

## HTML

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <title>Section 5: Advanced JavaScript: Objects and Functions</title>
  </head>

  <body>
    <h1>Section 5: Advanced JavaScript: Objects and Functions</h1>
    <script src="script.js"></script>
  </body>
</html>
```

## JS

### **// Lecture: Function constructor**

```
var john = {
  name: 'John',
  yearOfBirth: 1990,
  job: 'teacher'
};

var Person = function(name, yearOfBirth, job) {
  this.name = name;
  this.yearOfBirth = yearOfBirth;
  this.job = job;
}

Person.prototype.calculateAge = function() {
  console.log(2016 - this.yearOfBirth);
};

Person.prototype.lastName = 'Smith';

var john = new Person('John', 1990, 'teacher');
var jane = new Person('Jane', 1969, 'designer');
var mark = new Person('Mark', 1948, 'retired');
```

```
john.calculateAge();
jane.calculateAge();
mark.calculateAge();

console.log(john.lastName);
console.log(jane.lastName);
console.log(mark.lastName);
```

## **// Lecture: Object.create**

```
var personProto = {
  calculateAge: function() {
    console.log(2016 - this.yearOfBirth);
  }
};
```

```
var john = Object.create(personProto);
john.name = 'John';
john.yearOfBirth = 1990;
john.job = 'teacher';
```

```
var jane = Object.create(personProto, {
  name: { value: 'Jane' },
  yearOfBirth: { value: 1969 },
  job: { value: 'designer' }
});
*/
```

## **// Lecture: Primitives vs objects**

### **// Primitives**

```
var a = 23;
var b = a;
a = 46;
console.log(a);
console.log(b);
```

## **// Objects**

```
var obj1 = {  
  name: 'John',  
  age: 26  
};  
var obj2 = obj1;  
obj1.age = 30;  
console.log(obj1.age);  
console.log(obj2.age);
```

// Functions

```
var age = 27;  
var obj = {  
  name: 'Jonas',  
  city: 'Lisbon'  
};
```

```
function change(a, b) {  
  a = 30;  
  b.city = 'San Francisco';  
}
```

```
change(age, obj);
```

```
console.log(age);  
console.log(obj.city);
```

## **// Lecture: Passing functions as arguments**

```
var years = [1990, 1965, 1937, 2005, 1998];
```

```
function arrayCalc(arr, fn) {  
  var arrRes = [];  
  for (var i = 0; i < arr.length; i++) {  
    arrRes.push(fn(arr[i]));  
  }  
  return arrRes;  
}
```

```
function calculateAge(el) {
```

```

    return 2016 - el;
}

function isFullAge(el) {
    return el >= 18;
}

function maxHeartRate(el) {
    if (el >= 18 && el <= 81) {
        return Math.round(206.9 - (0.67 * el));
    } else {
        return -1;
    }
}

var ages = arrayCalc(years, calculateAge);
var fullAges = arrayCalc(ages, isFullAge);
var rates = arrayCalc(ages, maxHeartRate);

console.log(ages);
console.log(rates);

```

## **// Lecture: Functions returning functions**

```

function interviewQuestion(job) {
    if (job === 'designer') {
        return function(name) {
            console.log(name + ', can you please explain what UX design is?');
        }
    } else if (job === 'teacher') {
        return function(name) {
            console.log("What subject do you teach, ' + name + '?'");
        }
    } else {
        return function(name) {
            console.log('Hello ' + name + ', what do you do?');
        }
    }
}

```

```
var teacherQuestion = interviewQuestion('teacher');
var designerQuestion = interviewQuestion('designer');
```

```
teacherQuestion('John');
designerQuestion('John');
designerQuestion('jane');
designerQuestion('Mark');
designerQuestion('Mike');
```

```
interviewQuestion('teacher')('Mark');
```

## **// Lecture: IIFE**

```
function game() {
  var score = Math.random() * 10;
  console.log(score >= 5);
}
game();
```

```
(function () {
  var score = Math.random() * 10;
  console.log(score >= 5);
})();
```

```
//console.log(score);
```

```
(function (goodLuck) {
  var score = Math.random() * 10;
  console.log(score >= 5 - goodLuck);
})(5);
```

## **// Lecture: Closures**

```
function retirement(retirementAge) {
  var a = ' years left until retirement.';
  return function(yearOfBirth) {
    var age = 2016 - yearOfBirth;
    console.log((retirementAge - age) + a);
```

```
}  
}
```

```
var retirementUS = retirement(66);  
var retirementGermany = retirement(65);  
var retirementIceland = retirement(67);
```

```
retirementGermany(1990);  
retirementUS(1990);  
retirementIceland(1990);
```

```
//retirement(66)(1990);
```

```
function interviewQuestion(job) {  
  return function(name) {  
    if (job === 'designer') {  
      console.log(name + ', can you please explain what UX design is?');  
    } else if (job === 'teacher') {  
      console.log('What subject do you teach, ' + name + '?');  
    } else {  
      console.log('Hello ' + name + ', what do you do?');  
    }  
  }  
}
```

```
interviewQuestion('teacher')('John');
```

## **// Lecture: Bind, call and apply**

```
var john = {  
  name: 'John',  
  age: 26,  
  job: 'teacher',  
  presentation: function(style, timeOfDay) {  
    if (style === 'formal') {  
      console.log('Good ' + timeOfDay + ', Ladies and gentlemen! I\'m ' + this.name + ', I\'m a '  
+ this.job + ' and I\'m ' + this.age + ' years old.');
```

```
    }  
  }  
};
```

```
var emily = {  
  name: 'Emily',  
  age: 35,  
  job: 'designer'  
};
```

```
john.presentation('formal', 'morning');
```

```
john.presentation.call(emily, 'friendly', 'afternoon');
```

```
//john.presentation.apply(emily, ['friendly', 'afternoon']);
```

```
var johnFriendly = john.presentation.bind(john, 'friendly');
```

```
johnFriendly('morning');  
johnFriendly('night');
```

```
var emilyFormal = john.presentation.bind(emily, 'formal');  
emilyFormal('afternoon');
```

## **// Another cool example**

```
var years = [1990, 1965, 1937, 2005, 1998];
```

```
function arrayCalc(arr, fn) {  
  var arrRes = [];  
  for (var i = 0; i < arr.length; i++) {  
    arrRes.push(fn(arr[i]));  
  }  
  return arrRes;  
}
```

```
function calculateAge(el) {  
  return 2016 - el;  
}
```

```
function isFullAge(limit, el) {  
  return el >= limit;
```

```
}
```

```
var ages = arrayCalc(years, calculateAge);  
var fullJapan = arrayCalc(ages, isFullAge.bind(this, 20));  
console.log(ages);  
console.log(fullJapan);
```

## // CODING CHALLENGE

--- Let's build a fun quiz game in the console! ---

1. Build a function constructor called Question to describe a question. A question should include:
  - a) question itself
  - b) the answers from which the player can choose the correct one (choose an adequate data structure here, array, object, etc.)
  - c) correct answer (I would use a number for this)
2. Create a couple of questions using the constructor
3. Store them all inside an array
4. Select one random question and log it on the console, together with the possible answers (each question should have a number) (Hint: write a method for the Question objects for this task).
5. Use the 'prompt' function to ask the user for the correct answer. The user should input the number of the correct answer such as you displayed it on Task 4.
6. Check if the answer is correct and print to the console whether the answer is correct or not (Hint: write another method for this).
7. Suppose this code would be a plugin for other programmers to use in their code. So make sure that all your code is private and doesn't interfere with the other programmers code (Hint: we learned a special technique to do exactly that).



```

(function() {
  function Question(question, answers, correct) {
    this.question = question;
    this.answers = answers;
    this.correct = correct;
  }

  Question.prototype.displayQuestion = function() {
    console.log(this.question);

    for (var i = 0; i < this.answers.length; i++) {
      console.log(i + ': ' + this.answers[i]);
    }
  }

  Question.prototype.checkAnswer = function(ans) {
    if (ans === this.correct) {
      console.log('Correct answer!');

    } else {
      console.log('Wrong answer. Try again :)')
    }
  }

  var q1 = new Question('Is JavaScript the coolest programming language in the world?',
    ['Yes', 'No'],
    0);

  var q2 = new Question('What is the name of this course\'s teacher?',
    ['John', 'Micheal', 'Jonas'],
    2);

  var q3 = new Question('What does best describe coding?',
    ['Boring', 'Hard', 'Fun', 'Tediuous'],
    2);

  var questions = [q1, q2, q3];

  var n = Math.floor(Math.random() * questions.length);

  questions[n].displayQuestion();

  var answer = parseInt(prompt('Please select the correct answer.'));

```

```
questions[n].checkAnswer(answer);
})();
```

### --- Expert level ---

**8. After you display the result, display the next random question, so that the game never ends (Hint: write a function for this and call it right after displaying the result)**

**9. Be careful: after Task 8, the game literally never ends. So include the option to quit the game if the user writes 'exit' instead of the answer. In this case, DON'T call the function from task 8.**

**10. Track the user's score to make the game more fun! So each time an answer is correct, add 1 point to the score (Hint: I'm going to use the power of closures for this, but you don't have to, just do this with the tools you feel more comfortable at this point).**

**11. Display the score in the console. Use yet another method for this.**

```
(function() {
  function Question(question, answers, correct) {
    this.question = question;
    this.answers = answers;
    this.correct = correct;
  }

  Question.prototype.displayQuestion = function() {
    console.log(this.question);

    for (var i = 0; i < this.answers.length; i++) {
      console.log(i + ': ' + this.answers[i]);
    }
  }

  Question.prototype.checkAnswer = function(ans, callback) {
    var sc;

    if (ans === this.correct) {
      console.log('Correct answer!');
      sc = callback(true);
    } else {
      console.log('Wrong answer. Try again :');
      sc = callback(false);
    }
  }
})();
```

```

    }

    this.displayScore(sc);
}

Question.prototype.displayScore = function(score) {
    console.log('Your current score is: ' + score);
    console.log('-----');
}

var q1 = new Question('Is JavaScript the coolest programming language in the world?',
    ['Yes', 'No'],
    0);

var q2 = new Question('What is the name of this course\'s teacher?',
    ['John', 'Micheal', 'Jonas'],
    2);

var q3 = new Question('What does best describe coding?',
    ['Boring', 'Hard', 'Fun', 'Tediuous'],
    2);

var questions = [q1, q2, q3];

function score() {
    var sc = 0;
    return function(correct) {
        if (correct) {
            sc++;
        }
        return sc;
    }
}

var keepScore = score();

function nextQuestion() {

    var n = Math.floor(Math.random() * questions.length);
    questions[n].displayQuestion();

    var answer = prompt('Please select the correct answer.');
```

```
    if(answer !== 'exit') {  
        questions[n].checkAnswer(parseInt(answer), keepScore);  
        nextQuestion();  
    }  
}  
  
nextQuestion();  
  
})();
```