create…" – create something,
- "check…" – check something and return a boolean, etc,
- "is…" – predicate.*

* A predicate is a box that takes an argument and returns a Boolean value.

Function expression and invaction

```
var print = function (value) {
    console.log(value);
}

Print("hello");
```

Nested functions

```
function showPrimes(n) {
    function isPrime(n) {
        for (let i = 2; i < n; i++) {
            if ( n % i == 0) return false;
        }
        return true;
    }

    for (let i = 2; i < n; i++) {
        if (!isPrime(i)) {
            continue;
        }

        alert(i);  // a prime
    }
}
```

Callback functions

```
function ask(question, yes, no) {
  if (confirm(question)) {
      yes();
  } else {
      no();
  }
}


function showOk() {
  alert( "You agreed." );
}


function showCancel() {
  alert( "You canceled the execution." );
}

ask("Do you agree?", showOk, showCancel);
```

Arguments object

- Arguments is array like object
- function max() {

```
        var max = Number.NEGATIVE_INFINITY;
        for(var i = 0; i < arguments.length; i++) {
                if (arguments[i] > max) {
                        max = arguments[i];
                }
        }
        return max;
}
var largest = max(1, 10, 100, 2, 3, 1000, 4, 5, 10000, 6);
```

Es5 array methods

- forEach()
  ```
  var data = [1,2,3,4,5];
  var sum = 0;
  data.forEach(function(value, index, array) { sum += value; });
  ```

- map()
  ```
  var arr = [1, 2, 3];
  var mapedArr = arr.map(function(x) { return x*x; })
  ```

- filter()
  ```
  var arr = [1, 2, 3];
  var evenNumbers = arr.filter(function(x,i) { return x%2==0 });
  ```

- every()
  ```
  var arr = [1, 2, 3];
  var isLess = arr.every(function(x) { return x < 10 });
  ```

Es5 array methods

- **some()**
  var arr = [1, 2, 3, 15];
  var isExistsGreater = arr.some(function(x) { return x > 10 });

- **reduce()**
  var arr = [1, 2, 3];
  var sum = arr.reduce(function(x, y) { return x+y; }, 0)

- **reduceRight()**
  var arr = [1, 2, 3];
  var sum = arr. reduceRight(function(x, y) { return x+y; }, 0)

Any questions?

Thanks