Tips in JS

Arrow functions

`=>` called the anonymous function expression, it would **automatically return the result** (without `{}`), like lambda expression in Python.

```
hello = function() {
  return "Hello World!";
}
hello = () => {
  return "Hello World!";
}

hello = () => "Hello World";

concatenateTwoWords = (w1, w2) => w1 + " " + w2;
console.log(concatenateTwoWords("Hello", "World")); // Hello World

var func = x => x * x;
// concise body syntax, implied "return"

var func = (x, y) => { return x + y; };
// with block body, explicit "return" needed
```

`Var` and `Let`

```
function varTest() {
  var x = 1;
  {
    var x = 2;  // same variable!
    console.log(x);  // 2
  }
  console.log(x);  // 2
}
```

```
function varAndLet() {
  var x = 1;
  {
    let x = 2;
    console.log(x); // 2
  }
  console.log(x); // 1
}
```

```
function letTest() {
  let x = 1;
  {
    let x = 2;  // different variable
    console.log(x);  // 2
  }
  console.log(x);  // 1
}
```

```
function varAndLet() {
  let x = 1;
  {
    var x = 2;
    console.log(x); // Error
  }
  console.log(x); // 1
}
```

Call back

The purpose of call back function is **ensuring the call back function is executed after** the main function complete.

```
function first(){
 // Simulate a code delay
 setTimeout( function(){
  console.log(1);
 }, 500 );
}
function second(){
 console.log(2);
}
first();
second();
```

similar to `first(second);`

```
function first(callback) {
  setTimeout(function() {
    console.log(1);
    callback();
  }, 500);
}
first(function() {
  console.log(2);
});
```

```
function doHomework(subject, callback) {
 alert(`Starting my ${subject} homework.`);
 callback();
}

doHomework('math', function() {
 alert('Finished my homework');
});
```

Object Oriented Programming

Constructor

Use the capitalized letter to represent the constructor instead a normal function

```
function greetFunction() {
  console.log(`Hi, I am ${this.first_name} ${this.last_name}.`);
}

function Person(first_name, last_name) {
  this.first_name = first_name;
  this.last_name = last_name;
  this.greet = greetFunction;
};

var person = new Person("David", "Wang");
person.greet(); // Hi, I am David Wang.
```

Window and Navigator object

Get current url: window.location.href

Go back to previous page: window.history.back()

```
console.log('width: ' + window.innerWidth + ', height: ' + window.innerHeight);


console.log('appName = ' + navigator.appName);
console.log('appVersion = ' + navigator.appVersion);
console.log('language = ' + navigator.language);
console.log('platform = ' + navigator.platform);
console.log('userAgent = ' + navigator.userAgent);

window.history.back();
window.location.href;
```