

# 1

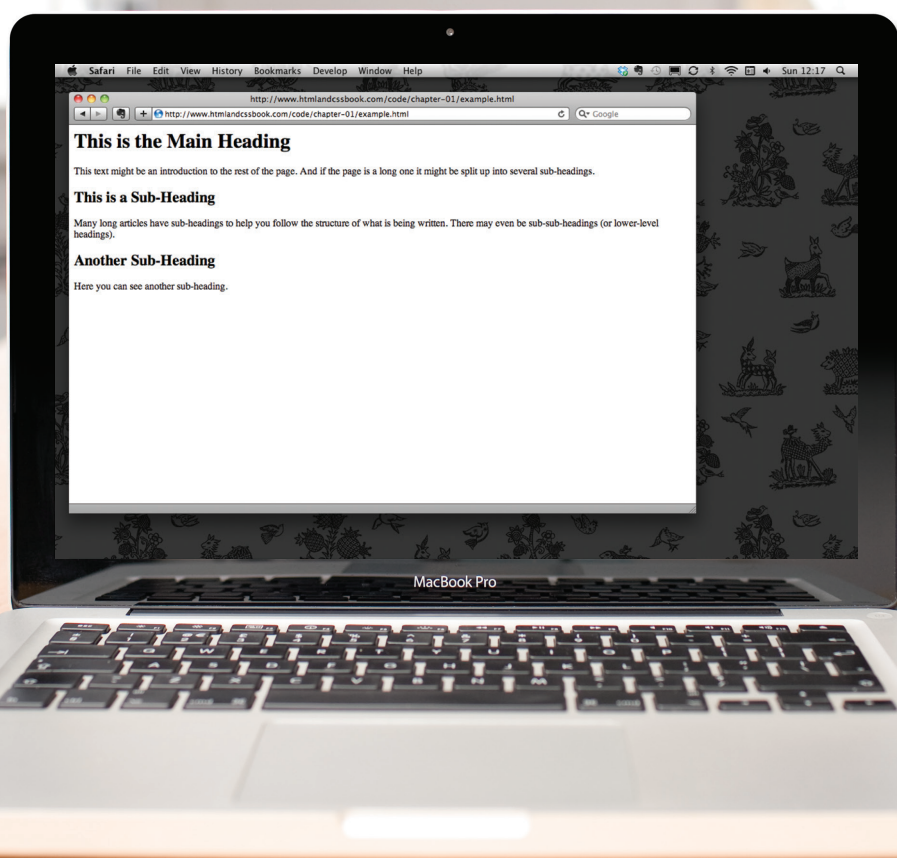
## LA STRUTTURA

- ▶ Che cosa si intende per struttura
- ▶ Che cosa si intende per markup HTML
- ▶ Tag ed elementi

Nella nostra vita quotidiana incontriamo ogni genere di documenti. Quotidiani, polizze assicurative, cataloghi di prodotti... la lista è lunga.

Molte pagine web si comportano come versioni elettroniche di questi documenti. Per esempio, i quotidiani presentano gli stessi articoli sia su carta, sia sui siti web; è possibile siglare un'assicurazione sul Web e molte aziende sono dotate di cataloghi online e di strumenti per il commercio elettronico. In tutti questi tipi di documenti, la struttura è molto importante per aiutare il lettore a comprendere i messaggi che state cercando di comunicargli e anche per navigare all'interno del documento. Pertanto, per poter imparare a scrivere pagine web, è molto importante capire come strutturare i documenti. In questo capitolo affronteremo i seguenti argomenti.

- Come il linguaggio HTML descrive la struttura di una pagina web.
- Imparare ad aggiungere a un documento dei tag o elementi.
- Scrivere una pagina web.



# IN QUALE MODO LE PAGINE USANO LA STRUTTURA

Riflettete un attimo sugli articoli che potete aver letto su un quotidiano: per ogni articolo trovate un titolo, del testo e, talvolta, anche alcune immagini. Se l'articolo è piuttosto corposo, potete trovarvi anche dei sottotitoli, che lo suddividono in più sezioni oppure citazioni chiave, tratte dall'articolo stesso. Questo forma una struttura che aiuta il lettore a comprendere quanto viene raccontato dall'articolo, all'interno del quotidiano.

La struttura è molto simile anche quando l'articolo viene visualizzato online (anche se in questo caso l'articolo può anche essere arricchito di contenuti audio o video). Questo concetto viene illustrato dalla pagina qui a destra, che rappresenta una copia di un articolo quotidiano e la corrispondente versione sul sito web.

Ora considerate un tipo molto differente di documento: un modulo assicurativo. I moduli spesso hanno dei titoli che suddividono le varie sezioni e ogni sezione contiene un elenco di domande, affiancate da aree all'interno delle quali potete compilare i dettagli oppure da caselle, da spuntare. Ancora una volta, questa struttura può essere molto simile online.



Read more on  
MediaGuardian.co.uk

Digital economy or bust  
Part 33: In which the team turn up the volume with inside track on The X Factor – and get a glimpse of the future

Coming up this week  
Monday: Shortlists for Student Media Awards announced  
Wednesday to Friday: Coverage of the RTS Cambridge Convention

## Interview Rio Carraeff

# Vevo revolutionary

Universal's former mobile chief is leading the music industry's fight to shake up online video. He reveals his frustration with MTV, and says why no one need own music if his site succeeds. Interview by **Mark Sweney**

If Rio Carraeff succeeds, perhaps only diehard fans will need to own music. His online music-video site, part-owned by the two largest record companies, also hopes to have the same impact as MTV and to be an answer to YouTube. Chuck those goals in with that of making the industry less dependent on the purchase of recordings, and for Carraeff there is clearly plenty to do.

Carraeff is the youthful chief executive of Vevo – launched in late 2009 with the backing of three of the four major groups, Sony Music, Universal Music and EMI – who is taking the venture international with a rollout starting in the UK and continental Europe. “Sex, music and sports are the only entertainment categories on the planet that people love that can build audiences at the scale of billions of people,” he says. “I’m in the business of connecting billions of people to music,” is his modestly stated aim.

With global CD sales plummeting by \$1.5bn last year, Carraeff's mission is clear. “We wouldn’t have created Vevo if we didn’t need it,” he says. “The industry felt it was necessary. If MTV was doing a great job paying royalties, if YouTube (best), there would have been no need. We have invested tens of millions to be responsible for our own destiny. We can’t sit back and say ‘I hope Apple or whoever figures this out’.”

Vevo's relationship with Google, the owner of the world's largest video-sharing platform YouTube, is clearly critical. Michael Grade called the company a “parasite” and Sir Martin Sorrell described it as a “freemium”. Despite the combative relationship the music industry has historically had with players in the digital space, Carraeff prefers to characterise Vevo's dealings with YouTube as “symbiotic”, although “declaration of independence” would be more appropriate.

“We said ‘let’s figure out how to work with them,’” he explains. “There are no duplicate copies [of music videos] on YouTube, there were thousands before, the official versions are only available from us. They don’t threaten us. YouTube is a place where people can upload any video in the world, we’re not trying to compete.” Carraeff points out that 50% of Vevo's traffic comes from YouTube searches, and 30% comes from recommendations of videos that users might like to watch that appear on the side of the YouTube web pages when a user is viewing clips.

### Free access

Vevo's business model is all about providing music videos that fans can access free, funded by advertising – or to put it another way – give consumers an alternative to owning songs. “I believe the future is access, not ownership, not iTunes as it is today,” he says. “We’re not trying to sell people music; our customers are not the small amount of people that want to buy music. We are about providing access: it is the only scalable model for the music industry; the question is, how do you do that and make money?”

Which raises the question of how well Vevo is actually doing. Carraeff doesn’t want to give away too much commercially but says it is already making “hundreds of millions of dollars” in revenue, although there are hosting costs to pay. More than half of gross revenue goes to content owners – the label, artist or licensor – with the remainder being kept by Vevo or paid to partners such as YouTube. He says that Vevo is “significantly ahead” of its original business plan – about 40% ahead to be precise – and is on track to achieve profitability “in the very early part of next year”.

Yet there are problems. Carraeff's business is dependent on advertising, and he is frustrated by the low rates that companies pay to run campaigns around music content. His contention is that advertising around music content is inferior and that Vevo's role is to “own” the prime content and then be able to position it as a pre-equivalent of BSkyB owning Premier League football.

“The audience that loves music is vast and promising; it should be treated as



Video vexations ... Rio Carraeff says ‘if MTV was doing it

We are about access: it is the only scalable model for the music industry; the question is, how do you do that and make money?’

### Curriculum vitae

#### Age 36

Education Did not go to university because “I started my first company when I was 18”

#### Career

2004 vice-president  
2005 general manager  
Universal Music Mobile division, responsible for new technology, Vevo

if it were (audience TV content and want to revert to low, I want music as a TV

Mark Sweney  
guardian.co.uk, Sunday 11 September 2011 20:00 BST  
Article history



If Rio Carraeff succeeds, perhaps only diehard fans will need to own music. His online music-video site, part-owned by the two largest record companies, also hopes to have the same impact as MTV and to be an answer to YouTube. Chuck those goals in with that of making the industry less dependent on the purchase of recordings, and for Carraeff there is clearly plenty to do.

Carraeff is the youthful chief executive of Vevo – launched in late 2009



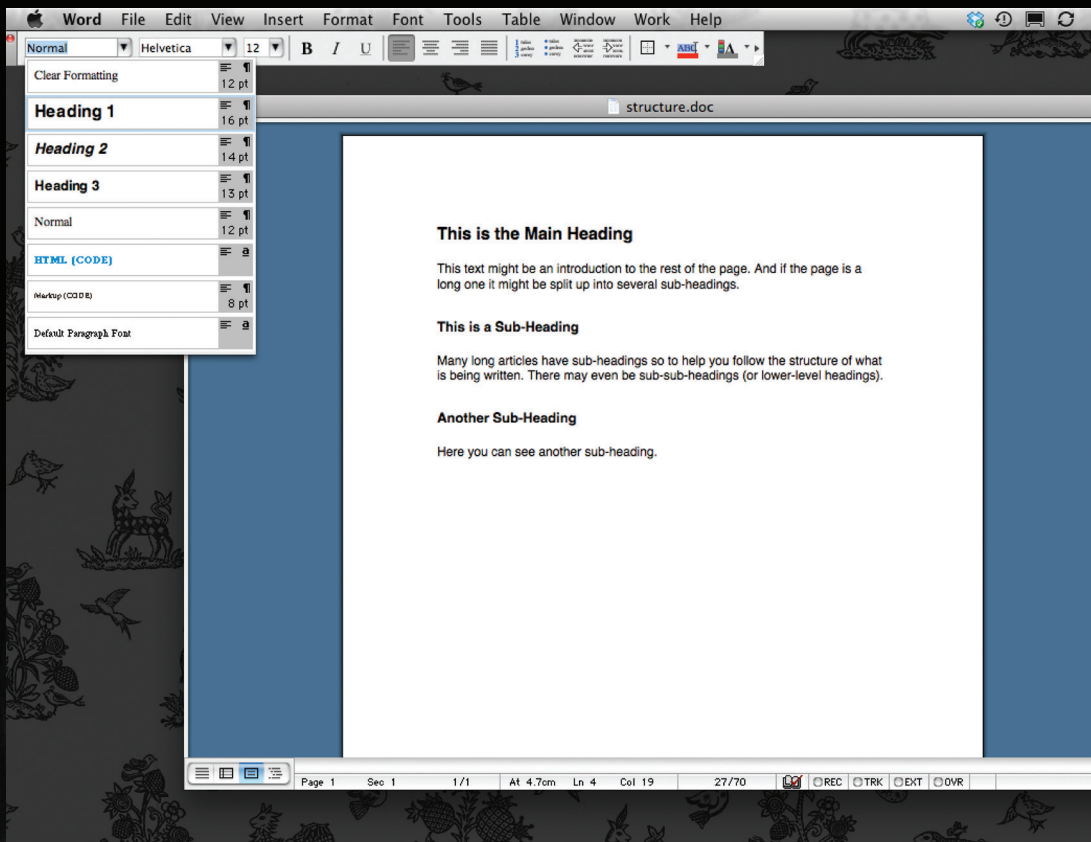
# STRUTTURARE I DOCUMENTI TESTUALI

L'uso di titoli e sottotitoli all'interno di un documento spesso riflette una gerarchia insita nell'informazione. Per esempio, un documento potrebbe iniziare con un grosso titolo, seguito da un'introduzione o dalle informazioni più importanti.

L'articolo può poi essere espanso grazie all'impiego di sottotitoli, più giù nella pagina. Quando si usa un word processor per creare un documento, si cerca di separare i testi per dotarli di una struttura. Ogni argomento potrebbe essere costituito da un nuovo paragrafo e ogni sezione può essere dotata di un titolo che descrive gli argomenti trattati.

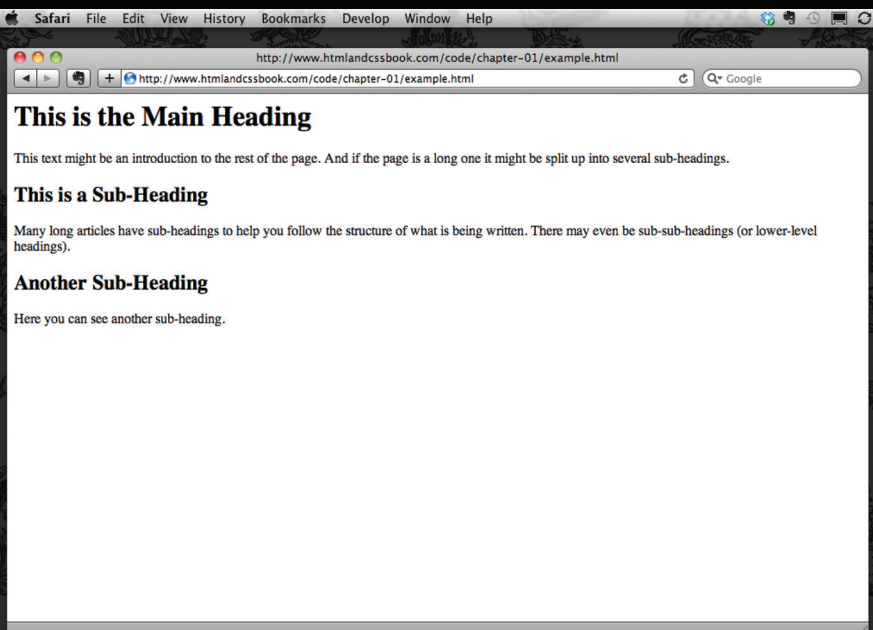
Qui a destra potete vedere un semplice documento in Microsoft Word. I vari stili impiegati per il documento, come i vari livelli di titolazione, sono indicati anche nella casella a discesa. Se utilizzate regolarmente Word, potete aver utilizzato la sua barra di formattazione per applicare queste caratteristiche al testo.





MacBook Pro

Nella pagina precedente, avete visto come applicare una struttura a un documento Word, per semplificarne la lettura. Utilizziamo questa stessa formattazione anche quando scriviamo le pagine web.



# HTML: UN LINGUAGGIO PER DESCRIVERE LA STRUTTURA DELLE PAGINE

Nella finestra del browser, potete vedere una pagina web che presenta esattamente gli stessi contenuti del documento Word presentato a pagina 18. Per descrivere la struttura di una pagina web, aggiungiamo dei codici alle parole che devono comparire all'interno della pagina.

Qui sotto potete vedere il codice HTML impiegato per creare questa pagina. Non preoccupatevi troppo del significato dei codici. Inizieremo a parlarne più in dettaglio a partire dalla prossima pagina. Notate solo che il codice HTML è rappresentato in blu, mentre il testo che compare effettivamente sulla pagina è rappresentato in nero.

```
<html>
  <body>
    <h1>This is the Main Heading</h1>
    <p>This text might be an introduction to the rest of
      the page. And if the page is a long one it might
      be split up into several sub-headings.</p>
    <h2>This is a Sub-Heading</h2>
    <p>Many long articles have sub-headings to help you
      follow the structure of what is being written.
      There may even be sub-sub-headings (or lower-level
      headings).</p>
    <h2>Another Sub-Heading</h2>
    <p>Here you can see another sub-heading.</p>
  </body>
</html>
```

Il codice HTML (in blu) è costituito da caratteri che devono trovarsi all'interno di parentesi angolari: questi vengono chiamati **elementi HTML**. Gli elementi sono normalmente costituiti da due **tag**: uno di apertura e uno di chiusura (il tag di chiusura è contraddistinto dal carattere barra, /). Ogni elemento HTML dice al browser qualcosa a proposito delle informazioni che sono contenute fra i tag di apertura e di chiusura.

# HTML USA GLI ELEMENTI PER DESCRIVERE LA STRUTTURA DELLE PAGINE

Diamo ora un'occhiata al codice presentato nella pagina precedente. Vi troviamo parecchi elementi distinti. Ogni elemento ha un tag di apertura e un tag di chiusura.

## CODICE

```
<html>
```

```
<body>
```

```
<h1>This is the Main Heading</h1>
```

```
<p>This text might be an introduction to the rest of  
the page. And if the page is a long one it might  
be split up into several sub-headings.</p>
```

```
<h2>This is a Sub-Heading</h2>
```

```
<p>Many long articles have sub-headings to help you  
follow the structure of what is being written.  
There may even be sub-sub-headings (or lower-level  
headings).</p>
```

```
<h2>Another Sub-Heading</h2>
```

```
<p>Here you can see another sub-heading.</p>
```

```
</body>
```

```
</html>
```

I tag si comportano come contenitori. Dicono qualcosa a proposito delle informazioni che sono situate fra il tag di apertura e il tag di chiusura.

## DESCRIZIONE

.....	Il tag di apertura <code>&lt;html&gt;</code> indica che quanto segue, fino al tag di chiusura <code>&lt;/html&gt;</code> è codice HTML.	.....
.....	Il tag <code>&lt;body&gt;</code> indica che quanto segue, fino al tag di chiusura <code>&lt;/body&gt;</code> deve essere mostrato nella finestra principale del browser.	.....
.....	Le parole fra <code>&lt;h1&gt;</code> e <code>&lt;/h1&gt;</code> rappresentano un titolo principale.	.....
.....	Un normale paragrafo di testo deve trovarsi compreso fra questi tag <code>&lt;p&gt;</code> e <code>&lt;/p&gt;</code> .	.....
.....	Le parole fra <code>&lt;h2&gt;</code> e <code>&lt;/h2&gt;</code> formano un sottotitolo.	.....
.....	Ecco un altro paragrafo fra il tag di apertura <code>&lt;p&gt;</code> e quello di chiusura <code>&lt;/p&gt;</code> .	.....
.....	Un altro sottotitolo fra i tag <code>&lt;h2&gt;</code> e <code>&lt;/h2&gt;</code> .	.....
.....	Un altro paragrafo fra i tag <code>&lt;p&gt;</code> e <code>&lt;/p&gt;</code> .	.....
.....	Il tag di chiusura <code>&lt;/body&gt;</code> indica la fine di ciò che deve comparire nella finestra del browser.	.....
.....	Il tag di chiusura <code>&lt;/html&gt;</code> indica la fine del codice HTML.	.....

# L'ASPETTO DEI TAG

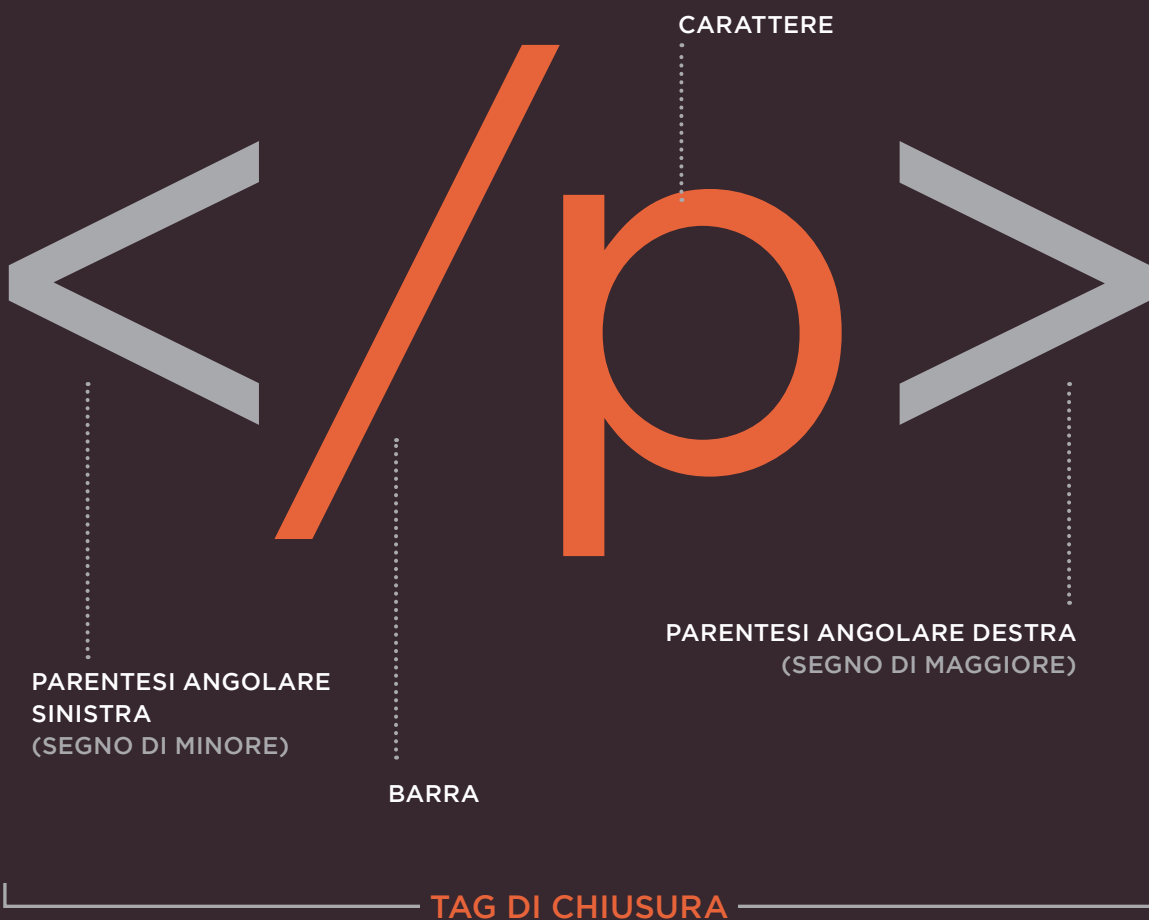


I caratteri fra le parentesi angolari indicano lo scopo del tag.

Per esempio, in questo tag, la `p` sta per paragrafo.

Il tag di chiusura si riconosce per la barra che segue il simbolo `<`.





I termini "tag" e "elemento" sono in sostanza sinonimi.

Per la precisione, un elemento è costituito dal tag di apertura

e dal tag di chiusura e anche da tutto ciò che si trova fra di essi.

# GLI ATTRIBUTI CI DICONO QUALCOSA DI PIÙ DEGLI ELEMENTI

Gli attributi forniscono informazioni aggiuntive sul contenuto di un elemento. Compaiono all'interno del tag di apertura dell'elemento e sono costituiti da due parti: un **nome** e un **valore**, separati dal segno di "uguale".



Il nome dell'attributo indica il tipo di informazione aggiuntiva che si intende fornire sul contenuto dell'elemento. L'attributo deve essere scritto in lettere minuscole.

Il valore è costituito dall'informazione o dall'impostazione dell'attributo. Dovrà essere collocato fra doppi apici. Attributi differenti possono avere valori differenti.

Qui viene utilizzato l'attributo `lang` per indicare la lingua utilizzata da questo elemento. Il valore di questo attributo in questa pagina specifica l'uso della lingua italiana.

HTML5 consente di utilizzare nomi di attributi in lettere maiuscole e di omettere i doppi apici, anche se si tratta di una pratica sconsigliata.



La maggior parte degli attributi può essere utilizzata solo su determinati elementi, anche se alcuni attributi (fra questi `lang`) possono trovarsi in qualsiasi elemento.

La maggior parte dei valori degli attributi ha un aspetto predefinito o segue un formato ben preciso. Parleremo dei valori consentiti quando introdurremo ciascun nuovo attributo.

Il valore dell'attributo `lang` è un modo abbreviato per specificare la lingua utilizzata all'interno dell'elemento, un codice che è compreso da tutti browser.

# I TAG BODY, HEAD E TITLE

## <body>

Abbiamo incontrato l'elemento `<body>` nel primo esempio che abbiamo creato. Tutto ciò che si trova all'interno di questo elemento viene rappresentato all'interno della finestra principale del browser.

## <head>

Prima dell'elemento `<body>` spesso si trova l'elemento `<head>`. Contiene *informazioni relative alla pagina*, invece che informazioni che vengono presentate nella parte principale della finestra del browser, evidenziata in blu nella pagina a fronte. Normalmente, all'interno dell'elemento `<head>` si trova un elemento `<title>`.

## <title>

Il contenuto dell'elemento `<title>` viene visualizzato nella parte superiore del browser, sopra la riga dell'indirizzo URL della pagina che si vuole visitare, oppure nella scheda di tale pagina (se il browser utilizza delle schede per consentire la visualizzazione in contemporanea di più pagine).

/chapter-01/body-head-title.html

HTML

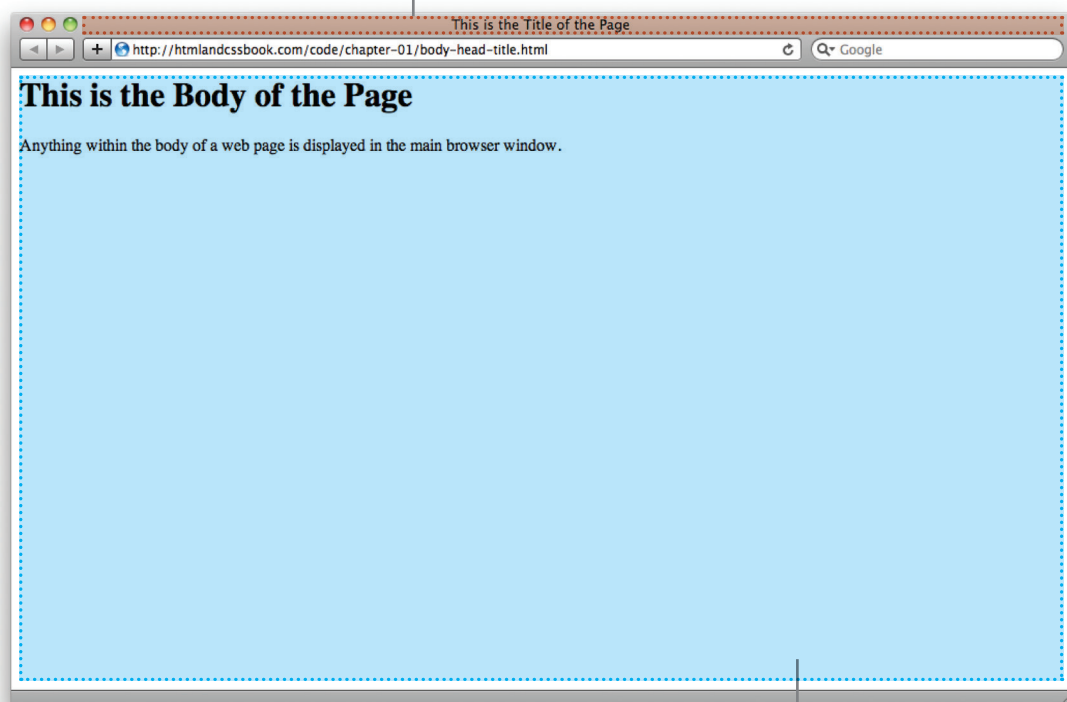
```
<html>
  <head>
    <title>This is the Title of the Page</title>
  </head>
  <body>
    <h1>This is the Body of the Page</h1>
    <p>Anything within the body of a web page is
      displayed in the main browser window.</p>
  </body>
</html>
```

OUTPUT

## This is the Body of the Page

Anything within the body of a web page is displayed in the main browser window.

Tutto ciò che si trova fra i tag `<title>` compare nella barra del titolo (o nella scheda) nella parte superiore della finestra del browser, che qui è evidenziata da una linea tratteggiata di colore arancio.



Tutto ciò che si trova tra i tag `<body>` comparirà nella finestra principale del browser, che qui è evidenziata in azzurro.

Forse sapete che la sigla HTML significa HyperText Markup Language. La parte *HyperText* fa riferimento al fatto che HTML consente di creare link ipertestuali che consentono ai visitatori di passare con facilità da una pagina all'altra.

Un linguaggio *markup* consente di produrre testo annotato, dove queste annotazioni forniscono ulteriori indicazioni sul contenuto di un documento. Se pensate a una pagina web, è come se aggiungessimo dei codici attorno al testo da

visualizzare; tali codici vengono poi utilizzati dal browser per controllare la formattazione della pagina.

In pratica, i tag che aggiungiamo costituiscono il markup, ovvero le annotazioni.

# CREARE UNA PAGINA WEB SU UN PC

Per creare la vostra prima pagina web su un PC, avviate il Blocco note. Lo potete trovare selezionando:

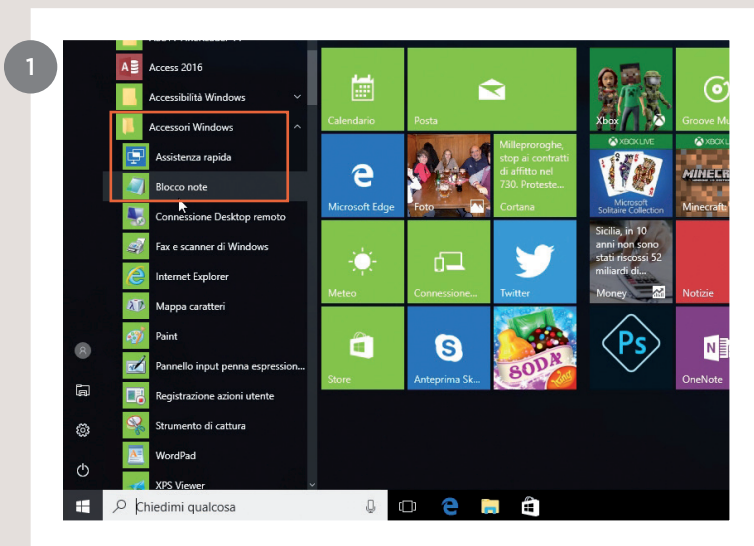
## Start

**Tutti i programmi (o Programmi)**

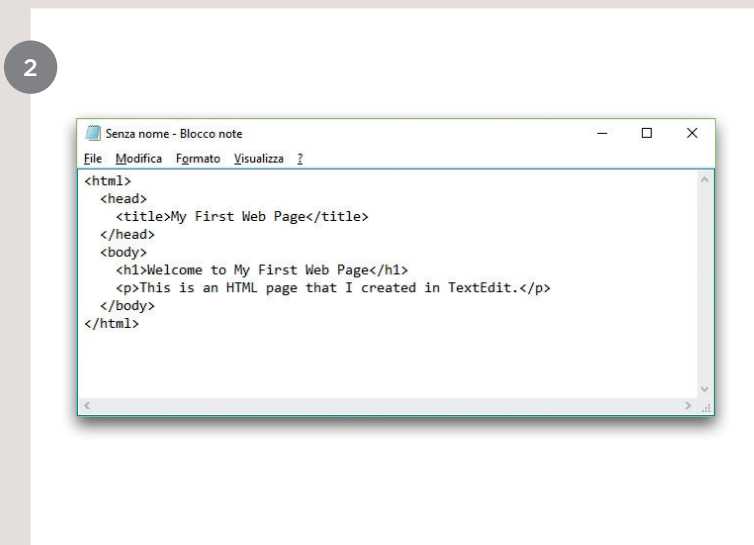
**Accessori Windows**

**Blocco note**

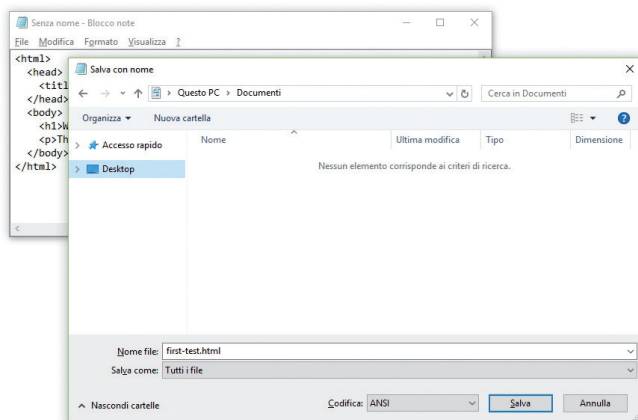
In alternativa potete utilizzare un editor gratuito chiamato Notepad++, che potete scaricare gratuitamente all'indirizzo web [notepad-plus-plus.org](http://notepad-plus-plus.org).



Digitate il codice rappresentato nella figura a destra.



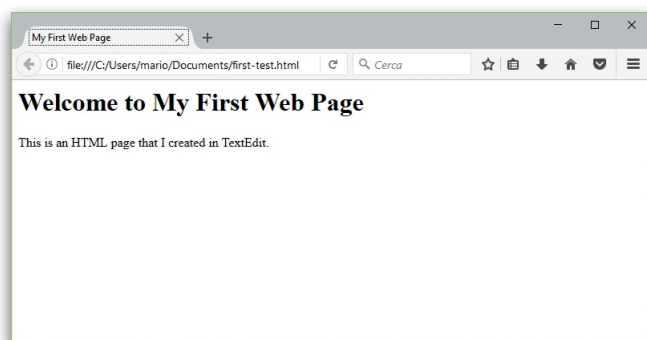
3



Aprirete il menu **File** e selezionate **Salva con nome**. Assegnate al file un nome facile da ricordare. Se lo desiderate, potete anche creare una cartella, da dedicare a tutti gli esempi che svilupperemo nel corso di questo libro.

Potete salvare questo file come `primo-test.html`. Assicuratevi che nella casella **Salva come** della finestra di dialogo **Salva con nome** sia riportata l'indicazione **Tutti i file**.

4



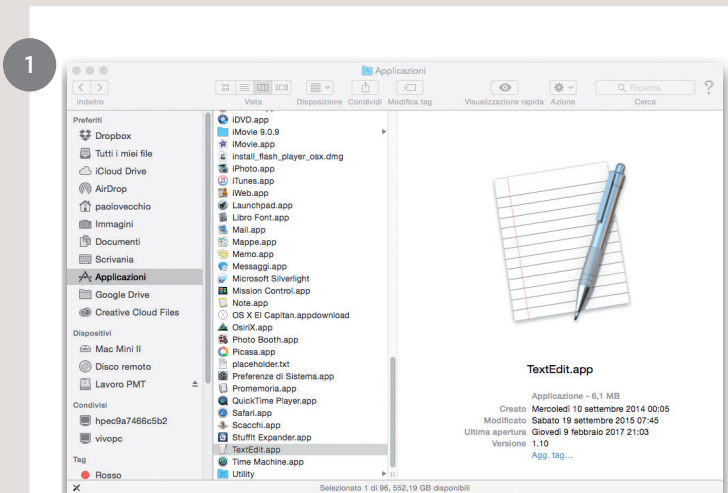
Avviate il browser web. Richiamate il suo menu **File** e selezionate **Apri**. Raggiungete poi il file che avete appena creato, selezionatelo e fate clic sul pulsante **Apri**. Il risultato dovrebbe avere un aspetto simile a quello rappresentato qui a sinistra.

Se l'aspetto fosse differente, trovate il file che avete appena creato sul computer e assicuratevi che usi l'estensione `.html`. Se riportasse l'estensione `.txt` dovreste tornare al Blocco note e salvare nuovamente il file, ma questa volta specificate le virgolette attorno al nome: `"primo-test.html"`.

# CREARE UNA PAGINA WEB SU UN MAC

Per creare la vostra prima pagina web su un Mac, avviate l'applicazione TextEdit. Questa dovrebbe trovarsi nella cartella **Applicazioni**.

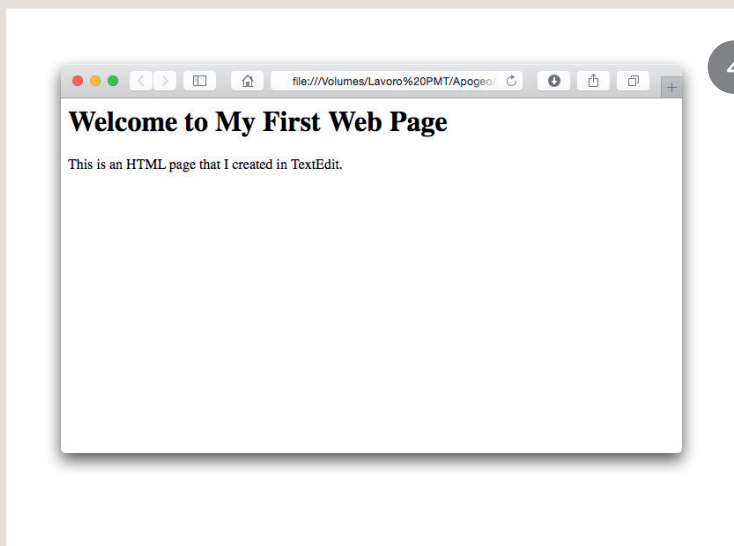
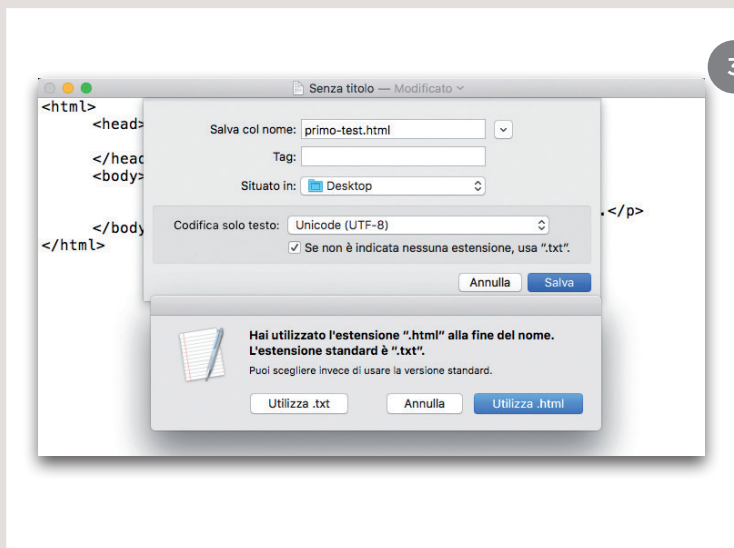
Potete anche scaricare un editor di testi gratuito adatto alla creazione di pagine web, chiamato TextWrangler disponibile all'indirizzo [barebones.com](http://barebones.com).



Digitate il codice rappresentato  
qui a destra.







Ora richiamate il menu **File** e selezionate il comando **Salva**. Scegliete di salvare il file in una posizione del computer facile da ricordare (per esempio il Desktop).

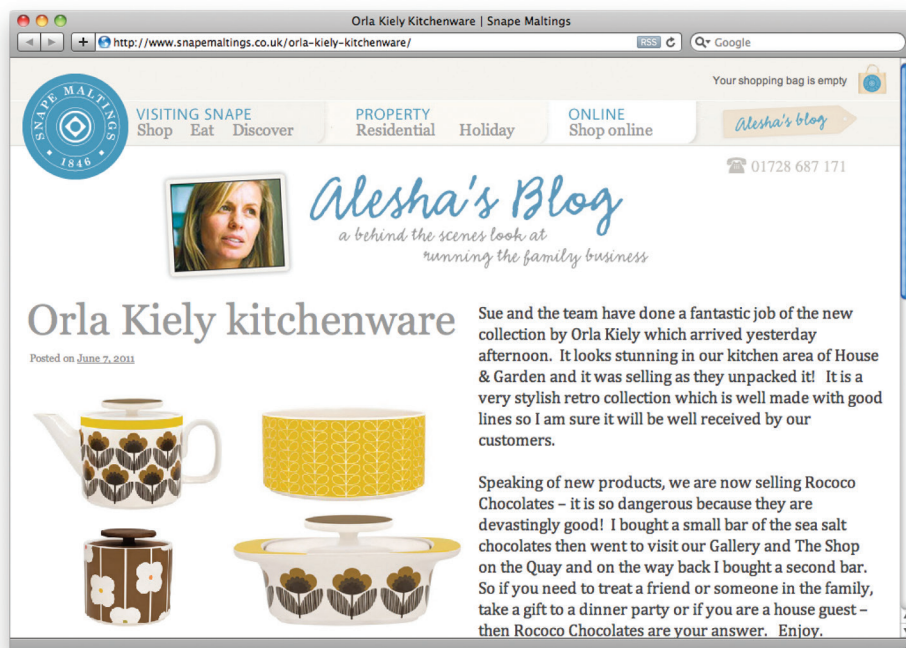
Se preferite, potete anche creare una cartella per gli esempi sviluppati nel corso di questo libro. Salvate questo file come `primo-test.html`. La finestra avrà un aspetto simile a quello rappresentato nella figura a sinistra.

Dovete selezionare il pulsante **Utilizza .html**.

A questo punto, avviate il browser web, richiamate il menu **File** e selezionate **Apri**. Raggiungete il file che avete appena creato, selezionatelo e fate clic sul pulsante **Apri**. Il risultato dovrebbe essere simile alla schermata rappresentata qui a sinistra.

Se l'aspetto fosse differente, potreste dover cambiare una delle impostazioni di TextEdit. Richiamate il menu di TextEdit e selezionate **Preferenze**. Poi nelle preferenze per **Apri e salva**, segname la casella **Visualizza file HTML come codice HTML anziché come testo formattato** e poi riprova a salvare il file.

# IL CODICE IN UN SISTEMA DI GESTIONE DEI CONTENUTI



Se us un sistema di gestione dei contenuti, una piattaforma blogging o un'applicazione di e-commerce, probabilmente vi conatterete tramite una sezione di amministrazione del sito, che offre la possibilità di controllarlo. Gli strumenti delle sezioni di amministrazione di questi siti permettono di intervenire sulle singole parti della pagina. Quindi, solo raramente vedrete gli elementi `<html>`, `<head>` o `<body>`.

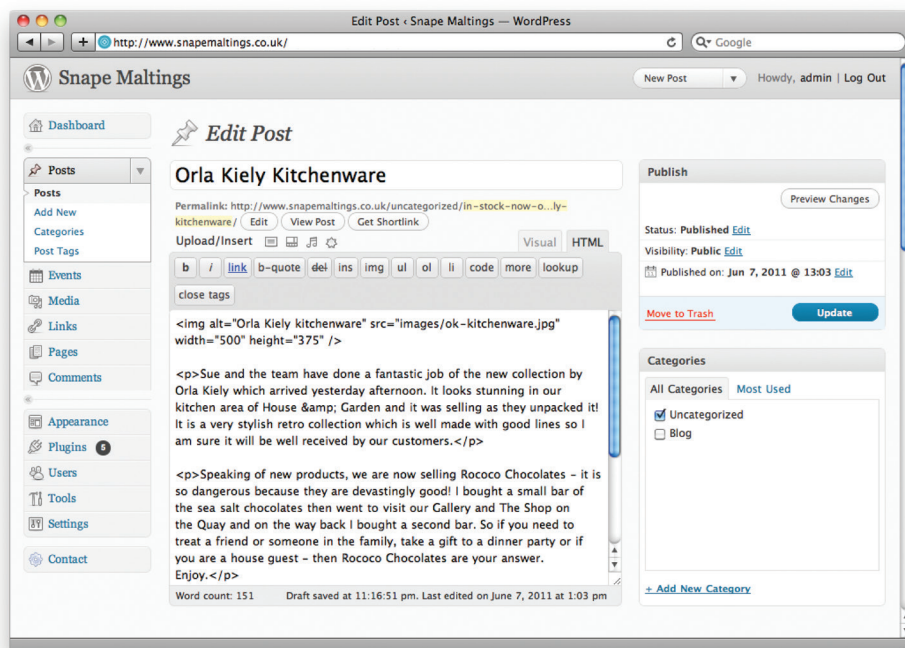
Osservando il sistema la pagina a fronte, vedrete una casella per specificare il titolo della pagina, un'altra casella

per specificare il contenuto principale dell'articolo, un modo per specificare la data di pubblicazione e anche qualcosa per indicare la sezione del sito alla quale appartiene la pagina.

Nel caso di un sito di e-commerce, potreste trovare caselle per specificare il nome del prodotto, una descrizione, un prezzo e la quantità disponibile.

Tutto questo, perché questi sistemi utilizzano un unico "modello" per controllare tutte le pagine di una sezione del sito. Per esempio, un sistema di

e-commerce potrebbe utilizzare lo stesso modello per presentare tutti i prodotti. Le informazioni che dovete fornire vengono poi collocate in specifiche posizioni del modello. Il vantaggio è che anche coloro che non sanno scrivere pagine web possono aggiungere informazioni al sito, con la possibilità anche di cambiare l'aspetto di alcuni elementi del modello, il quale si replicherà automaticamente su ogni pagina che lo utilizza. Se immaginate di avere un sito di e-commerce con mille articoli in vendita, la possibilità di intervenire solo sul modello è



decisamente molto più semplice che cambiare l'aspetto della pagina di ogni singolo prodotto. In questi sistemi, quando vi è un blocco di testo da modificare, per esempio un articolo, una voce del blog o la descrizione di un prodotto, spesso si opera in un editor di testi.

Gli editor di testi normalmente sono dotati di controlli simili a quelli che si trovano in un word processor, che forniscono varie opzioni per definire lo stile del testo, aggiungere link o inserire immagini. Dietro le quinte, questi editor aggiungono il

codice HTML al testo, proprio come abbiamo visto in queste pagine. Molti editor hanno anche un'opzione che consente di vedere (e modificare) il codice HTML prodotto

Quando avrete imparato a leggere e modificare questi codici, potrete avere un maggiore controllo su queste sezioni del sito web.

Nell'esempio qui sopra, l'editor di testi ha una scheda per la visualizzazione Visuale/HTML di ciò che viene specificato. Altri sistemi potrebbero avere

un pulsante (spesso dotato di parentesi angolari) per indicare il modo in cui si accede al codice.

Alcuni sistemi di gestione dei contenuti offrono strumenti che consentono anche di modificare i file del modello. Per modificarli dovrete seguire la documentazione, in quanto ogni sistema può utilizzare opzioni differenti. Occorre fare grande attenzione quando si interviene sul modello, poiché, cancellando erroneamente un codice o aggiungendo qualcosa nella posizione errata, il sito potrebbe smettere di funzionare.

# COME SONO REALIZZATI GLI ALTRI SITI

Quando il Web stava cominciando a decollare, uno dei modi più comunemente utilizzati per imparare a programmare in HTML e scoprire nuovi suggerimenti e tecniche consisteva nell'osservare il codice che componeva le pagine web.

Al giorno d'oggi vi sono molti libri e guide online che insegnano a utilizzare il codice HTML, ma è comunque possibile osservare il codice che viene inviato al browser dai server web.

Provate a osservare il codice di esempio di questo capitolo all'indirizzo [www.htmlandcssbook.com/view-source/](http://www.htmlandcssbook.com/view-source/).

Dopo aver aperto questa pagina, potete cercare il menu **Visualizza** nella finestra del browser e selezionare l'opzione **Visualizza sorgente** o **Visualizza codice** oppure fate clic sul tasto destro del mouse e scegliete **Visualizza sorgente pagina** (il titolo varia a seconda del browser utilizzato).

A volte l'opzione si trova in un menu Sviluppo, rivolto agli sviluppatori di pagine web, in quale deve essere attivato tramite un'apposita opzione nelle Preferenze.

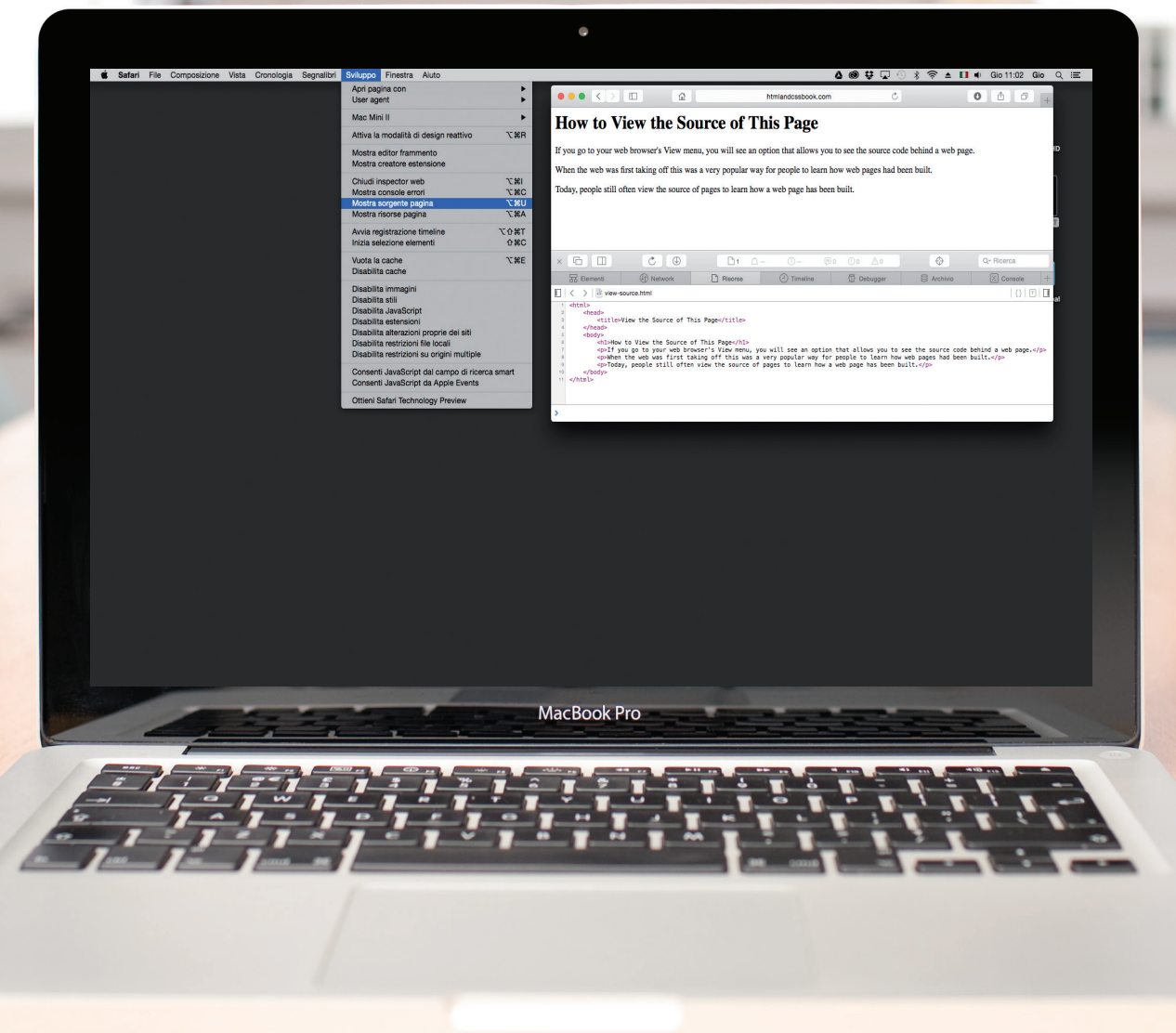
Comparirà un nuovo riquadro o finestra dove verrà visualizzato il codice sorgente utilizzato per creare questa pagina. Potete vedere il risultato nell'immagine rappresentata a destra.

In alto si può vedere la pagina e in basso il codice.

Inizialmente il codice può sembrare complicato, ma non scoraggiatevi. Entro la fine del prossimo capitolo sarete in grado di comprenderne appieno il significato.

Tutti gli esempi di questo libro sono contenuti nel sito web e potete utilizzare questa semplice tecnica su tutte le pagine d'esempio presentate, per vedere come sono state realizzate.

Potete anche scaricare tutto il codice sorgente di questo libro dallo stesso sito web, facendo clic sul link "Download" nella sezione "Code Samples".



- ▶ Le pagine HTML sono documenti testuali.
- ▶ Il linguaggio HTML utilizza tag, che sono codici di controllo situati all'interno di parentesi angolari. Si comportano come "contenitori" e spiegano come devono essere presentate le informazioni che racchiudono.
- ▶ I tag vengono anche chiamati elementi.
- ▶ I tag, normalmente, si presentano in coppie. Il tag di apertura indica l'inizio di un contenuto; il tag di chiusura ne denota la fine.
- ▶ I tag di apertura possono contenere degli attributi, che dicono qualcosa in più sul contenuto di tale elemento.
- ▶ Gli attributi richiedono un nome e un valore.
- ▶ Per imparare a programmare in HTML, occorre conoscere i tag disponibili, il loro significato e in quale posizione possono essere collocati.