



# SVILUPPATORE FRONT-END

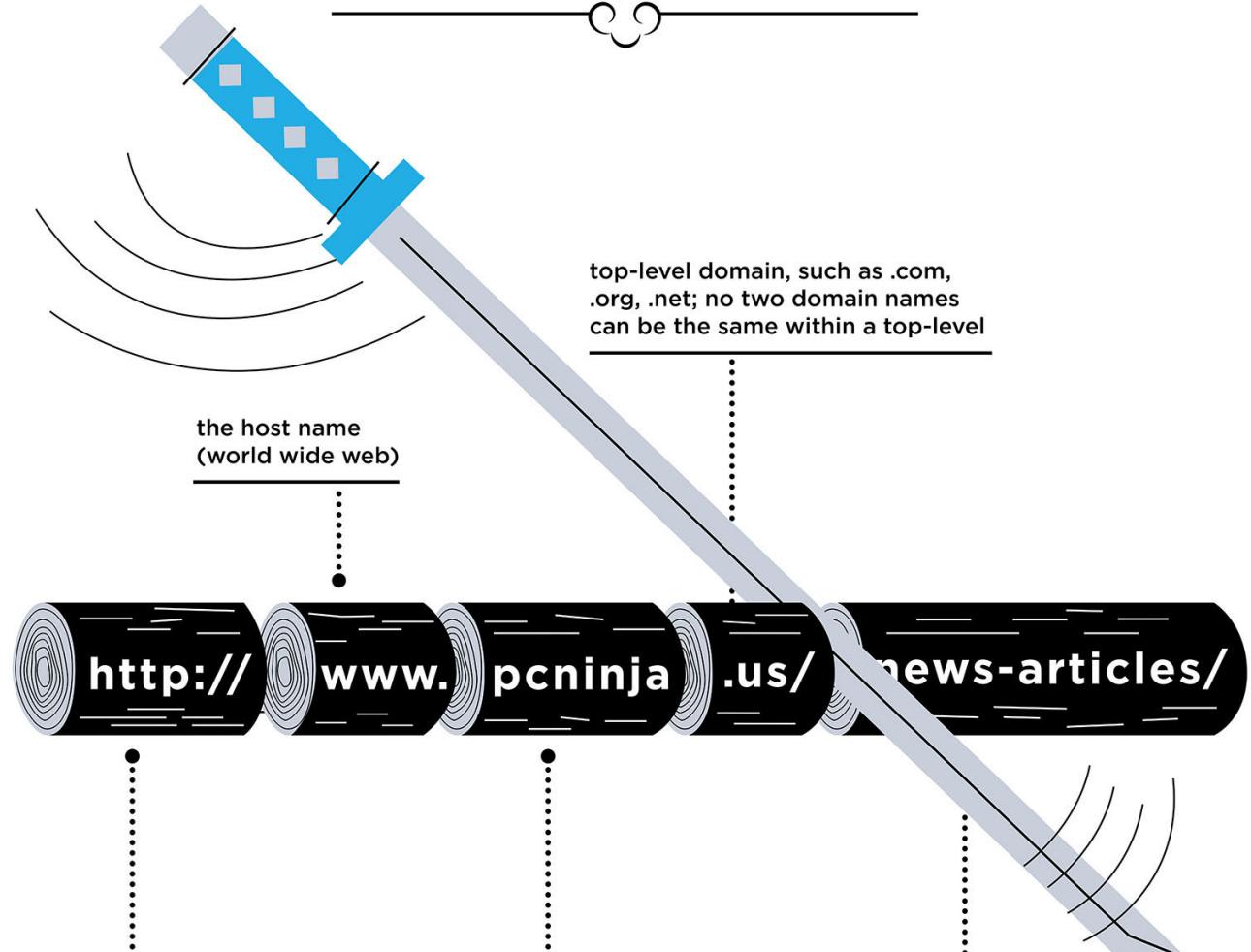
---

*Giulia Maddalena*

# INTERNET

## DISSECTING A URL

URL = UNIFORM RESOURCE LOCATOR, THE ADDRESS OF A SPECIFIC WEBSITE OR FILE ON THE INTERNET, LIKE A HOME ADDRESS



The URL really says, "Use the Web (http://) to find a host server named 'www' in the 'pcninja.us' network, and look in the 'news-articles' folder so I can pull out a particular file."



## HOW THE INTERNET WORKS

SO HOW EXACTLY DOES THE INTERNET WORK? What's going on in that cloudy, tangled web? A lot of little ninja are working at super speeds to bring you the data you seek!

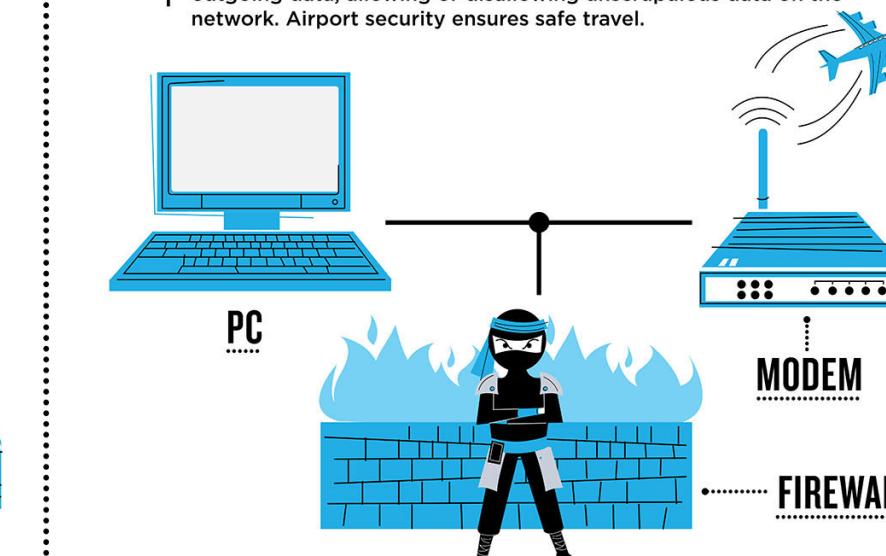
### STEP 1 | DEPARTURE

You type 'pcninja.us' into the web browser of your computer. Hop in the ninja mobile, and prepare for an adventure!



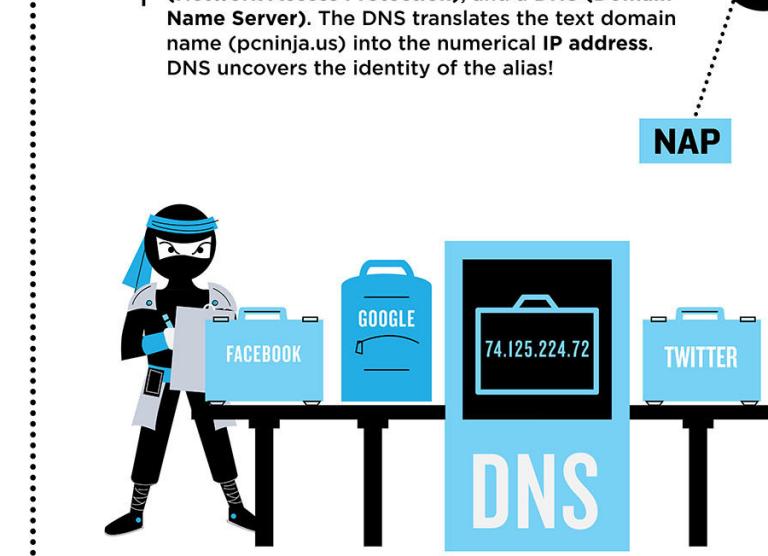
### STEP 2 | THE AIRPORT

Your computer is connected to the Internet through a **modem** and/or **router**, a jumping-off point to other networks in the world. **Firewalls**, in your browser and/or modem, monitor incoming and outgoing data, allowing or disallowing unscrupulous data on the network. Airport security ensures safe travel.



### STEP 3 | CUSTOMS

Your query is submitted to your **ISP** (Internet Service Provider). Within your ISP are multiple servers to store and transfer data, a **NAP** Server (Network Access Protection), and a **DNS** (Domain Name Server). The DNS translates the text domain name (pcninja.us) into the numerical IP address. DNS uncovers the identity of the alias!



### STEP 4 | BUCKLE UP

Your browser maps itself to the desired IP address and establishes the hypertext transfer protocol (http), or language used to communicate on the World Wide Web. Good idea to take a pocket translator!



### STEP 5 | TAKE OFF

With the destination identified and language acquired, the ninja finds the speediest way possible to reach the server with that IP address. It travels through the cloud, which is made up of all the other ISPs and servers to which all the other home and business networks connect, to form that network of networks, the Internet. Not a direct flight, but no long layovers with these multiple carriers!



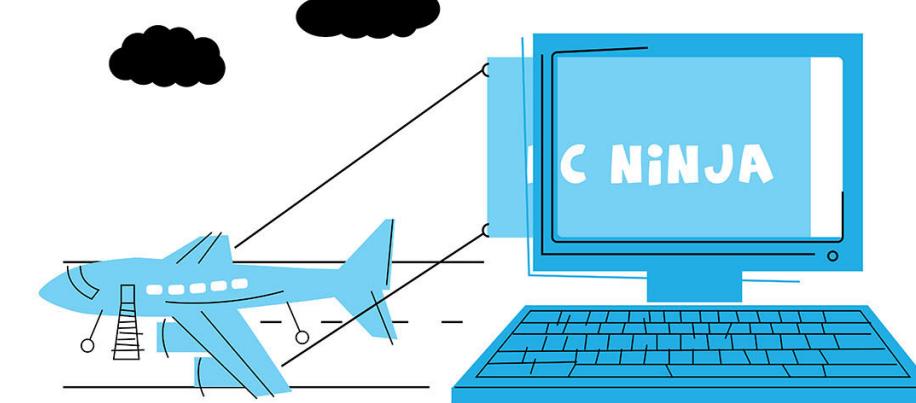
### STEP 6 | LANDING

Jumping from server to server on the Web, the ninja finally locates the target server hosting the target IP address for pcninja.us. A connection is established with that website and your computer. Please make sure your seats are in the upright position, we're ready to land!



### STEP 7 | INCOMING!

The ninja makes an even quicker return journey, bringing to your computer screen the graphical website of pcninja.us, which is full of data, pictures, and contact information. Thank you for flying with PC Ninja, and enjoy your browse!





# Niente pippe sulla storia del web

# ALCUNE DEFINIZIONI

---

- **Browser:** è un'applicazione per l'acquisizione, la presentazione e la navigazione di risorse sul web.  
*Oggi non c'è più il problema di quale browser ma di quale device.*
- **DNS:** è acronimo di Domain Name System è il sistema utilizzato per la conversione dei nomi a dominio nei corrispettivi indirizzi IP, ogni server ha un indirizzo IP.
- **HTTP:** è un protocollo per lo scambio di informazioni in internet.
- **Web Hosting:** è un servizio che permette di mettere il nostro sito nel web.

# STRUTTURA DEL TEAM

---

Ruoli necessari allo sviluppo di un sito web, web app o e-commerce:

- Ui designer: wireframe, iterazione ed usabilità
- Visual Design: fonts, colori, spazi, Visual, etc..
- Front-end Developer: codice lato client, cura qualsiasi aspetto grafico del sito
- Back-end Developer: codice lato server, permette al sito di esser dinamico e interattivo.



## COS'È LO SVILUPPATORE FRONT-END

---

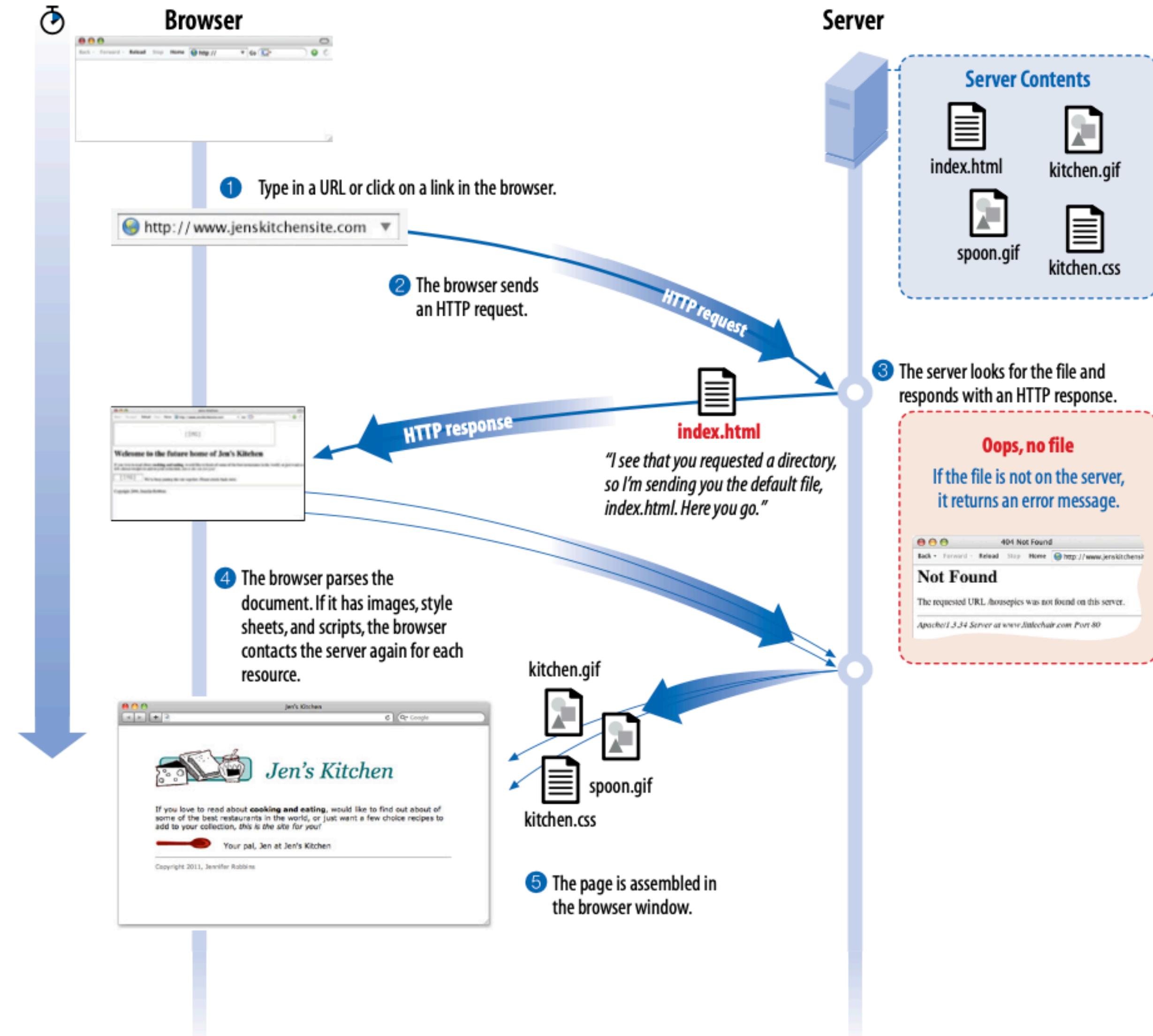
Front-end web development, also known as client-side development is the practice of producing HTML, CSS and JavaScript for a website or Web Application so that a user can see and interact with them directly. The challenge associated with front end development is that the tools and techniques used to create the front end of a website change constantly and so the developer needs to constantly be aware of how the field is developing.

# TECNOLOGIE UTILIZZATE

---

- HTML5
- CSS3
- Bootstrap / font awesome
- Responsive app
- Javascript
- Ajax
- jQuery
- CMS

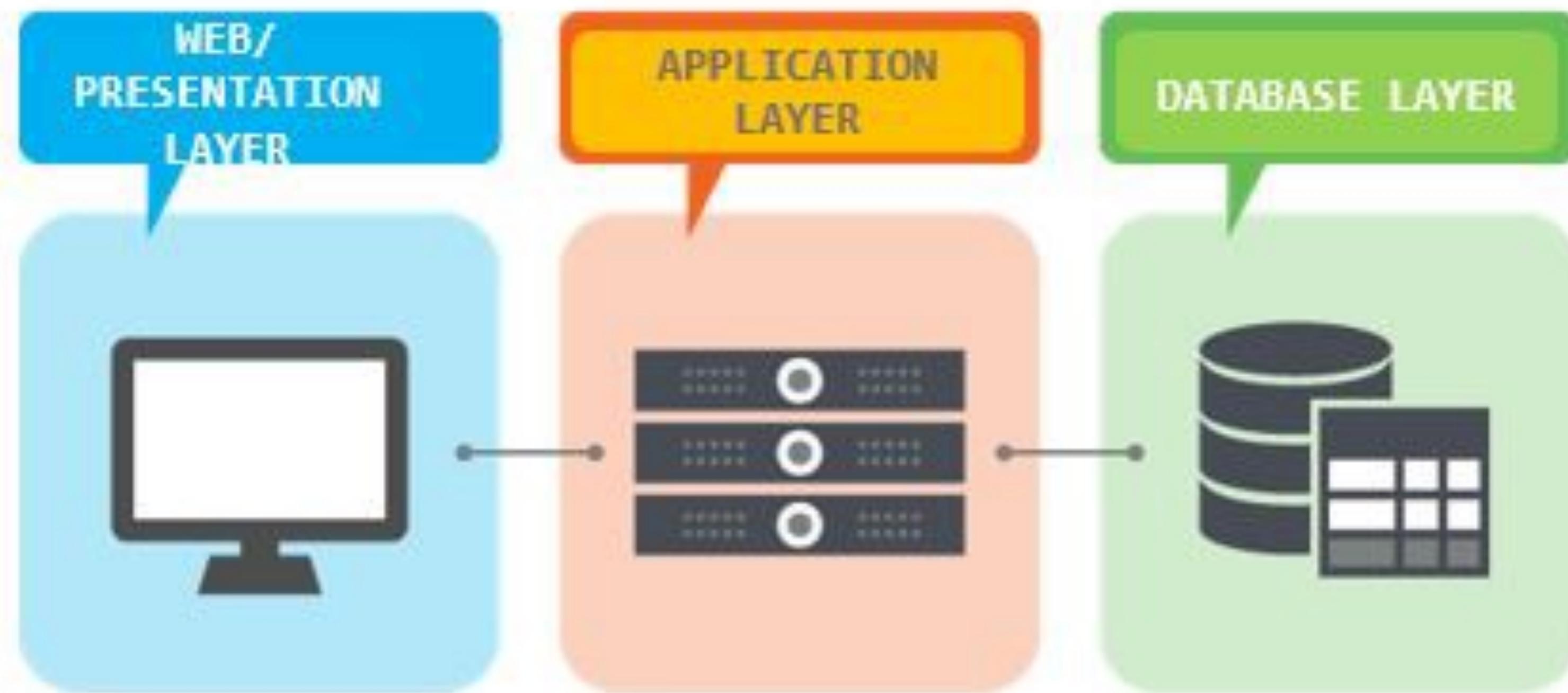
# ANATOMIA DI UNA PAGINA WEB



- 1 Richiesta di una pagina web digitando un url
- 2 Il browser invia la richiesta al server
- 3 Il server gestisce la richiesta
- 4 Il browser interpreta il documento html ed inserisce le eventuali immagini

# ANATOMIA DI UN SITO WEB

---



# HTML5

---

- È un linguaggio di markup, non un linguaggio di programmazione
- Formattazione della pagina tramite tag
- Definisce la struttura del contenuto
- Nuovi tag che definiscono il tipo di contenuto (audio, video, articolo, ecc..)
- Nuovi elementi per i form
- Supporto agli SVG e/o canvas

# REGOLE

---

- Suffisso nei file
- Niente spazi sui nomi dei file
- No caratteri speciali (?, \$, /, :, ;, etc... )
- I nomi dei file sono case-sensitive

# STRUTTURA DI BASE

---

```
1  <!DOCTYPE html>
    <html>
        2      <head>
            3          <meta charset="utf-8"> 4
            <title>Title here</title> 5
        </head>
        2      <body>
            6              Page content goes here.
        </body>
    </html>
```

1. Dichiarazione del tipo di documento
2. Root del documento
3. Informazioni descrittive
6. Tutto ciò che dev'esser visualizzato sul browser

# DOCTYPE

---

HTML 4.01 Strict	<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
HTML 4.01 Transitional	<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
HTML 4.01 Frameset	<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN" "http://www.w3.org/TR/html4/frameset.dtd">
XHTML 1.0 Strict (quick reference)	<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
XHTML 1.0 Transitional	<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
XHTML 1.0 Frameset	<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">
XHTML 1.1 – DTD	<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN" "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
XHTML Basic 1.1 (quick reference)	<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML Basic 1.1//EN" "http://www.w3.org/TR/xhtml-basic/xhtml-basic11.dtd">
HTML5	<!DOCTYPE HTML>

# ESERCIZIO 1

# TAG HTML

---

- Nomi differenti a seconda della loro funzione
- <tag attributo=“valore”> contenuto </tag>
- Tag vuoto<img width=“” src=“nome\_img.jpg” alt=“”>
- Annidamento e indentazione
- Tre tipi di tag
  - Elementi di blocco: div, p, table, etc...
  - Elementi inline: span, a, img
  - Liste
- Commenti <!-- commento -->

# **ESERCIZIO 2**

# ELEMENTI DI TESTO

---

- <h1>Black Goose Bistro</h1>
- <h2>The Restaurant</h2>
- <p></p>
- <em></em>

# ESEMPIO

---

The screenshot shows a web browser window with a light gray header bar. In the top left corner are three colored window control buttons (red, yellow, green). The title bar in the center contains the text "Black Goose Bistro" next to a small globe icon. To the right of the title bar are standard browser controls: a back arrow, a forward arrow, a refresh button, a star for bookmarks, and a magnifying glass for search.

The main content area of the browser displays the "Black Goose Bistro" website. The page has a white background with several sections outlined by red borders:

- Black Goose Bistro**: The main title at the top.
- The Restaurant**: A section describing the restaurant's offerings.
- Catering**: A section about catering services.
- Location and Hours**: Information about the restaurant's location and operating hours.

Within these sections, there is descriptive text and some blue highlighted text. For example, in the "Catering" section, the phrase "we'll handle the cooking" is highlighted in blue, and in the "Location and Hours" section, the word "Seekonk" is highlighted in blue.

# MARKUP SBAGLIATI

---

- a. <img "birthday.jpg">
- b. <i>Congratulations!</i>
- c. <a href="file.html">linked text</a href="file.html">
- d. <p>This is a new paragraph<\p>

# TABELLA TAG DI BASE

---

Table 1.1: List of basic tags

Tag	Description	Example
h1, ..., h6	Header tag h1 to h6	<h2> Hi </h2>
p	paragraphs (Line changes at the end)	<p> Hi </p>
span	No line change after span	<span>Hi</span> Bye.
div	make division between contents	<div> ... </div>
a	hyperlink	see <a href="#">Section 1.9</a>
center	Move content to center	<center> Hi </center>
br	Line break (no closing tag)	  or  
hr	horizontal line (no closing tag)	<hr /> or <hr>
pre	preserve formatting	<pre> ... </pre>
table	insert table	see <a href="#">Section 1.5</a>

# LIST

---

- Ogni elemento va racchiuso tra <li></li>
- Non ordinata <ul></ul>
- Ordinata <ol></ol>
- Lista descrittiva

```
<dl>  
  <dt></dt>  
  <dd></dd>  
</dl>
```

- Attributo type definisce lo stile dell'elenco
- Attributo start indica il valore di partenza
- Attributo reversed indica l'ordinamento decrescente

# ATTRIBUTO TYPE

---

Valore dell'attributo type	Descrizione	Codice	Output
type="1" (default)	numeri arabi	<pre>&lt;ol type="1"&gt;   &lt;li&gt;primo&lt;/li&gt;   &lt;li&gt;secondo&lt;/li&gt;   &lt;li&gt;terzo&lt;/li&gt; &lt;/ol&gt;</pre>	1. primo 2. secondo 3. terzo
type="a"	alfabeto minuscolo	<pre>&lt;ol type="a"&gt;   &lt;li&gt;primo&lt;/li&gt;   &lt;li&gt;secondo&lt;/li&gt;   &lt;li&gt;terzo&lt;/li&gt; &lt;/ol&gt;</pre>	a. primo b. secondo c. terzo
type="A"	alfabeto maiuscolo	<pre>&lt;ol type="A"&gt;   &lt;li&gt;primo&lt;/li&gt;   &lt;li&gt;secondo&lt;/li&gt;   &lt;li&gt;terzo&lt;/li&gt; &lt;/ol&gt;</pre>	A. primo B. secondo C. terzo
type="i"	numeri romani minuscoli	<pre>&lt;ol type="i"&gt;   &lt;li&gt;primo&lt;/li&gt;   &lt;li&gt;secondo&lt;/li&gt;   &lt;li&gt;terzo&lt;/li&gt; &lt;/ol&gt;</pre>	i. primo ii. secondo iii. terzo
type="I"	numeri romani maiuscoli	<pre>&lt;ol type="I"&gt;   &lt;li&gt;primo&lt;/li&gt;   &lt;li&gt;secondo&lt;/li&gt;   &lt;li&gt;terzo&lt;/li&gt; &lt;/ol&gt;</pre>	I. primo II. secondo III. terzo

# ELEMENTI HTML5

---

- Blackquote: citazione, testimonial, ecc...
- Figure e figcaption

```
<figure>
  
  <figcaption>An elephant at sunset</figcaption>
</figure>
```

- Sections and Articles: elementi di sezione.
- Aside <aside></aside>
- Navigation <nav></nav>
- Header e Footer
- Addresses <address></address> area di informazioni di contatto

# ESEMPI

---

```
<article>
  <header>
    <h1>More about WOFF</h1>
    <p>by Jennifer Robbins, <time datetime="11-11-2011"
       pubdate>November 11, 2011</time></p>
  </header>
  <p>...article content starts here...</p>
  <footer>
    <p><small>Copyright © 2012 Jennifer Robbins.</small></p>
    <nav>
      <ul>
        <li><a href="">Previous</a></li>
        <li><a href="">Next</a></li>
      </ul>
    </nav>
  </footer>
</article>
```

# INLINE ELEMENTS

---

- Keywords or visually emphasized text (bold) <strong>...</strong>
- Alternative voice (italic) <i>...</i>
- Incorrect text (strike-through) <s>...</s>
- Annotated text (underline) <u>...</u>
- Legal text; small print (smaller type size) <small>...</small>
- Line break <br>
- Emphasized <em></em>
- Subscript e Superscript <sub> e <sup>

# **ESERCIZIO 3**

# ELEMENTI GENERICI DIV E SPAN

---

- Usati quando non c'è alcun tag specifico
- Nessuna formattazione di default
- Utili per la strutturazione logica dei contenuti, lo stile o gli script

# ATTRIBUTI

---

Servono per meglio specificare la funzione o la tipologia dell'elemento, per memorizzare dati o per arricchire di significato il contenuto

- Sono coppie *chiave-valore* separate dal carattere = (uguale);
- I valori sono tipicamente racchiusi *tra virgolette*, ma è possibile anche utilizzare gli apici
- Si scrivono lasciando almeno uno spazio dopo il nome dell'elemento nel tag di apertura (o nell'unico tag nel caso di elementi non contenitori).

# ATTRIBUTI GLOBALE

---

- **ID:** identifica in modo univoco un elemento
- **Class:** classifica più elementi
- **Lang:** specifica la lingua
- **Style:** specifica uno stile in linea
- **Title:** fornisce un informazione aggiuntiva all'elemento

# ATTRIBUTI

---

Table 1.2: List of attributes

Name	Values	Description
id	user defined names	<p id='p_1'> Hi </p>
class	user defined names	<p class='p_class'> Hi </p>
style	CSS styles	<p style="color:red; font-weight:bold;"> Hi </p>
align	left, right, center	horizontal alignment
width	numeric value or % value	width of images and tables etc.
height	numeric value	height of images and tables etc.

# CARATTERI SPECIALI

---

Character	Description	Name	Number
	Character space (nonbreaking space)	&nbsp	&#160;
&	Ampersand	&amp;	&#038;
'	Apostrophe	&apos;	&#039;
<	Less-than symbol (useful for displaying markup on a web page)	&lt;	&#060;
>	Greater-than symbol (useful for displaying markup on a web page)	&gt;	&#062;
©	Copyright	&copy;	&#169;
®	Registered trademark	&reg;	&#174;
™	Trademark	&trade;	&#8482;

# META TAG

---

- Contengono i metadati della pagina (info contenutistiche)
- Da inserire tra i tag head
- Identificati dal tag <meta>

```
<head>
    <meta name="description" content="Free Web tutorials">
    <meta name="keywords" content="HTML,CSS,XML,JavaScript">
    <meta name="author" content="Web Master">
    <meta charset="UTF-8">
</head>
```

# **ESERCIZIO 4**



# LINGUAGGI DI PROGRAMMAZIONE

---

# LATO CLIENT E LATO SERVER

---

Gli **Script lato Client** vengono interpretati dal Browser. Il codice di scripting, inserito nel codice HTML, è contrassegnato dai tag <SCRIPT> e </SCRIPT>. In particolare:

1. il browser richiede una pagina al server
2. il server invia la pagina al browser
3. il browser invia la pagina al motore di script
4. il motore interpreta ed esegue lo script
5. il browser visualizza la pagina

# LATO CLIENT E LATO SERVER

---

Gli **Script lato Server** sono utilizzati per aggiungere interattività alle pagine web e permettono di interagire con altri programmi residenti sul server, oltre a poter interagire con eventuali database installati sul server. Gli script fanno parte della pagina, ma vengono eseguiti sul server. Al browser viene inviato solo codice HTML, quindi il codice sorgente non compare mai.

In particolare:

1. il browser richiede una pagina al server
2. il server invia lo script al motore
3. il motore interpreta ed esegue lo script
4. il server invia la pagina di risposta al browser in formato HTML
5. il browser visualizza la pagina

# LINGUAGGI DI PROGRAMMAZIONE WEB

---

## LATO CLIENT

- **Javascript:** è un linguaggio di scripting. Gli script Javascript servono a creare interattività nelle pagine Web ed è un linguaggio di scripting che viene eseguito localmente sul browser dell'utente, così il sito internet può rispondere alle interazioni dello stesso.

## LATO SERVER

- **PHP:** possiamo dire è il leader dei linguaggi lato server usati oggi su internet, il codice viene elaborato prima che raggiunga il browser dell'utente, quindi, quello che tutti vedono, è l'elaborazione finale e nessuno può sapere il codice PHP originale.

# LINGUAGGI DI PROGRAMMAZIONE WEB

---

- **ASP.net:** è il linguaggio di programmazione per i siti web su server Microsoft. Asp.net serve a creare contenuti dinamici ed interattivi ed è soprattutto utilizzato negli ambienti aziendali dove si utilizzano server Windows.
- **Python:** è un linguaggio di programmazione orientato agli oggetti, facile da usare e ottimo per l'apprendimento. Python è open source e viene usato anche da Google, Yahoo e, soprattutto, da Youtube proprio perchè semplice da leggere e da modificare per fare aggiornamenti e manutenzione.
- **Ruby:** è un linguaggio usato da siti famosi come Groupon, Shopify e Twitter. Si tratta di un linguaggio veloce e con meno ripetizioni, che viene interpretato dal server e poi inviato come pagina HTML completa al browser (come avviene per il PHP). Purtroppo, la maggior parte dei servizi di hosting non supportano Ruby quindi bisognerebbe sapere come configurare un proprio webserver.

# COME SCEGLIERE UN LINGUAGGIO DI PROGRAMMAZIONE

---

- **Velocità di apprendimento:** Quanto è semplice e veloce da apprendere?
- **Applicazioni:** Dove usare il linguaggio? Cosa ci puoi fare? Per quali scopi è usato?
- **Richiesta nel lavoro:** Quanto è richiesto nel mondo del lavoro? Per quali mansioni?
- **Risorse:** Esistono libri, tutorial e corsi per iniziare a studiarlo? Sono di qualità? Gratis o a pagamento?

# UTILIZZO DEI LINGUAGGI DI PROGRAMMAZIONE

---

Sito web	Linguaggio di programmazione lato client	Linguaggio/i di programmazione lato server
Google	JavaScript	C, C++, Go, Java, Python, PHP (HHVM)
Facebook	JavaScript	Hack, PHP (HHVM), Python, C++, Java, Erlang, D, XHP, Haskell
YouTube	JavaScript	C, C++, Python, Java, Go
Yahoo	JavaScript	PHP
Amazon	JavaScript	Java, C++, Perl
Wikipedia	JavaScript	PHP, Hack
Twitter	JavaScript	C++, Java, Scala, Ruby

# LINGUAGGI UTILIZZATI LATO FRONT-END

---

# HTML

---

HyperText Markup Language, commonly referred to as HTML, is the standard markup language used to create web pages. Web browsers can read HTML files and render them into visible or audible web pages. HTML describes the structure of a website semantically along with cues for presentation, making it a markup language, rather than a programming language.

# CSS

---

Cascading Style Sheets (CSS) is a style sheet language used for describing the look and formatting of a document written in a markup language. Although most often used to change the style of web pages and user interfaces written in HTML and XHTML, the language can be applied to any kind of XML document, including plain XML, SVG and XUL. Along with HTML and JavaScript, CSS is a cornerstone technology used by most websites to create visually engaging webpages, user interfaces for web applications, and user interfaces for many mobile applications.

# JAVASCRIPT

---

JavaScript is a high level, dynamic, untyped, and interpreted programming language. It has been standardized in the ECMAScript language specification. Alongside HTML and CSS, it is one of the three essential technologies of World Wide Web content production; the majority of websites employ it and it is supported by all modern web browsers without plug-ins. JavaScript is prototype-based with first-class functions, making it a multi-paradigm language, supporting object-oriented, imperative, and functional programming styles. It has an API for working with text, arrays, dates and regular expressions, but does not include any I/O, such as networking, storage or graphics facilities, relying for these upon the host environment in which it is embedded.

# **ULTIME NOZIONI DI HTML5**

# LINK E IPERTESTUALITÀ

---

- L'unico tag che permette i collegamenti è l'anchor `<a>...</a>`
- Con l'HTML5 qualsiasi elemento può esser un elemento anchor.
- L'attributo href indica al browser a quale risorsa punta il link

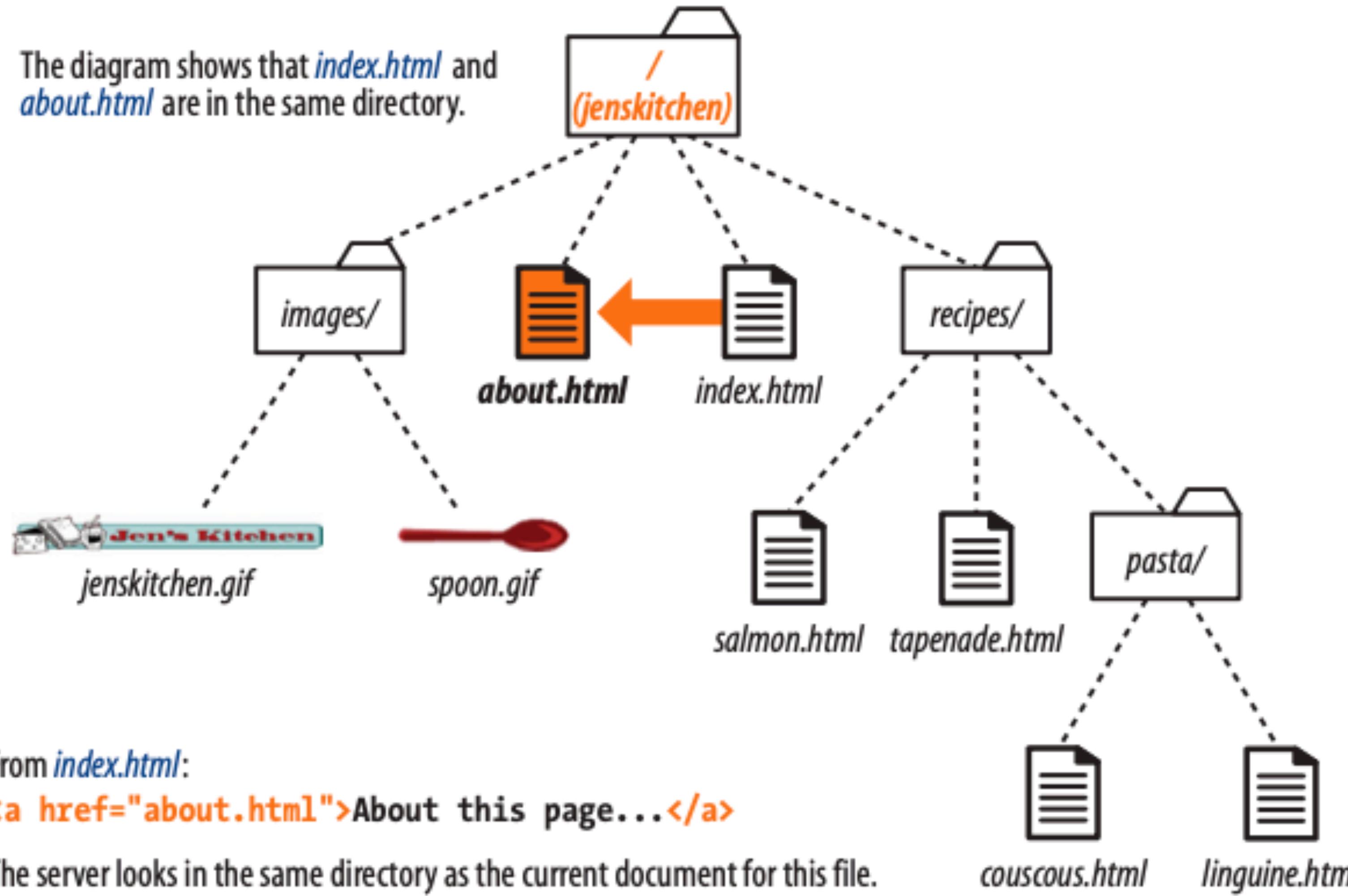
esempio: `<a href="http://www.demo.com/test">Go to page test</a>`

- Tipi di url
  - Assoluti: full url
  - Relativi: percorso da dove si trova il file
- # link ad un punto specifico della pagina
- Attributo `target="_blank"` apre il link in una nuova pagina

# ESEMPIO DI STRUTTURA

---

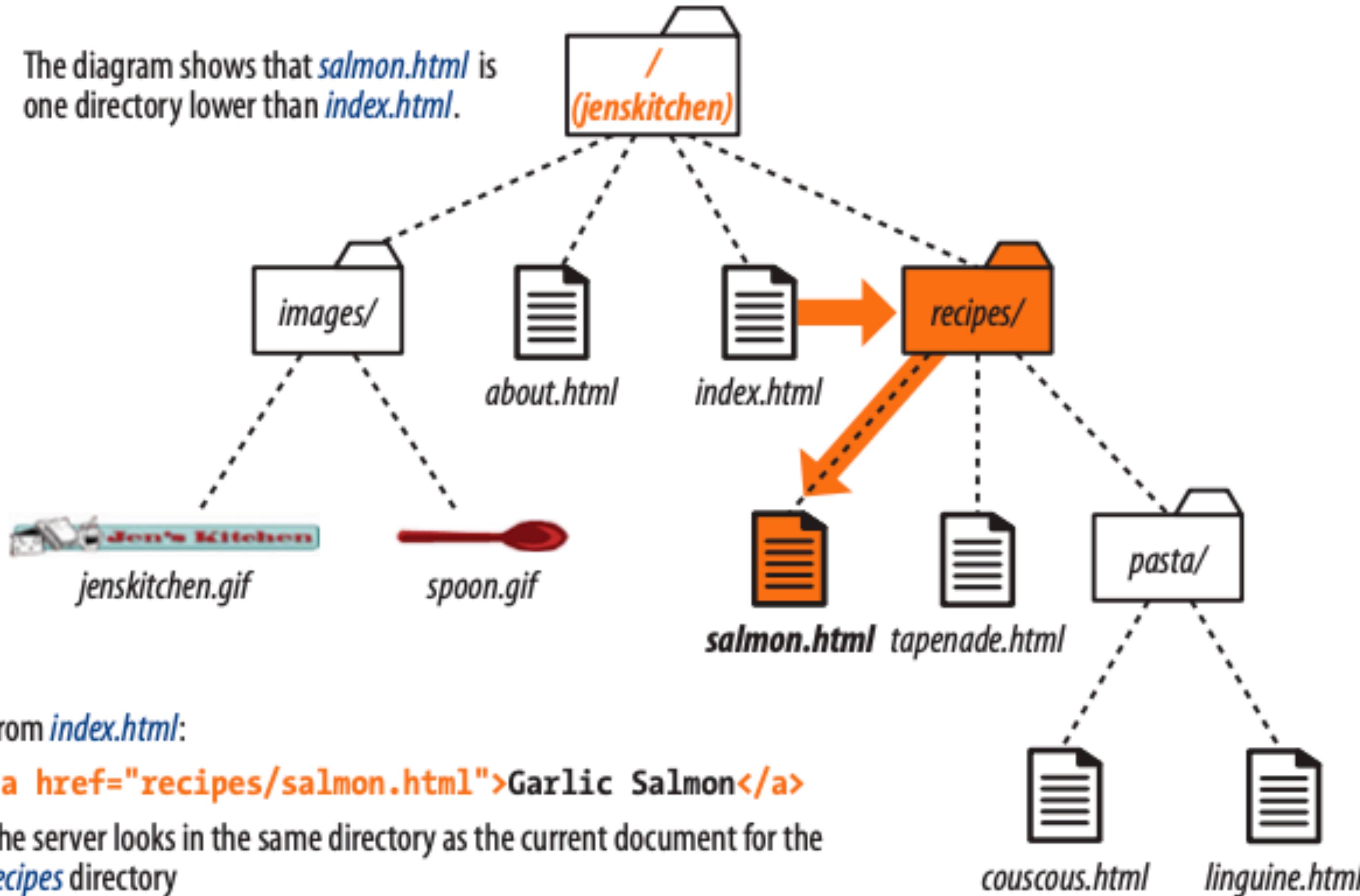
The diagram shows that *index.html* and *about.html* are in the same directory.



# ESEMPIO DI STRUTTURA

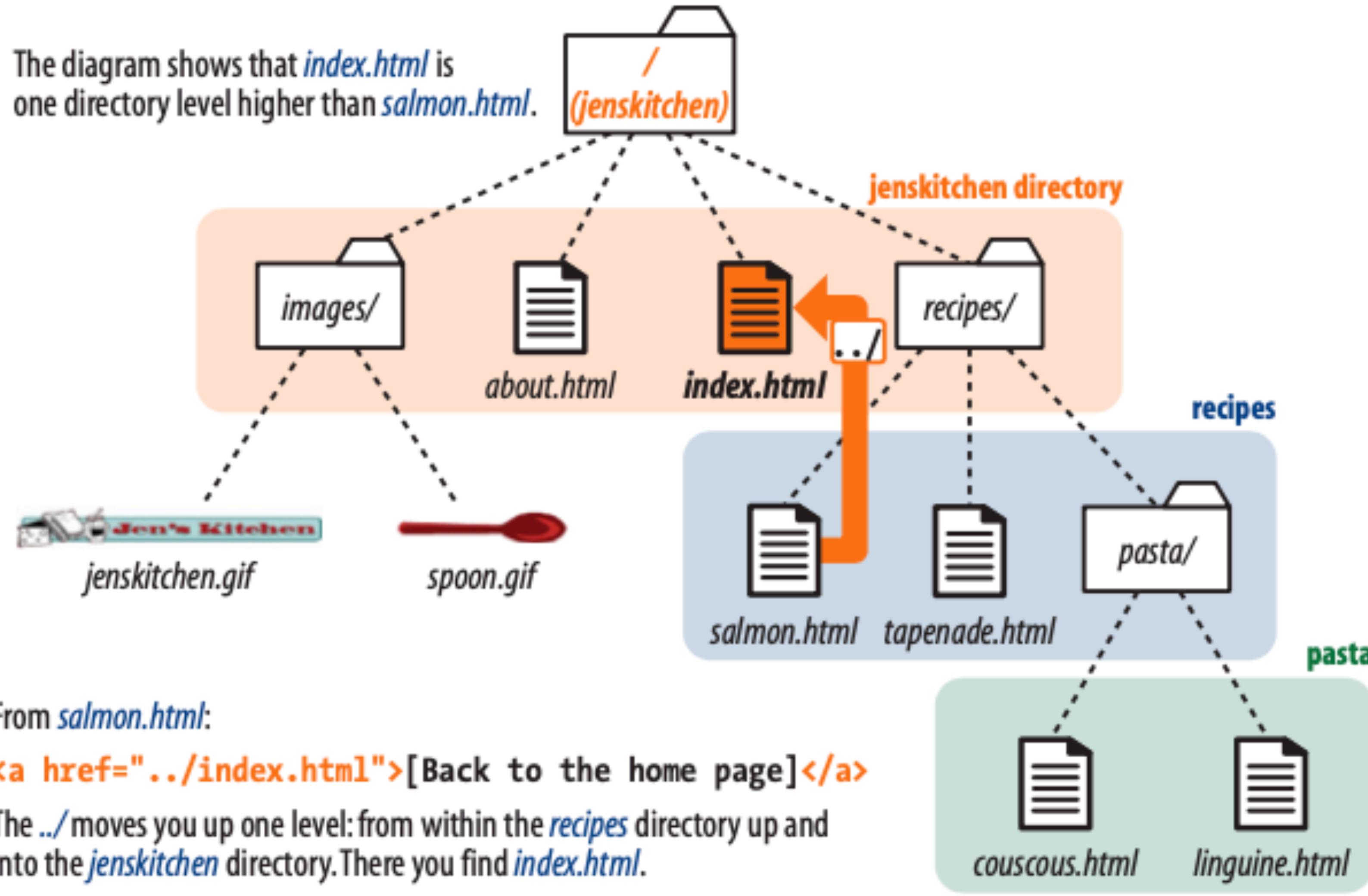
---

The diagram shows that *salmon.html* is one directory lower than *index.html*.



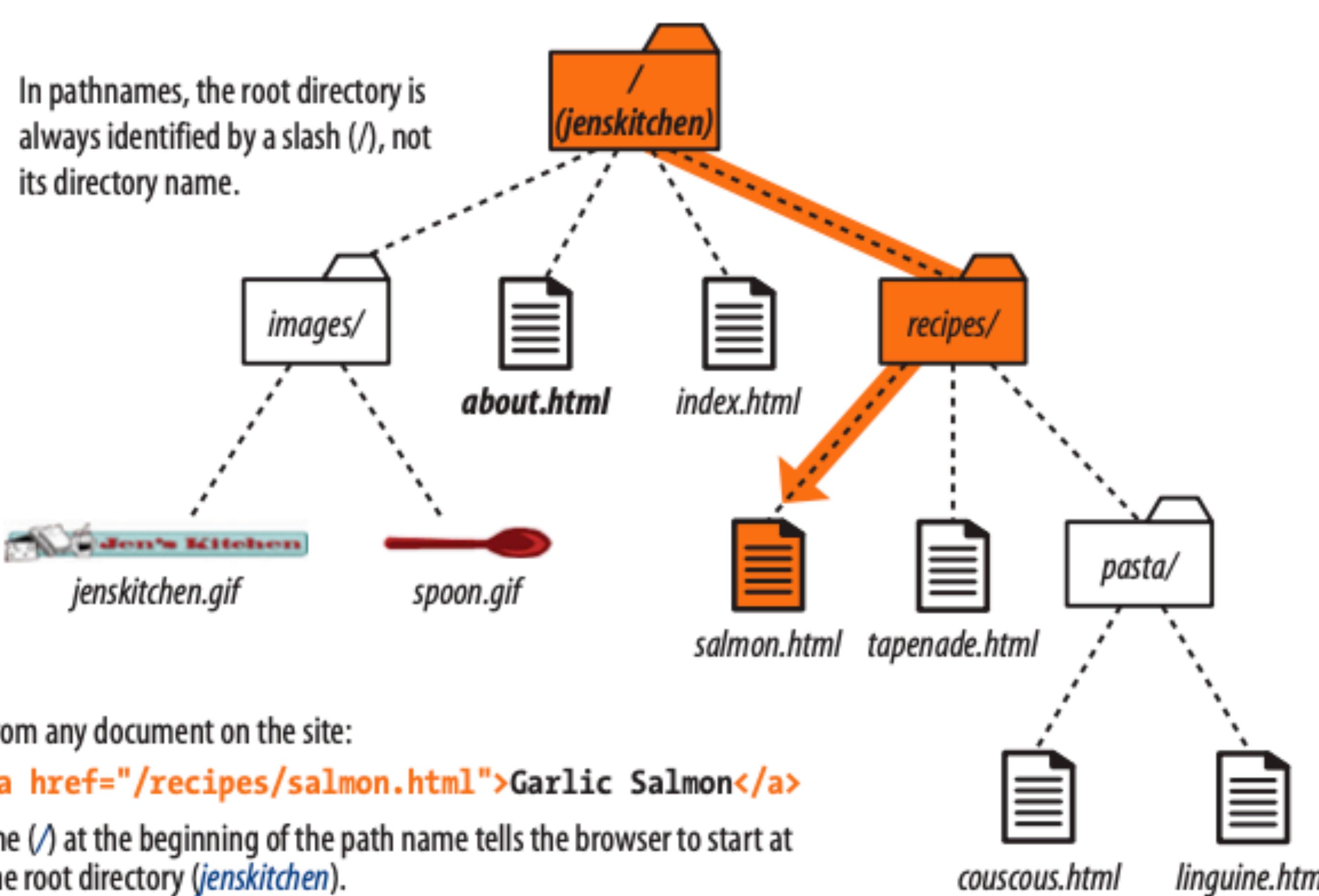
# ESEMPIO DI STRUTTURA

The diagram shows that *index.html* is one directory level higher than *salmon.html*.



# CONSIGLIO: SEMPRE MEGLIO PARTIRE DALLA ROOT

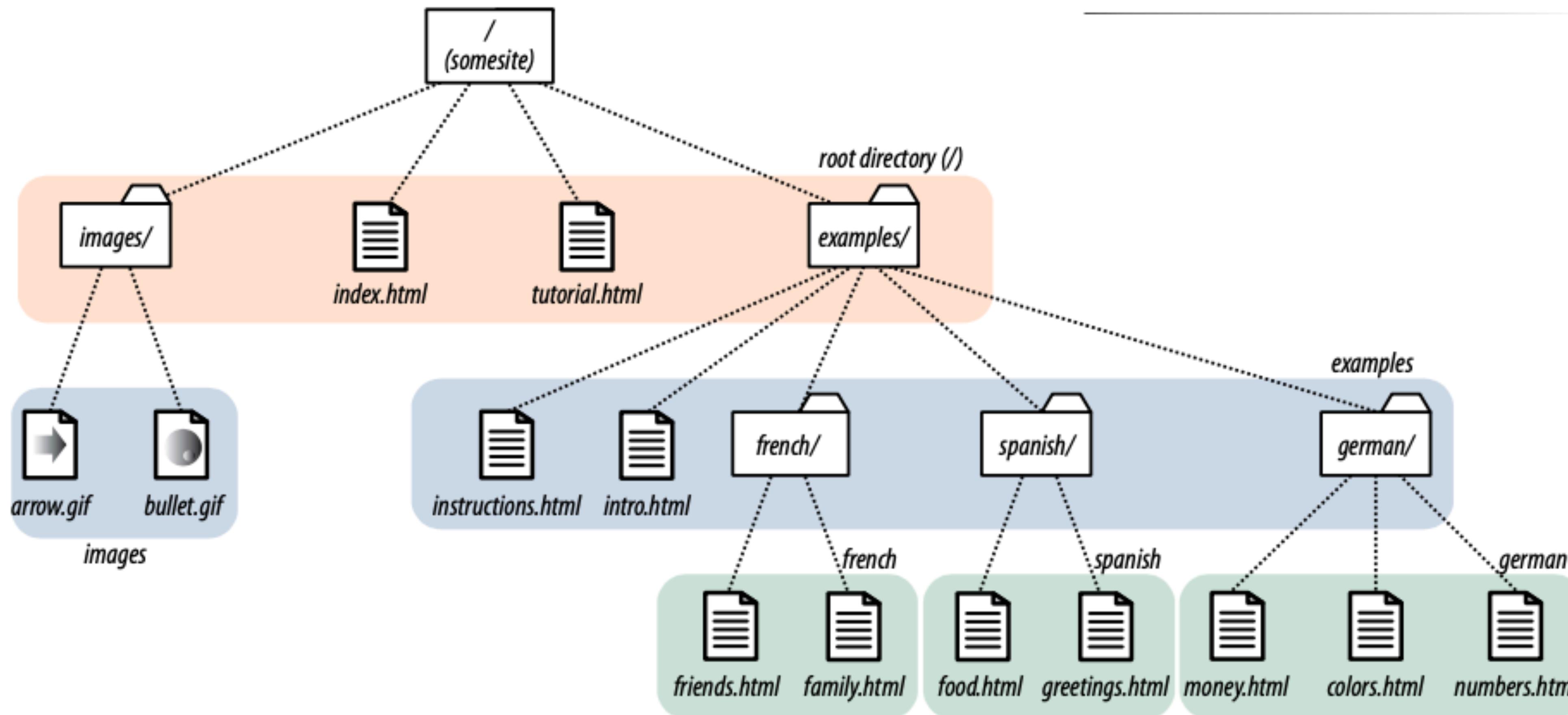
---



*Figure 6-10. Writing a relative URL starting at the root directory.*

# ESERCIZIO: LINK DA FAMILY A NUMBERS

**Figure 6-12.** The directory structure for the Test Yourself questions.



## ALTRI TIPI DI LINK

---

- Mail link <[a href="mailto:giulia@gmail.com"- Phone link <\[a href="tel:049838220"\]\(tel:049838220\)](mailto:giulia@gmail.com)

# IMMAGINI

---

- Formati: .gif , .jpeg o .png

```

```

- Attributo src: indica dove si trova l'immagine
- Attributo alt: fa vedere il testo in caso l'immagine non sia disponibile
- Attributi width e height indicano larghezza e altezza

# TABELLE

---

Table 1.3: Tags and attributes for creating tables

Tag	Description
table	beginning and end of table
tr	row of table
th	header cell
td	data cell
Attributes	
rowspan	number of rows to merge
colspan	number of columns to merge
border	width of border
cellpadding	width of whitespace between two border
cellspacing	width of whitespace within a border
bgcolor	background color
bordercolor	color of border
width	width of table (numeric or %)
height	height of table (numeric)
caption	caption for table

# TABELLE

---

<table>

Menu item	Calories	Fat (g)
Chicken Noodle Soup	120	2
Caesar Salad	400	26

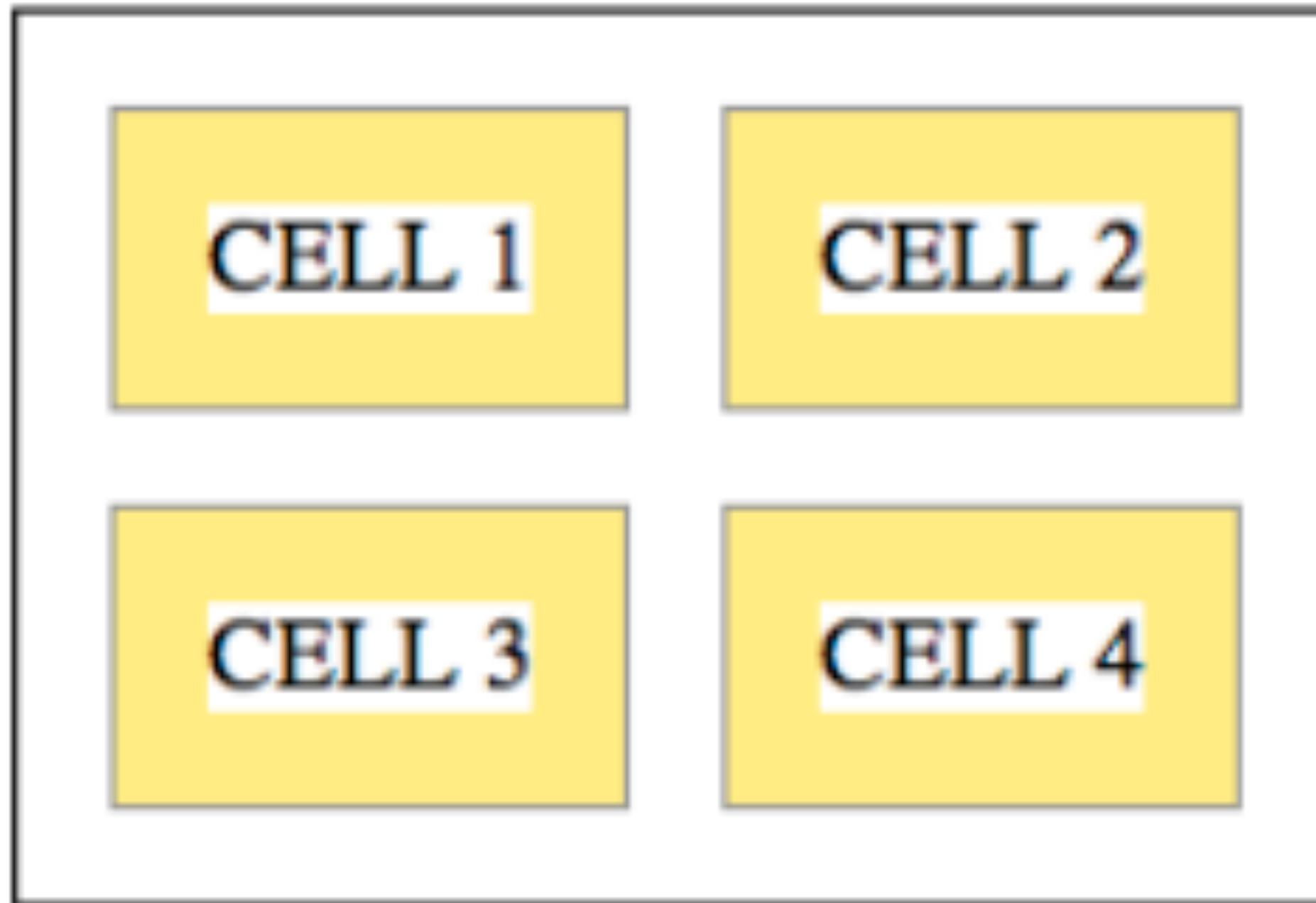
</table>

**Figure 8-3.** The elements that make up the basic structure of a table.

# CELL PADDING & CELL SPACING

---

Cell padding is the space between the edge of the cell and its contents.



Cell spacing is the space between cells.



# ESEMPIO TABELLA

---

```
<table border="1">  
  <tr>  
    <th>1</th>  
    <th>2</th>  
    <th>3</th>  
  </tr>  
  <tr>  
    <td>4</td>  
    <td colspan="2">  
      <b>celle unite</b>  
    </td>  
  </tr>  
</table>
```

1	2	3
4	<b>celle unite</b>	

# ESEMPIO TABELLA

```
<table border="1">
  <tr>
    <td>1</td>
    <td rowspan="2">
      <b>celle unite</b>
    </td>
    <td>3</td>
  </tr>
  <tr>
    <td>4</td>
    <td>6</td>
  </tr>
</table>
```

1	<b>celle unite</b>	3
4		6

# ESERCIZIO MUSEO LEONARDO

---

# CONCLUSIONE

---

*linkedin: /giuliamaddalena - www.giuliamadda.it*