Objects, JSON and DOM Events

SoftUni Team

Technical Trainers

Software University

SoftUni Foundation

Software University

http://softuni.bg

Table of Content

Have a Question?

SoftUni Foundation

sli.do

#JS-CORE

DOM Events

JavaScript reacting to HTML Events

Reacting to Events

- A JavaScript can be executed when an **event occurs**, like when a user clicks on an HTML element.
- To execute code when a user clicks on an element, **add JavaScript code** to an **HTML event attribute**:

```
onclick=JavaScript
```

- Examples of HTML events:
  - When a user clicks the mouse
  - When the mouse moves over an element
  - When an HTML form is submitted

Example

- In this example, a function is called from the event handler:

```
<!DOCTYPE html>
<html>
<body>
<h1 onclick="changeText(this)">Click on this text!</h1>
</body>
</html>
```

```
function changeText(id) {
    id.innerHTML = "Ooops!";
}
```

Event Attributes

- To assign events to HTML elements you can use event attributes.

```
<button onclick="displayDate()">Try it</button>
```

- The HTML DOM allows you to assign events to HTML elements using JavaScript:

```
<script>
document.getElementById("myBtn").onclick = displayDate;
</script>
```

Onmouseover and Onmouseout

- The onmouseover and onmouseout events can be used to trigger a function when the user mouses over, or out of, an HTML element:

```
<!DOCTYPE html>
<html>
<body>
<div onmouseover="mOver(this)" onmouseout="mOut(this)">
Mouse Over Me</div>
</body>
</html>
// TODO: write the functions to change the innerHTML
```

Objects

Collection of key-value pairs

What is an Object?

- You have already learned that JavaScript variables are containers for data values.
- This code assigns a **simple value** (Fiat) to a **variable** named car:

```
let car = "Fiat";
```

- Objects are variables too. But objects can contain many values.
- This code assigns **many values** (Fiat, 500, white) to a **variable** named car:

```
let car = {type:"Fiat", model:"500", color:"white"};
```

Defining and Object

- You define (and create) a JavaScript object with an object literal:

```
let person = {firstName:"John", lastName:"Doe", age:50};
```

- Spaces and line breaks are not important. An object definition can span multiple lines:

```
let person = {
    firstName:"John",
    lastName:"Doe",
    age:50
};
```

Object Properties

- The **name:values** pairs in JavaScript objects are called **properties**:

Property	Property Value
firstName	John
lastName	Doe
age	50

- You can access object properties in two ways:

```
let lastName = person.lastName; // John
let age = person["age"];        // 50
```

Object Methods

- Objects can also have methods.
- Methods are actions that can be performed on objects.
- Methods are stored in properties as function definitions.

```
let person = {
    firstName: "John",
    lastName : "Doe",
    fullName : function() {
        return this.firstName + " " + this.lastName;
    }
};
```

What to avoid

- When a JavaScript variable is declared with the keyword **"new"**, the variable is created as an object:

```
let x = new String();        // Declares x as a String object
let y = new Number();        // Declares y as a Number object
let z = new Boolean();       // Declares z as a Boolean object
```

- Avoid String, Number, and Boolean objects. They **complicate** your code and **slow down execution** speed.

JSON

JavaScript Object Notation

What is a JSON?

SoftUni Foundation

- JSON:
  - Stands for **J**ava**S**cript **O**bject **N**otation
  - It a **lightweight data** interchange **format**
  - Is **language independent** - syntax is derived from JavaScript object notation syntax, but the JSON format is text only
  - Is **"self-describing"** and easy to understand

Example

- This **JSON** syntax defines an employees object: an array of 3 employee records (objects):

```
{
"employees":[
    {"firstName":"John", "lastName":"Doe"},
    {"firstName":"Anna", "lastName":"Smith"},
    {"firstName":"Peter", "lastName":"Jones"}
]
}
```

Syntax rules

- In JSON:
  - Data is in **name/value** pairs
  - Data is **separated by commas**
  - **Curly braces** hold **objects**
  - **Square brackets** hold **arrays**

```
{
"employees":[{"firstName":"John" , "lastName":"Doe"}]
}
```

Parsing from strings

- A common use of JSON is to read data from a web server, and display the data in a web page.
- For simplicity, this can be demonstrated using a string as input.

```
let text = '{ "employees" : [' +
'{ "firstName":"John" , "lastName":"Doe" },' +
'{ "firstName":"Peter" , "lastName":"Jones" } ]}';
```

- Then, use the JavaScript built-in function JSON.parse() to convert the string into a JavaScript object:

```
var obj = JSON.parse(text);
```

Parsing from strings (2)

- Finally, use the new JavaScript object in your page:

```
<p id="demo"></p>
<script>
document.getElementById("demo").innerHTML =
obj.employees[1].firstName + " " + obj.employees[1].lastName;
</script>
```

Converting to string

- Use JSON.**stringify** to convert objects into a string:

```
let obj = { name: "John", age: 30, city: "New York" };
let myJSON = JSON.stringify(obj);
console.log(myJSON)
// {"name":"John","age":30,"city":"New York"}
```

- You can do the same for **arrays**

```
let arr = [ "John", "Peter", "Sally", "Jane" ];
let myJSON = JSON.stringify(arr);
console.log(myJSON)
// ["John","Peter","Sally","Jane"]
```

Live Exercises

Summary

- Objects are variables which can contain **many values**.
- We can access values using **.** or **[]**
- JSON is a lightweight data interchange format
- We can use **events** to change the **behavior of the HTML**

Questions?

SoftUni

Software University | SoftUni Svetlina | SoftUni Creative | SoftUni Digital | SoftUni Foundation | SoftUni Kids

SoftUni Diamond Partners

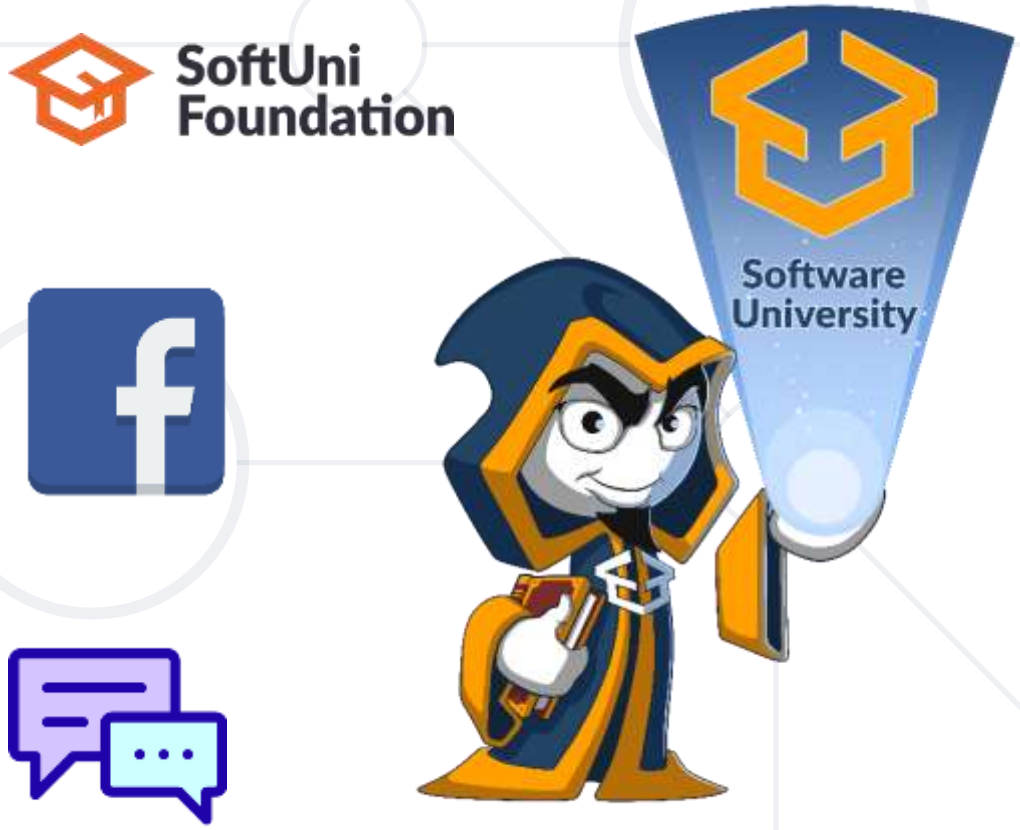
SoftUni Foundation

SoftUni Organizational Partners

SoftUni Foundation

Trainings @ Software University (SoftUni)

- Software University – High-Quality Education and Employment Opportunities
  - softuni.bg
- Software University Foundation
  - http://softuni.foundation/
- Software University @ Facebook
  - facebook.com/SoftwareUniversity
- Software University Forums
  - forum.softuni.bg



License

- This course (slides, examples, demos, videos, homework, etc.) is licensed under the "Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International" license