





Hello

Alessandro Pasqualini

Full-stack developer, appassionato di programmazione e sviluppo software in generale, amo le serie tv e i gatti

Responsive



Responsive

Da Wikipedia:

"Il **design responsivo**, ^[1] o *responsive web design* (**RWD**), indica una tecnica di <u>web design</u> per la realizzazione di <u>siti</u> in grado di adattarsi graficamente in modo automatico al dispositivo coi quali vengono visualizzati (computer con diverse <u>risoluzioni</u>, <u>tablet</u>, <u>smartphone</u>, <u>cellulari</u>, <u>web tv</u>), riducendo al minimo la necessità dell'utente di ridimensionare e scorrere i contenuti."



Design responsive vs Design adattivo

Design responsive

Un sito responsivo si adatta esclusivamente alla forma e dimensione dello schermo del dispositivo. La pagina web deve adattarsi in modo fluido alle dimensioni dello schermo.

Es. Facebook e Instagram



Design responsive vs Design adattivo

Design adattivo

Una soluzione adattiva è ottimizzata completamente per i device mobili, ciò garantisce una migliore user experiencees. Esistono più "versioni" della stessa pagina web: una per i dispotivi mobile, una per i dispositivi desktop, etc. E' possible anche che alcune funzionalità non siano presenti nella versione mobile.

es. Google e Amazon



Design responsive vs Design adattivo

Design adattivo

Una soluzione adattiva è ottimizzata completamente per i device mobili, ciò garantisce una migliore user experiencees. Esistono più "versioni" della stessa pagina web: una per i dispotivi mobile, una per i dispositivi desktop, etc. E' possible anche che alcune funzionalità non siano presenti nella versione mobile.

es. Google e Amazon



Mobile first vs Desktop first

Mobile first e **desktop first** sono due strategie di sviluppo responsive.

Mobile first

La pagina web viene sviluppata inizialmente per un dispositivo mobile e successivamente "adattata" per gli schermi più grandi.

Desktop first

Esattamente il contrario: la pagina è sviluppata concentrandosi su un dispositivo desktop e poi "adattata" per schermi più piccoli



Mobile first vs Desktop first

Quale scegliere?

Mobile first: se il traffico generato dal sito web proviene principalmente da dispositivi mobile.

Desktop first: se il traffico generato dal sito web proviene in larga parte da dispositivi desktop.

L'idea è che la pagina web deve essere ottimizzata per la maggior parte dei visitatori.



Mobile first vs Desktop first



Responsive in bootstrap



Appriccio mobile first

Bootstrap 4 è sviluppato utilizzando un approccio mobile first.

I suoi componenti e le sue classi sono sviluppate e ottimizzate per dispositivi mobile e poi «adattate» per dispositivi desktop.

La maggior parte dei componenti sono sviluppato usando la media query **min-width**

In alcuni casi, per alcuni componenti complessi, viene usata la media query **max-width** (tipica di approcci desktop-first)



Breakpoints

Un **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**

```
es.
@media (min-width: 768px) {
    ...
}
```



Breakpoints di bootstrap

```
// Extra small devices (portrait phones, less than 576px)
// No media guery for `xs` since this is the default in Bootstrap
// Small devices (landscape phones, 576px and up)
@media (min-width: 576px) { ... }
// Medium devices (tablets, 768px and up)
@media (min-width: 768px) { ... }
// Large devices (desktops, 992px and up)
@media (min-width: 992px) { ... }
// Extra large devices (large desktops, 1200px and up)
@media (min-width: 1200px) { ... }
```



Es. breakpoints

| | Extra small <576px | Small ≥576px | Medium ≥768px | Large ≥992px | Extra large ≥1200px |
|------------------|--------------------|-----------------|-------------------------|------------------------|------------------------|
| .container | 100% | 540px | 720px | 960px | 1140px |
| .container-sm | 100% | 540px | 720px | 960px | 1140px |
| .container-md | 100% | 100% | 720px | 960px | 1140px |
| .container-lg | 100% | 100% | 100% | 960px | 1140px |
| .container-xl | 100% | 100% | 100% | 100% | 1140px |
| .container-fluid | 100% | 100% | 100% | 100% | 100% |



Display

Bootstrap permette di modificare la proprietà display attraverso delle classi predefinite.

```
.d-{value}
.d-{breakpoint}-{value}
```

I breakpoint sono: sm, md, lg, e xl

Mentre i valori per **value** sono: none, inline, inline-block, block, table, table-cell, table-row, flex, inline



Display

Le classi

- .d-none
- .d-{breakpoint}-none

Sono usate per nascondere del contenuto in determinati dispositivi.

Quando usiamo i breakpoint bisogna ricordare che si applicano dal breakpoint scelto fino ad xl. Es. d-none nasconde tutto, d-md-none nasconde il contenuto per md, lg e xl ma lo farà vedere per xs e sm.

Utility class di bootstrap



Position

```
<div class="position-static">...</div>
<div class="position-relative">...</div>
<div class="position-absolute">...</div>
<div class="position-fixed">...</div>
<div class="position-sticky">...</div>
```

Con queste classi è possibile modificare la proprietà position degli elementi senza scriverla direttamente nei CSS



Float e clear

Example Button floated left

Example Button floated right

```
<div class="bg-info clearfix">
    <button type="button" class="btn btn-secondary float-left">...</button>
    <button type="button" class="btn btn-secondary float-right">...</button>
    </div>
```

Esercizio 1

Traveler

Rifare la pagina usando bootstrap e suoi componenti.

https://raw.githubuserco ntent.com/howtosrc/fro ntend-padova-21/master/es/es1.zip?to ken=ABCQIKV7Q6F2QZ WTWZDF4XLAY5K6K 0



Travel diaries

HOME BLOG RESUME CONTACT



I'M A GREAT TRAVELLER

Lorem ipsum dolor sit amet, consectetur adipticing ditt. Aliquam nec sagittis massa. Nulla facilist. Cras i darcu lorem, et seroper purus. Cum sociis natoque peneratibus et magnis dis parturient montes, nascetur ridiculus mus. Dius vel enim mi, in lobortis sem. Vestibulum luctus elit eu libero ultrices loi fermentum sem eguttis. Nulla imperente must ses deri digirismi noi aliquament aliquament mescence si demonte propriati a del propriati del propriati de la compania del compania

Vivamus sed libero nec mauris pulvinar facilisis ut non sem. Quisque mollis ullamcorper diam vel fauchus, vestibulum solicitudin facilisis fouglat. Nulla eusimos dostles hendrerit. Done cqui sord arcu, Vivamus fermentum mages a erat ullamcorper diginisismi, volume aliquam. Anenen pulvinar condimentum enim a diginismi. Vivamus si amet leicus is at ane adpission, advoluncia ged vital erisis. In al fringilia est. Cas id imperdiet in quam. Sed vestibulum, libero ut bibendum consectetur, eros ipsum ultrices nisl, in rutrum diam augue no tortor. Fueze ne massa et risus dapibus aliquam vitale nec diam.

Phasellus ligula massa, conque ac vulgutate non, dignissim at aque. Sed auctor fringilla quam quis portitor, Praesent vitae dignissim magan. Pellentesque quis sem purus, vei elementum im. Pellente habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Maecenas consecteur euromod uras. In hac habitasse platea dicturast. Quisque teriodant portitor vestibulum. Ut isculs, set modestie lacinia, ipsum mi adipiscing ligula, vel molits sem risus eu lectus. Nunc elit quam, rutrum ut dignissim sit amer. Qeresta as tem.

ABOUT THE AUTHOR



Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do elusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.









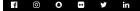


Usare font da file

```
@font-face {
    font-family: 'BallparkWeiner';
    src: url('fonts/ballpark.eot');
    src: url('fonts/ballpark.eot?#iefix') format('embedded-opentype'),
        url('fonts/ballpark.ttf') format('truetype'),
        url('fonts/ballpark.svg#BallparkWeiner') format('svg');
    font-weight: normal;
    font-style: normal;
}
```

I font sono contenuti dentro la cartella **fonts/** nella root del progetto.

Esercizio 2





JANE BLOGLIFE

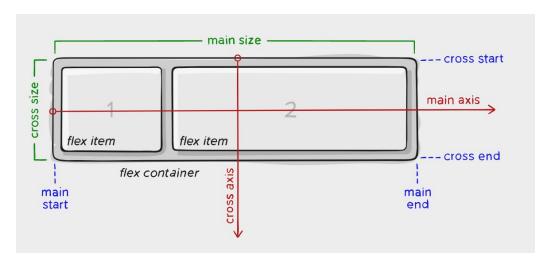
Welcome to the blog of Jane's world





- Flexbox è un modello di layout. Nasce con l'obiettivo di creare un modo efficiente per disporre, allineare e distribuire gli elementi in un container anche quando la loro dimensione è sconosciuta o dinamica.
- L'idea è dare al container la possibilità di modificare altezza e larghezza dei suoi figli.





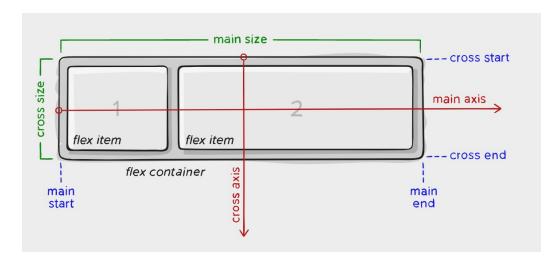
main axis

L'asse principale lungo la quale gli elementi vengono disposti

main-start | main-end

Gli elementi sono piazzati nel container partendo da «main-start» e arrivando fino a «main-end»





main size

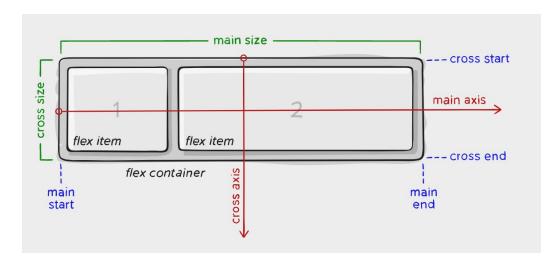
La dimensione (width, height) che corrispondente al main axis

- Height => asse X
- Width => asse Y

cross axis

L'asse perpendicolare alla main axis





cross-start | cross-end

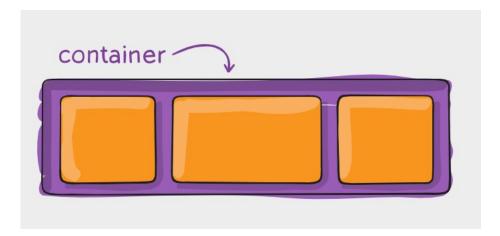
Gli elementi sono piazzati a partire da cross-start e fino a cross-end

cross size

L'asse perpendicolare alla main axis



Container

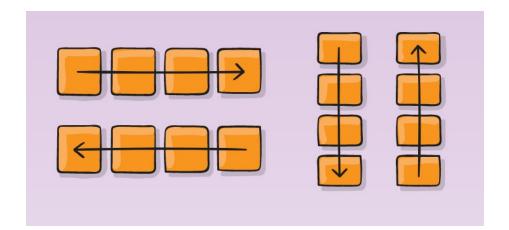


```
.container {
    display: flex; /* or inline-flex */
}
```

Il container è l'elemento che contiene gli elementi.

Ad esso si applica la proprietà **flex** (flex o inline-flex)

Si applica al container.



La proprietà flexdirection definisce il main-axis e di conseguenza tutte le altre proprietà.

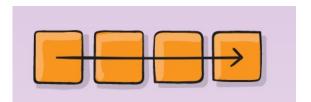
Si applica al container.

```
.container {
    flex-direction: row | row-reverse | column | column-reverse;
}
```



```
.container {
    flex-direction: row;
}
```

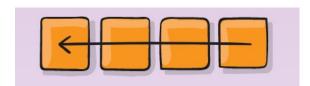
row => Figli disposti da
sinistra a destra (ltr)





```
.container {
    flex-direction: row-reverse;
}
```

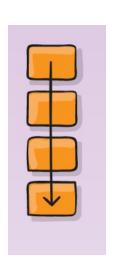
row-reverse => Figli
disposti da destra a
sinistra a destra (rtf)





```
.container {
    flex-direction: column;
}
```

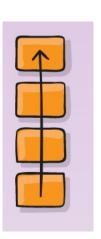
column => Figli disposti dall'altro verso il basso





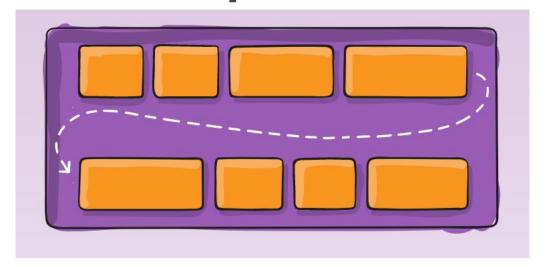
```
.container {
    flex-direction: column-reverse;
}
```

column-reverse => Figli
disposti dal basso verso
l'alto





Flex-wrap



```
.container {
    flex-wrap: nowrap | wrap | wrap-reverse;
}
```

Di default flex prova a disporre gli elementi lungo la riga, modificandone le dimensioni in base alle necessità

E' possibile variare questo comportamento con flex-wrap



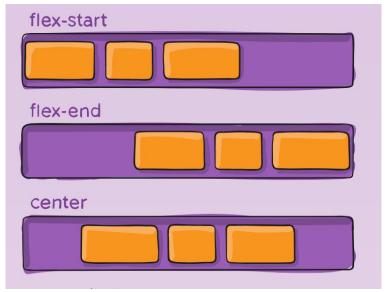
Flex-wrap

```
.container {
    flex-wrap: nowrap | wrap | wrap-reverse;
}
```

- nowrap => tutti gli elementi piazzati in una sola riga
- wrap => disponi gli elementi in diverse linee (dall'alto verso il basso)
- wrap-reverse => disponi gli elementi in diverse linee (dal basso verso l'altro)



Justify-content



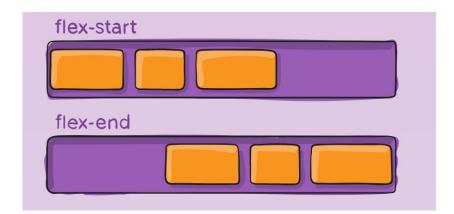
Definisce l'allineamento lungo la **main-axis**

```
.container {
    justify-content: flex-start | flex-end | center | space-between | space-around | space-evenly
}
```



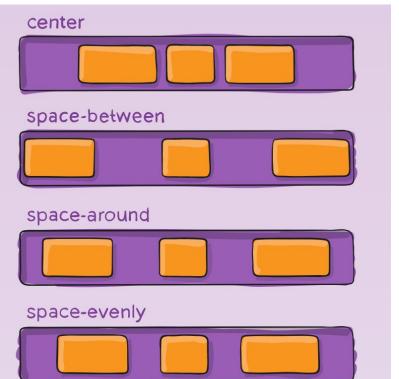
Justify-content

```
.container {
    flex-direction: flex-start | flex-end;
}
```





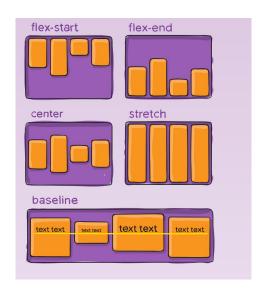
```
.container {
    flex-direction: center |
space-between | space-around |
space-evenly;
}
```







Align-items

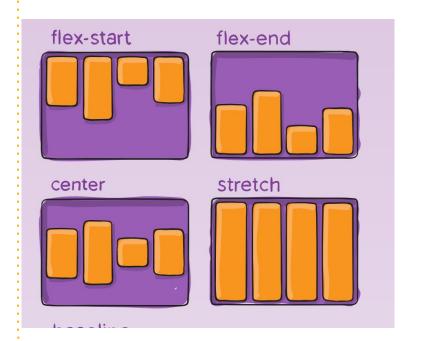


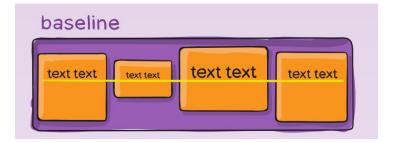
Definisce l'allineamento lungo la **cross-axis**

```
.container {
    align-items: stretch | flex-start | flex-end | center | baseline;
}
```



Align-items





Esercizio 3





Linda Hart 6765 Shady Ln Dr



Roger Medina 2420 Paddock Way



Dean Sullivan 6727 Airplane Ave



Elizabeth Austin 6685 Blossom Hill Rd

Ricreare il layout qui a fianco senza usare bootstrap e/o float. Usare flex e css «custom»



Thanks!

Any questions?

Per qualsiasi domanda contattatemi: alessandro.pasqualini.1105@gmail.com