

Mobile Web cap. II

Damn!! Other work!!

We must still work
on our site: it's
look well, but now
it's very slow....



Fabio Paccosi

Valerio Belli

Eugenio Paluello

Applab, 2013

La bellezza è nulla senza ottimizzazione

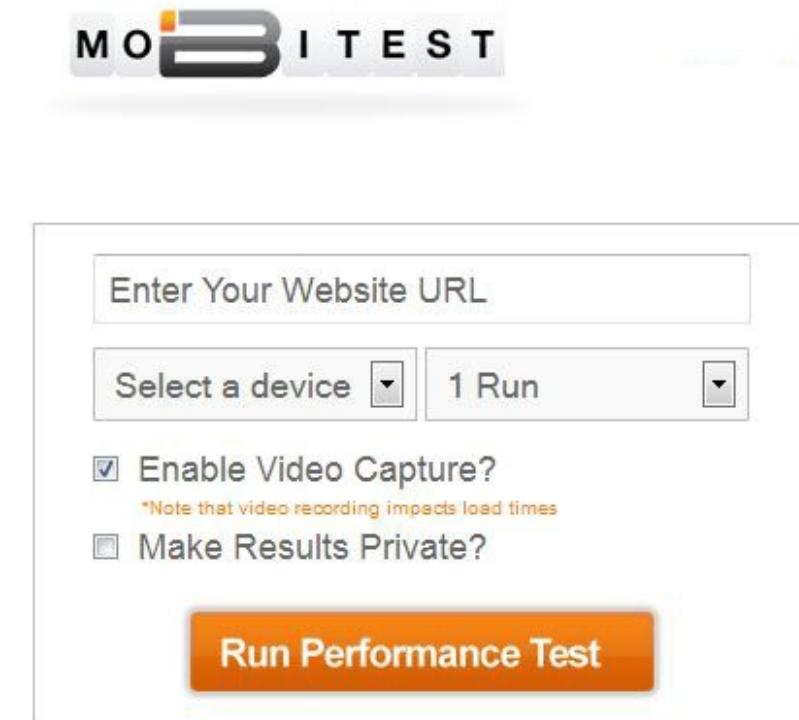
Il nostro sito, per quanto bello possa essere, deve essere progressivamente migliorato, altrimenti su alcuni device potrebbe risultare lento, come nel caso del nostro sfortunato amico, che ha aggiunto immagini al suo sito irresponsabilmente...



Mike

Per testare le performance del nostro sito, ci vengono in aiuto alcuni strumenti, come Blaze della Mobitest...

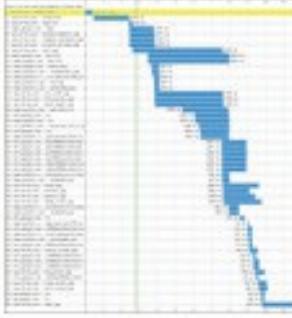
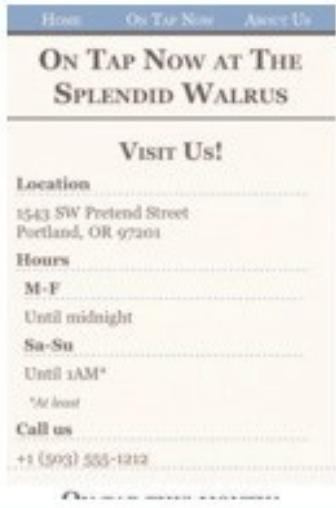
Disponibile qui:
<http://mobitest.akamai.com>



...un semplice tester per dispositivi mobile che lavora su un proxy server, noi gli diamo la pagina incriminata, lui fa il resto!!

Il tool della Mobitest ci dice:

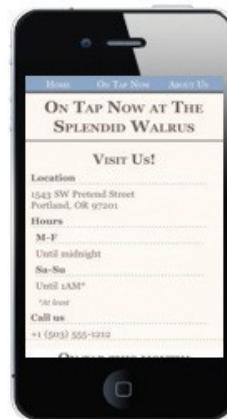
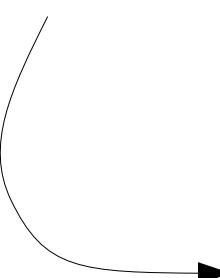
- Le dimensioni della pagina caricata
- Tempo di caricamento globale
- Whaterfall Chart delle risorse

LOAD TIME	PAGESIZE	WATERFALL CHART	SCREENSHOT
12.04s	3027.52kb		 <p>What is a waterfall chart?</p>

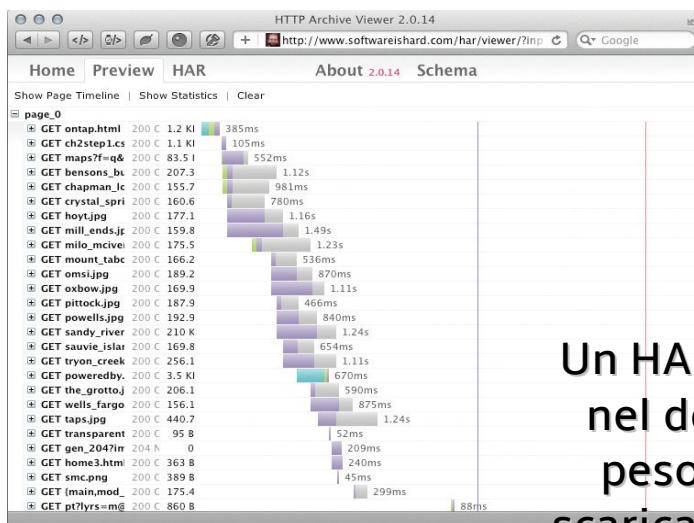
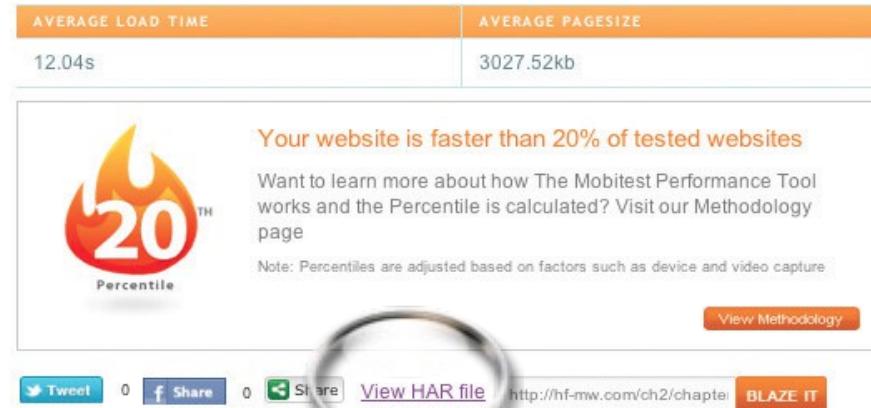
A woman with long dark hair, wearing a denim jacket over a white shirt and dark jeans, stands with her hands on her hips. A thought bubble originates from her head, pointing towards the waterfall chart in the table.

Il waterfall chart è una forma listata di visualizzazione dei dati che ci dice il tempo di caricamento di ogni risorsa impiegata dalla pagina (utile per scovare eventuali colli di bottiglia nel caricamento).

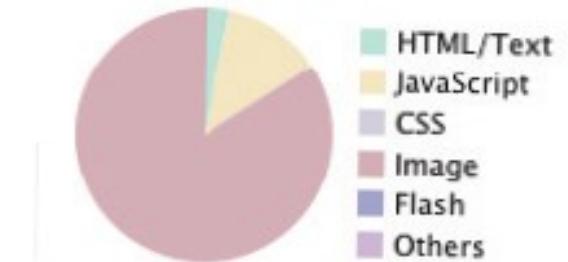
Se questo non fosse abbastanza per voi,
Blaze permette di vedere il file HAR del
vostro test...



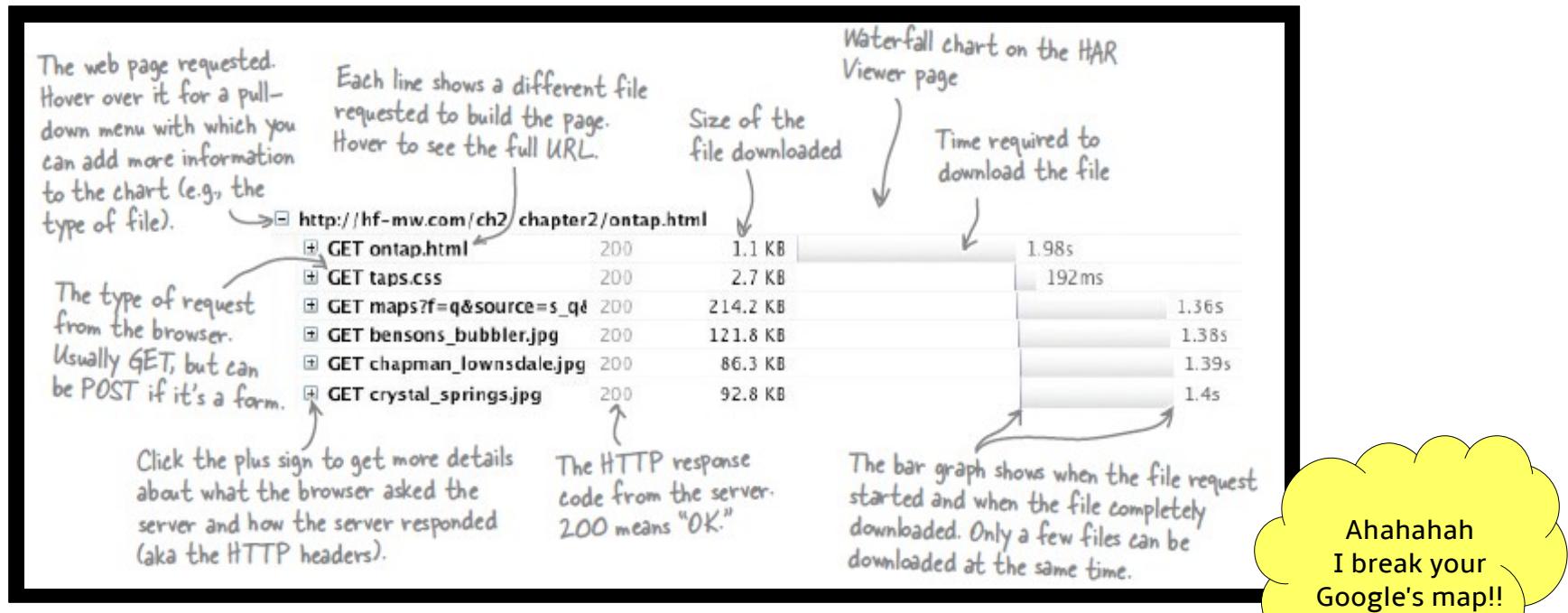
Performance Result Averages for iPhone in Canada, Ottawa



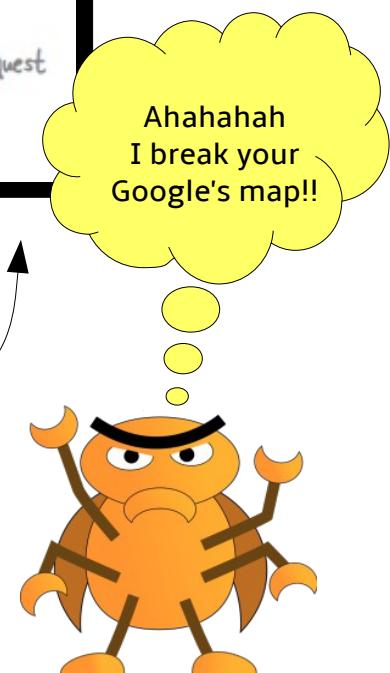
Un HAR (HTTP archive) file mostra più
nel dettaglio le risorse (estensione,
peso, provenienza) che sono state
scaricate e come il server risponde alla
richiesta, come il tempo necessario al
download.



Trova chi rallenta la pagina!



Abbiamo ora tutti i dati per capire cosa rallenta il caricamento della nostra pagina e per risolvere alcuni bugs che intaccano le funzionalità del sito su alcuni dispositivi mobile...



Infatti, quando navigate nella pagina *On Tap Now* di Mike da mobile, il sito non contiene nessuna mappa di Google come dovrebbe, nonostante venga scaricato il file *javascript* correlato...



Questo è un bug da eliminare!!



La mappa è inclusa nel file *ontap.html* all'interno del tag *iframe*, vediamo...

Look inside *ontap.html* to find this code.

```
<iframe id="map" width="300" height="300" frameborder="0" scrolling="no" marginheight="0" marginwidth="0" src="http://maps.google.com..."/></iframe>
```

This single *iframe* causes 47 files to be downloaded!

Extremely long URL abbreviated

Sortunatamente il problema non dipende dal file html, ma dal CSS...



taps.css

```
@media screen and (max-width:480px) {  
    :  
    :  
    :  
    #map {display:none;}  
}
```

There are many more rules in the CSS file.

The iframe has an id of map. This rule hides the Google Maps iframe by setting the display to none.

A hand-drawn style callout box surrounds a snippet of CSS code. The code uses media queries to target screens up to 480px wide. It contains several placeholder colons and a single rule for an element with the ID 'map', which is set to 'display: none;'. A handwritten note above the box says 'taps.css'. Another note to the right of the code states 'There are many more rules in the CSS file.' An arrow points from the word 'map' in the CSS to another handwritten note at the bottom right: 'The iframe has an id of map. This rule hides the Google Maps iframe by setting the display to none.'

Mike ha capito come usare i *media queries* per adattare il layout del sito ai dispositivi mobile, ma la direttiva che setta la vista dell'oggetto *iframe* con tag *map* è *none*, questo fa sì che la mappa venga scaricata ma non mostrata in fase di visualizzazione del sito.

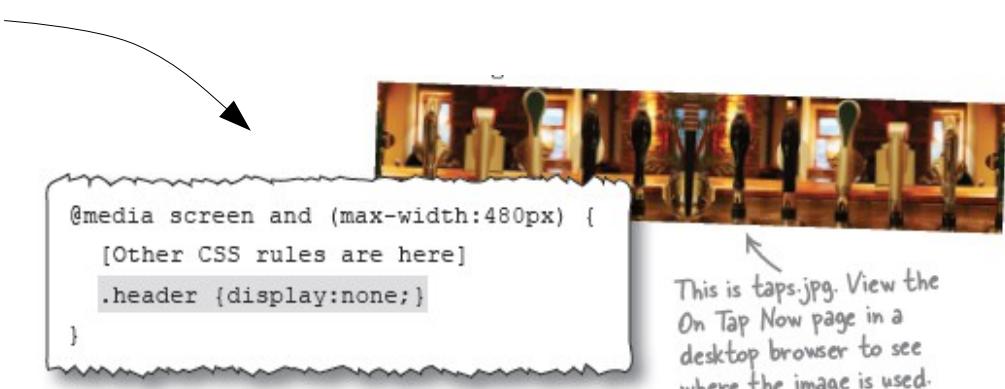
Per visualizzare la mappa utilizziamo il valore di proprietà *run-in*, che visualizza l'elemento in base al contesto.

E con le immagini di grandi dimensioni?

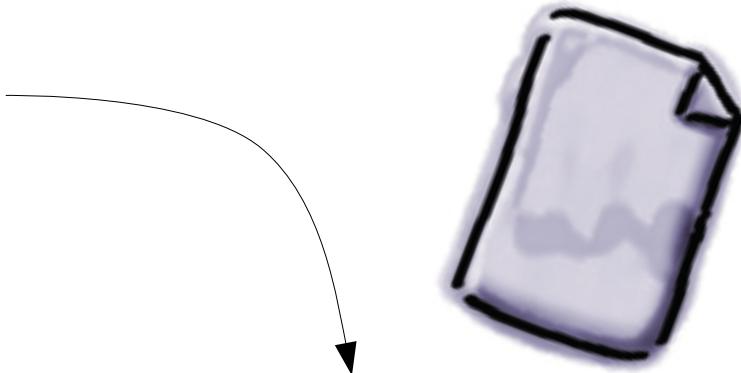


Gurdando il grafico del whaterfall chart notiamo che ci sono immagini di grandi dimensioni che rallentano il caricamento della pagina poiché sono ad elevata risoluzione e poco ottimizzate per i dispositivi mobile...

Grandi dimensioni non vuol dire che il dispositivo non visualizza i contenuti scaricati. Se avviene ciò, come succede a Mike con l'immagine *header*, succede la stessa cosa che capitava con la mappa di Google.



Il metodo delle immagini fluide che abbiamo visto nel capolo I potrebbe sembrare mobile friendly, ma scarica le immagini originali e poi le scala per adattarle allo schermo del dispositivo in uso, questo è un buon metodo solamente se nella nostra pagina abbiamo poche immagini...

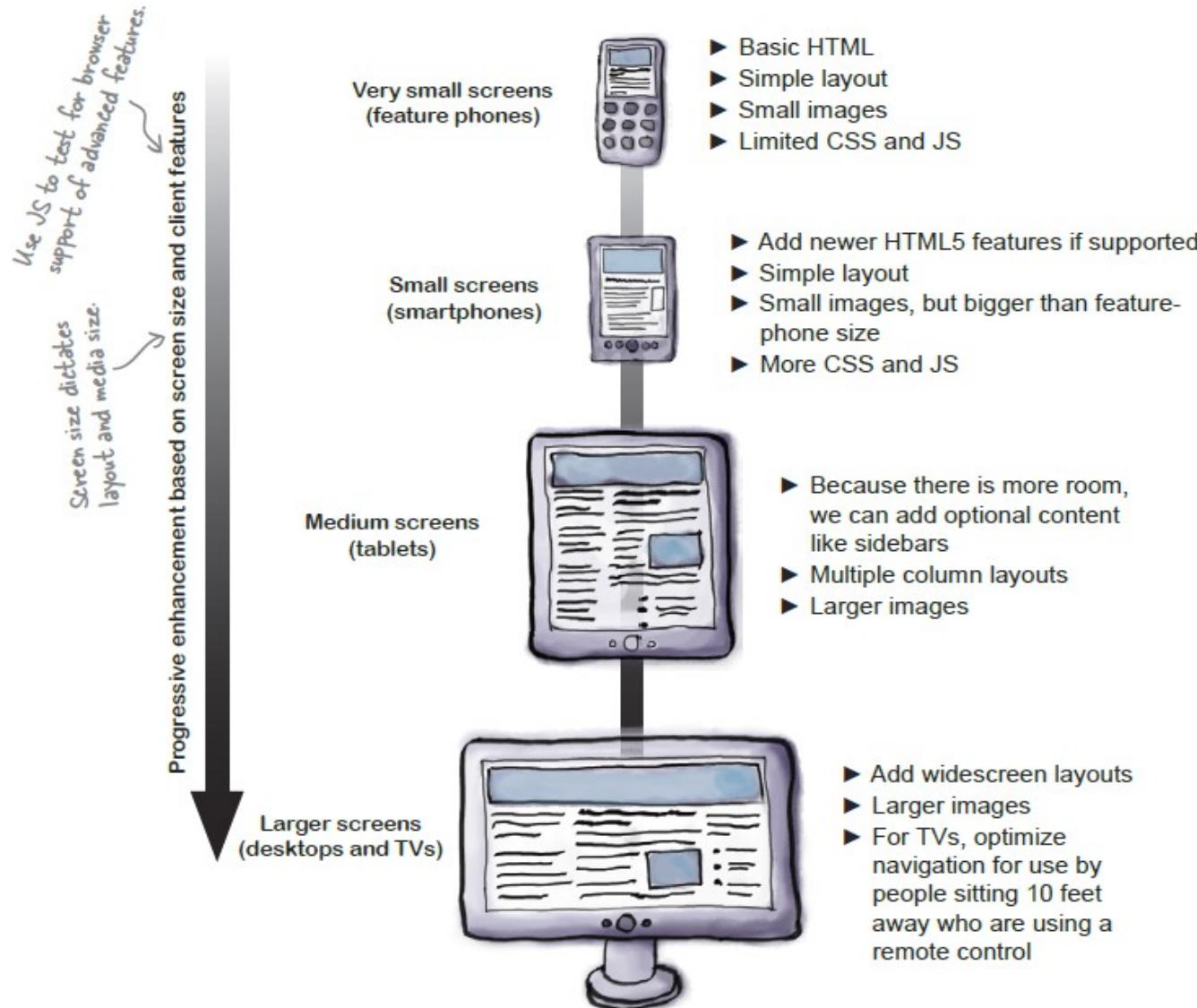


I grandi files sono ancora scaricati!!



La soluzione è creare dei layers per le nostre pagine web: l'idea è quella di far vedere a tutti il contenuto minimo della pagina poi, se si è inoltrata la richiesta da un browser più capace, vengono caricati ulteriori layers di stili e di interattività che ne migliorano l'esperienza per l'utente.

Mobile-first Responsive Web Design

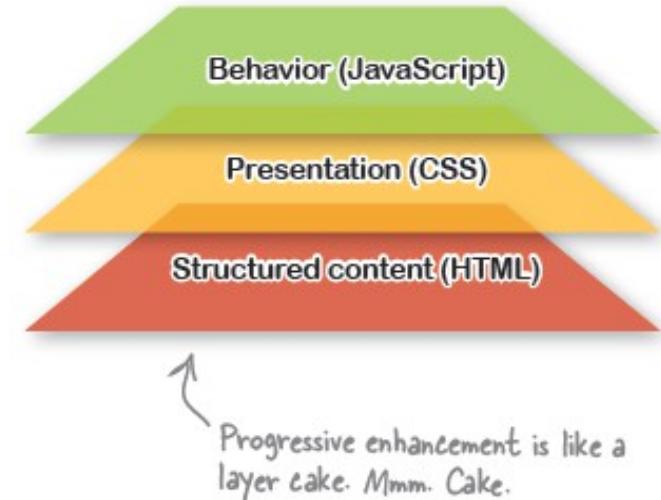


*These are just examples of enhancements. What you do depends on the project.

Si inizia dal template per dispositivi mobile, poi si amplia la struttura della nostra pagina web con progressivi miglioramenti...

La tecnica del web design con migliorie successive vede la pagina come una torta composta da strati diversi: al primo posto ci sono i contenuti combinati dai linguaggi di marcatura, leggibile da ogni browser del mondo.

Nel secondo strato troviamo la presentazione che usa fogli di stile CSS e più in alto il Javascript definisce il layer del funzionamento della pagina.



Quali sono i vantaggi di questa tecnica abbinata al mobile-first web design??



Una volta scelto accuratamente il contenuto del sito per mobile, possiamo ampliarlo per desktop o altri dispositivi, scegliendo cosa implementare o meno della versione precedente, cosa aggiungere di nuovo e gli aspetti da migliorare. E' semplice.

Avendo già usato la tecnica RWD, creare prima la nostra pagina per dispositivi mobile non è troppo complicato, dobbiamo prestare attenzione a:



- Rendere il file *HTML* il più semplice possibile e cambiare l'ordine del *CSS* in modo da caricare il layout per mobile prima.
- Usare il *display:none* del *CSS* con attenzione.
- Supportare origini diverse per i file nei tags ** in base alla risoluzione dello schermo, per caricare solo l'immagine della dimensione corretta.
- Aggiungere il file *Javascript* (come per la mappa di Google) solo quando il *browser* è in grado di supportarlo e la pagina è abbastanza grande.

Two columns
instead of
three

```
<div class="navigation">...</div>
<div class="header">...</div>
<h1>...
<div id="visit" class="column">...</div>
<div id="ontap" class="column">...</div>
<div class="footer">...</div>
```

Instead of a *<div>* called "main," Mike has created a new column with the id of "ontap." That *<div>* contains the list of current beers.

Comment out the Google Maps iframe by surrounding it with *<!--* and *-->*.

```
<!--
<iframe id="map" width="300" height="300" frameborder="0" scrolling="no"
marginheight="0" marginwidth="0" src="http://maps.google.com..."></iframe>
-->
```

Find the *iframe* in the *#visit* *<div>*.



The On Tap Now page looks so similar to the home page. How can visitors tell they're on a different page?

Abbiamo un problema con l'ordine del contenuto: per la *home page* è una cosa buona se, il primo oggetto visto, in versione mobile, è *Visit Us*, ma se questo è ripetuto ad ogni pagina al lettore risulta difficile capire se la pagina è cambiata o meno...

Cambiamo l'ordine della pagina in modo tale che le *On Tap Now Info* vengano viste prima del contenuto *Visit Us*...



Copy everything in the `<div>` with the visit id and paste it below the `ontap` `<div>`.

```
<div class="navigation">...</div>
<div class="header">...</div>
<h1>...</h1>
<div id="ontap" class="column">...</div>
<div id="visit" class="column">...</div>
<div class="footer">...</div>
```

ON TAP NOW AT THE
SPLENDID WALRUS

VISIT US!

Location

1543 SW Pretend Street
Portland, OR 97201

Hours

M-F

Until midnight

Sa-Su

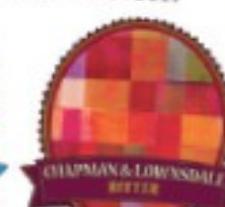
Until 1AM*

*drinks

Call us

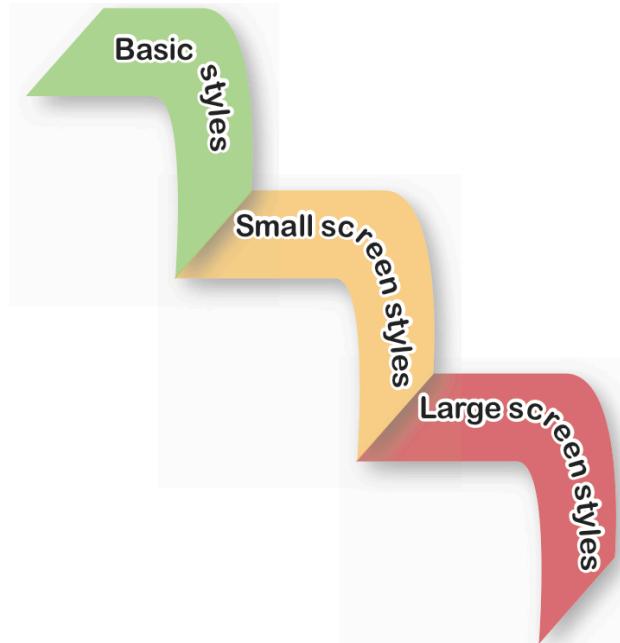
+1 (503) 555-1212

ON TAP THIS MONTH:



Mobile Media Queries

Prima di applicare tutte le media queries per lo schermo del dispositivo mobile dobbiamo settare lo stile di base (Colore, Tipo, ecc..), questo perchè alcuni browser mobile non supportano tutte le queries e, noi dobbiamo essere sicuri che almeno lo stile di base rimanga invariato.



Tieni in ordine il tuo codice

Nonostante si tenti di iniziare con una certa logica e organizzazione i file css sono spesso molto disordinati.

Per tenere le mobile media queries in ordine abbiamo bisogno di separare lo Stile di base dal layout.

Usare commenti condizionali con una media query

taps.css

```
/* Wider viewports/higher resolutions  
 (e.g. desktop) */  
  
@media screen and (min-width:481px) {  
  
    .column {  
        margin: 10px 1.04166667% 0 0;  
    }  
  
    #visit {  
        margin: 0 68.75% 0 0;  
    }  
  
    #points {  
        width: 25%;  
        float: right;  
    }  
  
    #main {  
        margin: 10px 27.0833333% 0  
26.0416667%;  
    }  
  
    #ontap {  
        width: 67%;  
        float: right;  
        margin: 10px 0 0 0;  
    }  
}
```

After you copy them, remove the rules and the surrounding media query from taps.css. We'll reapply the rules to the HTML document next.

layout.css

```
.column {  
    margin: 10px 1.04166667% 0 0;  
}  
  
#visit {  
    margin: 0 68.75% 0 0;  
}  
  
#points {  
    width: 25%;  
    float: right;  
}  
  
#main {  
    margin: 10px 27.0833333% 0  
26.0416667%;  
}  
  
#ontap {  
    width: 67%;  
    float: right;  
    margin: 10px 0 0 0;  
}
```

Copy these
rules to →
your new
file.

1) Crea un file di testo vuoto chiamandolo layout.css e copiaci all'interno le "desktop rules".

2) Aggiungiamo il link al nuovo documento.
Stiamo andando ad aggiungere il link al file layout.css per i browser che supportano le media queries.

```
<link rel="stylesheet" type="text/css" href="taps.css" />  
<link rel="stylesheet" type="text/css" href="layout.css" media="all and  
(min-width: 481px)" />
```

Add this link tag to ontap.html.

The 481px value for min-width was copied from the media query we removed from taps.css.

This is the media query syntax for link tags that you learned in Chapter 1.

```
<link rel="stylesheet" type="text/css" href="taps.css" />  
<link rel="stylesheet" type="text/css" href="layout.css" media="all and  
(min-width: 481px)">  
<!--[if (lt IE 9) & (!IEMobile)]-->  
<link rel="stylesheet" type="text/css" href="layout.css" media="all" />  
<![endif]-->
```

The conditional comment repeats the line above it, ensuring that desktop IE sees our layout.css file.

3)
Aggiungiamo i commenti condizionali di internet explorer.

La vecchia scuola dell'ottimizzazione

Nel corso degli anni il lavoro del web developer è cambiato molto passando dai primi periodi in cui spendevano moltissimo tempo ad ottimizzare le immagini, poi con le nuove reti sempre più veloci hanno iniziato a trascurare questo aspetto, fino all'arrivo dei dispositivi mobile dove è tornata questa esigenza.



Immagini nascoste????

Può accadere che la nostra waterfall chart ci mostri la presenza di un'immagine in background nonostante ci sia la pagina vuota, questo perché è possibile nascondere un'immagine per renderla disponibile solo al download.

Per fare in modo che l'immagine non venga scaricata quando non necessario dobbiamo inserire nella query la possibilità di eseguire il download solo se lo schermo super i 480 pixels.

Copy these lines from
taps.css and add them
to the end of layout.css.

```
{ .header {  
background:URL('images/taps.jpg') repeat-x;  
height: 300px;  
}
```

Delete these lines
from taps.css
after you add
them to layout.css.

Sencha.io Src

Le immagini css sono solo l'inizio dei nostri problemi in quanto all'Src può essere assegnato soltanto un valore a prescindere dal fatto che quell'immagine venga visualizzata su schermi diversi.

Tentiamo di modificarlo con il javascript.

There are 16 beer labels on the On Tap Now page that use an tag like this one for Benson's Bubbler.

Despite the need for multiple versions of this image depending on the screen size, HTML only allows one value for the src.

```

```

Il browser potrebbe non riuscire a raggiungere il server per ottenere l'immagine delle giuste dimensioni, in quel caso farà da solo, usando Senche.io Src

Set the first part of the src to
`http://src.sencha.io/`.

Replace with your domain and
path to the images.

After the slash, add the full URL of
the image you want to have resized.

```

```

Sencha.io Src will resize the image to fit the size of the device screen. For example, if an iPhone visits the site, the image will be constrained to its screen size of 320 by 480 pixels.

Come funziona Sencha.io Src?

Sencha.io Src usa un browser's user agent string per identificare con che versione del browser si sta accedendo, per poi controllare un database che contiene informazioni su migliaia di dispositivi tra cui anche la grandezza dello schermo.





Sorry, guys! I hate to spring a new requirement on you in the middle of your work, but one of my best customers has trouble seeing small text and is complaining that she can't zoom the page. Can you fix it?

viewport <meta> tag indicano al browser le dimensioni della pagina e il livello di zoom che deve essere applicato e, contengono delle istruzioni che possono impedire agli utenti di cambiare le dimensioni della pagina.

What type of tag is this?

```
<meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1" />
```

The content attribute contains a comma-separated list of instructions for the browser. See all of the options at <http://bit.ly/metaviewport>.

Width of the viewport. Can be set in pixels or can be set to "device-width," which tells the browser to match the viewport to the device resolution.

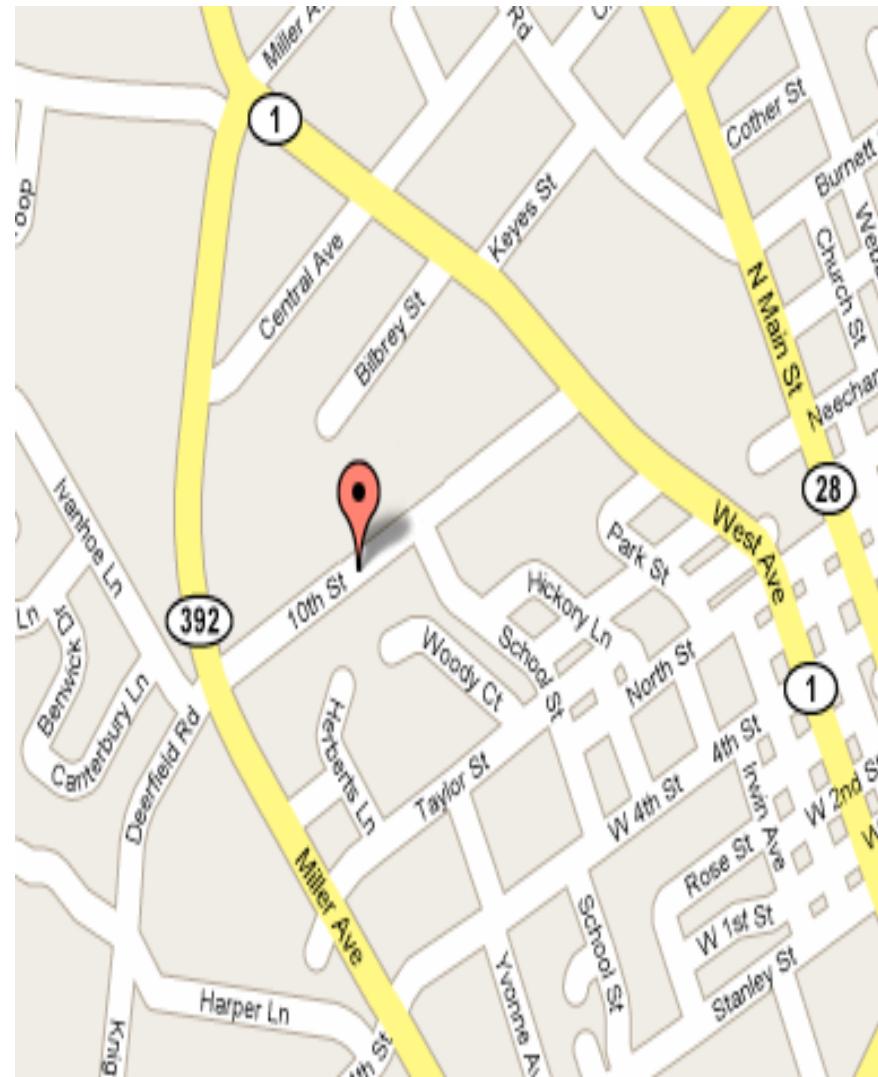
Sets the initial scale (or zoom level) of the page. Setting it to 1 means that the document should be displayed at its normal scale.

Declares a limit on how much the page can be scaled up. There is also a similar minimum-scale setting.

The maximum-scale is what is preventing the users from zooming the page.

Gestione della mappa

L'ultima cosa che ci rimane da fare è aggiungere la mappa di Google. Poiché quest'ultima è molto pesante ed è ottimizzata per funzionare bene sopra certe grandezze, verrà aggiunta solamente se lo schermo del dispositivo, per cui stiamo sviluppando il sito web, lo rende possibile. Tutto quello che diremo in questa parte varrà anche per tutti gli altri componenti di terze parti che vorrete utilizzare.



Aggiungiamo la mappa

Per questo scopo abbiamo due vie percorribili, una nettamente migliore dell'altra. La mappa può essere aggiunta sia via CSS, che Javascript. Il primo metodo ha lo svantaggio che le risorse necessarie per la visualizzazione di quest'ultima vengono scaricate, andando quindi ad appesantire di molto la pagina web.

Risulta perciò vantaggioso utilizzare il secondo metodo. Come primo passo cancelliamo il tag <iframe> che contiene la mappa.

Remember this iframe snippet that
we commented out? We're going to use
JavaScript to insert it into the page.



```
<!--  
  <iframe id="map" width="300" height="300" frameborder="0" scrolling="no"  
  marginheight="0" marginwidth="0" src="http://maps.google.com..."/></iframe>  
-->
```

Una mappa può essere sempre utile

Oggiorno molti dispositivi possono, attraverso qualche metodo di triangolazione del segnale telefonico od attraverso il gps, segnalare la loro posizione. Quindi è lecito andare ad utilizzare una mappa nel proprio sito web.

Un problema che porta con se è quello dell'ingombro. Quindi, se non vogliamo fare a meno di quest'ultima possiamo utilizzare alcuni espedienti, come per esempio, invece di visualizzarla, mettere il link sulla pagina, come descritto qui sotto.

We're going to need a container for the JavaScript to reference, so we'll add a <div> here.

The id on the <p> tag will allow us to insert the iframe above the link for wider screens. More on this soon.

Add the new <div> above the commented-out iframe code.

```
<div id="mapcontainer">
  <p id="maplink">
    <a href="http://g.co/maps/jm2pw">View Google Map</a>
  </p>
</div>

<!--
  <iframe id="map" width="300" height="300" frameborder="0"
  scrolling="no" marginheight="0" marginwidth="0"
  src="http://maps.google.com..."/>
-->
```

Costruiamo la nostra (pseudo) media query

Questo codice in JavaScript ci permette di inserire la mappa senza l'utilizzo di HTML, comportandosi similmente ad una media query. Come noto, questo script dovrà essere aggiunto alla pagina per il corretto funzionamento.

```
<script type="text/javascript">
var breakpoint = 481,
    id = 'mapcontainer',
    viewportWidth = window.innerWidth;
if (viewportWidth > breakpoint) {
  var mapElement = document.createElement('iframe');
  mapElement.id = 'map';
  mapElement.width = '300';
  mapElement.height = '300';
  mapElement.frameborder = '0';
  mapElement.scrolling = 'no';
  mapElement.marginheight = '0';
  mapElement.marginwidth = '0';
  mapElement.src = 'http://maps.google.com/maps?f=q&source=s_q&hl=en&geocode=&q=334+NW+11th+Ave,+Portland,+OR+97209&aq=&sll=37.0625,-95.677068&sspn=58.164117,80.332031&vpsrc=0&ie=UTF8&hq=&hnear=334+NW+11th+Ave,+Portland,+Oregon+97209&t=m&ll=45.525472,-122.68218&spn=0.01804,0.025749&z=14&output=embed';
  document.getElementById(id).insertBefore(mapElement, maplink);
}
</script>
```

Sets the breakpoint variable to 481 pixels. The breakpoint is the width at which the map will be added to the page.

Checks to see if the window viewport is larger than the breakpoint

These lines add all of the attributes to our new iframe element. The attributes and their values were copied from the Google Maps iframe snippet

This final step adds the iframe (mapElement) into the mapcontainer <div> (id) before the paragraph containing the link (maplink).

This variable is for the id of the element we want to add the map to. We're using a variable to make the <div> easier to change in the future.

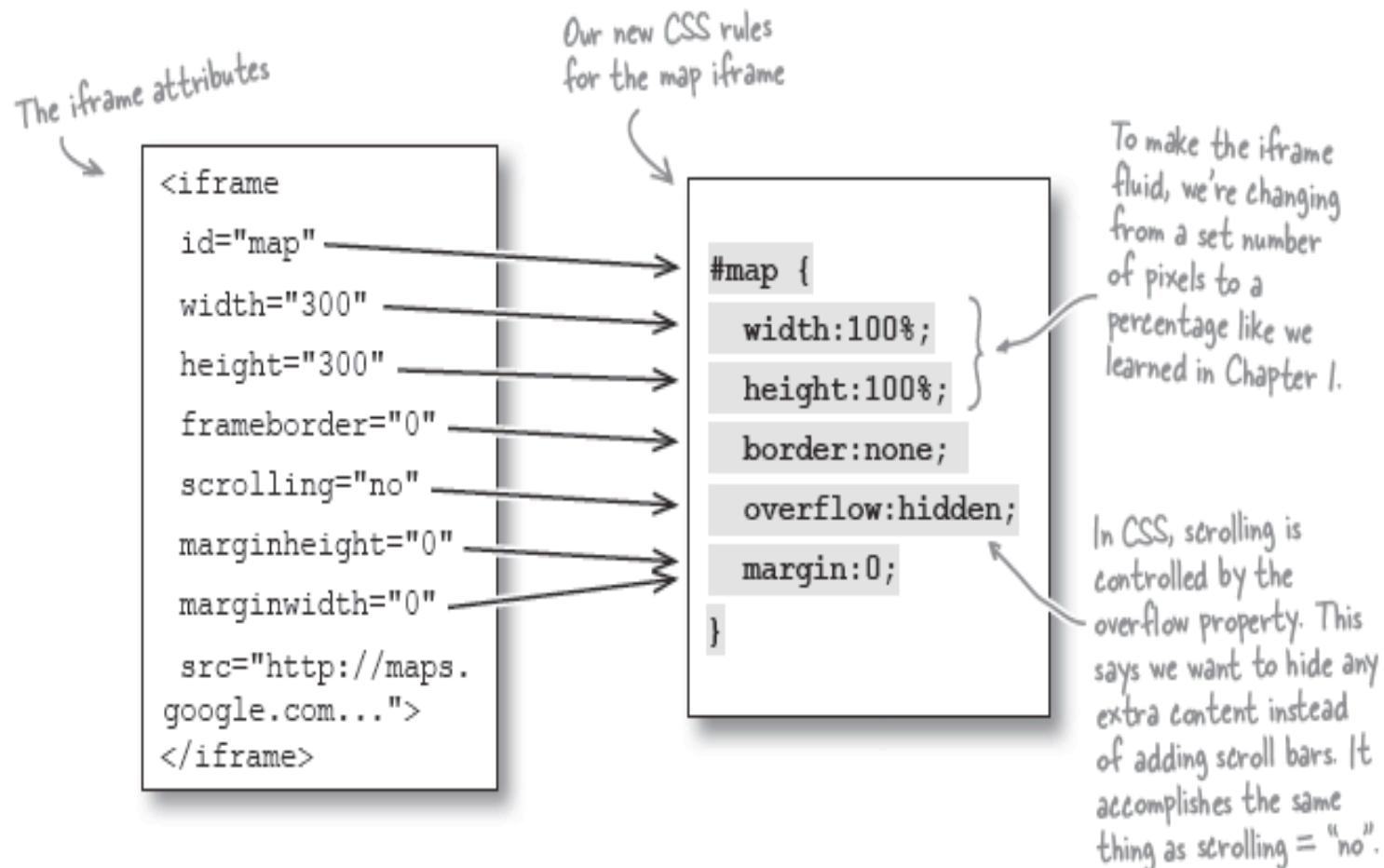
Adds a new iframe element and assigns it to the mapElement variable

The URL Google Maps provides is ugly. You probably don't want to retype it. Find a copy of this code in extras/map.js.

Una buona tecnica per utilizzare script in JavaScript, di minor importanza, all'interno delle pagine web è quella Inserirli in fondo alla pagina, prima della chiusura di <body>. In questo modo la pagina verrà caricata e l'utente potrà utilizzarla immediatamente.

Reattività dei componenti di terze parti

In generale, tutti i componenti di terze parti vengono creati per far sì che loro funzionino in tutti i casi possibili. Molto spesso questo viene fatto senza un occhio di riguardo al possibile impatto sul layout dei siti web. Quindi un'ottima tecnica per utilizzarli in qualsiasi caso è quella di "giocare" con i loro attributi, spostando quest'ultimi dalla pagina HTML al foglio di stile. Utilizziamo questo con la nostra mappa!

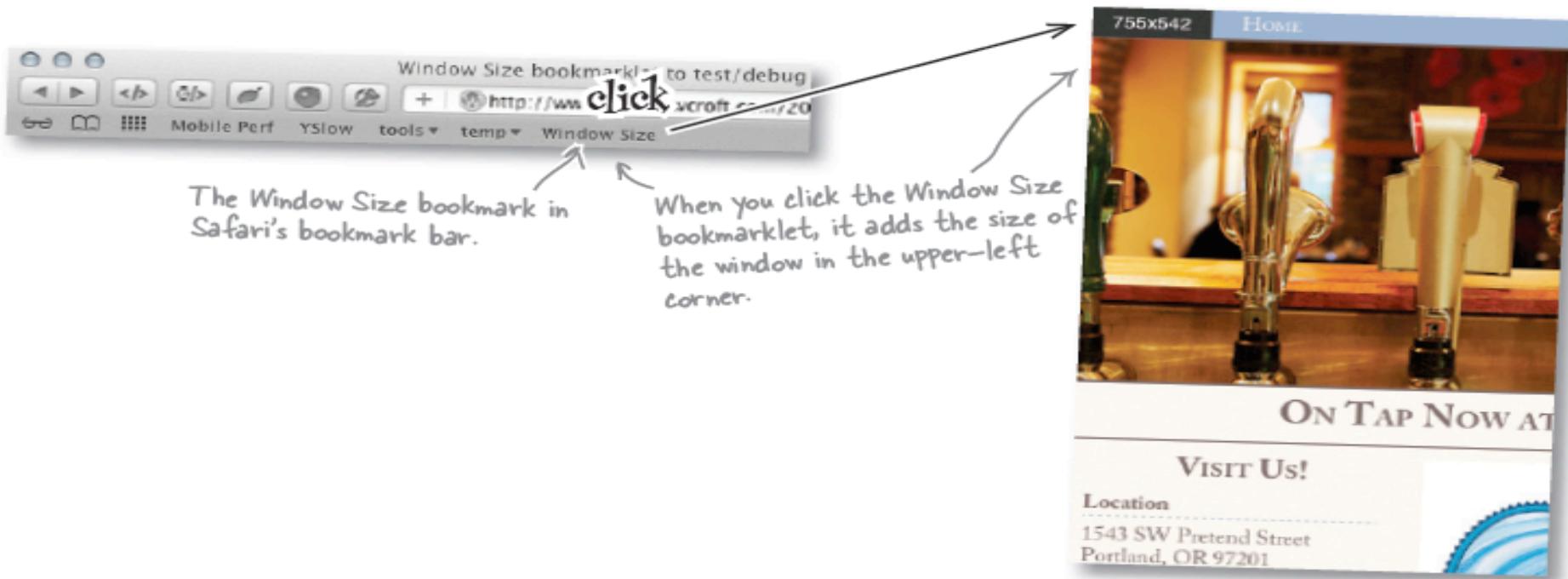


Una volta spostati bisogna cancellare i rispettivi elementi dallo script precedentemente creato.

Fatti guidare dal tuo contenuto

Un problema nell'utilizzo di una dimensione massima, che nel nostro caso è 480px, come breakpoint nelle media query è che questa non è egualmente utilizzata in tutti i dispositivi. Un approccio per evitare questo problema è "farsi guidare" dal proprio layout, cioè cambiare la dimensione della pagina e vedere i cambiamenti. Quest'ultima azione è difficile effettuarla senza degli appositi tool che facilitano il lavoro.

Uno di questi è un bookmark (scritto in JavaScript e raggiungibile a questo link <http://bit.ly/window-resize>) attraverso il quale è possibile scoprire la grandezza attuale della pagina. Altro tool è, invece, un'estensione di Chrome (<http://bit.ly/chrome-resizer>) che permette, oltre a conoscere la dimensione attuale della pagina anche di adattarla tra quelle più comuni. Ultimo tool è anch'esso un plugin (sia per Firefox che Chrome <http://bit.ly/webdevtoolkit>) che ha una funzione simile al precedente.



Nuovi breakpoint

Ritorniamo al nostro sito web da ottimizzare. In questo momento, se la larghezza rimane invariata, ci sono tra etichette di birra per ogni riga, però se questa aumenta c'è lo spazio per aggiungerne un'altra. Possiamo sfruttare questo spazio aggiungendo una nuova media query.

Setting the width of the list item (li) containing the beer labels to 25% will put four labels on each row.

```
@media screen and (min-width:1201px) {  
    .taplist li {  
        width: 25%;  
    }  
}
```

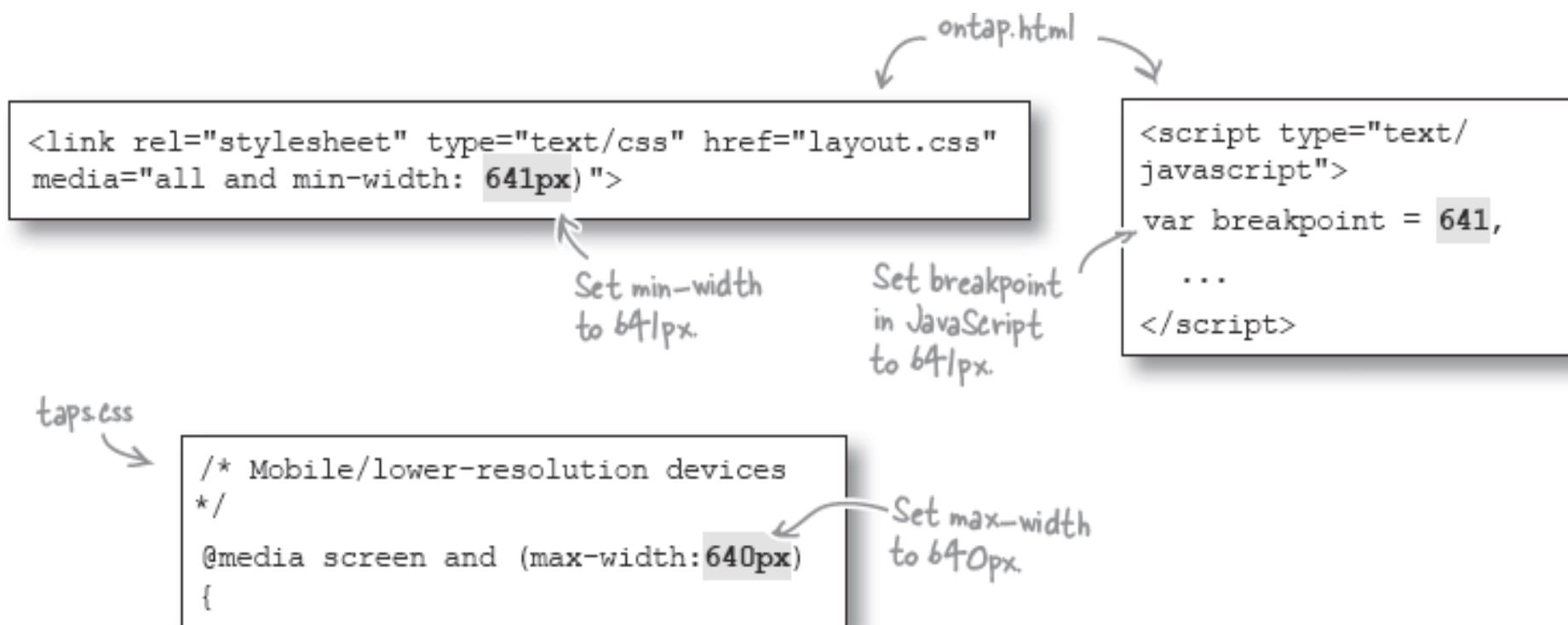
This change only happens if the window is bigger than 1,200 pixels.

Add these rules to layout.css

It's common to make images smaller proportionally as screens get wider.

Ultimi ritocchi

Un problema che si può osservare se si aggiunge un nuovo breakpoint a 640px è che la pagina, anche se viene visualizzata bene, è affollata. Quindi, invece di aggiungerne uno nuovo, possiamo trasformare quello da 480px in un da 640px. Facendo questo cambiamento andiamo a creare un layout formato da una sola colonna e a coprire la mappa. Inoltre abbiamo il vantaggio di andare a creare un layout con lo stesso stile per dispositivi con una risoluzione maggiore di 480px. Quindi applichiamo queste modifiche come descritto sotto.



Abbiamo finito!

You guys rock! The page is fast and looks great. Drinks are on the house.

Narrower views go to one column and hide the map.

Lightweight and fast on mobile

Widescreen view with four beer labels per row

On Tap Now at The Splendid Walrus

ON TAP THIS MONTH:

- MENSON'S BUBBLER MIN. ALCOHOLIC CITY
- CINNAMON DOWNSIDE BITTER
- CRYSTAL SPRINGS RESERVE
- HOYT
- REASON'S BUBBLER MIN. ALCOHOLIC BEER
- CRYSTAL SPRINGS RESERVE
- CRYSTAL SPRINGS RESERVE
- HOYT
- MILL ENDS PILS ALE
- MILD McIVY'S IRISH RED
- ROCKFORD SPARROW PILS ALE
- OMSI GINGER ALE

Performance Result Averages for iPhone in Canada, Ottawa

AVERAGE LOAD TIME	AVERAGE PAGESIZE
3.75s	386.80kB

Your website is faster than 70% of tested websites

Want to learn more about how The Mosaic Performance Tool works and the Percentile is calculated? Visit our Methodology page.

Percentiles are adjusted based on factors such as device and video capture.

View Methodology

http://hf-mw.com/ch2/ex/8/ontap.html

Bibliografia

- Head First Mobile Web – Chapter 2 – Responsible Responsiveness: Mobile-First Responsive Web