

Running JavaScript

Chapter 18

Concatenation: Operating On Strings

- **string concatenation:** using the + operator between a string and another value to make a longer string

- **Examples:**

```
'hello' + 42    is 'hello42'  
1 + "abc" + 2  is "1abc2"  
'abc' + 1 + 2  is 'abc12'  
1 + 2 + "abc"  is "3abc"  
"1" + 1        is "11"
```

2

Popup Box

- Alert box syntax:
`alert(<expression>);`

- Examples:
`alert("Hello, world!");`



3

What Is In The Variable?

- Use alert boxes to reveal the value of variables.

```
var x = 100;  
alert(x);
```



- Use string concatenation to make alert messages even more useful.

```
alert("x = [" + x + "]);
```



better!

4

Linking JavaScript File To XHTML File

```
<head>  
  <title>...</title>  
  <link rel="stylesheet" type="text/css" href="style.css" />  
  <script src="filename.js" type="text/javascript"></script>  
</head>
```

- Copy the `type` attribute and its corresponding value verbatim
- Use the `src` attribute to specify the location of a JavaScript file
 - Path location may be absolute or relative

5

Conditionals

Conditionals

- "If button is clicked, then close the popup box."
- "If Mario touches the flag, then end the level."
- "If a correct password has been entered, then reveal the top secret documents, otherwise contact the FBI."
- "If the coin collected brings the total to one hundred, make 1-up sound, otherwise make regular coin collection sound."

7

The if Statement

- **if statement:** a *control structure* that executes a block of statements only if a certain condition is true

- General syntax:

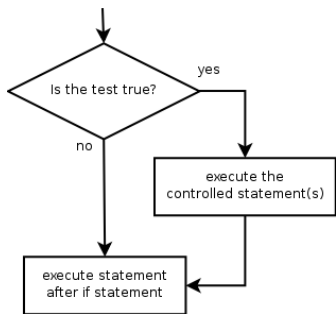
```
if (<test>) {  
    <statement(s)>;  
}
```

- Example:

```
var gpa = 3.25;  
if (gpa >= 3.0) {  
    alert("Good job! Have a cookie.");  
}
```

8

if Statement Flow Chart



9

The if/else Statement

- **if/else statement:** A control structure that executes one block of statements if a certain condition is true, and a second block of statements if it is false. We refer to each block as a *branch*.

- General syntax:

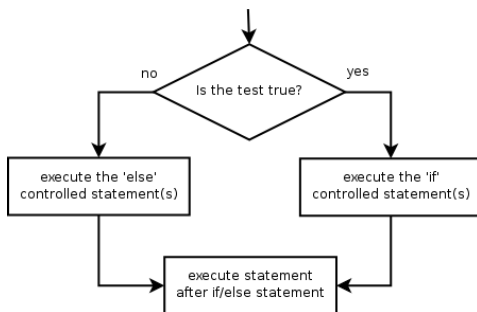
```
if (<test>) {  
    <statement(s)>;  
} else {  
    <statement(s)>;  
}
```

- Example:

```
var gpa = 3.25;  
if (gpa >= 3.0) {  
    alert("Good job! Have a cookie.");  
} else {  
    alert("No cookie for you!");  
}
```

10

if/else Statement Flow Chart



11

Relational Expressions

- The **<test>** used in an if or if/else statement must *evaluate* to a Boolean value (true or false).
- **Relational expressions** evaluate to Boolean values and use the following **relational operators**:

Operator	Meaning	Example	Value
==	equals	1 + 1 == 2	true
!=	does not equal	3.2 != 2.5	true
<	less than	10 < 5	false
>	greater than	10 > 5	true
<=	less than or equal to	126 <= 100	false
>=	greater than or equal to	5.0 >= 5.0	true

12

Evaluating Relational Expressions

- Relational operators have lower precedence than math operators.

```
5 * 7 >= 3 + 5 * (7 - 1)
5 * 7 >= 3 + 5 * 6
35 >= 3 + 30
35 >= 33
true
```

- Relational operators should not be "chained" as they can in algebra. **WARNING!** JavaScript will NOT complain if you do so and you may get unexpected results.

```
2 <= 1 <= 10
false <= 10
```

???

13

Errors In Coding

- Many students new to if/else write code like this:

```
var percent = 85;
if (percent >= 90) {
  alert("You got an A!");
}
if (percent >= 80) {
  alert("You got a B!");
}
if (percent >= 70) {
  alert("You got a C!");
}
if (percent >= 60) {
  alert("You got a D!");
}
else {
  alert("You got an F!");
}
```

- What will happen? What's the problem?
 - You may get too many popup boxes. Try it out!

14

Nested if/else Statements

- Nested if/else statement:** A chain of if/else that can select between many different outcomes based on several tests.

- General syntax:

```
if (<test>) {
  <statement(s)>;
} else if (<test>) {
  <statement(s)>;
} else {
  <statement(s)>;
}
```

- Example:

```
if (number > 0) {
  alert("Positive");
} else if (number < 0) {
  alert("Negative");
} else {
  alert("Zero");
}
```

15

Nested if/else Variations

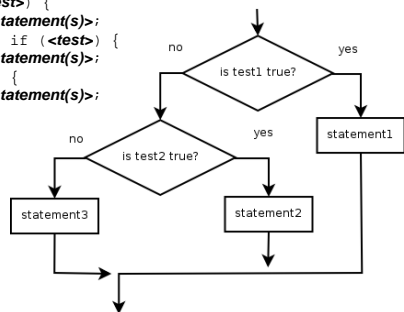
- A nested if/else can end with an if or an else.
 - If it ends with else, one of the branches must be taken.
 - If it ends with if, the program might not execute any branch.

<pre>if (<test>) { <statement(s)>; } else if (<test>) { <statement(s)>; } else { <statement(s)>; }</pre>	<pre>if (<test>) { <statement(s)>; } else if (<test>) { <statement(s)>; } else (if) (<test>) { <statement(s)>; }</pre>
--	--

16

Nested if/else Flow Chart

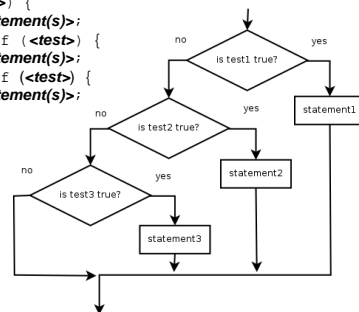
```
if (<test>) {
  <statement(s)>;
} else if (<test>) {
  <statement(s)>;
} else {
  <statement(s)>;
}
```



17

Nested if/else if Flow Chart

```
if (<test>) {
  <statement(s)>;
} else if (<test>) {
  <statement(s)>;
} else if (<test>) {
  <statement(s)>;
}
```



18

Nested if/else Variations

```

if (place == 1) {
    alert("You win the gold medal!");
} else if (place == 2) {
    alert("You win a silver medal!");
} else if (place == 3) {
    alert("You earned a bronze medal.");
}
    
```

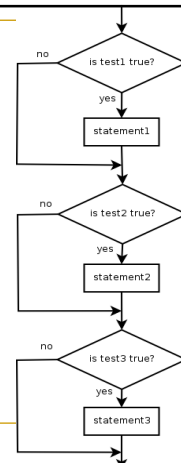
- Are there any cases where this code will not print a message?
 - Yes, if the `place` variable is not 1, 2, or 3.
- How could we modify it to print a message to non-medalists?
 - Add an `else` clause.

19

Sequential if Flow Chart

```

if (<test>) {
    <statement(s)>;
}
if (<test>) {
    <statement(s)>;
}
if (<test>) {
    <statement(s)>;
}
    
```



Summary: if/else Structures

- Choose exactly 1 set of statements

```

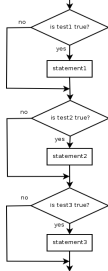
if (<test>) {
    <statement(s)>;
} else if (<test>) {
    <statement(s)>;
} else {
    <statement(s)>;
}
    
```



- Choose 0, 1, or more set of statements

```

if (<test>) {
    <statement(s)>;
}
if (<test>) {
    <statement(s)>;
}
if (<test>) {
    <statement(s)>;
}
    
```



- Choose 0 or 1 set of statements

```

if (<test>) {
    <statement(s)>;
} else if (<test>) {
    <statement(s)>;
} else if (<test>) {
    <statement(s)>;
}
    
```



21

Which if/else Construct To Use?

- Reading the user's GPA and printing whether the student is on the dean's list (3.8 to 4.0) or honor roll (3.5 to 3.7)
- Printing whether a number is even or odd
- Printing whether a user is lower-class, middle-class, or upper-class based on their income
- Determining whether a number is divisible by 2, 3, and/or 5
- Printing a user's grade of A, B, C, D, or F based on their percentage in the course

22

Which if/else Construct To Use?

- Reading the user's GPA and printing whether the student is on the dean's list (3.8 to 4.0) or honor roll (3.5 to 3.7)
 - `if / else if`
- Printing whether a number is even or odd
 - `if / else`
- Printing whether a user is lower-class, middle-class, or upper-class based on their income
 - `if / else if / else`
- Determining whether a number is divisible by 2, 3, and/or 5
 - `if / if / if`
- Printing a user's grade of A, B, C, D, or F based on their percentage in the course
 - `if / else if / else if / else if / else`

23

That Thing Called Style

- As with HTML, you are required to indent your code properly.
 - Indent code within opening and closing curly braces.
- You should spend time on thinking or coding. You should **NOT** be wasting time looking for that missing closing brace.
- So code with style!

24