

BCPR280

Software Engineering 2

Dr Luofeng Xu

Lesson 03

Introduction to JavaScript

- Link JavaScript to a HTML5 webpage
 - [Linkingjsandhtml.pdf](#)
- Demo
 - [lightbulb](#)
- Exercise
 - [displaydate](#)

Introduction to JavaScript

- Hoisting
 - http://www.w3schools.com/js/js_hoisting.asp
 - JavaScript default behavior of moving declarations to the top of the current scope.
 - Variable & function declarations
 - But not function expression

Homework

- Cards1 & cards2
- Read and practice with JavaScript functions
 - http://www.w3schools.com/js/js_function_definition.asp

Lesson 04

Notices

- <http://jsfiddle.net/>

Introduction to JavaScript

- array.length isn't necessarily the number of items in the array.
 - `var a = ["dog", "cat", "hen"];`
 - `a[100] = "fox";`
 - `a.length;` `// 101`
 - `typeof(a[90])` `// undefined`

Introduction to JavaScript

- A better way to iterate over an array
 - `for (var i = 0, item; item = a[i++];) {`
 - `// Do something with item`
 - `}`
- When a contains "falsy" values,
 - `for (var i = 0, len = a.length; i < len; i++) {`
 - `// Do something with a[i]`
 - `}`

- Emulating private methods with closures
 - <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Closures>

Functions

- http://www.w3schools.com/js/js_function_definition.asp
- https://developer.mozilla.org/en-US/docs/Web/JavaScript/A_re-introduction_to_JavaScript
- function declaration
- function expression
- function construction
- Self-invoking function
- Closure
 - http://www.w3schools.com/js/js_function_closures.asp
 - <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Closures>
- arguments object
 - argumentsObjectDemo.js
- Inner function

Functions

- Demo
 - jsFunctionDemo.js

Objects in JavaScript

- https://developer.mozilla.org/en-US/docs/Web/JavaScript/Introduction_to_Object-Oriented_JavaScript
- https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/A_re-introduction_to_JavaScript
- JavaScript does not have class statement; instead, the class functionality is accomplished by object prototypes.
- JavaScript objects can be thought of as simple collections of name-value pairs.
- Demo
 - jsObjectDemo.js

Prototype

- http://www.w3schools.com/js/js_object_prototypes.asp
- Every JavaScript object has a prototype. The prototype is also an object.
- All JavaScript objects inherit their properties and methods from their prototype.
- Demo
 - jsObjectDemo.js

Relationships

- Inheritance
 - https://developer.mozilla.org/en-US/docs/Web/JavaScript/Inheritance_and_the_prototype_chain
- Composition
- Aggression
- Demo
 - `objectRelationshipDemo.js`

Homework

- Research on how to use professional version control tools, e.g., GitHub
- Learn and practice on JavaScript object, prototype and relationships