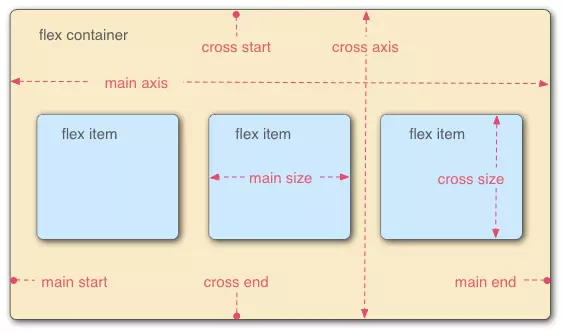
[Flexbox Froggy - Một trò chơi để học CSS flexbox](http://flexboxfroggy.com/#vi)

**https://viblo.asia/p/huong-dan-day-du-ve-css-flexbox-maGK7J9a5j2**

**justified-content**

justified-content dùng để căn chỉnh vị trí của các items so với trục chính(main axis). Có 6 giá trị có thể dùng đối với thuộc tính justified-content:

* flex-start: sẽ đặt item bắt đầu từ main start (và đây cũng là giá trị mặc định)
* flex-end:sẽ đặt item bắt đầu từ main end
* center: sẽ đặt tất cả item ở giữa trục main axis
* space-between: sẽ chia đều khoảng cách thừa và thêm nó vào giữa các item
* space-around: sẽ chia khoảng cách ở đầu và cuối. Khoảng cách ở đầu và cuối sẽ bằng 1 nửa khoảng cách ở giữa 2 item với nhau
* space-evenly: sẽ chia khoảng cách đều khoảng cách giữa các item với item, item và main start, item với main end bằng nhau

**align-items**

Thuộc tính align-items dùng để xác định cách mà các flex item được đặt trong container dọc theo chiều cross axis.

* align-items: stretch: Chiều dài của item sẽ bằng chiều dài của cross axis.
* align-items: flex-start: Item được đặt ở điểm bắt đầu của cross start(trên cùng bên trái), và kích thước item không bị thay đổi.
* align-items: flex-end: Item được đặt ở điểm bắt đầu của cross end(dưới cùng bên trái)
* align-items: center: Item được đặt ở giữa điểm bắt đầu của cross start và điểm bắt đầu của cross end (ở giữa bên trái)
* align-items: baseline: Item sẽ được đặt dữ theo các ký tự thuộc item đó. Mục đích chính là căn chỉnh dữa liệu dòng văn bản của các item.

#### flex-direction

flex-direction dùng để chỉ định hướng hiển thì của các item, việc thay đổi hướng hiển thị flex cũng thể có thể cho phép ta thay đổi vị trí của các flex item.

##### flex-direction: row

flex-direction: row là giá trị mặc định khi sử dụng flexbox, không thực hiện bất kỳ thay đổi nào, chỉ đặt các item từ trái qua phải theo trục chính.

##### flex-direction: row-reverse

##### Giống với tên gọi, flex-direction: row-reverse ngược lại với row, các item sẽ được đặt từ phải qua trái.

##### flex-direction: column

##### Khi chúng ta xét flex-direction: column, lúc này trục chính sẽ đi từ trên xuống dưới vậy nên giờ đây các items sẽ được xếp chồng lên nhau.

##### flex-direction: column-reverse

##### Khi đó các items sẽ được xếp chống lên nhau nhưng theo chiều ngược lại. Hay để ý sẽ thấy ở ví dụ trên (1) sẽ ở trên cùng, nhưng khi sử dụng column-reverse (1) sẽ ở dưới cùng.

Chú ý rằng khi hướng flex là hàng dọc, **justify-content** thay đổi thành chiều dọc và **align-items** to chiều ngang.

Một thuộc tính CSS khác mà bạn có thể áp dụng cho các hạng mục riêng biệt là **align-self**. Thuộc tính CSS này chấp nhận các giá trị tương tự như **align-items** và giá trị của nó đối với một hạng mục riêng biệt.

Ồ không! Những chú ếch đều được dồn vào một hàng duy nhất theo các lá bông súng. Tách rời chúng ra bằng cách sử dụng thuộc tính **flex-wrap**, thuộc tính này chấp nhận các giá trị sau:

* **nowrap**: Mỗi hạng mục được dồn vào một hàng duy nhất.
* **wrap**: Các hạng mục quấn xung quanh để tạo thêm dòng.
* **wrap-reverse**: Các hạng mục quấn xung quanh để tạo thêm dòng ngược lại.

Cả hai thuộc tính **flex-direction** và **flex-wrap** thường được sử dụng với nhau vì thế từ viết tắt **flex-flow** được tạo ra để kết hợp chúng. Từ viết tắt này chấp nhận giá trị của một trong hai thuộc tính ngăn cách bởi một khoảng trắng.

Ví dụ, bạn có thể sử dụng **flex-flow: row wrap** để thiết lập hàng ngang và quấn xung quanh chúng.

Những chú ếch đang lan rộng trên cả ao, nhưng các lá bông súng đang chụm lại ở phía trên. Bạn có thể dùng **align-content** để thiết lập cách các dòng có khoảng cách xa nhau như thế nào. Những thuộc tính này có giá trị như sau:

* **flex-start**: Các dòng được dồn về phía trên cùng của hộp chứa.
* **flex-end**: Các dòng được dồn về phía dưới cùng của hộp chứa.
* **center**: Các dòng được dồn về trung tâm của hộp chứa.
* **space-between**: Các dòng được trình bày với khoảng cách bằng nhau giữa chúng.
* **space-around**: Các dòng sẽ được trình bày với khoảng cách bằng nhau xung quanh chúng.
* **stretch**: Các dòng sẽ được kéo dài để phù hợp với hộp chứa.

Điều này có thể gây nhầm lẫn, nhưng **align-content** xác định khoảng cách giữa các dòng, trong khi **align-items** xác định cách các hạng mục như một toàn thể được xếp trong hộp chứa. Khi chỉ có một dòng, **align-content** không có hiệu lực.