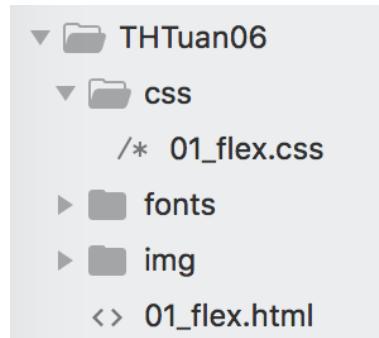


FLEX LAYOUT – POSITION

I. Chuẩn bị

Tạo cấu trúc thư mục như sau



II. Flex layout

II.1. Định dạng trên dòng với flex

Bước 1. Tạo trang 01_flex.html có nội dung như sau:

```
.container>.item[id=item$]{$}*8
```

Bước 2. Định dạng như sau:

```
1 .container{
2   display:flex;
3 }
4 .item{
5   background-color: #453;
6   color:#fff;
7 }
```

12345678

Nhận xét:

- Kết quả các item được hiển thị chung trên một dòng, width dãn theo nội dung
- Điều chỉnh width của .item thành 10% (tức tổng là 80%<100%), xem hiển thị kết quả.
- Điều chỉnh width của .item thành 20% (tức tổng là 160%>100%), xem hiển thị kết quả tự chia đều trong 100% đảm bảo tất cả trên 1 dòng → không bị nhảy

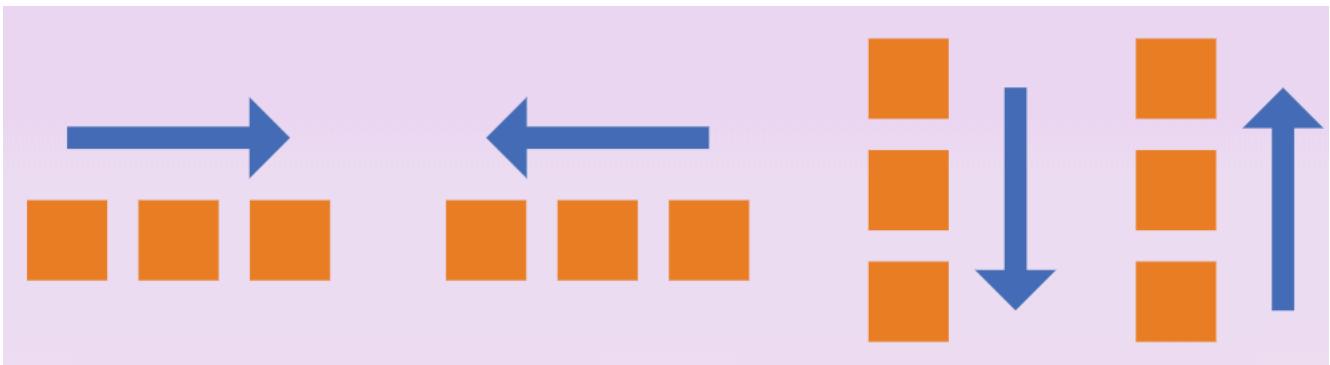
Bước 3. Điều chỉnh margin các thành phần:

```
.item{  
    background-color: #453;  
    color:#fff;  
    width:20%;  
    margin:2%;  
}
```

Bước 4. Sau đó thử tăng margin lên từ từ → không bị nhảy

II.2. Thứ tự các item

flex-direction: row | row-reverse | column | column-reverse;



II.3. Tự động xuống dòng khi đủ 100% width

flex-wrap: nowrap | wrap | wrap-reverse;

```
.container{  
    display:flex;  
    flex-wrap:wrap;  
}  
.item{  
    background-color: #453;  
    color:#fff;  
    width:20%;  
    margin:2%;  
}
```

Shorthand:

flex-flow: <'flex-direction'> || <'flex-wrap'>

II.4. Canh lề theo chiều ngang

justify-content: flex-start | flex-end | center | space-between | space-around | space-evenly;

flex-start (default): Các item hiển thị từ đầu dòng

flex-end: Các item hiển thị từ cuối dòng

center: Các item hiển thị dồn vào giữa dòng

Trường hợp không wrap

```
.container{  
    display:flex;  
    justify-content: center;  
}  
.item{  
    background-color: #453;  
    color:#fff;  
    width:10%;  
}
```

Trường hợp có wrap

```
.container{  
    display:flex;  
    flex-wrap:wrap;  
    justify-content: center;  
}  
.item{  
    background-color: #453;  
    color:#fff;  
    width:20%;  
}
```

space-between: item hiển thị ở góc đầu dòng, item hiển thị ở góc cuối dòng và có khoảng cách giữa các item

Trường hợp không wrap

```
.container{  
    display:flex;  
    /*flex-wrap:wrap;*/  
    justify-content: space-between;  
}  
.item{  
    background-color: #453;  
    color:#fff;  
    width:10%;  
}
```

Trường hợp wrap

```
.container{  
    display:flex;  
    flex-wrap:wrap;  
    justify-content: space-between;  
}  
.item{  
    background-color: #453;  
    color:#fff;  
    width:20%;  
}
```

space-around: có khoảng cách xung quanh các item.

Trường hợp không có wrap

```
.container{  
    display:flex;  
    /*flex-wrap:wrap;*/  
    justify-content: space-around;  
}  
.item{  
    background-color: #453;  
    color:#fff;  
    width:10%;  
}
```

Trường hợp có wrap:

```
.container{  
    display:flex;  
    flex-wrap:wrap;  
    justify-content: space-around;  
}  
.item{  
    background-color: #453;  
    color:#fff;  
    width:20%;  
}
```

space-evenly: có khoảng cách giữa 2 item bất kỳ.

Trường hợp không có wrap

```
.container{  
    display:flex;  
    /*flex-wrap:wrap;*/  
    justify-content: space-evenly;  
}  
.item{  
    background-color: #453;  
    color:#fff;  
    width:10%;  
}
```

Trường hợp có wrap

```
.container{  
    display:flex;  
    flex-wrap:wrap;  
    justify-content: space-evenly;  
}  
.item{  
    background-color: #453;  
    color:#fff;  
    width:20%;  
}
```

1.1. Canh lề theo các item

Điều chỉnh độ cao các item khác nhau:

```
#item1{ height:10vh; } #item4{ height:40vh; }
#item2{ height:20vh; } #item5{ height:50vh; } #item7{ height:70vh; }
#item3{ height:30vh; } #item6{ height:60vh; } #item8{ height:80vh; }
```

flex-start: Theo item đầu tiên

flex-end: Theo item cuối cùng

center: canh giữa theo độ cao item dài nhất

baseline: theo thẻ cha

stretch (default): dãn ra theo thẻ có độ dài dài nhất. Lưu ý để demo thuộc tính này không được chỉnh độ cao theo id mà độ cao được dãn ra do nội dung các item.

Lưu ý: chỉ chạy với nhiều dòng

flex-start: bắt đầu từ dòng đầu tiên

flex-end: bắt đầu từ dòng cuối

center: canh giữa theo độ cao thẻ cha

space-between: có khoảng cách các dòng lẻ, bắt đầu dòng đầu

space-around: các dòng có cách khoảng

stretch (default): dãn độ cao các dòng để vừa độ cao thẻ cha

II.5. Thứ tự các item

order: <integer>;

Các item sẽ được sắp xếp theo thứ tự tăng dần

II.6. Độ rộng các item

flex-grow: <number>;

Lưu ý: không được giá trị âm

Độ rộng gấp đôi những item khác thì để giá trị 2 (các item mặc định lúc đầu bằng nhau)

flex-shrink: <number>;

Lưu ý: không được giá trị âm

Độ rộng 1/2 những item khác thì để giá trị 2 (các item mặc định lúc đầu bằng nhau)

flex-basis: <length> | auto;

Chỉnh độ rộng

Shorthand

flex: none | ['<'flex-grow'> '<'flex-shrink'>? || '<'flex-basis'>']

II.7. Canh lề mỗi item

align-self: auto | flex-start | flex-end | center | baseline | stretch;

III. Position – Popup – Menu nhiều cấp

III.1. Thuộc tính position

Thể hiện vị trí của thành phần HTML

static: mặc định lần lượt theo dòng của trang. Các thuộc tính top, left, bottom, right không hoạt động.

relative: tương tự thuộc tính static nhưng các thuộc tính vị trí hoạt động và có toạ độ so với vị trí gốc ban đầu.

absolute: không theo dòng của trang, toạ độ vị trí được tính theo thẻ chứa bên ngoài có position là relative (mặc định luôn có thẻ body có position: relative bên ngoài)

fixed: không theo dòng của trang, toạ độ vị trí được tính theo khung trình duyệt, không bị ảnh hưởng bởi tác dụng thanh cuộn

sticky: mặc định ban đầu có tính chất như relative nhưng khi đạt đến vị trí (do cuộn) thì sẽ trở thành fixed.

Bước 1. Tạo thêm 2 tập tin 02_position.html, 02_position.css lần lượt trong thư mục THTuan06 và thư mục css

Bước 2. Tạo cấu trúc html như sau

header+nav.nav_sticky+(section.content>.item_absolute)+footer+.top_ico

Bước 3. Xây dựng giao diện ban đầu

```

header{
    height:20vh;
    background-color: yellow;
}
.nav_sticky{
    height: 10vh;
    background-color: green;
}
.content{
    background-color: pink;
    padding-top:10px;
}
.item_absolute{
    height: 10vh;
    background-color: black;
}
footer{
    height: 20vh;
    background-color: aqua;
}
.top_ico{
    width:50px;
    height:50px;
    background-color: blue;
}

```



Bước 4. Điều chỉnh css cho .item_absolute có thuộc tính position:absolute, quan sát trên trình duyệt, sau đó điều chỉnh width:5vh, sau đó thêm top:0px cho .item_absolute, quan sát trên trình duyệt

Nhận xét:

- .item_absolute ẩn đi (do width trở về 0), độ cao của thẻ .content bị giảm xuống không còn dãn theo thẻ .item_absolute (mặc dù width của .item_absolute vẫn là 10vh → Giống

nhiều float, position:absolute sẽ làm độ dài thẻ con không giống thẻ cha và width mặc định trở thành 0.

```
.item_absolute{  
    height: 10vh;  
    background-color: black;  
    position: absolute;  
}
```

- Khi điều chỉnh width 5vh cho .item_absolute ta thấy nằm ngay dưới thẻ .content (top mặc định được tính toán) nhưng sau đó điều chỉnh top:0px thì .item_absolute sẽ có toạ độ theo body nên nằm sát phần bên trên (bởi vì body luôn có position là relative và nó chứa thẻ .item_absolute bên trong (không phải là thẻ chứa trực tiếp của .item_absolute)) → toạ độ thành phần có position:absolute sẽ theo thành phần có position:relative chứa nó (không cần phải chứa trực tiếp). Lưu ý rằng .item_absolute vẫn chịu tác động của scroll, nếu độ dài các thẻ dài ra, khi scroll xuống thì nó vẫn có thể bị che khuất, cần nhớ điều này để so sánh với giá trị fixed của thuộc tính position.

```
.item_absolute{  
    height: 10vh;  
    width:5vh;  
    background-color: black;  
    position: absolute;  
}
```

```
.item_absolute{  
    height: 10vh;  
    width:5vh;  
    top:0px;  
    background-color: black;  
    position: absolute;  
}
```

Bước 5. Điều chỉnh css cho.content với position:relative, height:100vh

Nhận xét:

- .item_absolute hiển thị toạ độ theo .content và không tính padding-top → toạ độ của thành phần absolute sẽ theo thành phần relative gần nhất.

Bước 6. Điều chỉnh css cho .top_ico cho phép luôn luôn hiện góc dưới trình duyệt và thêm nội dung để nhấn vào sẽ lên trên đầu trang web

```
.top_ico{  
    width:50px;  
    height:50px;  
    background-color: blue;  
    position:fixed;  
    bottom:0;  
    right:0;  
}
```

Tạo liên kết đến header, tạo id cho thẻ header, thẻ a liên kết nằm trong .top_ico

```
<header id="header"></header>
<nav class="nav_sticky"></nav>
<section class="content">
    <div class="item_absolute"></div>
</section>
<footer></footer>
<div class="top_ico"><a href="#header">Top</a></div>
```

Bước 7. Điều chỉnh menu để khi menu được vào phần trên sẽ dùng lại và nằm trên top (không bị che khuất)

```
.nav_sticky{
    height: 10vh;
    background-color: green;
    top:0;
    position: sticky;
    z-index: 999;
}
```

III.2. Popup

Những cửa sổ hiện lên trong lúc người dùng sử dụng web và có thể che các phần còn lại (nằm trên), thường do khi người dùng nhấp vào 1 liên kết hay một thành phần nào đó. Ví dụ sau đây chỉ thể hiện trạng thái khi 1 popup hiển thị lên, việc xử lý canh chỉnh và xử lý nhấp nút người dùng sẽ bỏ qua.

Bước 1. Tạo giao diện có cấu trúc như sau

```
header+section.content+footer+.popup
```

Bước 2. Chính height cho các thành phần

```
header{
    height: 20vh;
    background-color: yellow;
}
.content{
    height: 100vh;
    background-color: aqua;
}
.footer{
    height: 30vh;
    background-color: pink;
}
.popup{
    height: 30vh;
    width: 30vh;
    background-color: black;
}
```

Bước 3. Chính toạ độ cho popup hiện đè lên các thành phần khác

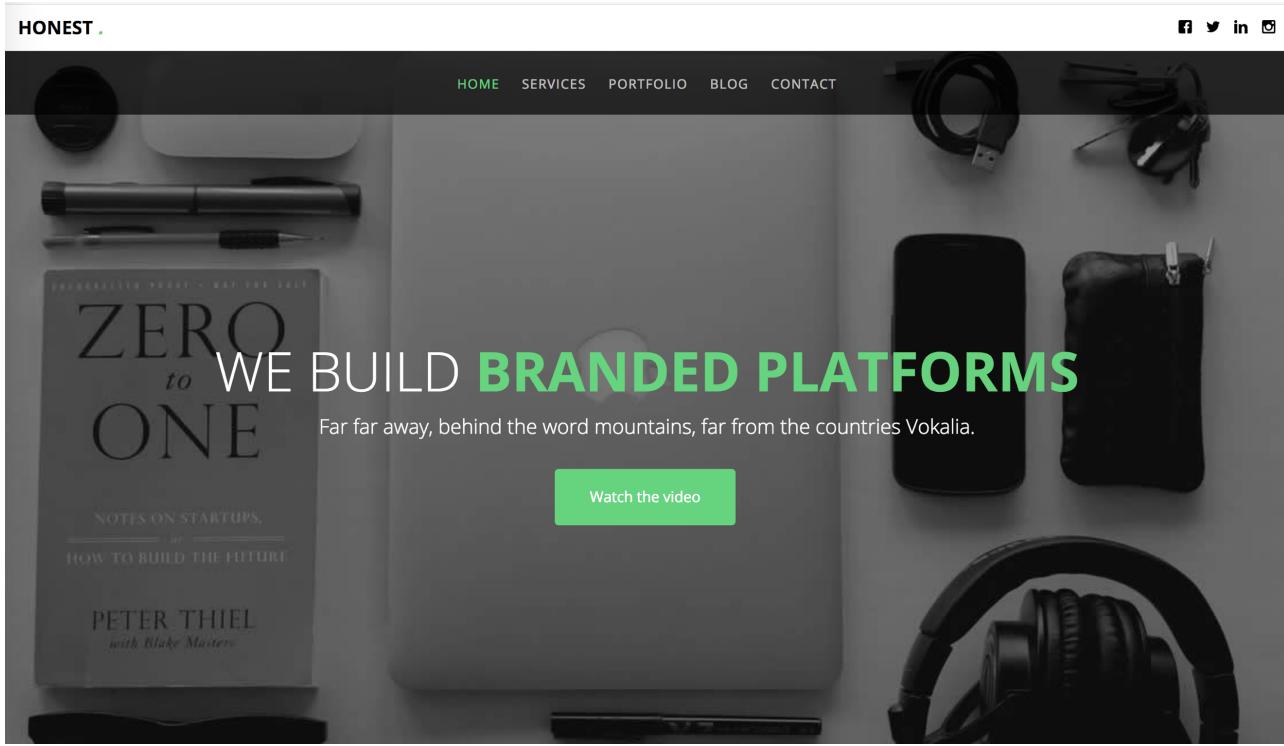
```
.popup{  
    height: 30vh;  
    width:30vw;  
    background-color: black;  
    position:fixed;  
    top:35vh;  
    left:35vw;  
}
```

Bước 4. Tạo lớp che các phần của trang web

```
.popup{  
    height: 30vh;  
    width:30vw;  
    background-color: black;  
    position:fixed;  
    top:35vh;  
    left:35vw;  
    z-index: 2;  
}  
.wrapper{  
    height: 100vh;  
    width:100vw;  
    background-color: RGBA(0,0,0,0.5);  
    position:fixed;  
    top:0px;  
    z-index: 1;  
}  
  
<header></header>  
<section class="content"></section>  
<footer></footer>  
<div class="popup"></div>  
<div class="wrapper"></div>
```

Lưu ý thuộc tính z-index thể hiện chiều sâu sẽ làm cho .wrapper nằm bên dưới của .popup và tạo thành màn che các thành phần bên dưới. Trong thực tế có thể tương tác dùng js để canh chỉnh cho kích thước, không cho phép cuộn các phần bên dưới.

IV. Thiết kế giao diện trang honest



The screenshot shows the homepage of the honest website. At the top, there's a dark header bar with the word "HONEST" in white. Below it is a large, dark background image featuring a book titled "ZERO to ONE", a pair of headphones, and a smartphone. Overlaid on this image is the text "WE BUILD BRANDED PLATFORMS" in large, bold, green letters. Below this, a smaller text reads "Far far away, behind the word mountains, far from the countries Vokalia." A green button labeled "Watch the video" is positioned in the center. At the very bottom of the page, there's a white footer section with the title "Creative Minds" and some placeholder Latin text.

Creative Minds

Quaerat quisquam quasi minima at molestias porro accusantium qui atque doloremque ab, odio tempora obcaecati cumque veritatis quo, assumenda nihil dolor optio enim fugit accusamus laborum. Autem neque dignissimos, corrupti, fugiat labore nisi illo, delectus, alias natus optio distinctio culpa!

✓ Quaerat quisquam
quasi

✓ Accusantium qui atque
doloremque

✓ Expedita atque quod

✓ Quaerat quisquam
quasi

✓ Accusantium qui atque
doloremque

✓ Expedita atque quod



The inset image shows a creative workspace setup. It includes a world map, a camera, a pencil, a notebook, and a pair of glasses. The overall aesthetic is professional and focused on creativity.



Never Stop Learning

Expertise & Beliefs

Dignissimos asperiores vitae velit veniam totam fuga molestias accusamus alias autem provident. Odit ab aliquam dolor eius.

 Lorem ipsum dolor sit amet, consectetur adipisicing elit. Repudiandae accusamus, fugiat ut! Ducimus accusamus, quos. Esse ea iste, expedita atque quod quo maxime, saepe! Architecto unde aliquid atque excepturi dolores dolore fugit rem recusandae amet a, molestiae et! Dolore, laborum?

 Quaerat quisquam quasi minima at molestias porro accusantium qui atque doloremque ab, odio tempora obcaecati cumque veritatis quo, assumenda nihil dolor optio enim fugit accusamus laborum. Autem neque dignissimos, corrupti, fugiat labore nisi illo, delectus, alias natus optio distinctio culpa!

✓ Quaerat quisquam
quasi

✓ Accusantium qui atque

✓ Expedita atque quod

✓ Quaerat quisquam
quasi

✓ Accusantium qui atque

✓ Expedita atque quod





Quality Over Quantity

Services

Dignissimos asperiores vitae velit veniam totam fuga molestias accusamus
alias autem provident. Odit ab aliquam dolor eius.



Data Analytics

Far far away, behind the word
mountains, far from the countries
Vokalia..

[Learn more](#)



Web Development

Far far away, behind the word
mountains, far from the countries
Vokalia..

[Learn more](#)



Branding

Far far away, behind the word
mountains, far from the countries
Vokalia..

[Learn more](#)





eCommerce Development

Far far away, behind the word mountains, far from the countries Vokalia..

[Learn more](#)



Design & UX

Far far away, behind the word mountains, far from the countries Vokalia..

[Learn more](#)



Strategy

Far far away, behind the word mountains, far from the countries Vokalia..

[Learn more](#)

How We Work



How We Work

Dignissimos asperiores vitae velit veniam totam fuga molestias accusamus alias autem provident. Odit ab aliquam dolor eius.

1

Design

Dignissimos asperiores vitae velit veniam totam fuga molestias accusamus alias autem provident.

2

Develop

Dignissimos asperiores vitae velit veniam totam fuga molestias accusamus alias autem provident.

3

Launch

Dignissimos asperiores vitae velit veniam totam fuga molestias accusamus alias autem provident.





We have high quality services that you
will surely love!

Facts

Dignissimos asperiores vitae velit veniam totam fuga molestias accusamus
alias autem provident. Odit ab aliquam dolor eius.

196

CLIENTS

97

PROJECTS

12402

COFFEE

12202

LINE OF CODES

Subscribe

Be the first to know about the new templates.

Your Email

SUBSCRIBE

ABOUT US

Lorem ipsum dolor sit amet consectetur adipisciing elit. Tempore eos molestias quod sint ipsum possimus temporibus officia iste perspiciatis consequatur in fugiat.

[Learn more...](#)

SERVICES

Data Analytics
Web Development
Branding & Identity
eCommerce Development
Design & UX
Strategt

GET IN TOUCH

+1 234 567 890
info@yoursite.com
[Live Chat](#)

© All Rights Reserved.
Designed by [GetTemplates.co](#) Demo Images: Unsplash

