Opdrachten – dag 1

introduction to JavaScript - comparisons with the logical Or Operator

Quoting Strings with Single Quotes

```
var myStr = "<a href=\"http://www.example.com\" target=\"_blank\">Link</a>"
;
```

→

```
var myStr = '<a href="http://www.example.com" target="_blank">Link</a>';
```

Escape Sequences in Strings

```
var myStr = 'FirstLine\n\t\\SecondLine\nThirdLine';
```

```
// myStr:
FirstLine
        \SecondLine
ThirdLine
```

Opdrachten – dag 2 en 4

Write reusable JavaScript - Assignment with a Returned value

Accessing Object Properties with Dot Notation

```
// Setup
var testObj = {
  "hat": "ballcap",
  "shirt": "jersey",
  "shoes": "cleats"
};

// Only change code below this line

var hatValue = testObj.hat;      // Change this line
var shirtValue = testObj.shirt;    // Change this line
```

Accessing Object Properties with Bracket Notation

```
// Setup
var testObj = {
  "an entree": "hamburger",
  "my side": "veggies",
  "the drink": "water"
};

// Only change code below this line

var entreeValue = testObj["an entree"];   // Change this line
var drinkValue = testObj["the drink"];    // Change this line
```

Accessing Object Properties with Variables

```
// Setup
var testObj = {
  12: "Namath",
  16: "Montana",
  19: "Unitas"
};

// Only change code below this line

var playerNumber = 16;       // Change this line
var player = testObj[playerNumber];   // Change this line
```

Using Objects for Lookups

```
// Setup
function phoneticLookup(val) {
  var result = "";

  // Only change code below this line
    var lookup = {
    "alpha":"Adams",
    "bravo":"Boston",
    "charlie":"Chicago",
    "delta":"Denver",
    "echo":"Easy",
    "foxtrot":"Frank",
  };
  return lookup[val];

  // Only change code above this line
  return result;
}

phoneticLookup("charlie");
```

Testing Objects for Properties

```
function checkObj(obj, checkProp) {
  // Only change code below this line
  if(obj.hasOwnProperty(checkProp)) {
    return obj[checkProp];
  }
  else {
    return "Not Found";
  }
  // Only change code above this line
}
```

Manipulating Complex Objects

Let op: You will need to place a comma after every object in the array, unless it is the last object in the array

```
var myMusic = [
  {
    "artist": "Billy Joel",
    "title": "Piano Man",
    "release_year": 1973,
    "formats": [
      "CD",
      "8T",
      "LP"
    ],
    "gold": true
  },
  {
    "artist": "Pearl Jam",
    "title": "Yield",
    "release_year": 1998,
    "formats": [
      "CD",
      "8T",
      "LP",
      "iTunes"
    ]
  }
];
```

Opdrachten - dag 3

Build JavaScript objects - Record Collection

Accessing Object Properties with Dot Notation

```
// Setup
var testObj = {
  "hat": "ballcap",
  "shirt": "jersey",
  "shoes": "cleats"
};

// Only change code below this line

var hatValue = testObj.hat;      // Change this line
var shirtValue = testObj.shirt;    // Change this line
```

Accessing Object Properties with Bracket Notation

```
// Setup
var testObj = {
  "an entree": "hamburger",
  "my side": "veggies",
  "the drink": "water"
};

// Only change code below this line

var entreeValue = testObj["an entree"];   // Change this line
var drinkValue = testObj["the drink"];    // Change this line
```

Accessing Object Properties with Variables

```
// Setup
var testObj = {
  12: "Namath",
  16: "Montana",
  19: "Unitas"
};

// Only change code below this line

var playerNumber = 16;       // Change this line
var player = testObj[playerNumber];   // Change this line
```

Using Objects for Lookups

```
// Setup
function phoneticLookup(val) {
  var result = "";

  // Only change code below this line
    var lookup = {
    "alpha":"Adams",
    "bravo":"Boston",
    "charlie":"Chicago",
    "delta":"Denver",
    "echo":"Easy",
    "foxtrot":"Frank",
  };
  return lookup[val];

  // Only change code above this line
  return result;
}

phoneticLookup("charlie");
```

Testing Objects for Properties

```
function checkObj(obj, checkProp) {
  // Only change code below this line
  if(obj.hasOwnProperty(checkProp)) {
    return obj[checkProp];
  }
  else {
    return "Not Found";
  }
  // Only change code above this line
}
```

Manipulating Complex Objects

Let op: You will need to place a comma after every object in the array, unless it is the last object in the array

```
var myMusic = [
  {
    "artist": "Billy Joel",
    "title": "Piano Man",
    "release_year": 1973,
    "formats": [
      "CD",
      "8T",
      "LP"
    ],
    "gold": true
  },
  {
    "artist": "Pearl Jam",
    "title": "Yield",
    "release_year": 1998,
    "formats": [
      "CD",
      "8T",
      "LP",
      "iTunes"
    ]
  }
];
```

iterate - profile lookup

Iterate with JavaScript While Loops

The first type of loop we will learn is called a `while` loop because it runs "while" a specified condition is true and stops once that condition is no longer true.

```
// Setup
var myArray = [];

// Only change code below this line
var i = 5;
while(i >= 0) {
  myArray.push(i);
  i--;
}
```