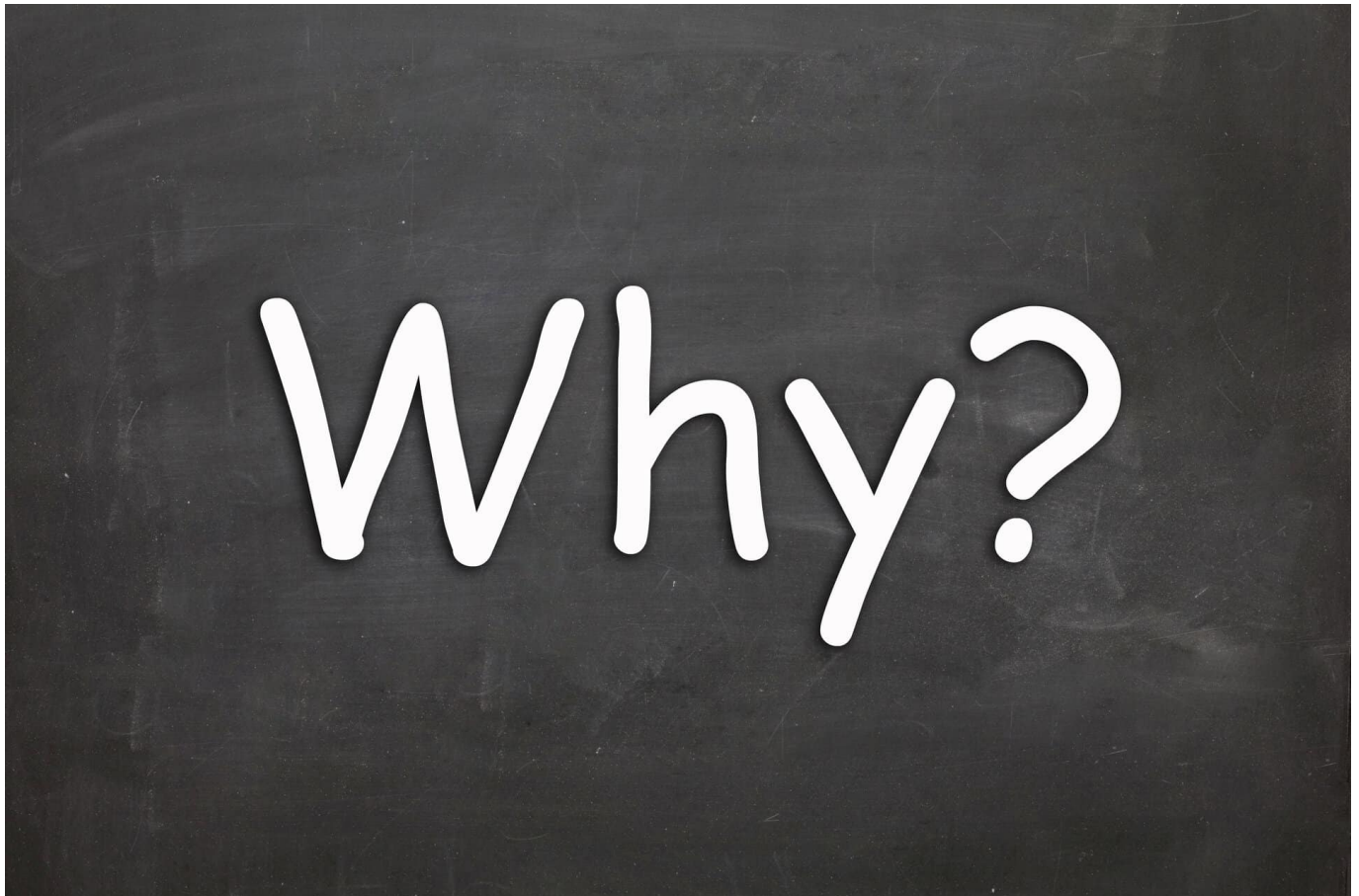


Mục tiêu:

1. Hiểu về Flexbox và cách sử dụng nó để tạo layout linh hoạt.
2. Sử dụng các thuộc tính của Flexbox để tạo layout dễ quản lý và bảo trì.
3. Thực hành cải thiện layout của trang web với Flexbox.

Tổng quan về Flexbox

Khi phát triển những layout phức tạp đòi hỏi tư duy bố cục tổng thể trước khi đi vào chi tiết. Tức là sắp xếp các phần tử con trong một khối hộp cha(container) sao cho chúng tạo ra một layout linh hoạt, dễ quản lý và dễ bảo trì.



Trong những hoàn cảnh như vậy, chúng ta cần sử dụng các công cụ hỗ trợ như Flexbox hoặc CSS Grid (thay vì cố gắng căn chỉnh kích thước từng khối hộp con, chúng ta căn chỉnh vị trí các khối hộp con theo một bố cục được quyết định bởi khối hộp cha(container)).

Flexbox (Flexible Box) là một mô hình bố cục trong CSS, được thiết kế để giúp sắp xếp các phần tử trong một container dễ dàng hơn. Flexbox rất mạnh mẽ trong việc quản lý không gian giữa các phần tử và giúp tạo ra các layout linh hoạt, **đặc biệt khi kích thước của các phần tử trong container không cố định.**

Khái niệm cơ bản

Flexbox dựa trên hai khái niệm chính:

Container (flex container): Phần tử cha có chứa các phần tử cần sắp xếp.

Items (flex items): Các phần tử con bên trong container, sẽ được sắp xếp theo Flexbox.

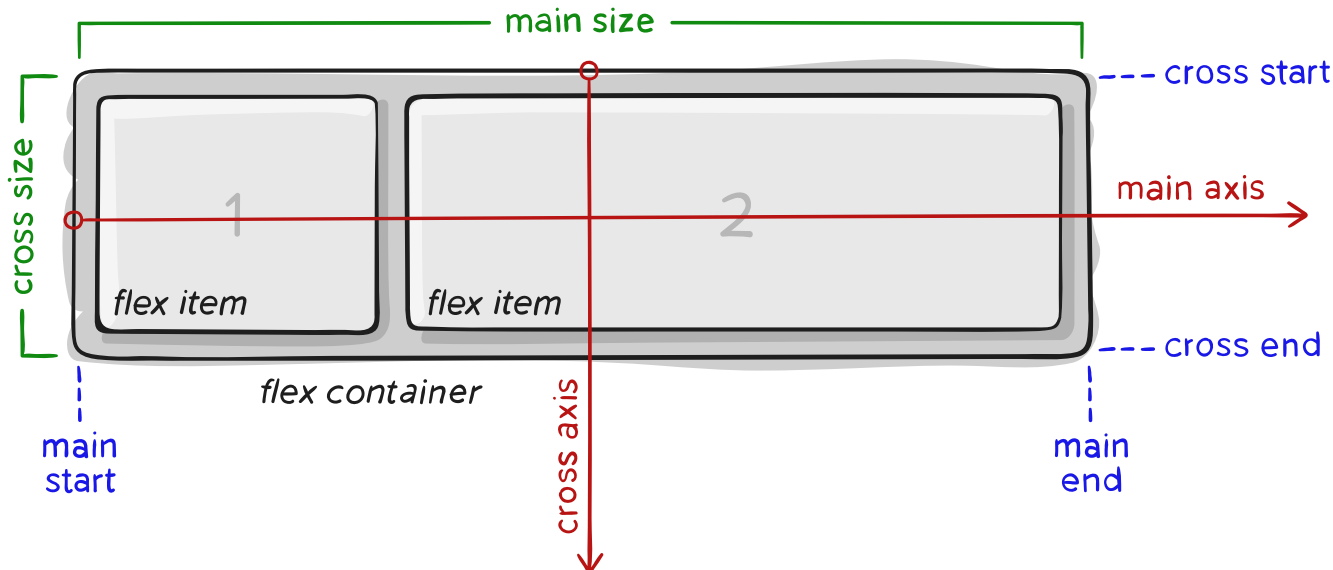
Khi một phần tử được thiết lập thành `display: flex`, nó trở thành một flex container. Tất cả các phần tử con trực tiếp của nó trở thành flex-item.

Lưu ý: Flexbox hoạt động bằng cách sắp xếp các phần tử con bên trong một container theo một hướng cụ thể (dọc hoặc ngang), và điều chỉnh không gian giữa các phần tử để tạo ra layout linh hoạt.

Đây là tư duy xây dựng layout mang tính tổng quan, không cố gắng điều chỉnh chi tiết từng phần tử như trước đây chúng ta đã từng làm.

```
<style>
  .container {
    display: flex;
  }
</style>
<div class="container">
  <div class="item">Item 1</div>
  <div class="item">Item 2</div>
  <div class="item">Item 3</div>
</div>
```

Cùng quan sát hình ảnh dưới đây để hiểu rõ hơn về container và items trong Flexbox:



Trong đó:

- **Main axis:** trục chính, là hướng mà các phần tử con được sắp xếp, hãy cẩn thận vì nó không phải lúc nào cũng nằm ngang. Nó có thể là chiều dọc hoặc chiều ngang tùy thuộc vào thuộc tính `flex-direction`. Nhưng mặc định thì nó là chiều ngang.
- **Cross axis:** trục phụ, là hướng vuông góc với trục chính.