

CS 105

Week 4, Part 2

Grades are on Compass 2g!

Activities: 0, 1, 2, 3, 4

Labs: 0, 1, 2, 3

MPs: 0

Upcoming Grades/Data

MPs: 1, 2

i>Clicker Confirmations

Midterm #1

Tuesday, Oct. 7, 2014

8:00pm – 9:30pm

Midterm #1 Conflict Exam Signup
Posted on course website tonight!

CS 105: Improved

9am Lecture Video Recorded

TA Lecture Notes

Lecture Review

```
var x = [4, 6, 8, 10];  
print( x[2] );
```

What is printed to the screen?

A) 2

D) 8

B) 4

E) 10

C) 6

```
var x = [4, 6, 8, 10];  
print( x[2] );
```

What is printed to the screen?

A) 2

D) 8

B) 4

E) 10

C) 6

```
var x = [4, 6, 8, 10];
```

```
{  
    print(x[i]);  
}
```

Fill in the blank:

A) while (i < x.length)

B) for (i < x.length)

C) for (var i = 0; i < x.length; i++)

```
var x = [4, 6, 8, 10];
```

```
{  
    print(x[i]);  
}
```

Fill in the blank:

A) while (i < x.length)

B) for (i < x.length)

C) for (var i = 0; i < x.length; i++)

JavaScript

JavaScript

!=

Java

```
import javax.swing.*;

public class Hello extends JFrame
{
    public Hello()
    {
        super("hello");
        super.setDefaultCloseOperation(
            WindowConstants.EXIT_ON_CLOSE);
        super.add(new JLabel("Hello, world!"));
        super.pack();
        super.setVisible(true);
    }

    public static void main(final String[] args)
    {
        new Hello();
    }
}
```

```
import javax.swing.*;

public class Hello extends JFrame
{
    public Hello()
    {
        super("hello");
        super.setDefaultCloseOperation(
            WindowConstants.EXIT_ON_CLOSE);
        super.add(new JLabel("Hello, world!"));
        super.pack();
        super.setVisible(true);
    }

    public static void main(final String[] args)
    {
        new Hello();
    }
}
```

```
import javax.swing.*;

public class Hello extends JFrame
{
    public Hello()
    {
        super("hello");
        super.setDefaultCloseOperation(
            WindowConstants.EXIT_ON_CLOSE);
        super.add(new JLabel("Hello, world!"));
        super.pack();
        super.setVisible(true);
    }

    public static void main(final String[] args)
    {
        new Hello();
    }
}
```

```
import javax.swing.*;

public class Hello extends JFrame
{
    public Hello()
    {
        super("hello");
        super.setDefaultCloseOperation(
            WindowConstants.EXIT_ON_CLOSE);
        super.add(new JLabel("Hello, world!"));
        super.pack();
        super.setVisible(true);
    }

    public static void main(final String[] args)
    {
        new Hello();
    }
}
```

JavaScript == JS

JavaScript == ECMAScript

JS == ECMAScript

JavaScript != Java

Java vs. JavaScript

- JavaScript is built for the Internet
 - Gmail is [partially] written in JavaScript
- Java is built for standalone applications
 - Android apps are written in Java

Console: Goodbye!

HTML

HyperText Markup Language

HTML

- Not a programming language

HTML

- Not a programming language
 - No conditionals
 - No loops
 - No variables

HTML

- HTML is a *markup* language

HTML

- HTML is a *markup* language
 - Describes how something should be displayed

HTML

- HTML is a *markup* language
 - Describes how **something** should be displayed

HTML

- HTML is a *markup* language
 - Describes how **web pages** should be displayed

HTML

- HTML is a *markup* language
 - Describes how **web pages** should be displayed
 - Structured, rule-based language

```
<html>
```

```
</html>
```

```
<html>
```

```
</html>
```

```
<html>
```

```
  <head>
```

```
  </head>
```

```
  <body>
```

```
  </body>
```

```
</html>
```

```
<html>
```

```
  <head>
```

```
</head>
```

```
  <body>
```

```
</body>
```

```
</html>
```

```
<html>
```

```
  <head>
```

```
  </head>
```

```
  <body>
```

```
    Hello, world!
```

```
  </body>
```

```
</html>
```

HTML

- All HTML pages have the `<html>` element as the top-most level.

HTML

- All HTML pages have the `<html>` element as the top-most level.
- The `<html>` element has exactly two children:
 - `<head>`
 - `<body>`

<body>

- Everything in the **<body>** element is displayed as part of the web page.

```
<html>
  <head>
</head>
  <body>
    <div>
      Hello, world!
    </div>
  </body>
</html>
```

```
<html>
  <head>
</head>
  <body>
    <div>
      Hello, world!
    </div>
  </body>
</html>
```

<div>

- The HTML **<div>** element defines a **division** (or section) of the web page.

Attributes

- All HTML elements can have **attributes**, which give extra meaning to an HTML element.
 - Attributes are always of the form
`name="value"`
 - `<div id="main">`

```
<html>
  <head>
</head>
  <body>
    <input type="button" value="Add"
      onclick="sayHello()">
    </input>
  </body>
</html>
```

<input>

- The HTML **<input>** element places a component on the web page that allows the user to interact with the webpage.
 - If the input has the attribute **type="button"**, a button appears on the web page.

onclick Attributes

- One HTML attribute that is applied to `<input>` elements is **onclick**.
 - The value of the **onclick** attribute is the JavaScript code you want to run when the element is clicked.
 - Usually, **onclick** will call a function.

```
<html>
  <head>
    <script src="file.js"></script>
  </head>
  <body>
    <input type="button" value="Add"
          onclick="add()" />
  </body>
</html>
```