Javascript Forms

Web Authoring and Design

Benjamin Kenwright

Outline

- Review
- Javascript and Object Orientated Concepts
- Javascript and Forms
- Summary
- Review/Discussion

25th

December

Last Day – Quizzes

Grade Taken

Presentation/Demonstration
(Individual Project)

- 1pm – 6pm
- Room 24-401
- Presentation (slides)
- Answer questions

Question

- How can you create an Array in JavaScript?

Answer

- You can define arrays using the array literal as follows −

```
var x = [];
var y = [1, 2, 3, 4, 5];
```

Question

- What order does f receive its arguments?

Language: Javascript

```
1 f("foo");
2 setTimeout(function() { f("bar");}, 0);
3 f("baz");
```

a) foo bar baz
b) foo baz bar
c) bar foo baz
d) foo baz

Answer

b) foo baz bar

```
<html>
<head></head>
<body>

<script>

function f(ss)
{
    alert(ss);
}

f("foo");
setTimeout(function() { f("bar");}, 0);
f("baz");

</script>

</body>
</html>
```

Question

- Which of the following is correct about the expression 1==“1.0” in Javascript?

a) It returns false. The two values have different types.
b) It returns true. The two values are converted to strings and are equal.
c) It returns false. The two values are converted to strings and are not equal.
d) It returns true. The two values are converted to numbers and are equal.

Answer

b) It returns true. The two values are converted to strings and are equal.

```
<html>
<head></head>
<body>
<script>

if ( 1==“1.0” )
{
    alert("equality test");
}

</script>
</body>
</html>
```

What arguments will fn receive, in order?

Language: Javascript

```
1  for(var i = 0; i < 3; i++) {
2    setTimeout(function() {
3      fn(i);
4    }, 0);
5  }
6
7  for(let i = 0; i < 3; i++) {
8    setTimeout(function() {
9      fn(i);
10   }, 0);
11 }
```

a) `3, 3, 3, 3, 3, 3`

b) `3, 3, 3, 0, 1, 2`

c) `0, 1, 2, 3, 3, 3`

d) `0, 1, 2, 0, 1, 2`

Answer

a) 3, 3, ….

Reason:

https://stackoverflow.com/questions/5226285/settimeout-in-for-loop-does-not-print-consecutive-values

Defining functions

Function Declarations:

```
function functionName(parameters (optional))
{
        FunctionBody
}
```

Arguments

Function without arguments

```
function greetings()
{
        alert("Good morning");
}
```

Function with arguments

```
function findSum(a,b)
{
        alert("Sum is : " + (a + b));
}
```

Calling a Function

You can call Javascript functions by simply call the function name

```
function greetings()
{
        alert("Good Morning");
}
greetings(); //just give the function name on
```

Return Values

```
<script>
    function findSum(x,y)
    {
        var result = x+y;
        return result;
    }
    var sum = findSum(25,30);
    alert ("Sum is : "+ sum);
</script>
```

Question

- Is Javascript an Object Orientated Language?

a) Yes

b) No

Answer

- a) Yes

JavaScript is an object oriented programming language

How to create and use objects in Javascript

```
var student = {
        name: "Jack",
        age: 12,
        getName: function()
        {
                alert(this.name);
        }
};
```

Call Object function

```
var student = {
        name: "Jack",
        age: 12,
        getName: function()
        {
                alert(this.name);
        }
};
```

```
student.getName();
```

Object constructor

```
function studentConstructor() {
        this.name = "Jack";
        this.age = 12;
        this.getName = function()
        {
                alert(this.name);
        }
}
```

How to create a New Instance from a Constructor

```
var student = new studentConstructor();
student.getName(); // will alert "Jack"
```

Literal notation in Javascript Object

```
var student = {
        name: "Jack",
        age: 12,
        getName: function()
        {
                alert(this.name);
        }
};
student.getName(); //will alert "Jack"
```

Object.create() in Javascript Object

```
var student = {
        name: "Jack",
        age: 12,
        getName: function()
        {
                alert(this.name);
        }
};
var st = Object.create(student);
st.getName();
```

Accessing Object Properties in JavaScript

We can access Object Properties in JavaScript using either the dot(.) notation or the bracket[] notation

```
var student = {};
student.name = "Jack"; // access via . notation
student["age"] = 12; // access via [] brackets
alert(student["name"]); // will alert "Jack"
alert(student.age); // will alert 12
alert(student.name); // will alert "Jack"
alert(student["age"]); // will alert 12
```

Object contain another Object

```
var student = {
        name: "Jack",
        age: 12,
        classTeacher: {
        firstname: "John",
        lastname: "Peter"
        }
};
alert(student.classTeacher.firstname);
alert(student['classTeacher']['firstname']);
alert(student.classTeacher['firstname']);
alert(student['classTeacher'].firstname);
```

Forms
Trigger/Call Javascript

```
<!DOCTYPE html>
<html>
<body>

<p>Enter value and click "Submit" to submit the form:</p>

<form id="frm1">
  Enter value: <input type="text" name="fname"><br>
  <input type="button" onclick="myFunction()">
</form>

<script>
function myFunction() {
    //.......
}
</script>

</body>
</html>
```

This Week

- Review Slides
- Read Associated Chapters
- Work through Javascript Examples
  - ▷Update GitHub Account/Webpage
- **Group Project (Christmas Theme)**
- **Website & Javascript Game**

- **Demonstrate Individual Projects**

Summary

- Overview of Objects in Javascript
- Javascript Forms
- Hands-On/Practical

Questions/Discussion

Research (Game Mechanics)
Javascript Key Event Triggers
(e.g., Moving shape left and right when key is pressed)