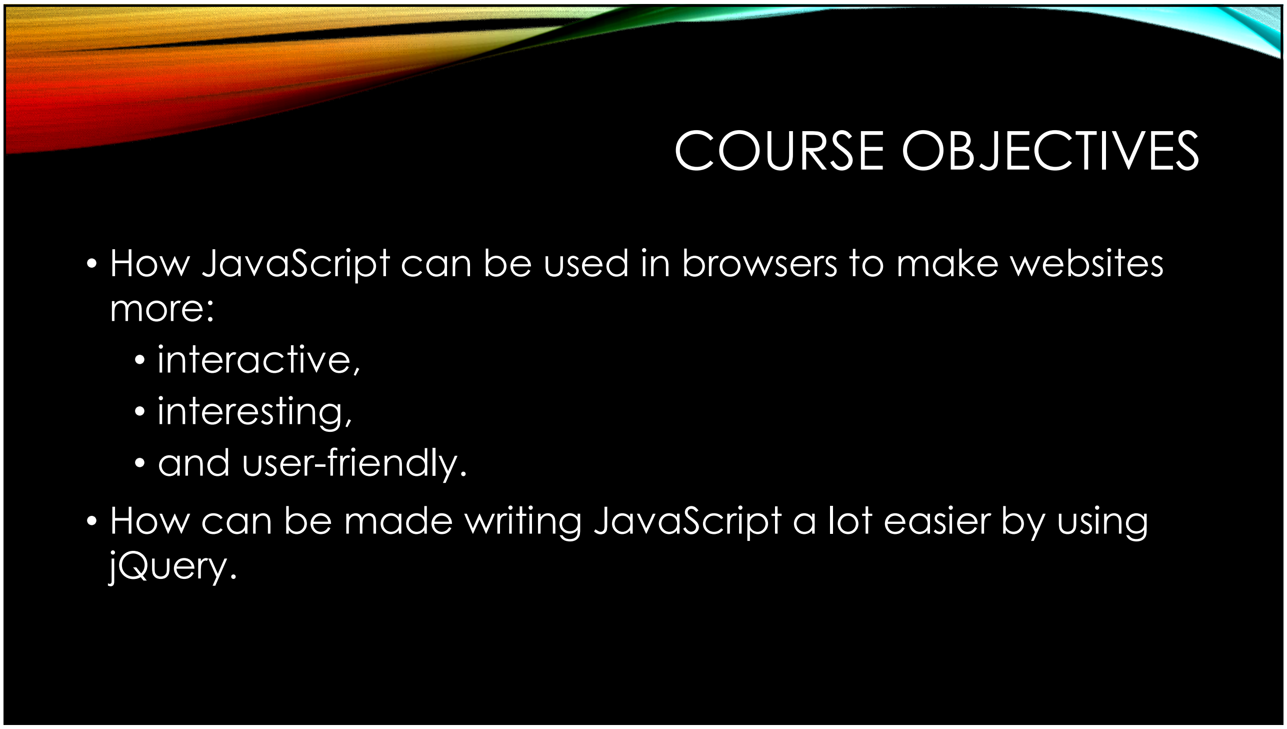
JAVASCRIPT

Haider M. Habeeb

REFERENCES

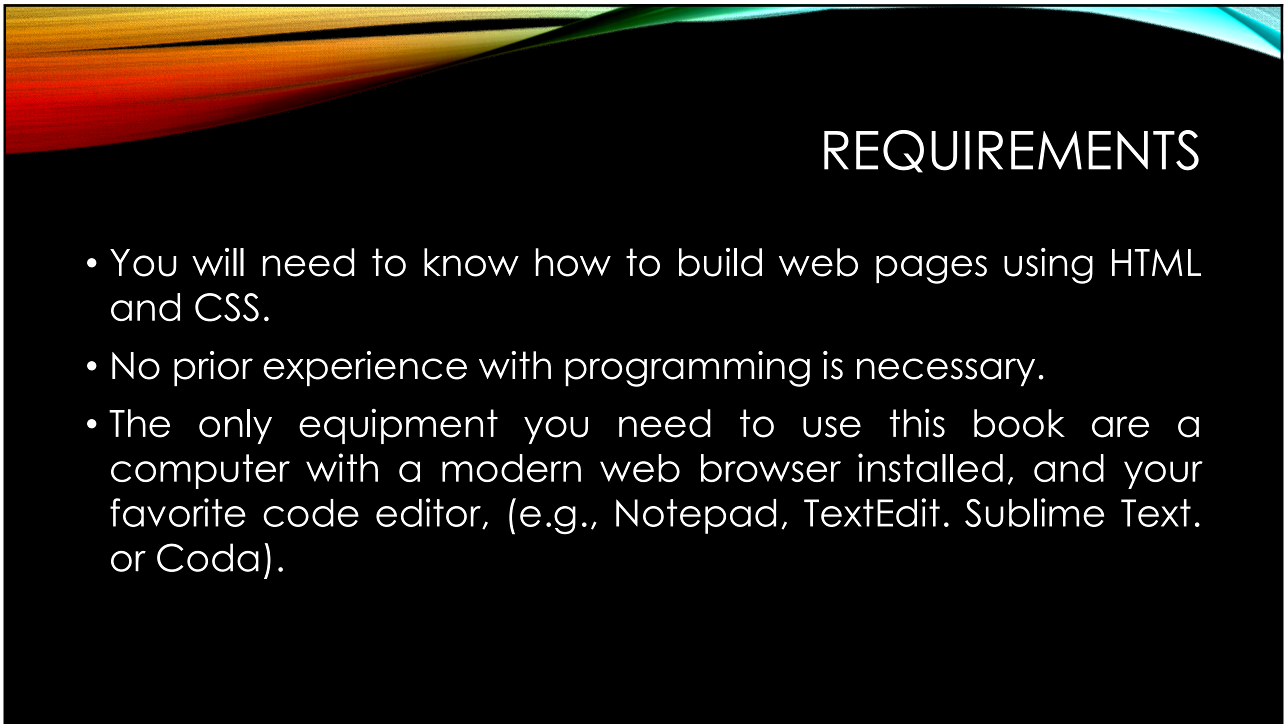
- Jon Duckett. 2014. *JavaScript and JQuery: Interactive Front-End Web Development* (1st ed.). Indianapolis, Indiana: John Wiley & Sons Publishing

- Refsnes, H. and Refsnes, S. and Refsnes, K.J. and Refsnes, J.E. and Henthorne, K.D. 2010. *Learn JavaScript and Ajax with w3Schools.* Indianapolis, Indiana: Wiley Publishing



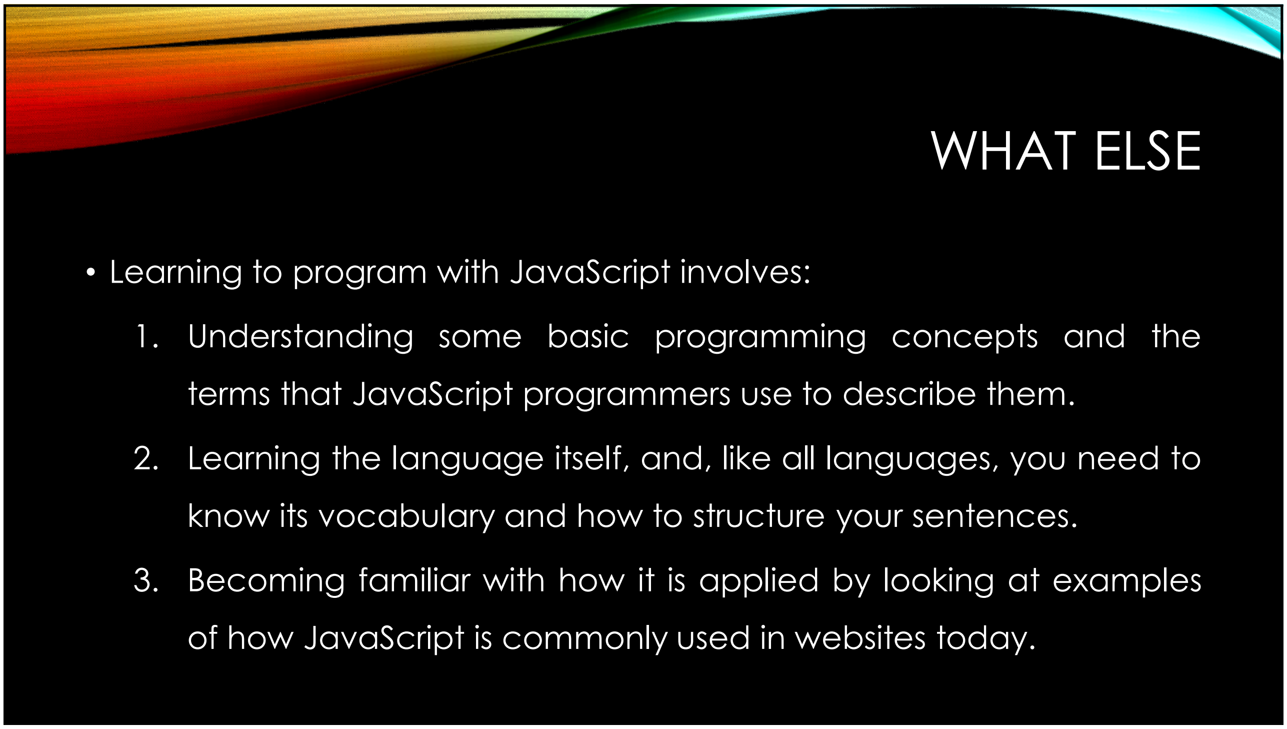
COURSE OBJECTIVES

- How JavaScript can be used in browsers to make websites more:
  - interactive,
  - interesting,
  - and user-friendly.
- How can be made writing JavaScript a lot easier by using jQuery.



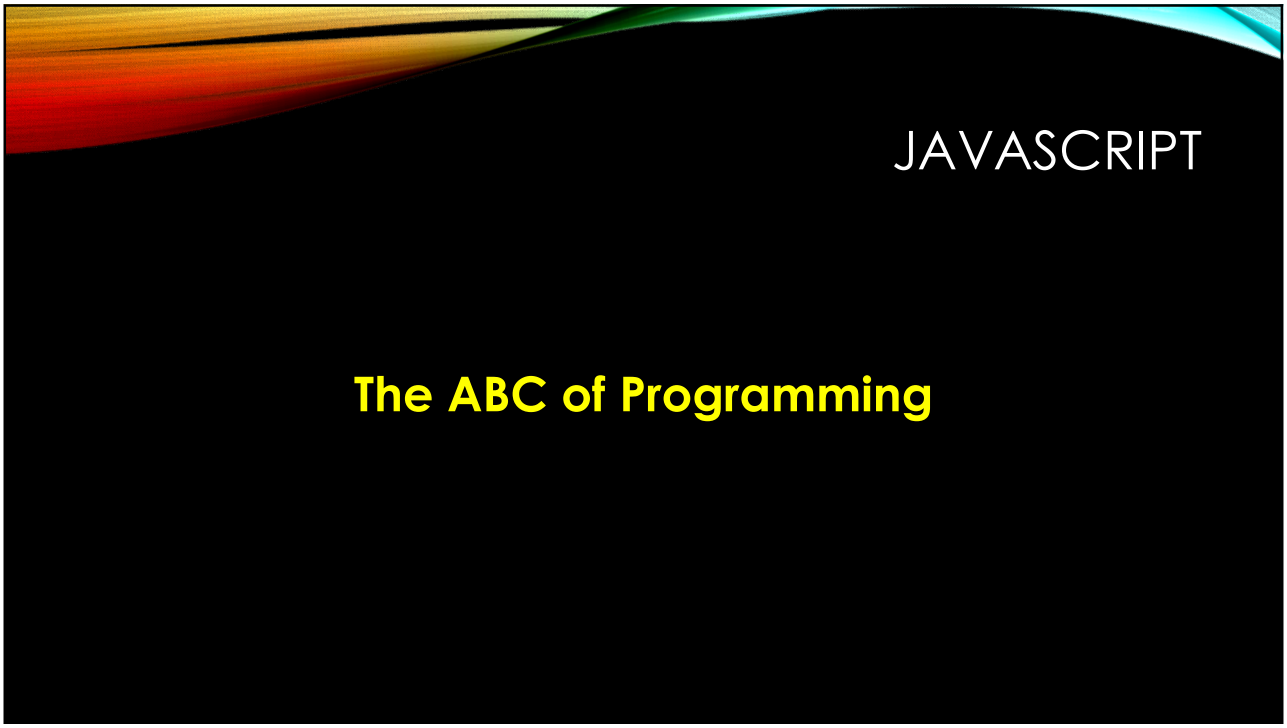
REQUIREMENTS

- You will need to know how to build web pages using HTML and CSS.
- No prior experience with programming is necessary.
- The only equipment you need to use this book are a computer with a modern web browser installed, and your favorite code editor, (e.g., Notepad, TextEdit. Sublime Text. or Coda).



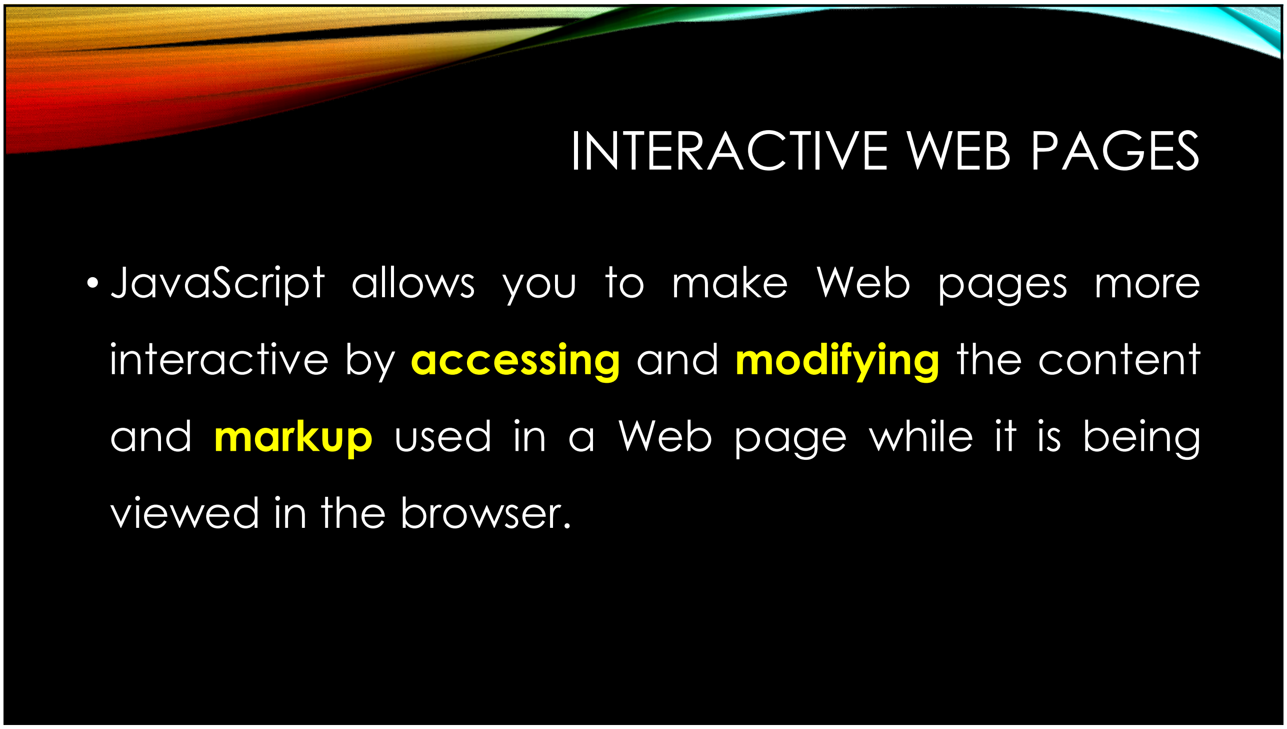
WHAT ELSE

- Learning to program with JavaScript involves:
  1. Understanding some basic programming concepts and the terms that JavaScript programmers use to describe them.
  2. Learning the language itself, and, like all languages, you need to know its vocabulary and how to structure your sentences.
  3. Becoming familiar with how it is applied by looking at examples of how JavaScript is commonly used in websites today.



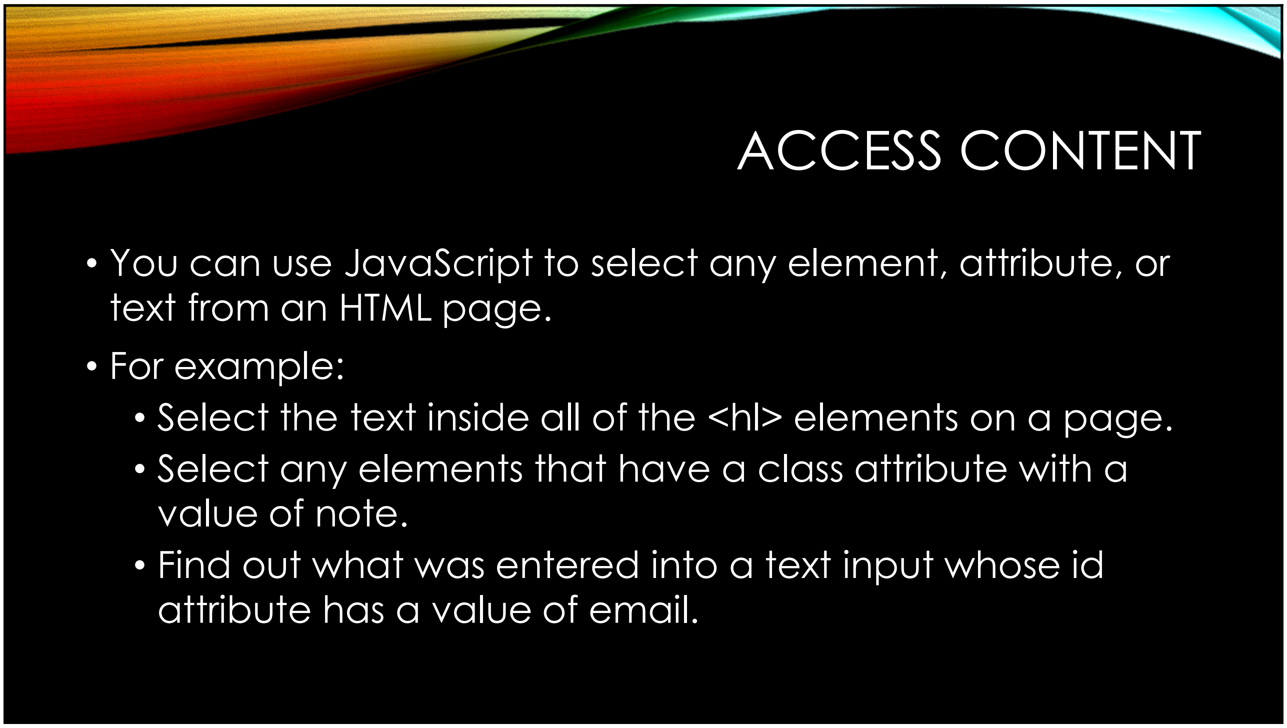
JAVASCRIPT

The ABC of Programming



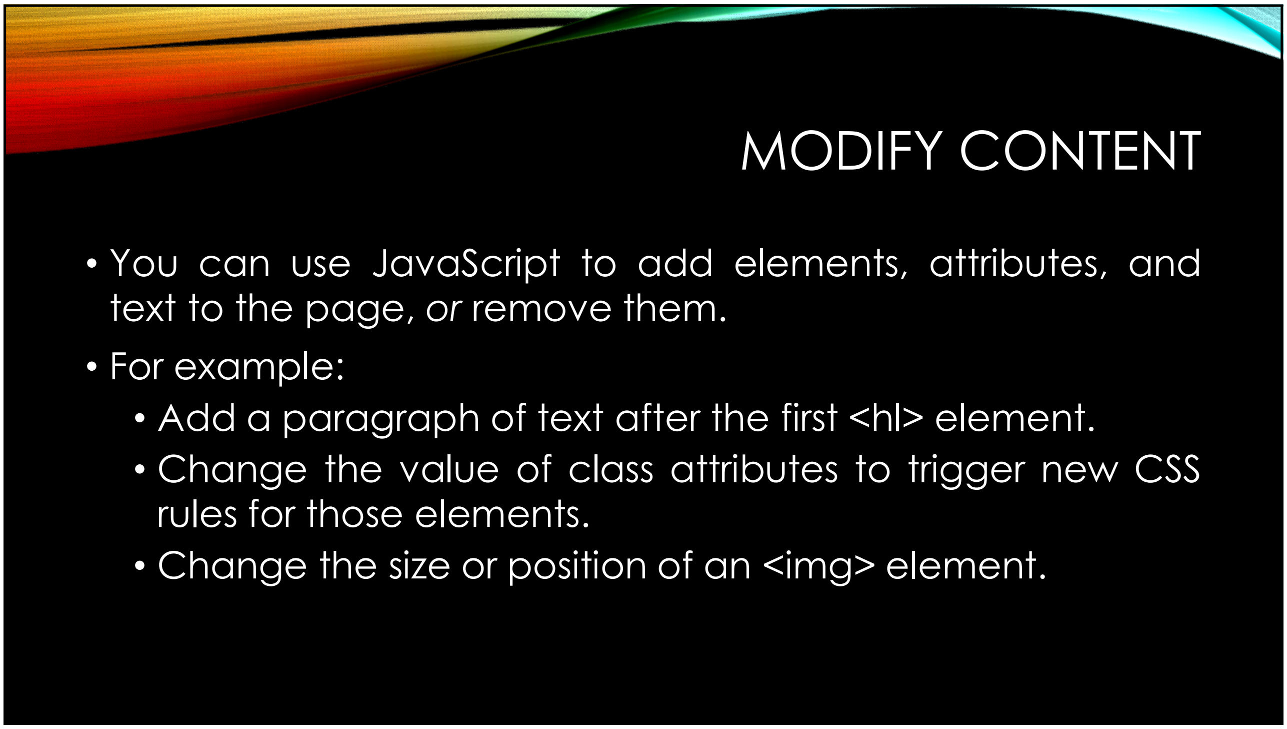
INTERACTIVE WEB PAGES

- JavaScript allows you to make Web pages more interactive by **accessing** and **modifying** the content and **markup** used in a Web page while it is being viewed in the browser.



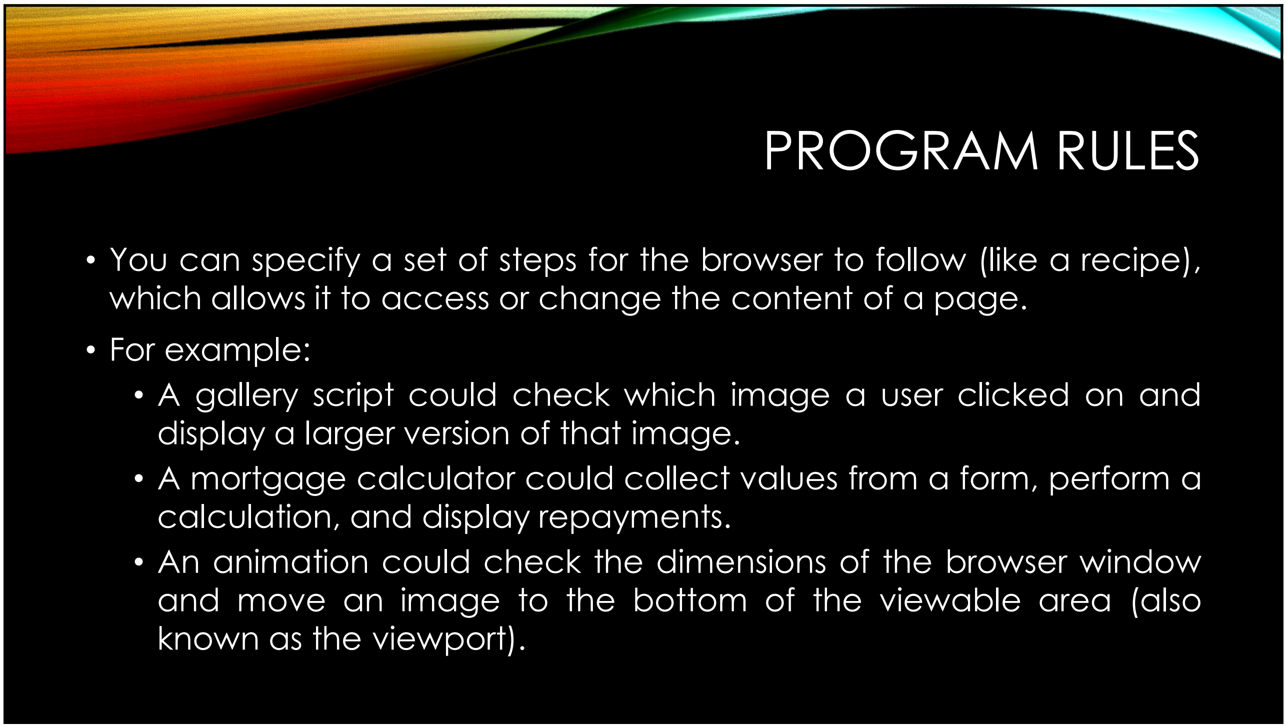
ACCESS CONTENT

- You can use JavaScript to select any element, attribute, or text from an HTML page.
- For example:
  - Select the text inside all of the <hl> elements on a page.
  - Select any elements that have a class attribute with a value of note.
  - Find out what was entered into a text input whose id attribute has a value of email.



MODIFY CONTENT

- You can use JavaScript to add elements, attributes, and text to the page, *or* remove them.
- For example:
  - Add a paragraph of text after the first <hl> element.
  - Change the value of class attributes to trigger new CSS rules for those elements.
  - Change the size or position of an <img> element.



PROGRAM RULES

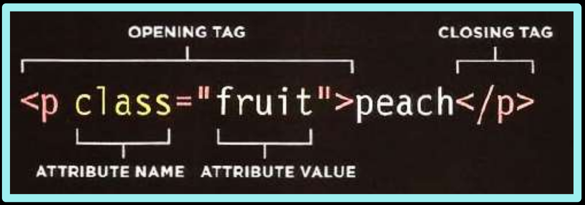
- You can specify a set of steps for the browser to follow (like a recipe), which allows it to access or change the content of a page.
- For example:
  - A gallery script could check which image a user clicked on and display a larger version of that image.
  - A mortgage calculator could collect values from a form, perform a calculation, and display repayments.
  - An animation could check the dimensions of the browser window and move an image to the bottom of the viewable area (also known as the viewport).

REACT TO EVENTS

- You can specify that a script should run when a specific event has occurred. For example, it could be run when:
  - A button is pressed
  - A link is clicked (or tapped) on
  - A cursor hovers over an element
  - Information is added to a form
  - An interval of time has passed
  - A web page has finished loading

A QUICK REFRESHER

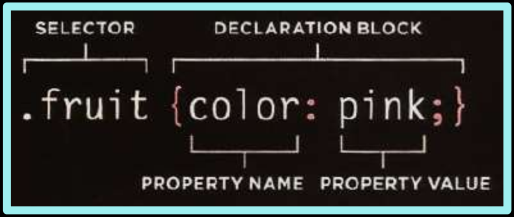
- **HTML Element:**
- An element consist of the opening and closing tags.
- There are a few empty elements with no content.
- Opening tags can carry attributes.
- Attributes have a name and value.
- The value is usually given in quotes.



```
<p class="fruit">peach</p>
```

A QUICK REFRESHER

- **CSS Rules:**
  - CSS uses rules to indicate how the content of elements should be displayed in the browser.
  - Each rule has a selector and declaration block.
  - Each declaration has a property and a value.



```
.fruit {color: pink;}
```

ABC OF PROGRAMING LANGUAGE

A What is a script and how do I create one?

B How do computers fit in with the world around them?

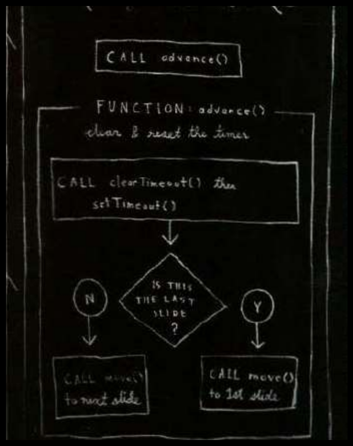
C How do I write a script for a web page?

A

- **What is a script and how do I create one?**
  - A script is a series of instructions that a computer can follow to achieve a goal.
  - You could compare scripts to any of the following:
    - RECIPES
    - HANDBOOKS
    - MANUALS

WRITING A SCRIPT

- To write a script, you need to first state your goal and then list the tasks that need to be completed in order to achieve it.
- Start with the big picture of what you want to achieve, and break that down into smaller steps.



SOLVE PROBLEM

1: DEFINE THE GOAL

- First, you need to define the task you want to achieve. (think of a puzzle)

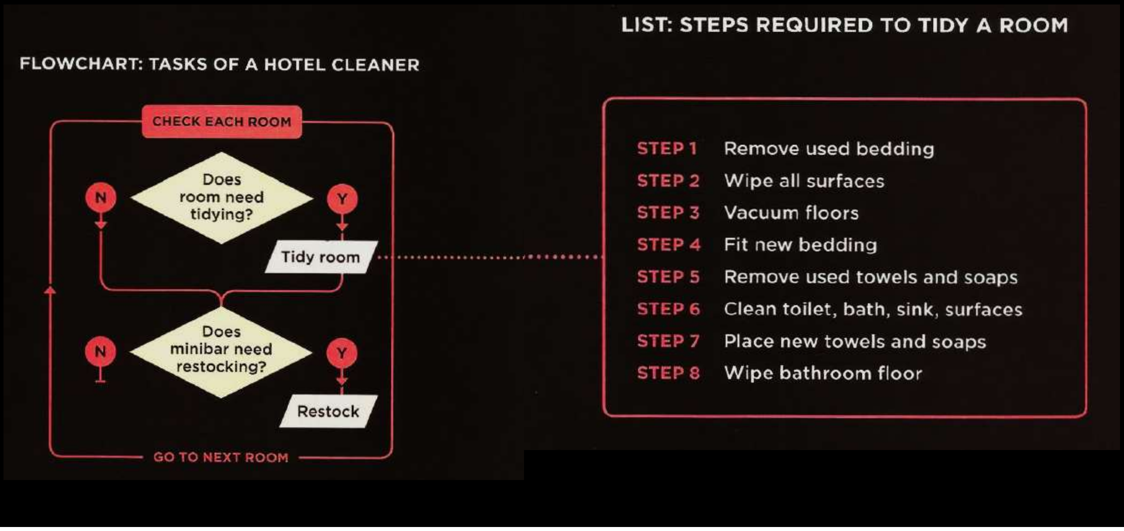
2: DESIGN THE SCRIPT

- To design a script you split the goal out into a series of tasks.

3: CODE EACH STEP

- Each of the steps needs to be written in a programming language that the computer understands.

DESIGN THE SCRIPT: TASKS & STEPS



LIST: STEPS REQUIRED TO TIDY A ROOM

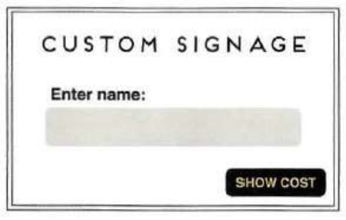
STEP 1	Remove used bedding
STEP 2	Wipe all surfaces
STEP 3	Vacuum floors
STEP 4	Fit new bedding
STEP 5	Remove used towels and soaps
STEP 6	Clean toilet, bath, sink, surfaces
STEP 7	Place new towels and soaps
STEP 8	Wipe bathroom floor

FROM STEPS TO CODE

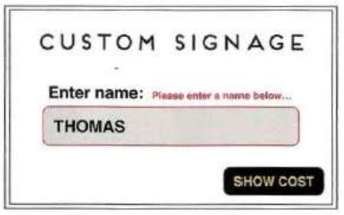
- Every step for every task shown in a flowchart needs to be written in a language the computer can understand and follow.
- **Vocabulary**: The words that computers understand.
- **Syntax**: How you put those words together to create instructions computers can follow.

EXAMPLE

- Calculates the cost of a name plaque.

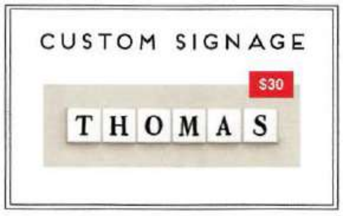

CUSTOM SIGNAGE

Enter name:


CUSTOM SIGNAGE

Enter name: Please enter a name below...

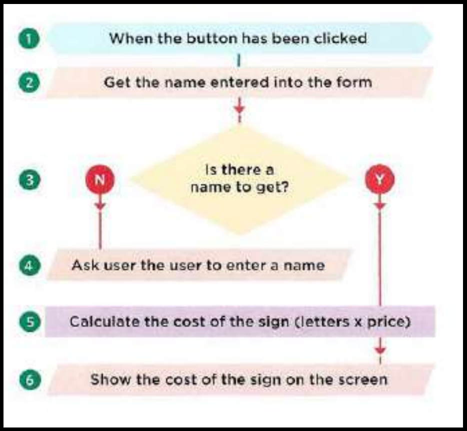
THOMAS


CUSTOM SIGNAGE

$30

EXAMPLE

- Tasks that have to be performed in order to achieve the goals:
    1. The script is triggered when the button is clicked.
    2. It collects the name entered into the form field.
    3. It checks that the user has entered a value.
    4. If the user has not entered anything, a message will appear telling them to enter a name.
    5. If a name has been entered, calculate the cost of the sign by multiplying the number of letters by the cost per letter.
    6. Show how much the plaque costs.


1
When the button has been clicked
2
Get the name entered into the form
3
N
Is there a name to get?
Y
4
Ask user the user to enter a name
5
Calculate the cost of the sign (letters x price)
6
Show the cost of the sign on the screen

SUMMARY

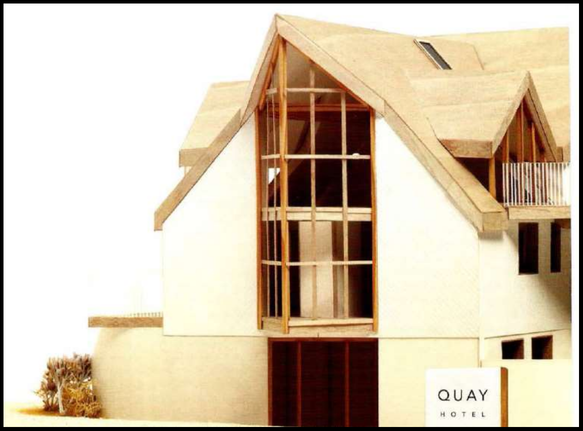
- What is a script and how do I create one?
    1. A script is a series of instructions that the computer can follow in order to achieve a goal.
    2. Each time the script runs, it might only use a subset of all the instructions.
    3. Computers approach tasks in a different way than humans, so your instructions must let the computer solve the task programmatically.
    4. To approach writing a script, break down your goal into a series of tasks and then work out each step needed to complete that task (a flowchart can help).

B

- **How do computers fit in with the world around them?**

MODEL

- Computers create models of the world using data
- Here is a model of a hotel, along with some model trees, model people, and model cars. To a human, it is clear what kind of real - world object each one represents.

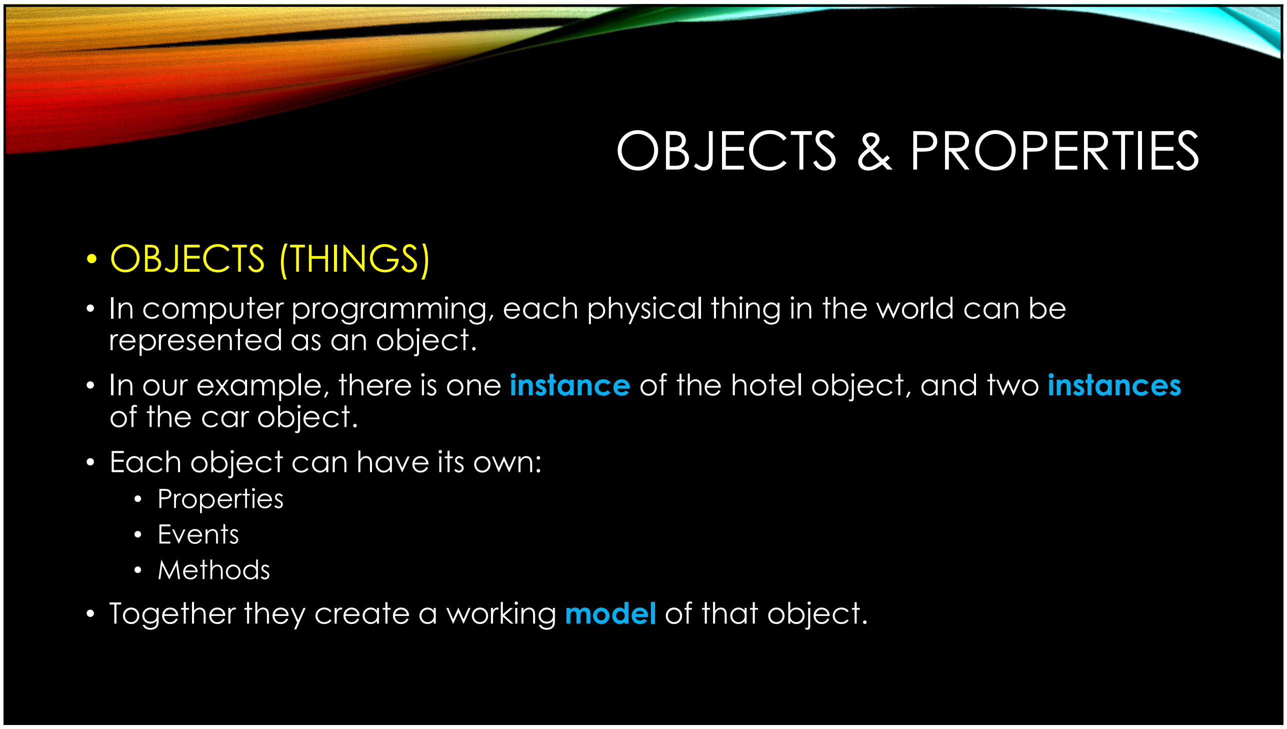


MODEL

- A computer has no predefined concept of what a hotel or car is. It does not know what they are used for. Your laptop or phone will not have a favorite brand of car, nor will it know what star rating your hotel is.
- So how do we use computers to create hotel booking apps, or video games where players can race a car?
- **The answer:** is that *programmers create a very different kind of model*, especially for computers.
- Programmers make these models using **data**.
- the **data** is all the computer needs in order to follow the instructions you give it to carry out its tasks.

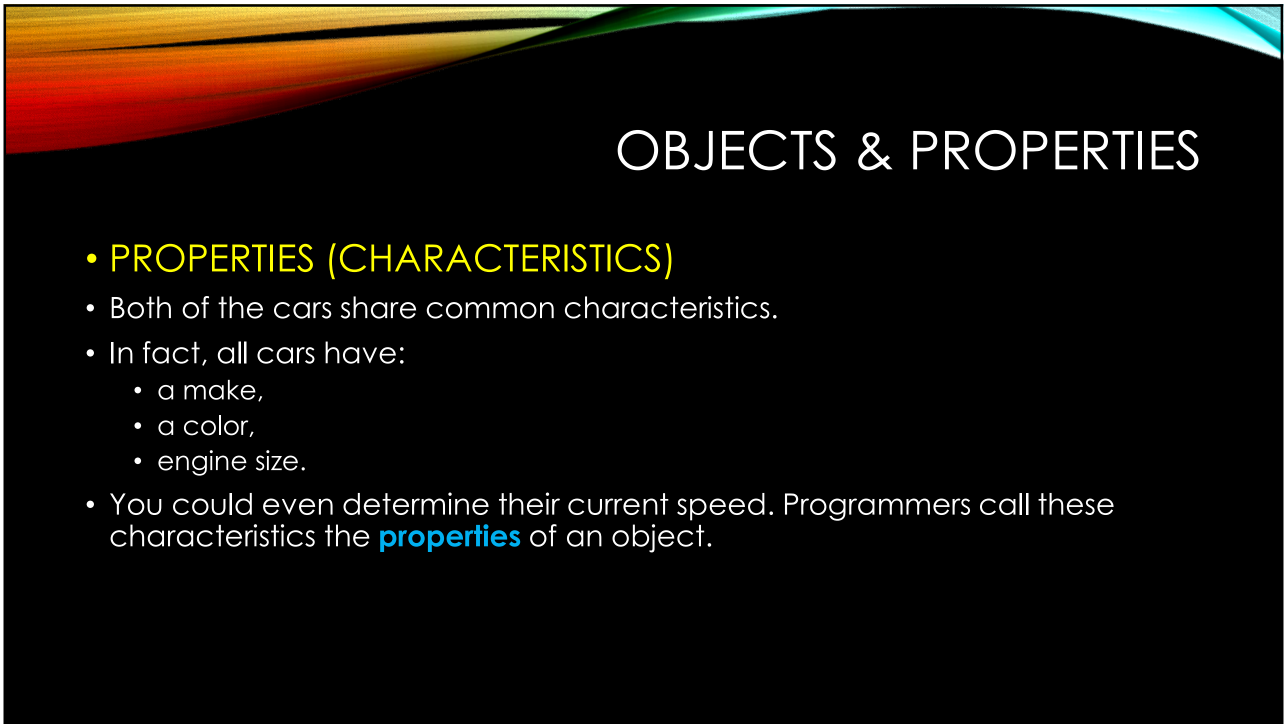
OBJECTS & PROPERTIES

- If you could not see the picture of the hotel and cars, the data in the information boxes alone would still tell you a lot about this scene.



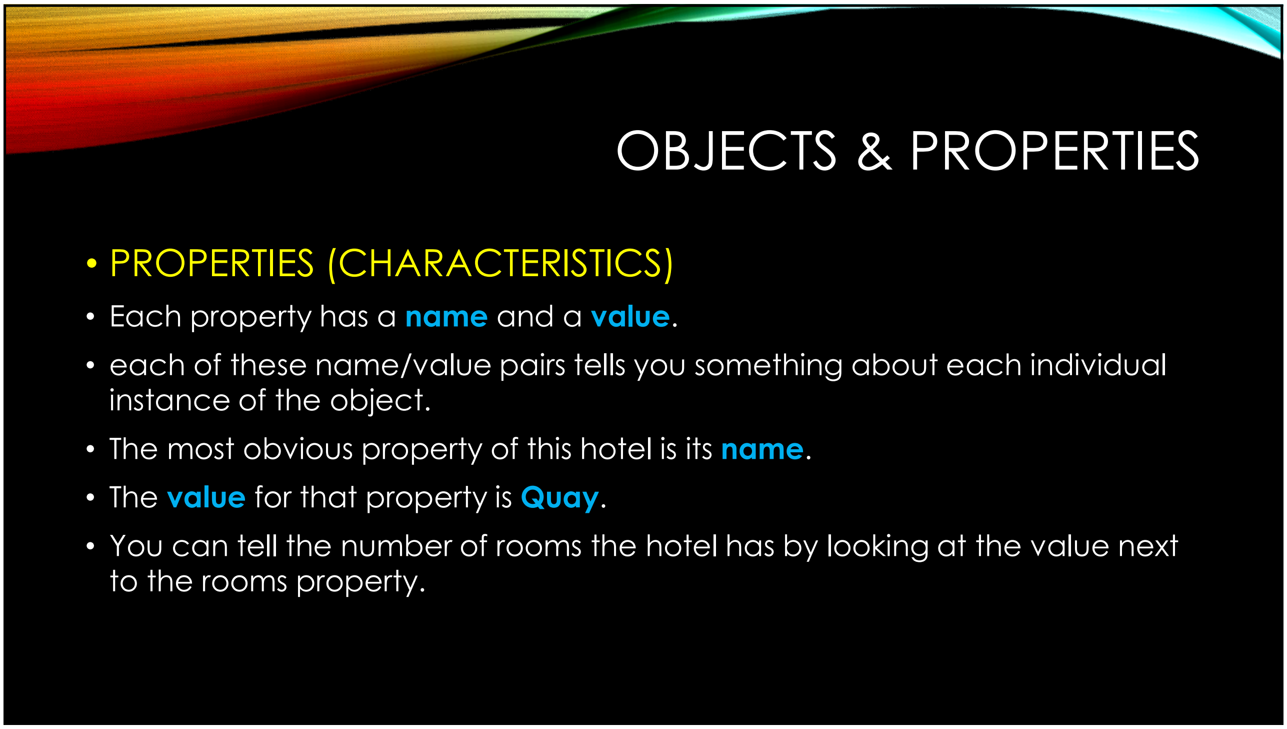
OBJECTS & PROPERTIES

- OBJECTS (THINGS)
- In computer programming, each physical thing in the world can be represented as an object.
- In our example, there is one **instance** of the hotel object, and two **instances** of the car object.
- Each object can have its own:
  - Properties
  - Events
  - Methods
- Together they create a working **model** of that object.



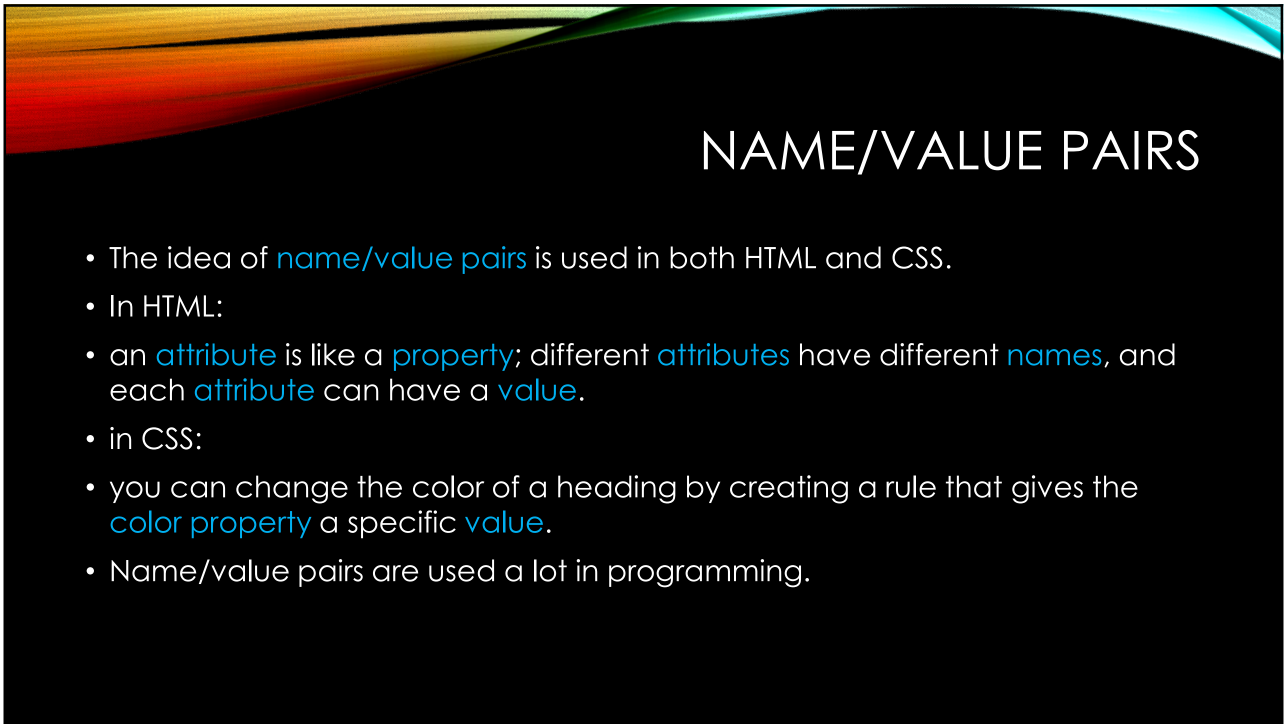
OBJECTS & PROPERTIES

- PROPERTIES (CHARACTERISTICS)
- Both of the cars share common characteristics.
- In fact, all cars have:
  - a make,
  - a color,
  - engine size.
- You could even determine their current speed. Programmers call these characteristics the **properties** of an object.



OBJECTS & PROPERTIES

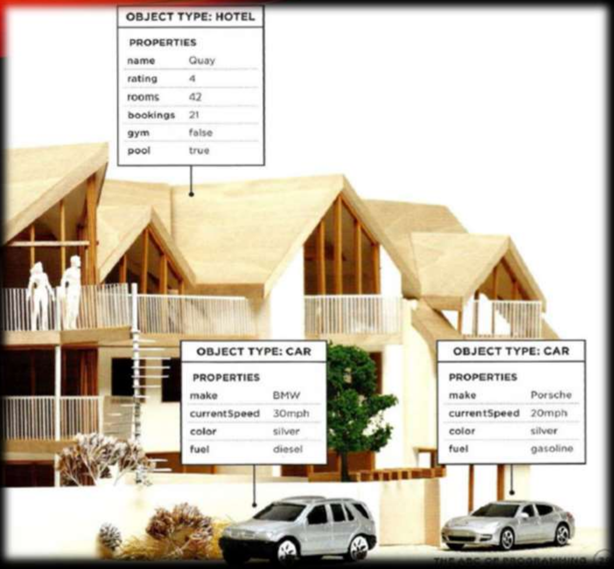
- PROPERTIES (CHARACTERISTICS)
- Each property has a **name** and a **value**.
- each of these name/value pairs tells you something about each individual instance of the object.
- The most obvious property of this hotel is its **name**.
- The **value** for that property is **Quay**.
- You can tell the number of rooms the hotel has by looking at the value next to the rooms property.



NAME/VALUE PAIRS

- The idea of name/value pairs is used in both HTML and CSS.
- In HTML:
- an attribute is like a property; different attributes have different names, and each attribute can have a value.
- in CSS:
- you can change the color of a heading by creating a rule that gives the color property a specific value.
- Name/value pairs are used a lot in programming.

EXAMPLE



OBJECT TYPE: HOTEL	
PROPERTIES	
name	Quay
rating	4
rooms	42
bookings	21
gym	false
pool	true

OBJECT TYPE: CAR	
PROPERTIES	
make	BMW
currentSpeed	30mph
color	silver
fuel	diesel

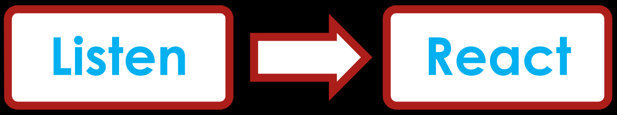
OBJECT TYPE: CAR	
PROPERTIES	
make	Porsche
currentSpeed	20mph
color	silver
fuel	gasoline

- **HOTEL OBJECT**
- The hotel object uses property names and values to tell you about this particular hotel'
- **CAR OBJECTS**
- The car objects both share the same properties, but each one has different values for those properties.

EVENTS

- In the real world, people interact with objects. These interactions can change the values of the properties in these objects.
- Example: The car has been designed to respond differently when the driver interacts with each of the two different pedals:
  - The accelerator makes the car go faster
  - The brake slows it down

EVENT MODEL



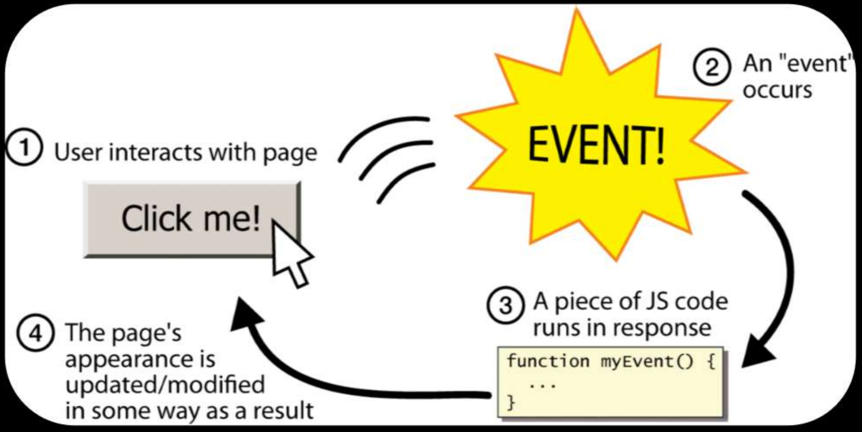
When ______ happens, do ______.

Source: Events in JavaScript by kirupa

EVENT MODEL

- **WHAT DOES AN EVENT DO?**
- Programmers choose which events they respond to.
- When a specific event happens, that event can be used to trigger a specific section of the code.
- Scripts often use different events to trigger different types of functionality.
- So a script will state which events the programmer wants to respond to, and what part of the script should be run when each of those events occur.

EVENT MODEL



1 User interacts with page

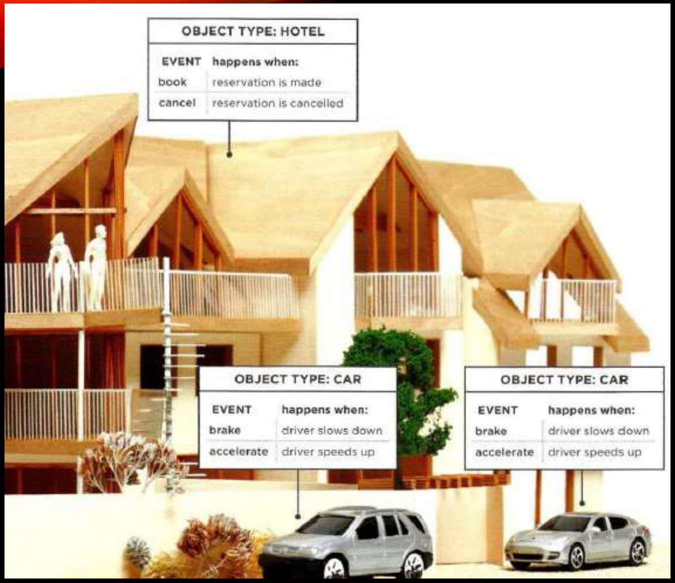
2 An "event" occurs

3 A piece of JS code runs in response

```
function myEvent() {
  ...
}
```

4 The page's appearance is updated/modified in some way as a result

Source: Web Programming Step by Step, 2nd Edition - *Chapter 8: JavaScript*



OBJECT TYPE: HOTEL

EVENT	happens when:
book	reservation is made
cancel	reservation is cancelled

OBJECT TYPE: CAR

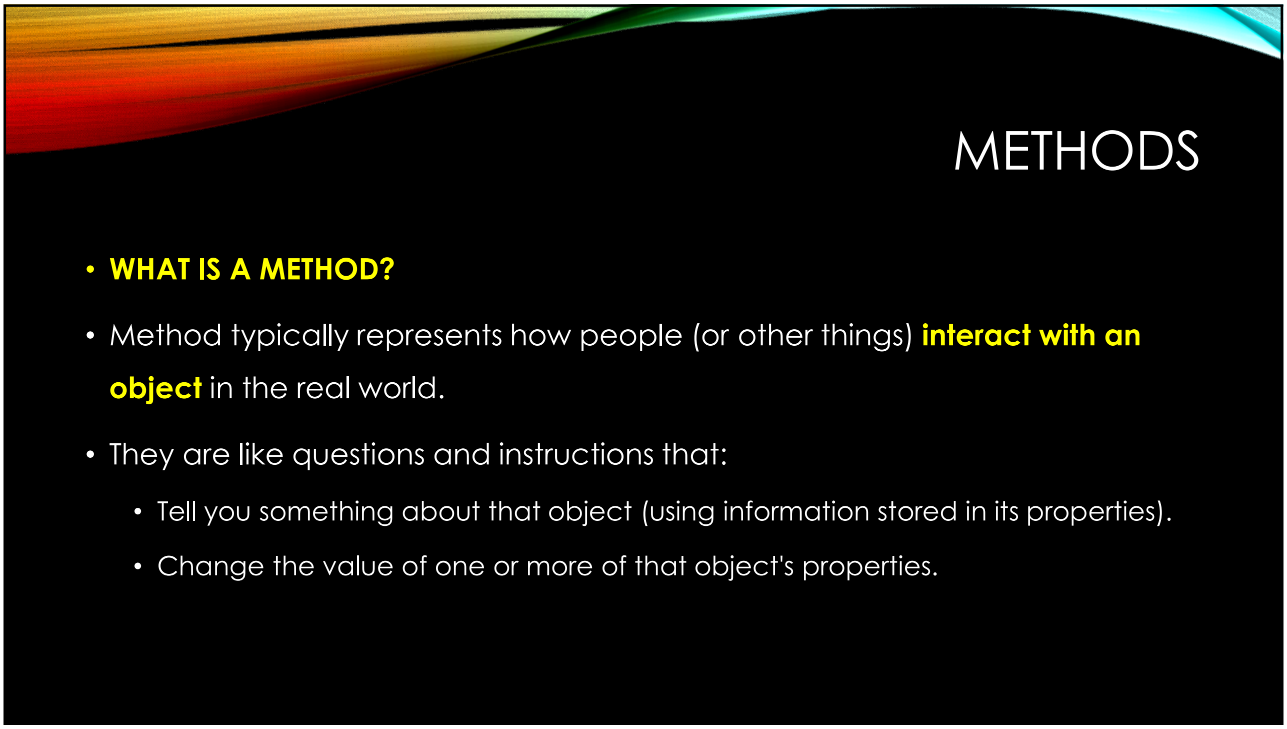
EVENT	happens when:
brake	driver slows down
accelerate	driver speeds up

OBJECT TYPE: CAR

EVENT	happens when:
brake	driver slows down
accelerate	driver speeds up

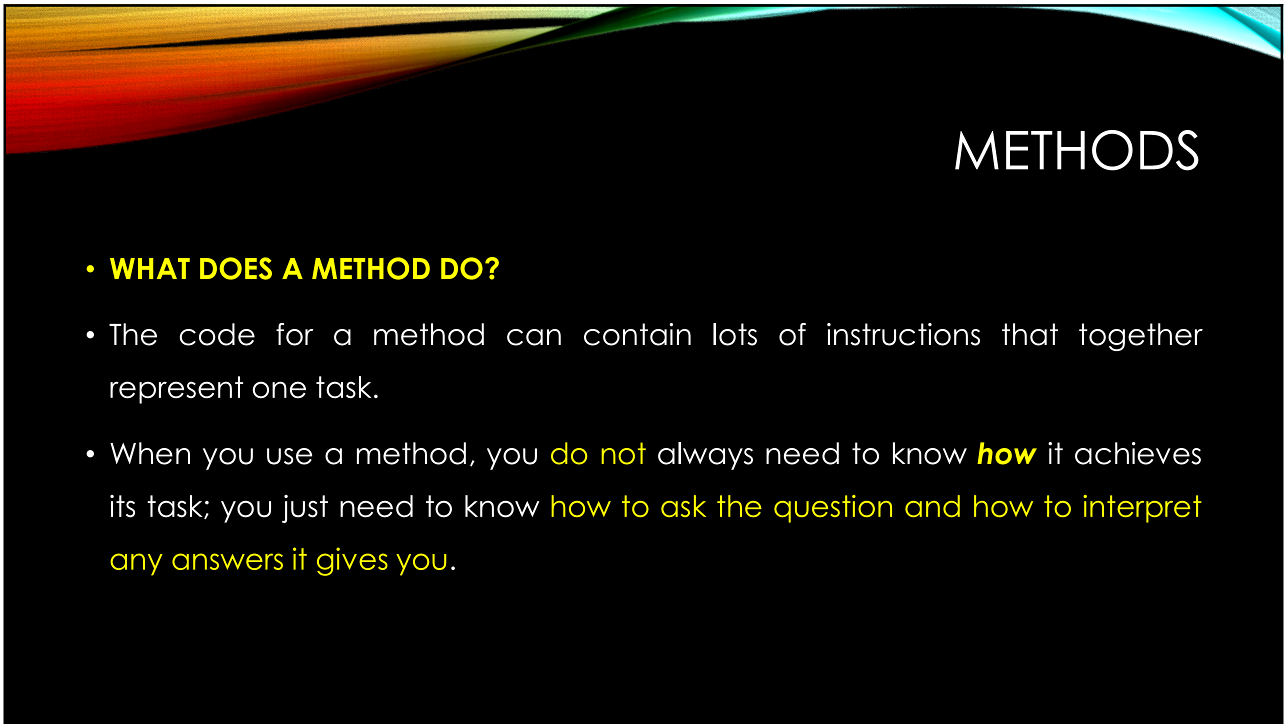
EVENT

- **HOTEL OBJECT**
- A hotel will regularly have bookings for rooms.
- Each time a room is reserved, an event called **book** can be used to trigger code that will increase the value of the **bookings property**.
- Likewise, a **cancel** event can trigger code that decreases the value of the **bookings property**.



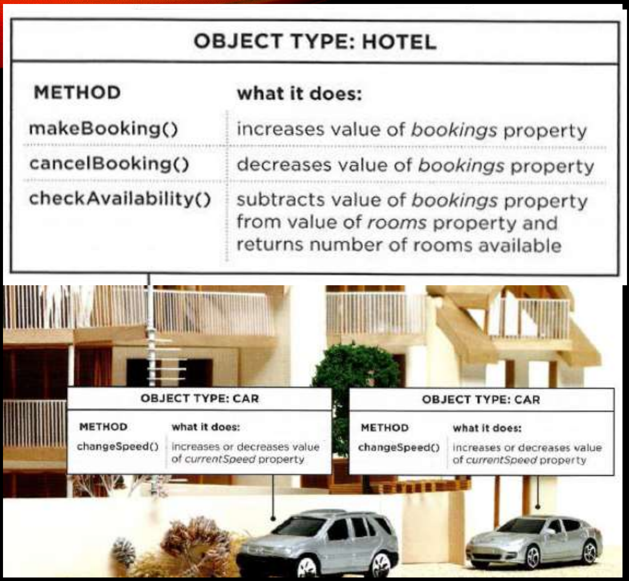
METHODS

- **WHAT IS A METHOD?**
- Method typically represents how people (or other things) **interact with an object** in the real world.
- They are like questions and instructions that:
  - Tell you something about that object (using information stored in its properties).
  - Change the value of one or more of that object's properties.



METHODS

- **WHAT DOES A METHOD DO?**
- The code for a method can contain lots of instructions that together represent one task.
- When you use a method, you do not always need to know ***how*** it achieves its task; you just need to know how to ask the question and how to interpret any answers it gives you.



OBJECT TYPE: HOTEL

METHOD	what it does:
makeBooking()	increases value of *bookings* property
cancelBooking()	decreases value of *bookings* property
checkAvailability()	subtracts value of *bookings* property from value of *rooms* property and returns number of rooms available

OBJECT TYPE: CAR

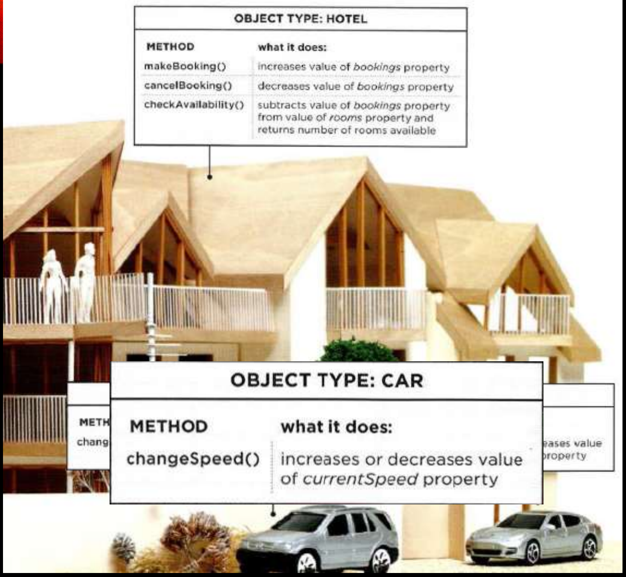
METHOD	what it does:
changeSpeed()	increases or decreases value of *currentSpeed* property

OBJECT TYPE: CAR

METHOD	what it does:
changeSpeed()	increases or decreases value of *currentSpeed* property

METHODS

- **HOTEL OBJECT**
- Hotels will commonly be asked if any rooms are free.
- To answer this question, a **method** can be written that subtracts the number of bookings from the total number of rooms.
- **Methods** can also be used to increase and decrease the value of the bookings property when rooms are booked or cancelled.



OBJECT TYPE: HOTEL

METHOD	what it does:
makeBooking()	increases value of *bookings* property
cancelBooking()	decreases value of *bookings* property
checkAvailability()	subtracts value of *bookings* property from value of *rooms* property and returns number of rooms available

OBJECT TYPE: CAR

METHOD	what it does:
changeSpeed()	increases or decreases value of *currentSpeed* property

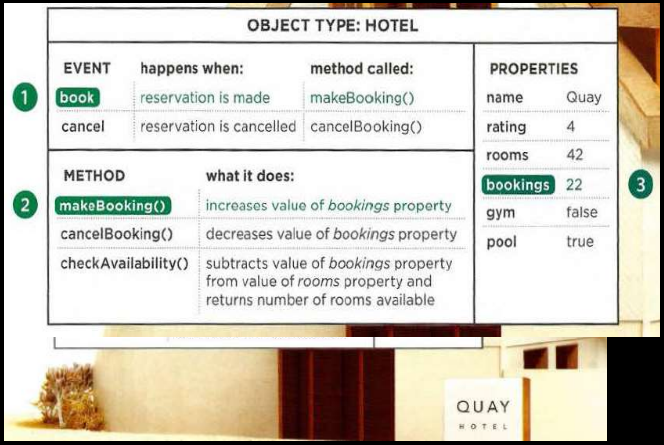
METHODS

- **CAR OBJECTS**
- The value of the **currentSpeed** property needs to go up and down as the driver accelerates and brakes.
- The code to increase or decrease the value of the currentSpeed property could be written in a **method**, and that method could be called **changeSpeed ()**.

PUTTING IT ALL TOGETHER

- Computers use data to create models of things in the real world.
- The events, methods, and properties of an object all relate to each other:
- Events can trigger methods, and methods can retrieve or update an object's properties.

PUTTING IT ALL TOGETHER


1
2
3
QUAY
HOTEL

OBJECT TYPE: HOTEL

EVENT	happens when:	method called:
book	reservation is made	makeBooking()
cancel	reservation is cancelled	cancelBooking()

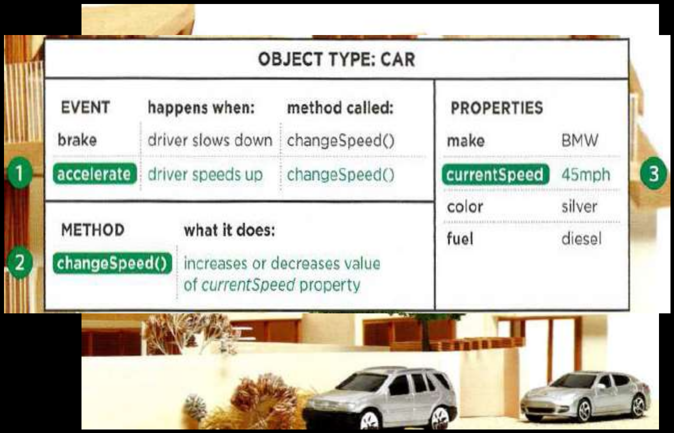
METHOD	what it does:
makeBooking()	increases value of *bookings* property
cancelBooking()	decreases value of *bookings* property
checkAvailability()	subtracts value of *bookings* property from value of *rooms* property and returns number of rooms available

PROPERTIES	
name	Quay
rating	4
rooms	42
bookings	22
gym	false
pool	true

- **HOTEL OBJECT**

1. When a reservation is made, the book event fires.
2. The book event triggers the makeBooki ng () method, which increases the value of the bookings property.
3. The value of the bookings property is changed to reflect how many rooms the hotel has available.

PUTTING IT ALL TOGETHER



OBJECT TYPE: CAR

EVENT	happens when:	method called:
brake	driver slows down	changeSpeed()
accelerate	driver speeds up	changeSpeed()

METHOD	what it does:
changeSpeed()	increases or decreases value of *currentSpeed* property

PROPERTIES	
make	BMW
currentSpeed	45mph
color	silver
fuel	diesel

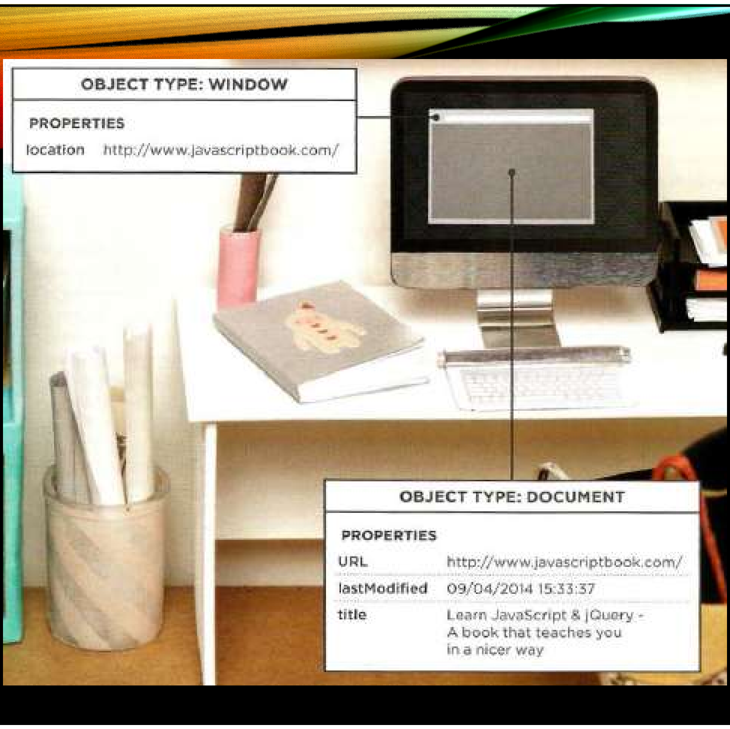
- **CAR OBJECTS**

1. As a driver speeds up, the accel erate event fires.
2. The accelerate event calls the changeSpeed () method. which in turn increases the value of the currentSpeed property.
3. The value of the currentSpeed property reflects how fast the car is traveling.

WEB BROWSERS & OBJECTS

- **Web browsers are programs built using objects**
- **WINDOW OBJECT**
- On the right-hand page you can see a model of a computer with a browser open on the screen.
- The browser represents each window or tab using a window object.
- The location property of the window object will tell you the URL of the current page.

DOCUMENT OBJECT



OBJECT TYPE: WINDOW

PROPERTIES

location	http://www.javascriptbook.com/

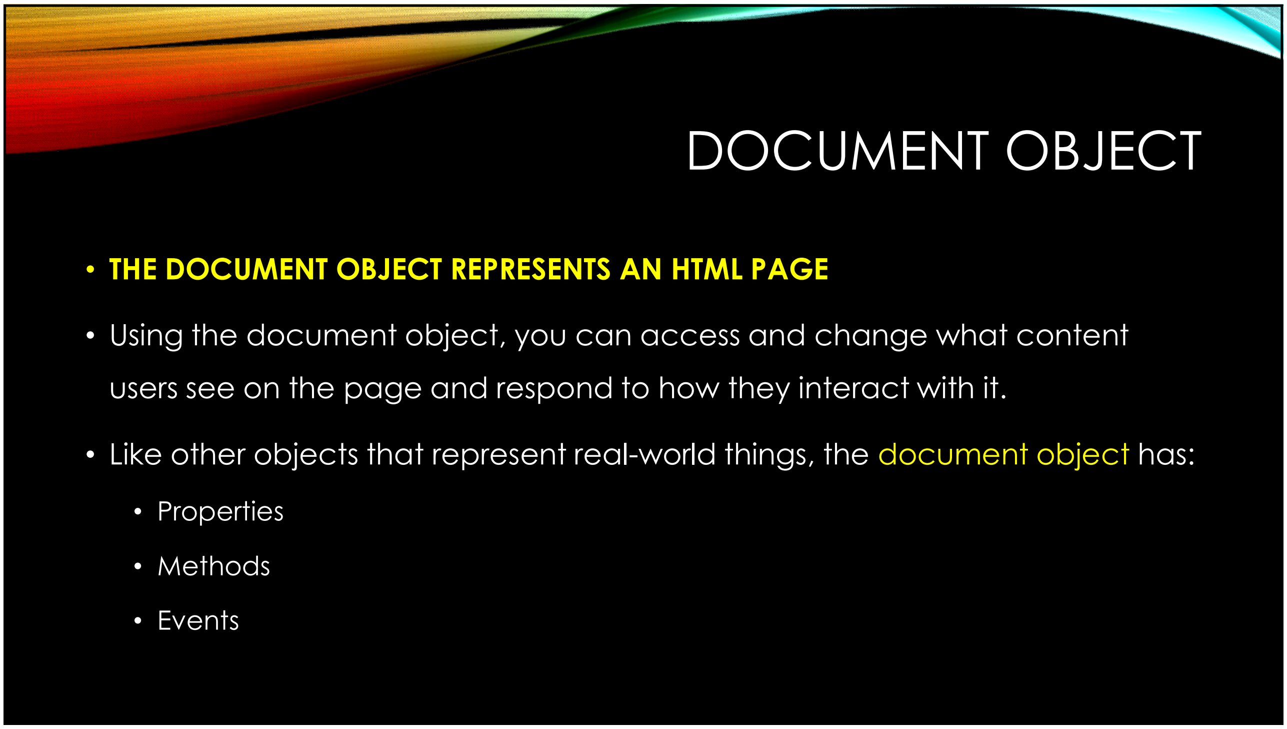
OBJECT TYPE: DOCUMENT

PROPERTIES

URL	http://www.javascriptbook.com/
lastModified	09/04/2014 15:33:37
title	Learn JavaScript & jQuery - A book that teaches you in a nicer way

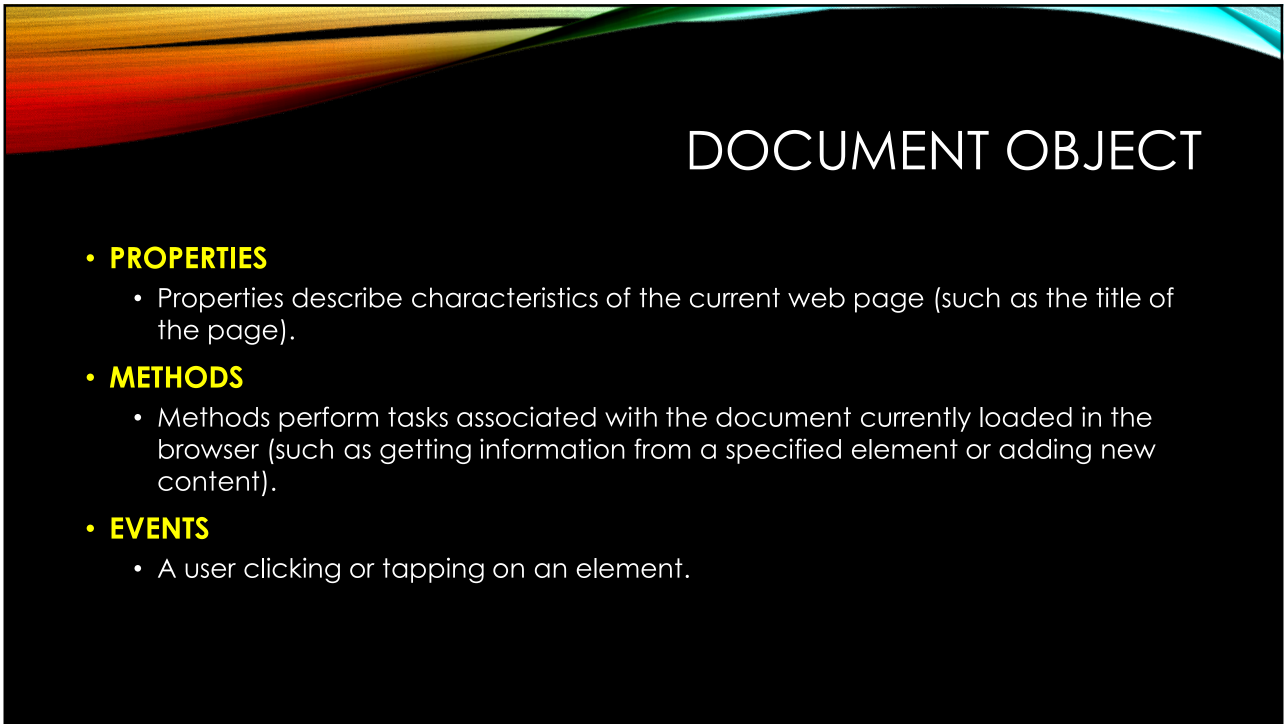
DOCUMENT OBJECT

- **DOCUMENT OBJECT**
- The current web page loaded into each window is modelled using a document object.
- The title property of the document object tells you what is between the opening <title> and closing </title> tag for that web page, and
- The Last Modified property of the document object tells you the date this page was last updated.



DOCUMENT OBJECT

- **THE DOCUMENT OBJECT REPRESENTS AN HTML PAGE**
- Using the document object, you can access and change what content users see on the page and respond to how they interact with it.
- Like other objects that represent real-world things, the document object has:
  - Properties
  - Methods
  - Events



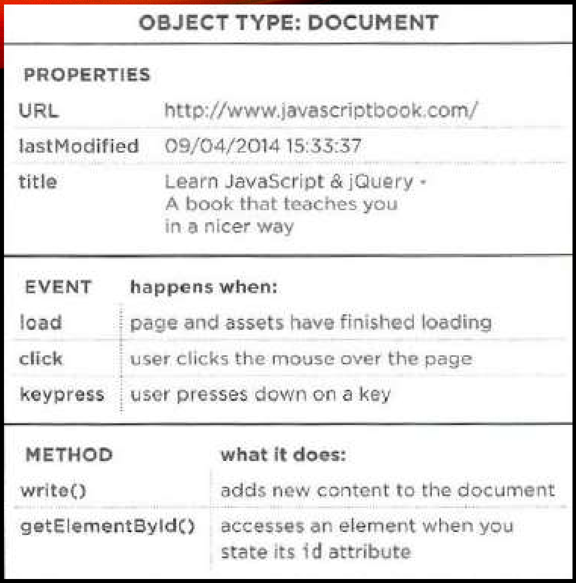
DOCUMENT OBJECT

- **PROPERTIES**
  - Properties describe characteristics of the current web page (such as the title of the page).
- **METHODS**
  - Methods perform tasks associated with the document currently loaded in the browser (such as getting information from a specified element or adding new content).
- **EVENTS**
  - A user clicking or tapping on an element.

DOCUMENT OBJECT

- The document object is just one of a set of objects that all major browsers support.
- When the browser creates a model of a web page, it not only creates a document object, but it also creates a new object for each element on the page.
- Together these objects are described in the **Document Object Model** (DOM).

DOCUMENT OBJECT



OBJECT TYPE: DOCUMENT

PROPERTIES	
URL	http://www.javascriptbook.com/
lastModified	09/04/2014 15:33:37
title	Learn JavaScript & jQuery - A book that teaches you in a nicer way

EVENT	happens when:
load	page and assets have finished loading
click	user clicks the mouse over the page
keypress	user presses down on a key

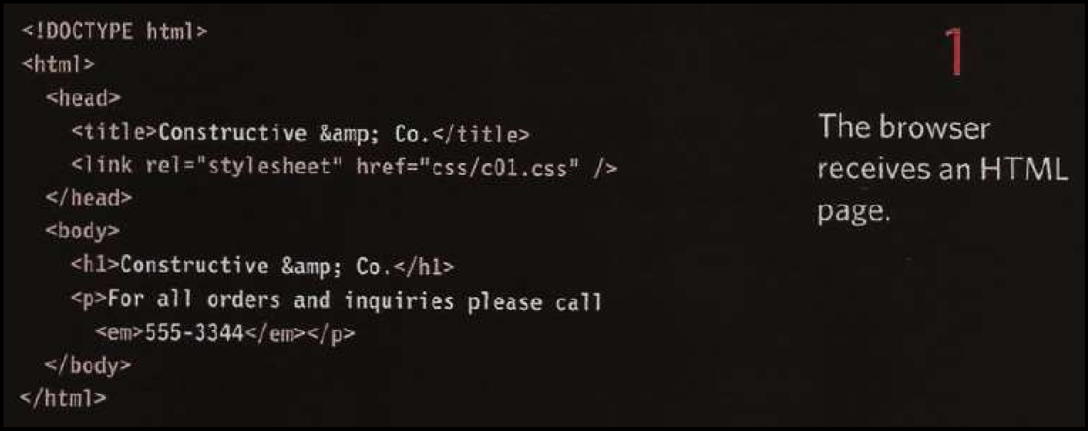
METHOD	what it does:
write()	adds new content to the document
getElementById()	accesses an element when you state its id attribute

BROWSER &WEB PAGE

- **HOW A BROWSER SEES A WEB PAGE**
  1. Receive a page as html code
  2. Create a model of the page and store it in memory
  3. Use a rendering engine to show the page on screen

- All major browsers use a JavaScript interpreter to translate your instructions (in JavaScript) into instructions the computer can follow.

BROWSER &WEB PAGE

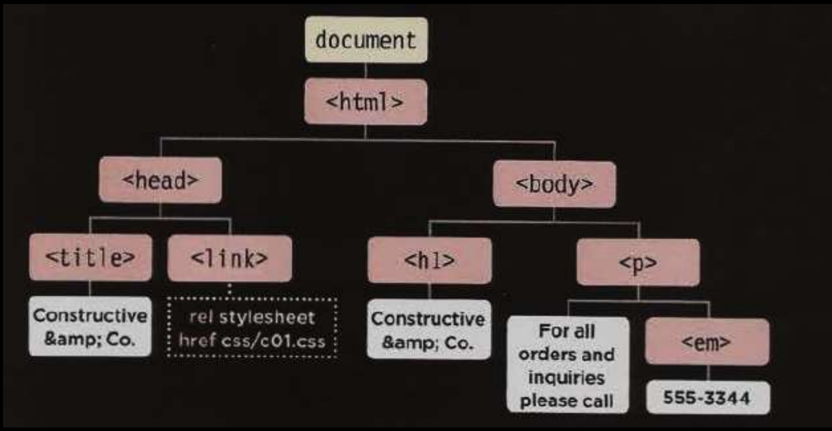


```
<!DOCTYPE html>
<html>
  <head>
    <title>Constructive &amp; Co.</title>
    <link rel="stylesheet" href="css/c01.css" />
  </head>
  <body>
    <h1>Constructive &amp; Co.</h1>
    <p>For all orders and inquiries please call
      <em>555-3344</em></p>
  </body>
</html>
```

1

The browser receives an HTML page.

BROWSER &WEB PAGE



document

<html>

<head>

<body>

<title>

<link>

<h1>

<p>

Constructive &amp; Co.

rel stylesheet href css/c01.css

Constructive &amp; Co.

For all orders and inquiries please call

<em>

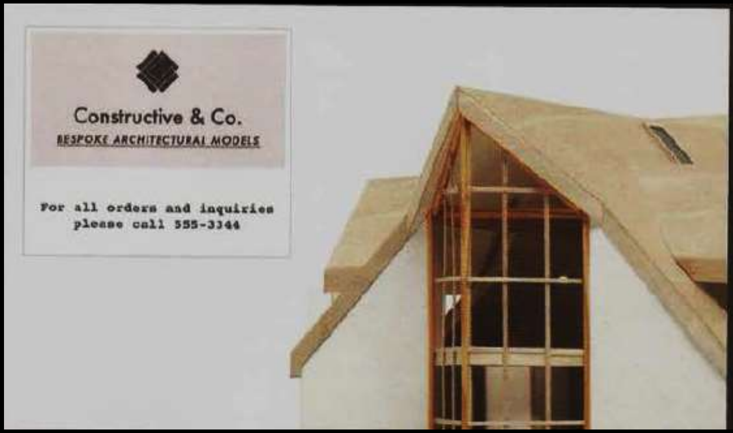
555-3344

2

It creates a model of the page and stores it in memory.

● OBJECT
● ELEMENT
● TEXT
⁙ ATTRIBUTES

BROWSER &WEB PAGE


Constructive & Co.
BESPOKE ARCHITECTURAL MODELS
For all orders and inquiries please call 555-3344

3

It shows the page on screen using a rendering engine.

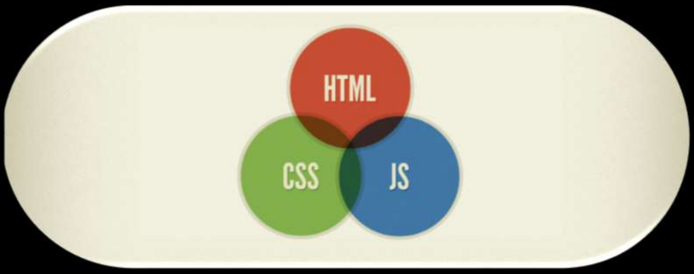
SUMMARY

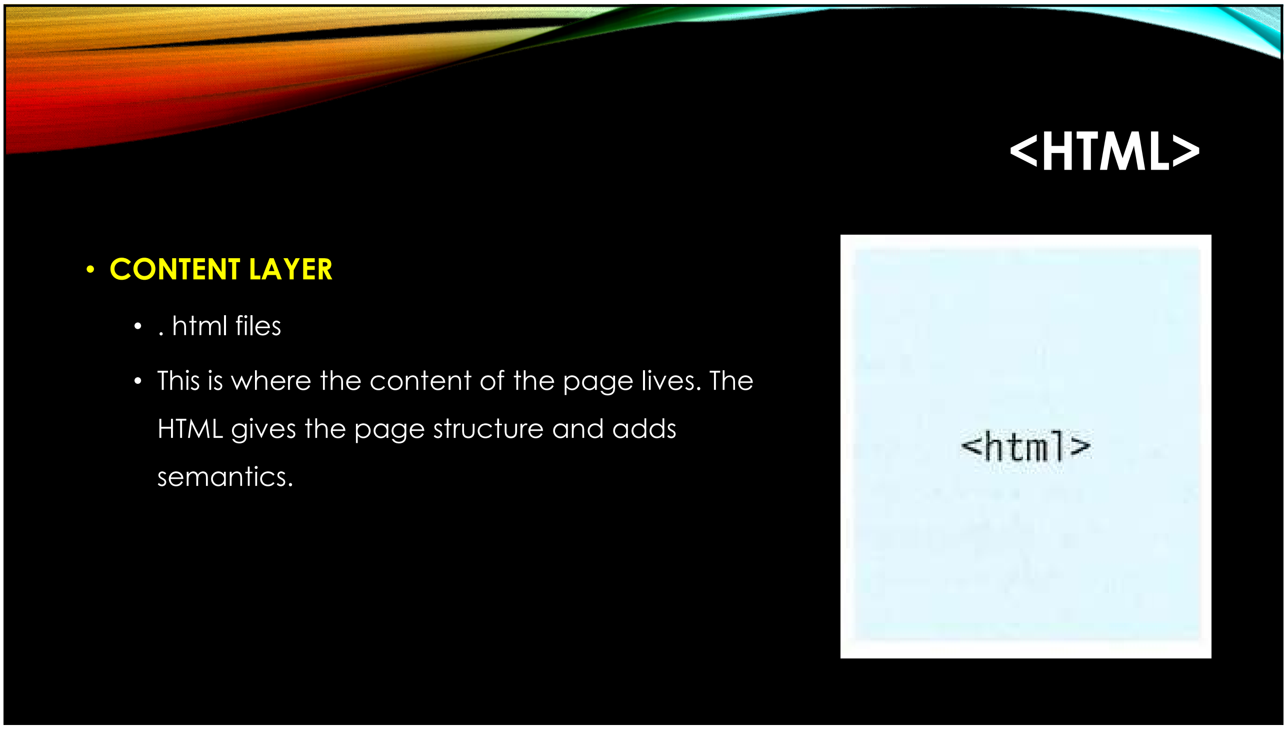
- How do computers fit in with the world around them?

1. Computers creates models of the world using data.
2. The models using objects to represents physical things. Objects can have properties, methods, and events.
3. Programmers can write code to say "When this event occurs, run that code".
4. Web browsers use HTML markup to create a model of the web page.
5. To make web pages interactive, write code that uses the browser's model of the web page.

C

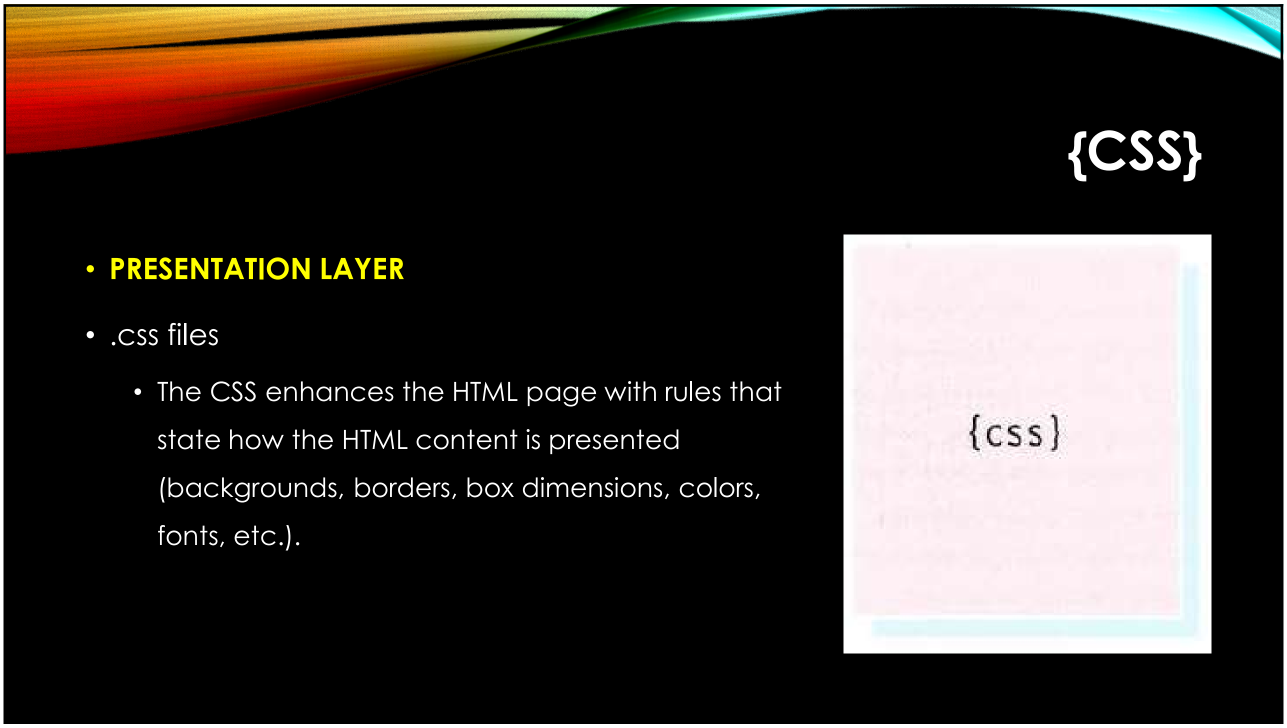
- **How do I write a script for a Web page?**
- First, you need to know how JS will fit together with the HTML and CSS in your web pages.





<HTML>

- **CONTENT LAYER**
  - . html files
  - This is where the content of the page lives. The HTML gives the page structure and adds semantics.

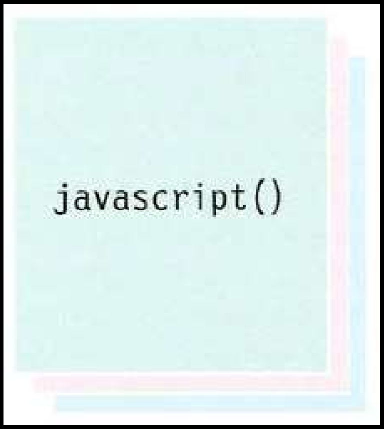


{CSS}

- **PRESENTATION LAYER**
- .css files
  - The CSS enhances the HTML page with rules that state how the HTML content is presented (backgrounds, borders, box dimensions, colors, fonts, etc.).

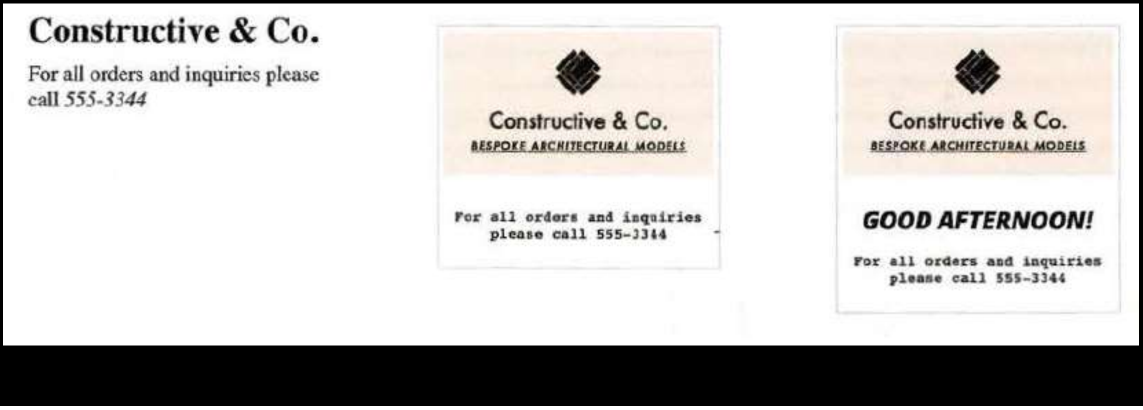
JAVASCRIPT()

- **BEHAVIOR LAYER**
- .js files
  - This is where we can change how the page behaves, adding interactivity. We will aim to keep as much of our JavaScript as possible in separate files.



PROGRESSIVE ENHANCEMENT

- These three layers form the basis of a popular approach to building web pages called progressive enhancement.


Constructive & Co.

For all orders and inquiries please call 555-3344

Constructive & Co.

BESPOKE ARCHITECTURAL MODELS

For all orders and inquiries please call 555-3344

Constructive & Co.

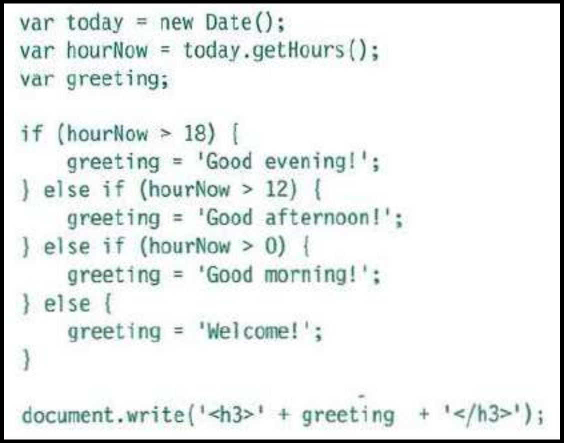
BESPOKE ARCHITECTURAL MODELS

GOOD AFTERNOON!

For all orders and inquiries please call 555-3344

CREATING A BASIC JAVASCRIPT

- This example adds a greeting into an HTML page. The greeting changes depending on the time of day.



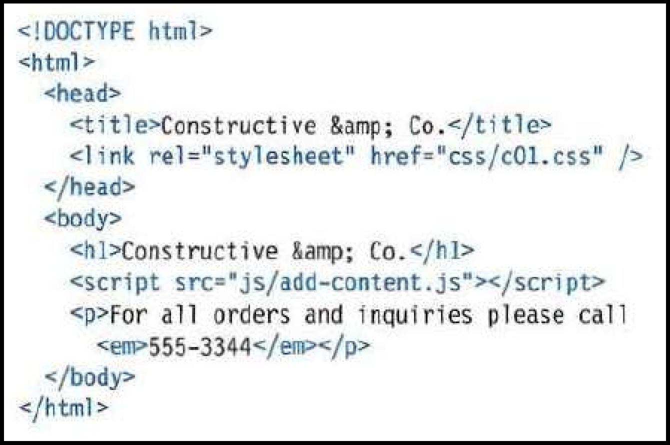
```
var today = new Date();
var hourNow = today.getHours();
var greeting;

if (hourNow > 18) {
    greeting = 'Good evening!';
} else if (hourNow > 12) {
    greeting = 'Good afternoon!';
} else if (hourNow > 0) {
    greeting = 'Good morning!';
} else {
    greeting = 'Welcome!';
}

document.write('<h3>' + greeting  + '</h3>');
```

LINKING HTML TO JS

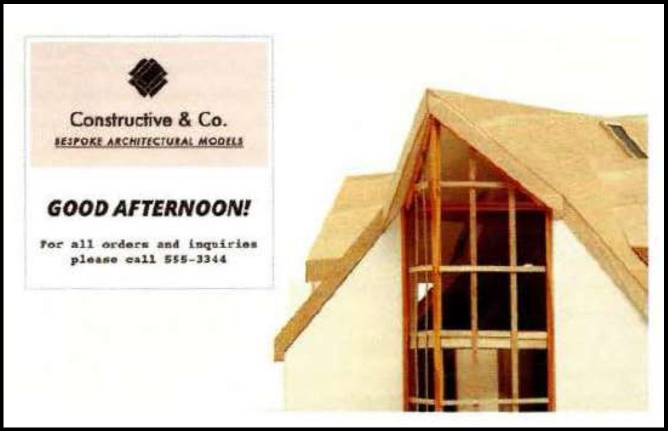
- When you want to use JavaScript with a web page, you use the HTML <script> element to tell the browser it is coming across a script.
- Its **src** attribute tells people where the JavaScript file is stored.



```
<!DOCTYPE html>
<html>
  <head>
    <title>Constructive &amp; Co.</title>
    <link rel="stylesheet" href="css/c01.css" />
  </head>
  <body>
    <h1>Constructive &amp; Co.</h1>
    <script src="js/add-content.js"></script>
    <p>For all orders and inquiries please call
      <em>555-3344</em></p>
  </body>
</html>
```

RESULT

- After applying CSS rules
- **Please note:** Internet Explorer sometimes prevents JavaScript running when you open a page stored on your hard drive. If this affects you, please try Chrome, Firefox, Opera, or Safari instead.



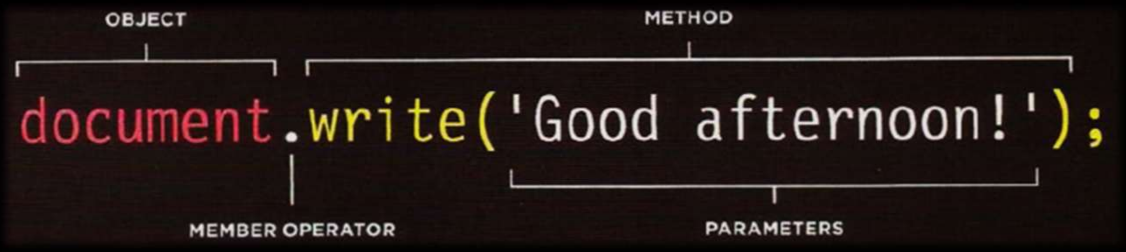
Constructive & Co.
BESPOKE ARCHITECTURAL MODELS

GOOD AFTERNOON!

For all orders and inquiries
please call 555-3344

OBJECTS & METHODS

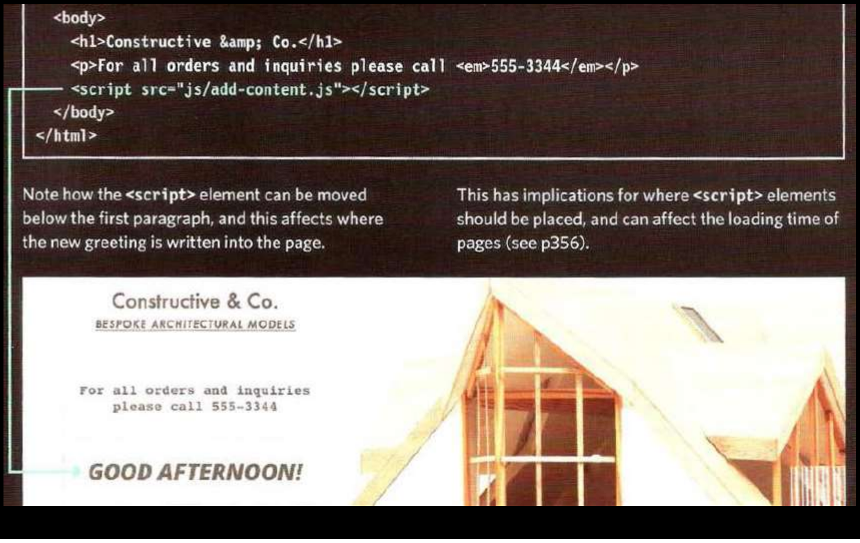
- **How to use objects and methods?**



```
document.write('Good afternoon!');
```

RUNNING JAVASCRIPT

- JavaScript runs where it is found in the HTML.



```
  <body>
    <h1>Constructive &amp; Co.</h1>
    <p>For all orders and inquiries please call <em>555-3344</em></p>
    <script src="js/add-content.js"></script>
  </body>
</html>
```

Note how the <script> element can be moved below the first paragraph, and this affects where the new greeting is written into the page.

This has implications for where <script> elements should be placed, and can affect the loading time of pages (see p356).

Constructive & Co.

BESPOKE ARCHITECTURAL MODELS

For all orders and inquiries please call 555-3344

GOOD AFTERNOON!

SUMMARY

- **How do I write a script for a Web page?**
  - It is best to keep JavaScript code in its own JavaScript file. JavaScript files are text files (like HTML pages and CSS style sheets), but they have the . j s extension.
  - The HTML <script> element is used in HTML pages to tell the browser to load the JavaScript file (rather like the <link> element can be used to load a CSS file).
  - If you view the source code of the page in the browser, the JavaScript will not have changed the HTML, because the script works with the model of the web page that the browser has created.

Thank you!