

Umberto Emanuele

# HTML & CSS

## 3 Giorno

---

Novembre 2023



---

Responsive design & develop

Si definisce Responsive Design l'approccio di sviluppo per adattare il contenuto di una pagina web o di un app a qualunque schermo utilizzabile.

Non è un linguaggio fisico o figurato (come ad esempio il Material Design) ma solo un modo di utilizzare HTML e CSS.

## Mobile first

il «vecchio» adagio dell'era mobile.

Oggi è uno standard.

Non è un linguaggio fisico o figurato (come ad esempio il Material Design) ma solo un modo di utilizzare HTML e CSS.



*Gli utenti mobile hanno da tempo superato gli utenti desktop.*



*Da Marzo 2021 Google indicizza solo le versioni mobile*

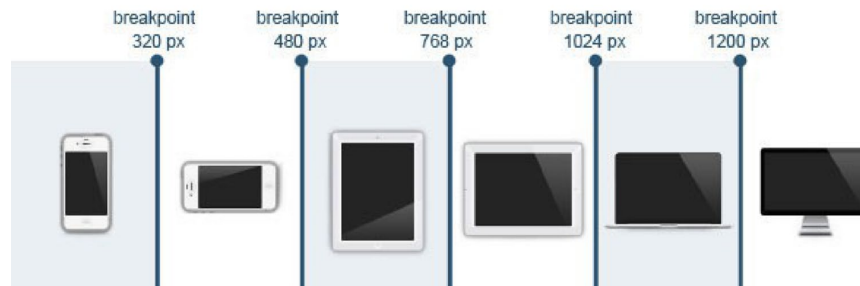
# Media queries

Principi di responsive design



Il breakpoint è un punto, su una linea ideale che parte da 0, in cui si verifica una qualche modifica (tramite i CSS) al layout della pagina.

I breakpoint si definiscono con valori numerici e unità di misura nel contesto delle media queries inserite nei CSS.



```
@media screen and (max-width: 767px) {  
  main {  
    font-size: 1.5rem;  
  }  
}
```

Ad ogni breakpoint c'è un sostanziale mutamento del layout.



```
@media screen and (min-width: 450px) {  
  body {  
    font-family: Arial;  
  }  
}  
  
@media print {  
  body {  
    font-family: Georgia;  
  }  
}
```

<b>All</b>	Valore predefinito
<b>Screen</b>	Per schermi dei dispositivi
<b>Print</b>	Per stampanti
<b>Speech</b>	Per screen readers



```
@media screen and (min-width: 450px) {  
  body {  
    font-family: Arial;  
  }  
}
```

```
@media print {  
  body {  
    font-family: Georgia;  
  }  
}
```

Width / Height	Larghezza / Altezza del viewport
Min - width	Larghezza minima
Max - width	Larghezza massima
Min - height	Altezza minima
Max - height	Altezza massima
Orientation	Portrait / Landscape

---

Rendere elementi responsive

```
img.responsive { /* ingrandisce e rimpicciolisce */  
  width: 100%;  
  height: auto;  
}
```

```
img.responsive { /* rimpicciolisce solamente */  
  max-width: 100%;  
  height: auto;  
}
```

```

```

```
<div class="responsive">
  <table>
    <tr>
      <td>Cella 1</td>
    </tr>
  </table>
</div>
```

```
div.responsive {
  overflow-x:auto
}
```

**Pro:** soluzione più semplice e crossbrowser.

**Contro :** costringe allo scrolling orizzontale.

## Soluzione 1

Utilizzare griglie di un grid framework per gestire affiancamenti di label e input.

## Soluzione 2

Utilizzare flexbox + mediaqueries.

## Soluzione 3

Impostare le proprietà display con inline/block/inline-block/table per i tag label e input.

Variare in base al viewport.

<b>Column -count</b>	Numero di colonne da creare
<b>Column -width</b>	Larghezza minima ipotizzata
<b>Column -gap</b>	Distanza fra le colonne
<b>Column -fill</b>	Bordi di riempimento

---

# **Animazioni 1: introduzione, CSS Transform 2D e 3D**



## No JS, no Flash!

Dopo anni di sintassi e tecnologie supplementari, a volte usate solo per piccole animazioni, abbellimenti grafici o semplici animazioni, molti effetti possono essere realizzati utilizzando solo nuove proprietà CSS3.

## Pro

- Ampio supporto dei browser più recenti
- Degradabili senza problemi
- Zero tecnologie di terze parti

<code>translateX()</code> / <code>translateY()</code> / <code>translate(x,y)</code>	Sposta un oggetto sull'asse X o Y
<code>rotate(gradi)</code>	Ruota un oggetto in senso orario o antiorario
<code>scaleX(1)</code> / <code>scaleY(1)</code> / <code>scale(1,1)</code>	Ridimensiona un oggetto in altezza/larghezza
<code>skewX(gradi)</code> / <code>skewY(gradi)</code> / <code>skew(gradi, gradi)</code>	Distorce un oggetto sull'asse X o Y
<code>matrix()</code>	Indica tutte le trasformazioni contemporaneamente
<b>Proprietà</b>	
<code>transform -origin</code>	Proprietà. Sposta l'asse X e Y

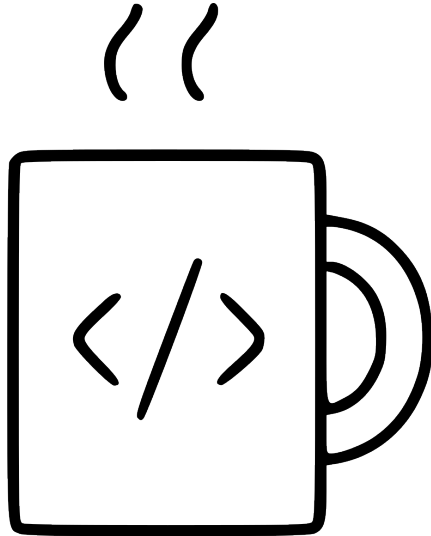
---

```
div { transform: translate(30px, 70px); } /* sposta di 30px da sinistra e 70px dall'alto */
div { transform: rotate(-20deg); } /* ruota di 20 gradi a sinistra */
div { transform: scaleX(2); } /* raddoppia la larghezza */
div { transform: scaleY(3); } /* triplica l'altezza */
div { transform: scale(0.7, 0.7); } /* riduce del 30% le dimensioni */
div { transform: skewX(35deg); } /* inclina di 35 gradi verso sinistra */
div { transform: skewY(-15deg); } /* inclina di 15 gradi verso il basso */
div { transform: skew(15deg 15deg); transform-origin: 20% 25%; } /* distorce su 2 assi e sposta gli assi */
```

---

<b>translateZ()</b>	Sposta un oggetto sull'asse Z
<b>rotateZ (<i>gradi</i>)</b>	Ruota un oggetto sull'asse Z
<b>rotate3D(x,y,z)</b>	Imposta una rotazione 3D
<b>scaleZ(1)</b>	Ridimensiona un oggetto sull'asse Z
<b>Proprietà</b>	
<b>transform -style</b>	Flat   preserve-3d
<b>perspective</b>	Distorce un oggetto sull'asse X o Y

**transform:** perspective(100px) rotateZ(50deg) // rotateX/rotateY/rotateZ



**PAUSA**

Ci vediamo alle ore 14.00

---

## **Animazioni 2: CSS Transitions**

**CSS Transitions** prevedono solo uno start ed un end.

A differenza delle trasformazioni, si interviene su **come** venga elaborata una proprietà, non su **cosa** faccia.

<b>transition -property</b>	Quale proprietà far animare
<b>transition -duration</b>	Quanto dura la transizione
<b>transition -timing -function</b>	Velocità di movimento della transizione
<b>transition -delay</b>	Eventuale ritardo di avvio (s/ms)
<b>transition</b>	Sintassi abbreviata

```
div {  
  width: 200px;  
  height: 100px;  
  transition-property: width;  
  transition-duration: 2.5s;  
}  
  
div:hover {  
  width: 400px;  
}
```

```
div {  
  width: 200px;  
  height: 100px;  
  transition-property: width;  
  transition-duration: 2.5s;  
}  
  
div:hover {  
  transform: scaleX(3);  
}
```



```
div {  
  width: 200px;  
  height: 100px;  
  background-color: ■ red;  
  transition: width 3s, background-color 3s;  
}  
  
div:hover {  
  width: 400px;  
  background-color: ■ blue;  
}
```

---

<b>Ease</b>	Parte piano, accelera, rallenta. default
<b>Linear</b>	Velocità costante
<b>Ease-in</b>	Bassa velocità iniziale
<b>Ease-out</b>	Bassa velocità finale
<b>Ease-in-out</b>	Bassa velocità iniziale e finale
<b>Cubic -bezier(n,n,n,n)</b>	Valori customizzati

---

## **Animazioni 3: CSS Animations**

Micro filmati di oggetti in movimento e/o trasformazione. Prevedono punto di partenza ed arrivo, step intermedi tramite keyframes e ripetizioni cicliche. Vanno precostruite e poi richiamate.

<b>animation -name</b>	Chiamata al nome dell'animazione
<b>animation -duration</b>	Durata dell'animazione
<b>animation -timing -function</b>	Velocità di movimento dell'animazione
<b>animation -delay</b>	Eventuale ritardo di avvio (s/ms)
<b>animation -iteration -count</b>	Numero di ripetizioni
<b>animation -direction</b>	Svolgimento in avanti o contrario
<b>animation -fill -mode</b>	Stili applicati ad inizio o fine animazione
<b>animation</b>	Sintassi abbreviata

```
@keyframes nomeAnimazione { /* definizione dell'animazione*/
  from { /* punto di partenza - from/0%/0 */}
  50% { /* eventuale step intermedio (Keyframe selector */}
  to { /* punto di arrivo - to/100% */}
}

.oggetto {
  /* name, duration , timing-function, delay, iteration-count, direction, fill-mode*/
  animation: nomeAnimazione 0.5s cubic-bezier(0.1,0.25,0.1,1) 0s infinite alternate both;
}
```

I prefissi vendor (o prefissi browser) sono applicabili a tante proprietà che non sono sempre applicate “alla lettera” dai browser, sperimentali oppure per supportare caratteristiche proprietarie (non W3C standard).

<b>Chrome, Safari, iOS, Android</b>	<code>-webkit</code>
<b>Firefox</b>	<code>-moz</code>
<b>Internet Explorer</b>	<code>-ms-</code>
<b>Opera</b>	<code>-o-</code>

---

## **Animazioni 4: librerie esterne e menu di navigazione**

---

<b>Animate CSS</b>	<a href="https://daneden.github.io/animate.css/">https://daneden.github.io/animate.css/</a>
<b>Hover CSS</b>	<a href="http://ianlunn.github.io/Hover/">http://ianlunn.github.io/Hover/</a>
<b>Animista</b>	<a href="http://animista.net/">http://animista.net/</a>
<b>Motion UI</b>	<a href="https://zurb.com/playground/motion-ui">https://zurb.com/playground/motion-ui</a>
<b>All Animation</b>	<a href="https://all-animation.github.io">https://all-animation.github.io</a>
<b>Woah</b>	<a href="http://www.joerezendes.com/projects/Woah.css/">http://www.joerezendes.com/projects/Woah.css/</a>





*shaping the skills of tomorrow*

challengenetwork.it

