



匠人学院
jiangren.com.au

HTML & CSS

www.jiangren.com.au

Objectives



- **Flexbox** – a power tool to arrange elements 1-dimensional
- **Responsive Web Design** – cross-screen compatible design
- **Sass** – the most popular CSS pre-processor
- **Bootstrap** – the most popular UI component
- **UI Design Principle**

Flexbox



WHAT IS FLEXBOX?

FLEXBOX

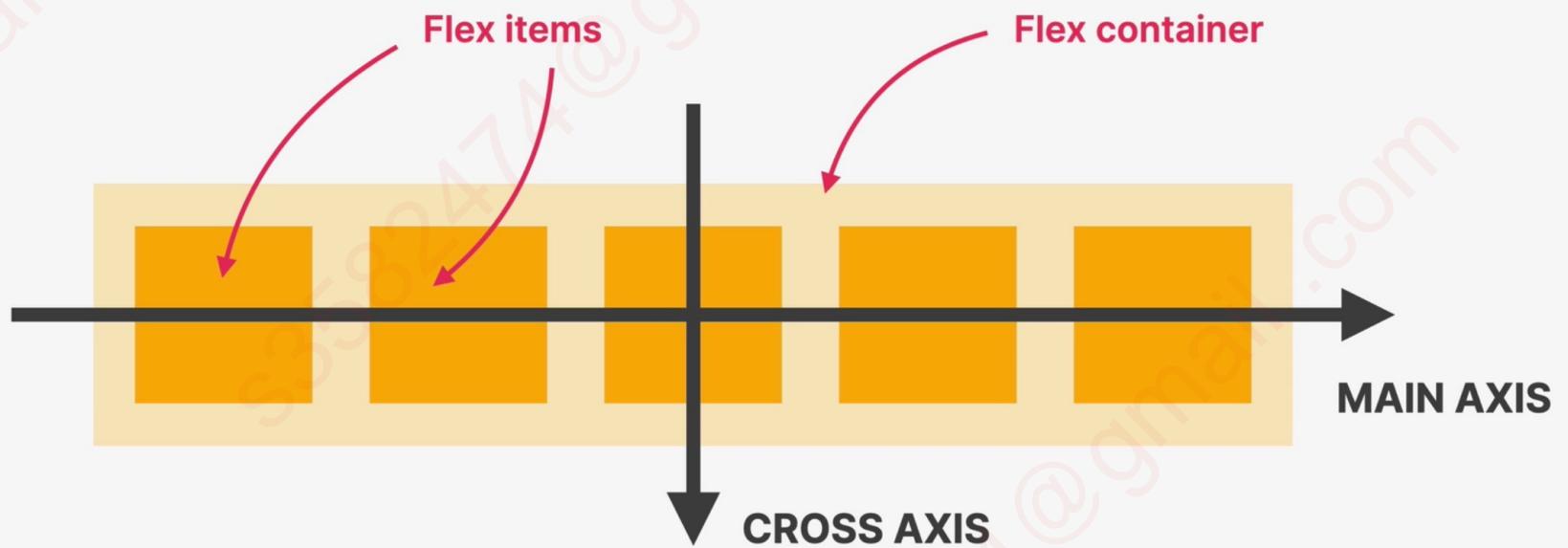


- 👉 Flexbox is a set of related **CSS properties** for **building 1-dimensional layouts**
- 👉 The main idea behind flexbox is that empty space inside a container element can be **automatically divided** by its child elements
- 👉 Flexbox makes it easy to automatically **align items to one another** inside a parent container, both horizontally and vertically
- 👉 Flexbox solves common problems such as **vertical centering** and creating **equal-height columns**
- 👉 Flexbox is perfect for **replacing floats**, allowing us to write fewer and cleaner HTML and CSS code

Flexbox

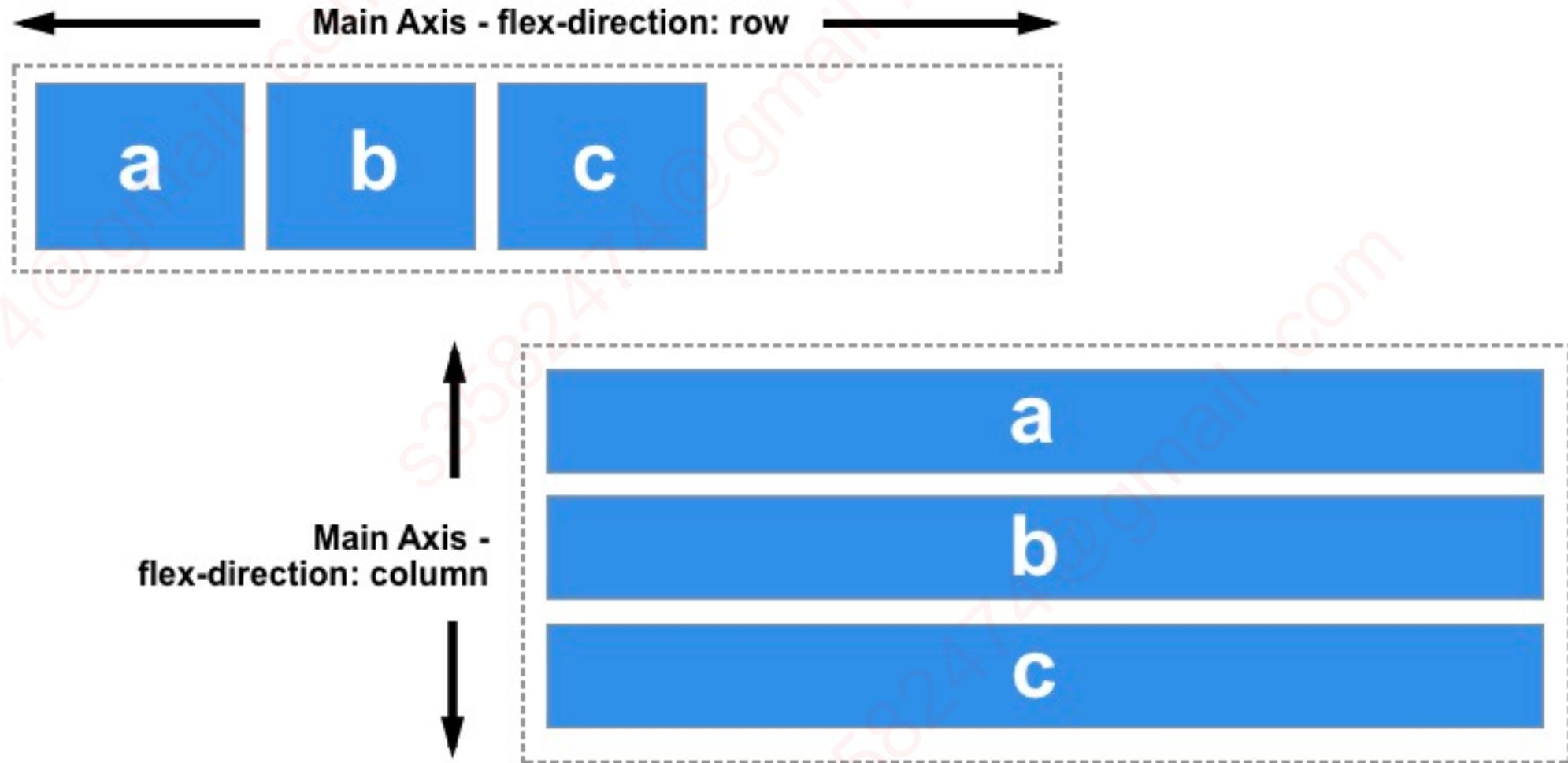


FLEXBOX TERMINOLOGY



`display: flex`

Flexbox - The main axis



Flexbox



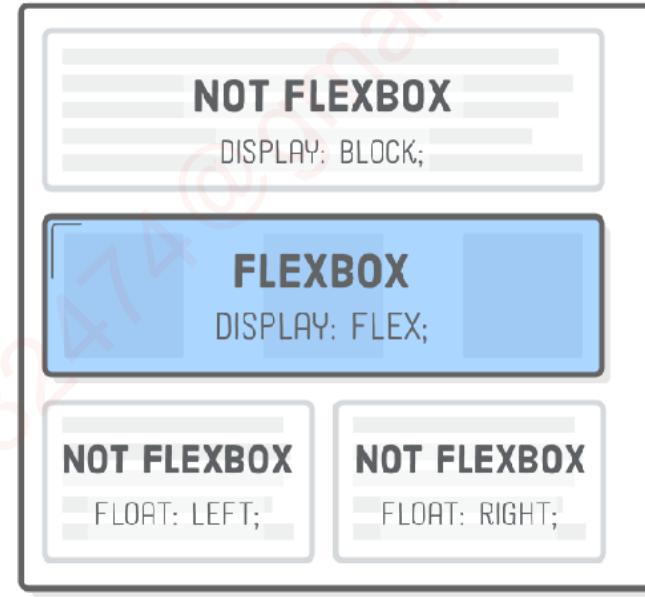
“FLEX CONTAINER”



“FLEX ITEMS”

<https://cssreference.io/flexbox/>

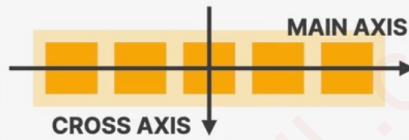
Flex containers



Flexbox



FLEX CONTAINER



FLEX ITEMS

1 `gap: 0 | <length>`

👉 To create **space between items**, without using margin

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

👉 To align items along **main axis (horizontally)**, by default

3 `align-items: stretch | flex-start | flex-end | center | baseline`

👉 To align items along **cross axis (vertically)**, by default

4 `flex-direction: row | row-reverse | column | column-reverse`

👉 To define which is the **main axis**

5 `flex-wrap: nowrap | wrap | wrap-reverse`

👉 To allow items to **wrap into a new line** if they are too large

6 `align-content: stretch | flex-start | flex-end | center | space-between | space-around`

👉 Only applies when there are **multiple lines** (`flex-wrap: wrap`)

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

👉 To **overwrite align-items** for individual flex items

2 `flex-grow: 0 | <integer>`

👉 To allow an element to **grow** (0 means no, 1+ means yes)

3 `flex-shrink: 1 | <integer>`

👉 To allow an element to **shrink** (0 means no, 1+ means yes)

4 `flex-basis: auto | <length>`

👉 To define an item's width, **instead of the width property**

5 `flex: 0 1 auto | <int> <int> <len>`

👉 **Recommended** shorthand for `flex-grow`, `-shrink`, `-basis`.

6 `order: 0 | <integer>`

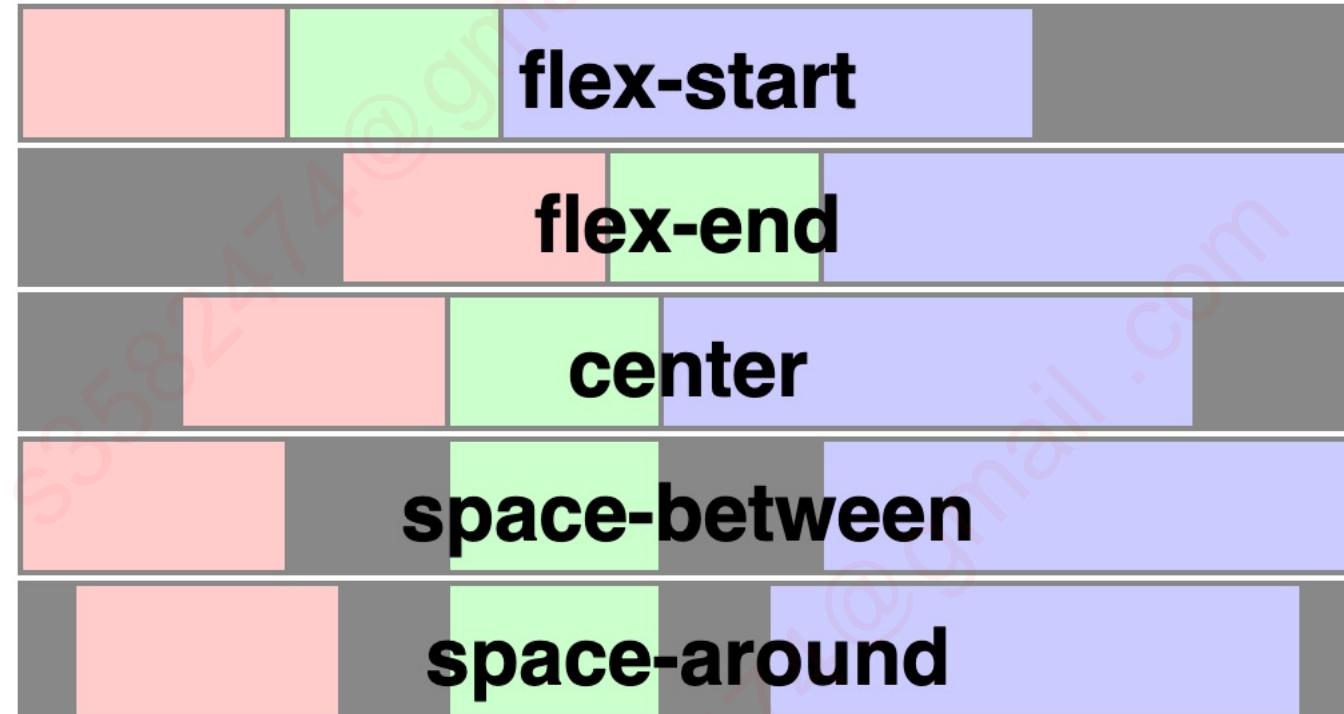
👉 Controls order of items. -1 makes item **first**, 1 makes it **last**

Aligning a flex item



justify-content :

- center
- flex-start
- flex-end
- space-around
- space-between

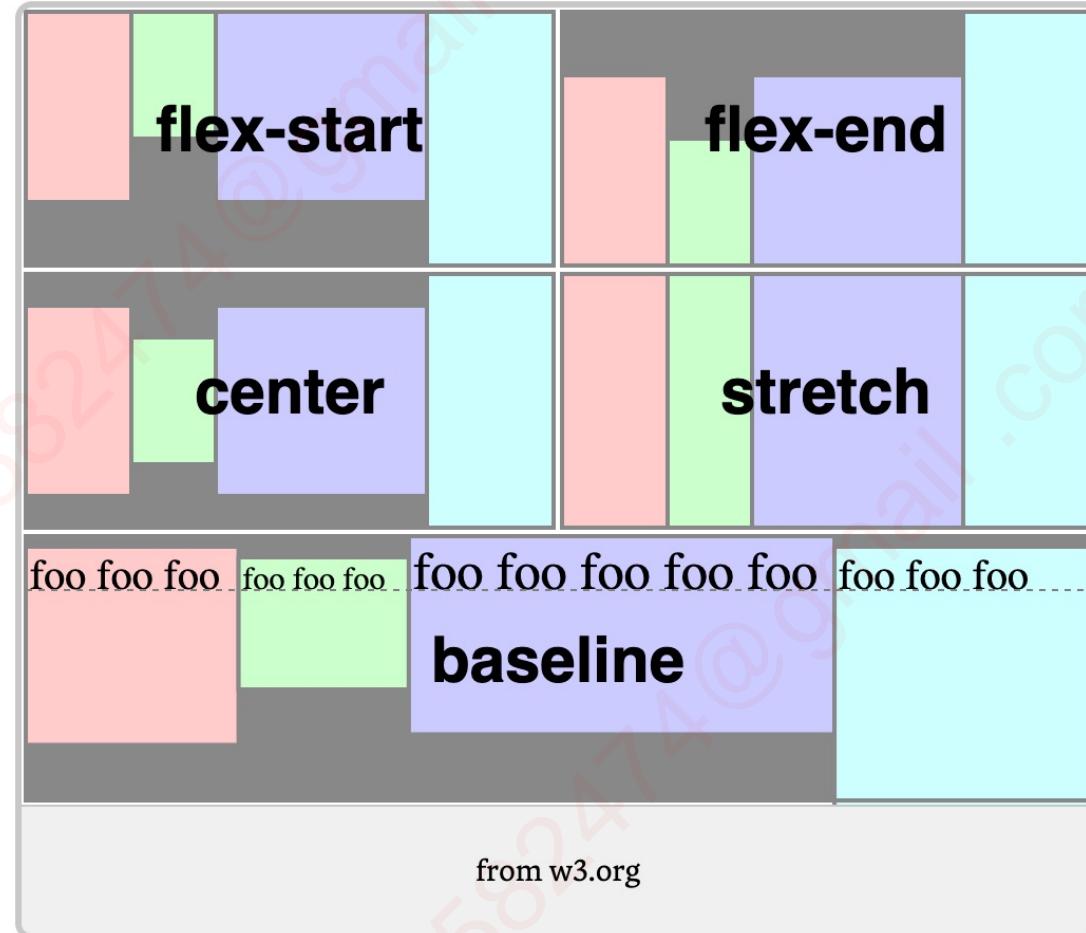


cross-axis (vertical) alignment



align-items:

- center
- flex-start (top)
- flex-end (bottom)
- stretch
- baseline

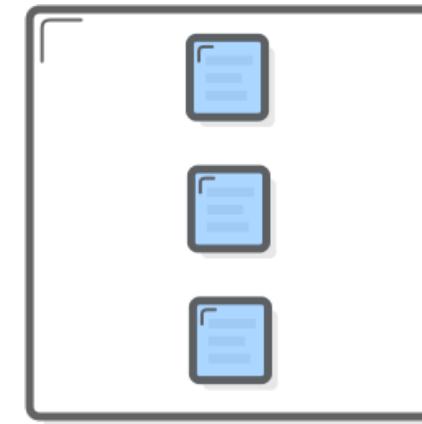


flex container direction



ROW

FLEX-DIRECTION: ROW;



COLUMN

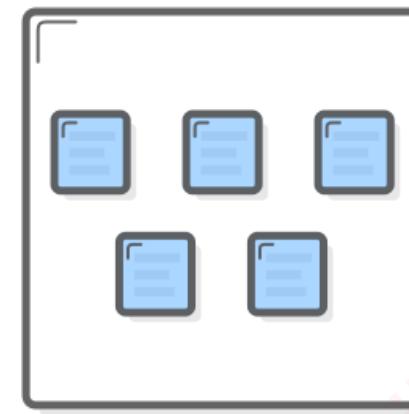
FLEX-DIRECTION: COLUMN;

wrapping flex items



NO WRAPPING

FLEX-WRAP: NOWRAP;



WITH WRAPPING

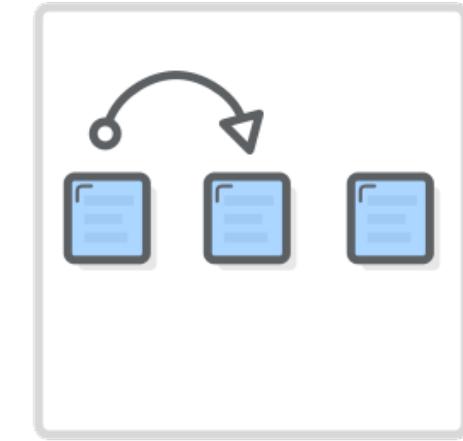
FLEX-WRAP: WRAP;

Flex item order



FLEX-DIRECTION

(WHOLE CONTAINER)



ORDER

(INDIVIDUAL ITEMS)

<https://cssreference.io/flexbox/>

Lab



<https://cssreference.io/flexbox/>



Flexbox - Practice



Practice :
Main width: 900px

Responsive Web Design



The image displays four side-by-side screenshots of the 'think with Google' website, illustrating its responsive design across various devices. The top navigation bar remains consistent, while the main content area adapts to the screen width. Each screenshot shows a different section of the site, such as 'YouTube Collaborations', 'New Research on Micro-Moments and Profitable ROI', and 'Inside Google Marketing: Banner Ads Can Be Creative and Effective'. The layout includes a mix of text, images, and call-to-action buttons.

Demo – Ominifood

Media query



HOW MEDIA QUERIES WORK (WITH MAX-WIDTH)

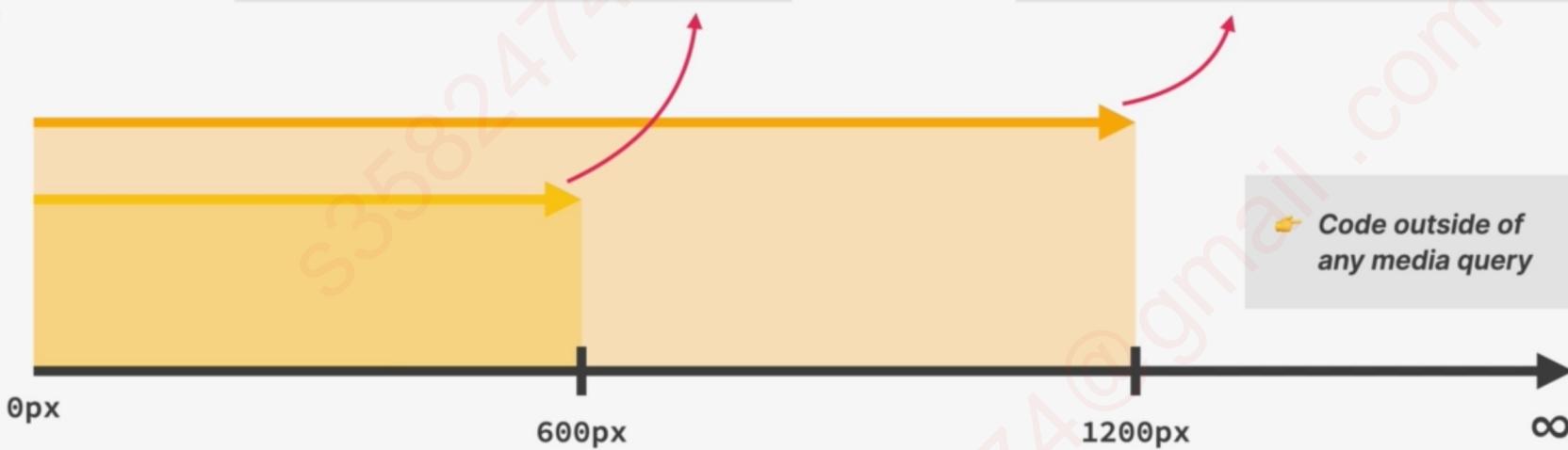
Maximum width at which media query still applies

`@media (max-width: 600px)`

👉 “Is width \leq 600px ?”

`@media (max-width: 1200px)`

👉 “Is width \leq 1200px ?”



Which one applies?



HOW MEDIA QUERIES WORK (WITH MAX-WIDTH)

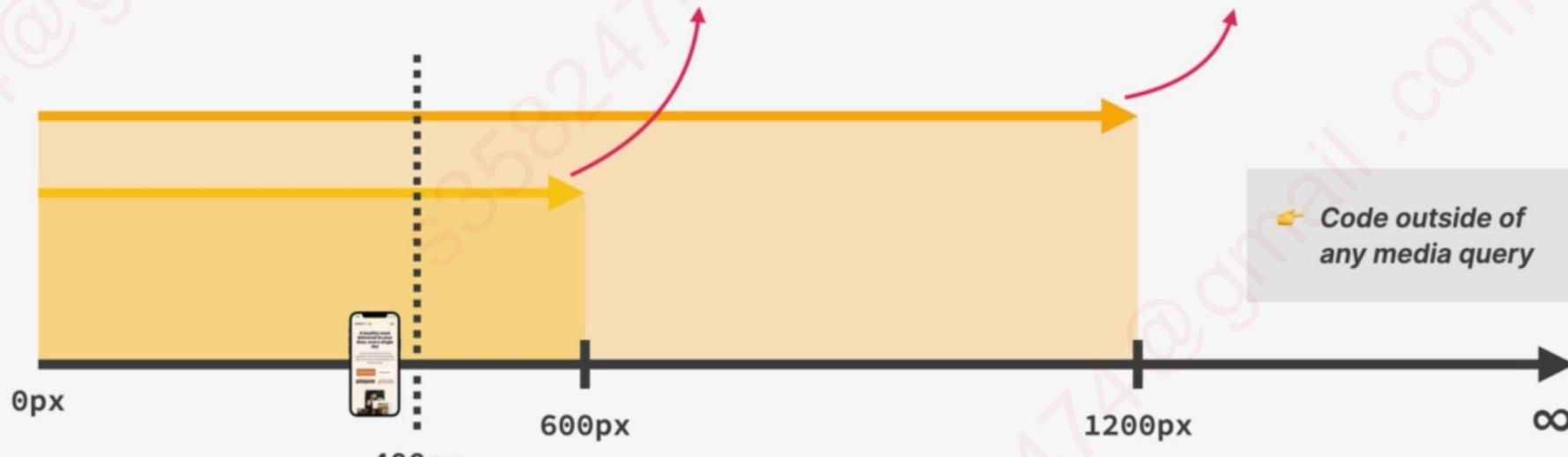
Maximum width at which media query still applies

`@media (max-width: 600px)`

👉 "Is width <= 600px ?"

`@media (max-width: 1200px)`

👉 "Is width <= 1200px ?"

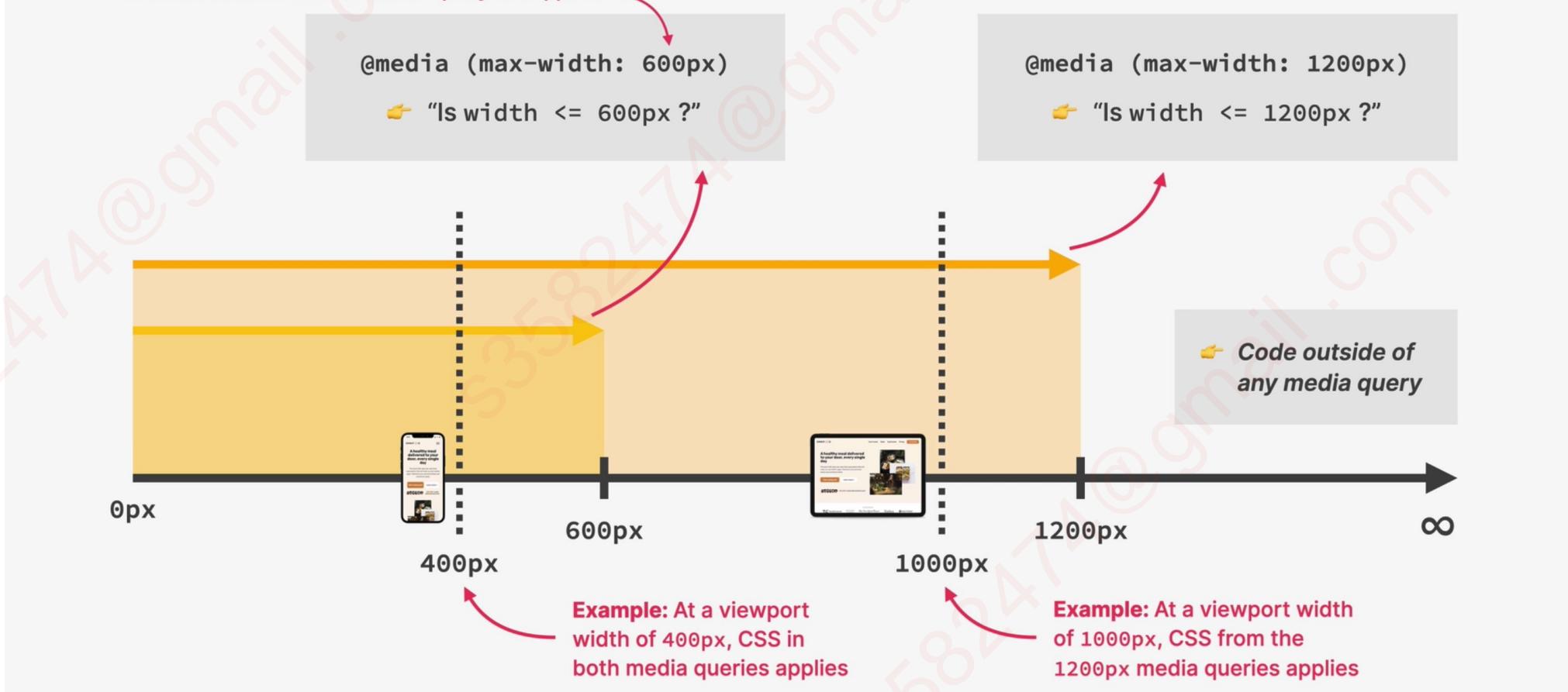


Example: At a viewport width of 400px, CSS in both media queries applies

Which one applies?



HOW MEDIA QUERIES WORK (WITH MAX-WIDTH)



Breakpoints



STRATEGIES FOR SELECTING BREAKPOINTS

BAD

Based on popular devices



GOOD

Based on screen width ranges



PERFECT

When design breaks down



Responsive Web Design



Set the viewport

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

- Use the meta viewport tag to control the width and scaling of the browser's viewport.
- Include width=device-width to match the screen's width in device-independent pixels.
- Include initial-scale=1 to establish a 1:1 relationship between CSS pixels and device-independent pixels.
- Ensure your page is accessible by not disabling user scaling.

Responsive Web Design



Flexible Images

- img {
- max-width:100%;
- }

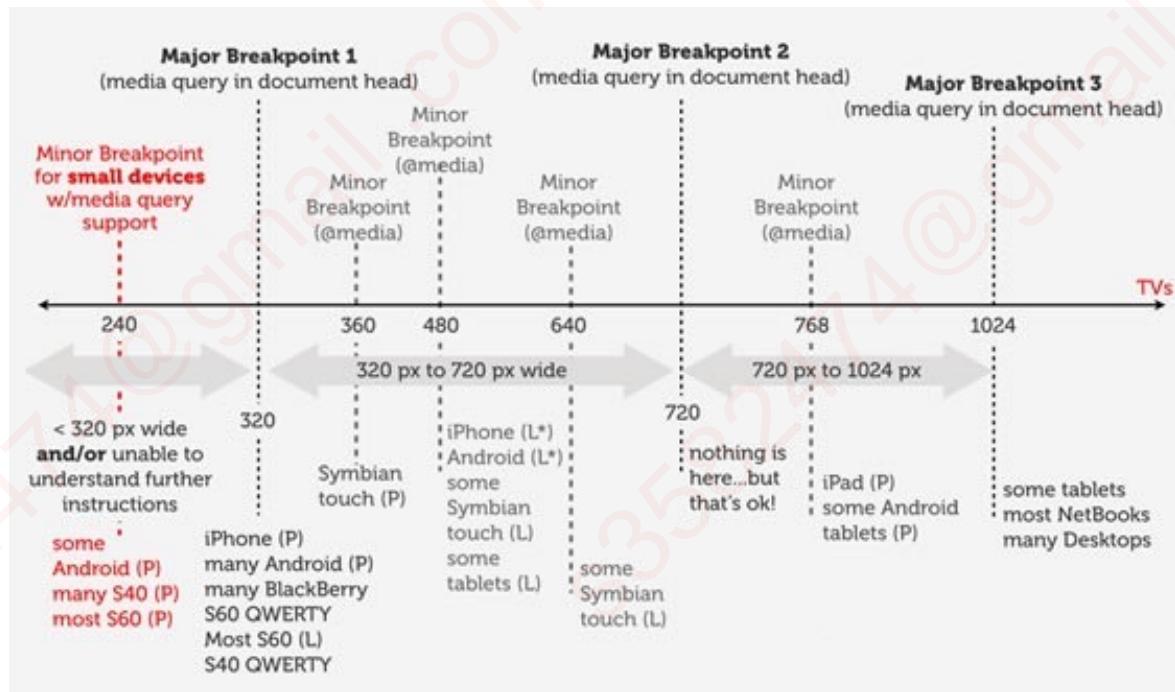
```
/* iPhone and Smartphones (portrait and landscape) */
@media screen and (min-device-width : 320px) and (max-device-width: 480px) {
    //你的样式放在这里...
}
```

CSS3 Media Queries

- @media screen and (max-width: 600px) {
 //你的样式放在这里....
• }

- @media screen and (min-width: 900px) {
 //你的样式放在这里...
• }

Responsive Web Design



```
/*===== Mobile First Method =====*/  
  
/* Custom, iPhone Retina */  
@media only screen and (min-width : 320px) {  
  
}  
  
/* Extra Small Devices, Phones */  
@media only screen and (min-width : 480px) {  
  
}  
  
/* Small Devices, Tablets */  
@media only screen and (min-width : 768px) {  
  
}  
  
/* Medium Devices, Desktops */  
@media only screen and (min-width : 992px) {  
  
}  
  
/* Large Devices, Wide Screens */  
@media only screen and (min-width : 1200px) {  
  
}
```

sass



What's SASS / SCSS?

CSS



Sass
Syntactically Awesome Style Sheets

Does NOT run in the Browser
Extends CSS (during Development)
Compiled to CSS before Production

sass



SASS Features

Nested Rules

Helper Functions

Inheritance

Mixins

Partials

Conditions & Loops

Variables

Css Preprocessor



Less



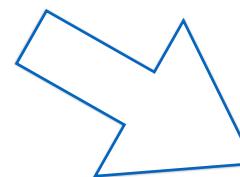
Sass



PostCSS



PostCSS



Sass



`npm install sass -g`

`sass source/stylesheets/index.scss build/stylesheets/index.css`



Sass CSS预处理器 preprocessor



Sass

```
nav {  
    ul {  
        margin: 0;  
        padding: 0;  
        list-style: none;  
    }  
  
    li { display: inline-block; }  
  
    a {  
        display: block;  
        padding: 6px 12px;  
        text-decoration: none;  
    }  
}
```



```
nav ul {  
    margin: 0;  
    padding: 0;  
    list-style: none;  
}  
  
nav li {  
    display: inline-block;  
}  
  
nav a {  
    display: block;  
    padding: 6px 12px;  
    text-decoration: none;  
}
```

Sass: 命令



nested: 嵌套缩进的css代码，它是默认值。

expanded: 没有缩进的、扩展的css代码。

compact: 简洁格式的css代码。

compressed: 压缩后的css代码。

```
sass --style compressed ./style.scss ./style.min.css
```

可以查看下压缩后的效果，生产环境Production时候使用

SASS监听某个文件或目录

```
// watch a file
```

```
sass --watch input.scss:output.css
```

```
// watch a directory
```

```
sass --watch app/sass:public/stylesheets
```

Lab

Sass

根Scss文件

index.scss

main.scss



Imports

Sass

创建variables.scss

```
@import 'variables';
```





Variables

变量以\$开头

Sass

variables.scss

```
$primaryColor: #eefeffcc;  
  
body {  
    background: $primaryColor;  
}
```

```
// Breakpoints  
// Usage: as @include media($small)  
$mobileSm: new-breakpoint(max-width $mobileSm);  
$tiny: new-breakpoint(max-width $tinyMobileMax);  
$mobile: new-breakpoint(max-width $mobileMax);  
$tinyOnly: new-breakpoint(max-width $screenSmMin);  
$small: new-breakpoint(min-width $screenSmMin);  
$medium: new-breakpoint(min-width $screenMdMin);  
$mediumOnly: new-breakpoint(max-width $screenMdMin);  
$deviceMax: new-breakpoint(min-width $deviceMaximum);  
$large: new-breakpoint(min-width $screenLgMin);
```

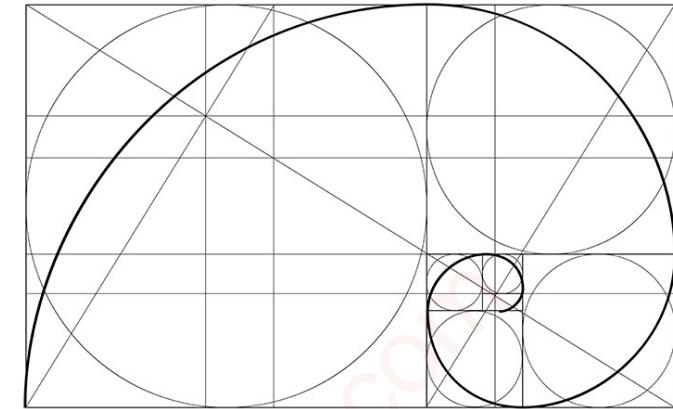
```
// Media queries breakpoints  
$screenSmMin: 768px !default; // Small  
$screenMdMin: 992px !default; // Medium  
$screenLgMin: 1200px !default; // Large
```

Math

Sass

```
$container-width: 100%;  
  
.container {  
    width: $container-width;  
}  
  
.col-4 {  
    width: $container-width / 4;  
}
```

golden ratio



Golden Ratio

$$A/B = (A+B)/A = 1.618033987 = \Phi$$



$$A/B = (A+B)/A = 1.618033987 = \Phi$$

width:300px and another of $300 * 1.6180 \approx 485\text{px}$

Nesting

Sass

```
a.myAnchor {  
    color: blue;  
    &:hover {  
        text-decoration: underline;  
    }  
    &:visited {  
        color: purple;  
    }  
}
```



@extend 继承



Sass

```
.class1 {  
    border: 1px solid #ddd;  
}  
  
.class2 {  
    @extend .class1;  
    font-size:120%;  
}
```

Mixins



使用@Mixin命令， 定义一个代码块。

```
@mixin media($queryString){
```

```
}
```

使用@include命令， 调用这个mixin。

```
div {  
    @include media;  
}
```

Mixins: Lab



写一个media queries的 Mixin

```
// media queries
@mixin MQ($canvas) {
  @if $canvas==S {
    @media only screen and (max-width: $S) {
      @content;
    }
  }
  @else if $canvas==M {
    @media only screen and (max-width: $M) {
      @content;
    }
  }
  @else if $canvas==L {
    @media only screen and (max-width: $L) {
      @content;
    }
  }
}
```



要求

后面所有的css均使用Sass

Popular CSS Library



[Bootstrap](#)



[Foundation](#)



[Pure CSS](#)



[Mobi CSS](#)



[Semantic UI](#)

推荐阅读 [The 5 Most Popular Frontend Frameworks of 2017 Compared](#)



[Bootstrap](#)





- Bootstrap4 是 Bootstrap 的最新版本，与 Bootstrap3 相比拥有了更多的具体的类以及把一些有关的部分变成了相关的组件。同时 Bootstrap.min.css 的体积减少了40%以上。Bootstrap4 使用flexbox 布局，Bootstrap3使用float, width布局
- Bootstrap4 放弃了对 IE8 以及 iOS 6 的支持，现在仅仅支持 IE9 以上 以及 iOS 7 以上版本的浏览器。如果对于其中需要用到以前的浏览器，那么请使用 Bootstrap3。

Bootstrap



	Extra small <576px	Small ≥576px	Medium ≥768px	Large ≥992px	Extra large ≥1200px
Max container width	None (auto)	540px	720px	960px	1140px
Class prefix	.col-	.col-sm-	.col-md-	.col-lg-	.col-xl-
# of columns	12				
Gutter width	30px (15px on each side of a column)				
Nestable	Yes				
Column ordering	Yes				

```
<div class="row">
  <div class="col-sm-8">col-sm-8</div>
  <div class="col-sm-4">col-sm-4</div>
</div>
<div class="row">
  <div class="col-sm">col-sm</div>
  <div class="col-sm">col-sm</div>
  <div class="col-sm">col-sm</div>
</div>
```

Bootstrap: Container



Bootstrap comes with three different containers:

- `.container`, which sets a `max-width` at each responsive breakpoint
- `.container-fluid`, which is `width: 100%` at all breakpoints
- `.container-{breakpoint}`, which is `width: 100%` until the specified breakpoint

Bootstrap: row col



	Extra small ≤576px	Small ≥576px	Medium ≥768px	Large ≥992px	Extra large ≥1200px
Max container width	None (auto)	540px	720px	960px	1140px
Class prefix	.col-	.col-sm-	.col-md-	.col-lg-	.col-xl-
# of columns	12				
Gutter width	30px (15px on each side of a column)				
Nestable	Yes				
Column ordering	Yes				

```
<div class="row">
  <div class="col-sm-8">col-sm-8</div>
  <div class="col-sm-4">col-sm-4</div>
</div>
<div class="row">
  <div class="col-sm">col-sm</div>
  <div class="col-sm">col-sm</div>
  <div class="col-sm">col-sm</div>
</div>
```



UI Design principles

Consistency

User familiarity
Minimal surprise



Typography

- primary typeface,
- eventually secondary typeface (no need for a third one—two typefaces are usually enough, really),
- chosen font families (regular, italic, bold, semibold, light, ...),
- text styles examples (headings, paragraphs, info texts,).

Google Fonts

<https://fonts.google.com/>

Typefaces

Source Sans Pro

Source Sans Pro is an open-source sans serif typeface created for legibility in UI design. With a variety of weights that read easily at all sizes, Source Sans Pro provides clear headers as well as highly-readable body text.

Inspired by twentieth century American geometric typeface design, its slender but open letters offer a clean and friendly simplicity. Advanced hinting allows Source Sans Pro to render well on Windows systems which run ClearType, and across browsers and devices. Moreover, it supports a variety of languages and alphabets, including Western and European language, Vietnamese, pinyin, Romanization of Chinese, and Navajo.

Aa

ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz
0123456789



Color Palette

- primary colours,
- secondary colours,
- complementary colours,
- alerts colours (warnings, errors, information validations, ...),
- additional colour elements (for example gradients or semi-transparent overlays).

Light Marvel #85D7FF	Light Purple #A389F4	Light Pink #FF7CE5	Light Orange #FF9E7C
Marvel #1FB6FF	Purple #7E5BEF	Pink #FF49DB	Orange #FF7849
Dark Marvel #009EB	Dark Purple #592DEA	Dark Pink #FF16D1	Dark Orange #FF5216
Light Green #29EB7F	Light Yellow #FFD55F		
Green #13CE66	Yellow #FFCB2C		
Dark Green #0F9F4F	Dark Yellow #F8B700		

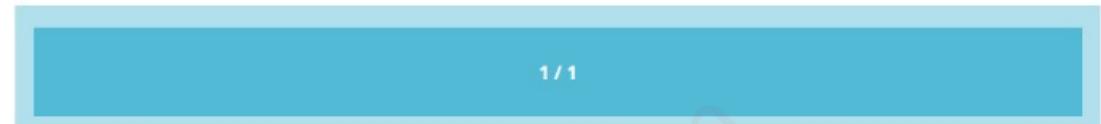


Spacing and Positioning

- grid system (sizes, gutters, nesting, ...),
- spacing rules and spacing patterns,
- spacing helpers (reusable constants for vertical and horizontal gutters between UI elements which can be used in the code).

Grid sizes

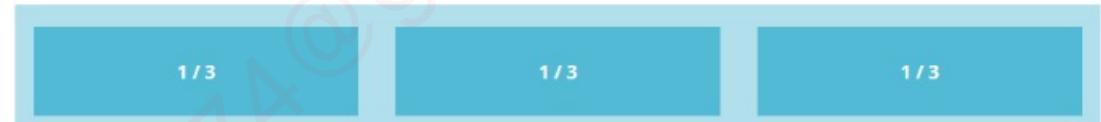
Size 1 of 1



Size 1 of 2

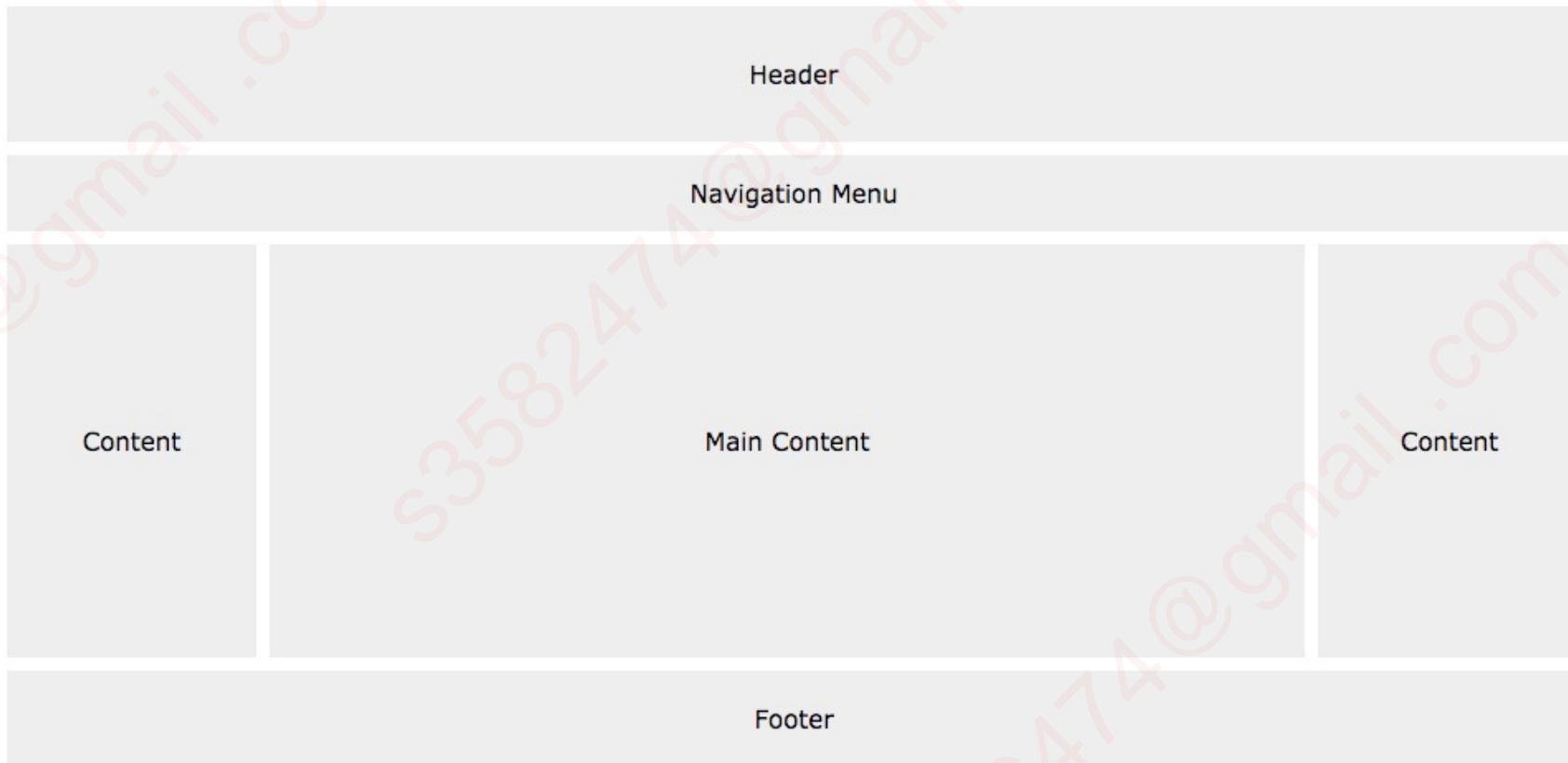


Size 1 of 3





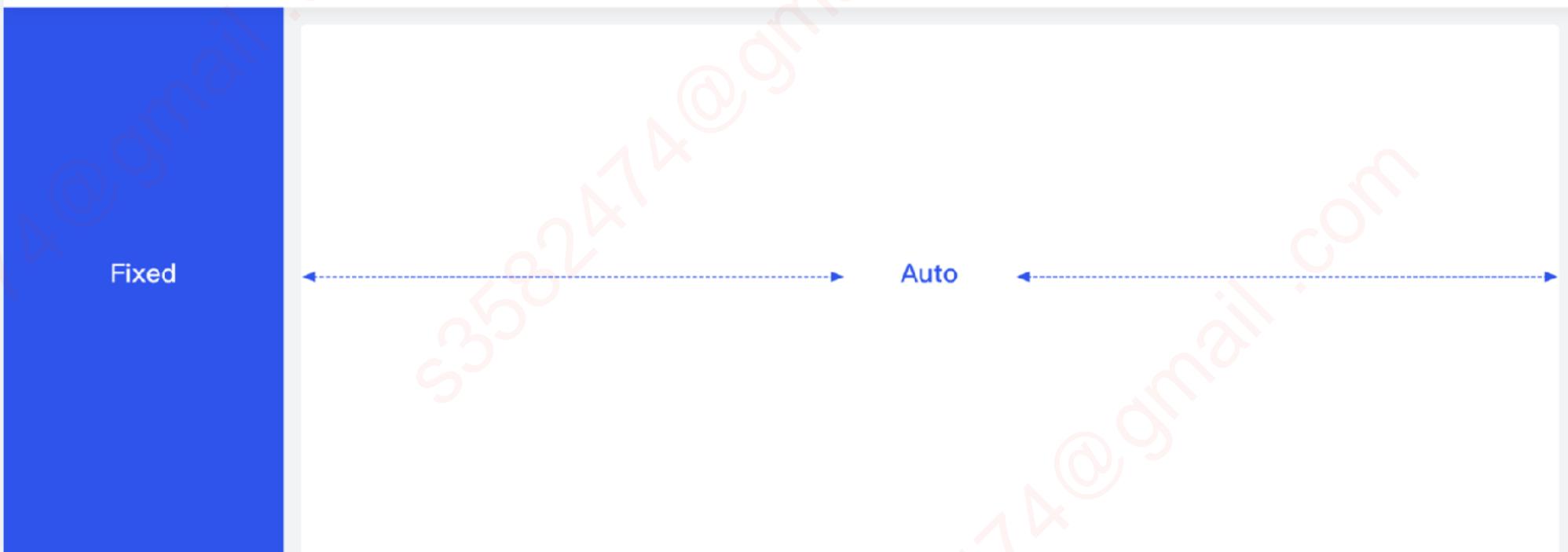
Layout





Layout

导航目录一 导航目录二 导航目录三 导航目录四



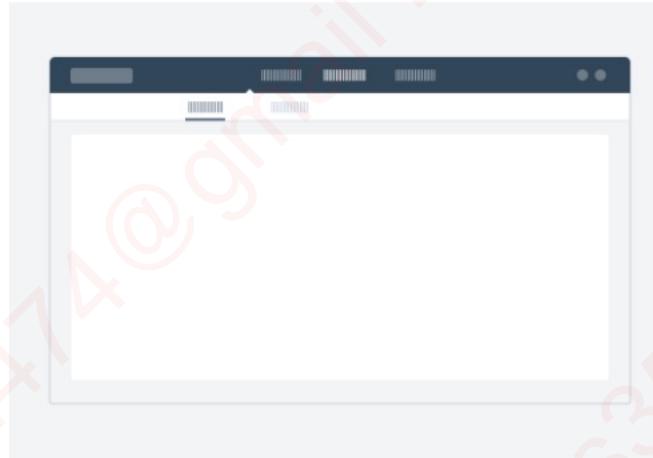


UI Elements

- navigation (main nav, secondary nav, breadcrumbs, pagination, ...),
- buttons (primary/secondary action buttons, tools buttons, ...),
- alert messages (warnings, errors, information, validations, ...),
- forms (fields, radio buttons, checkboxes, poplists, infield validation, ...),
- markers (person types markers, patient states markers, document types, ...),
- additional elements (calendar, patient timeline, patients lists, documents lists, cards, modal panels,).



Navigation



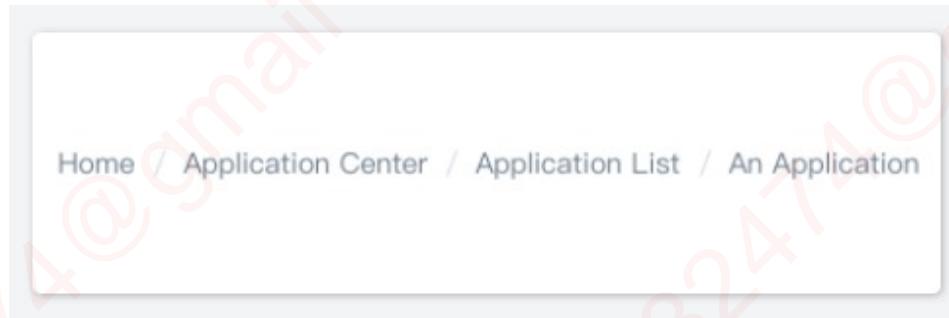
Top Navigation



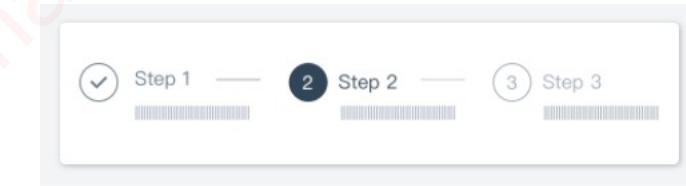
Side Navigation



Navigation



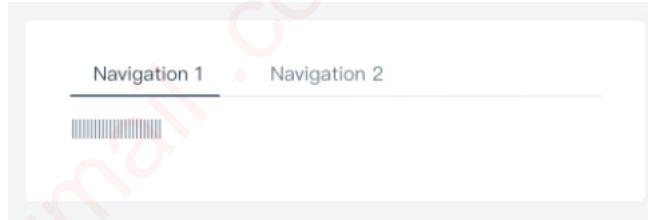
Breadcrumb



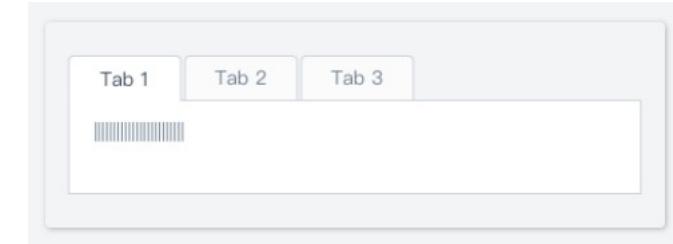
Steps



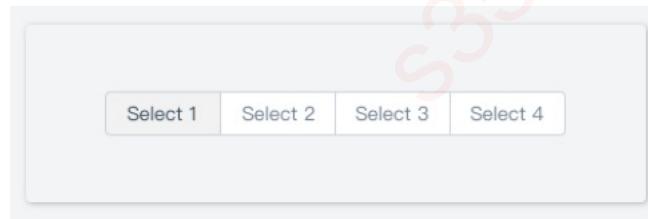
Navigation



Basic



Card



Pills

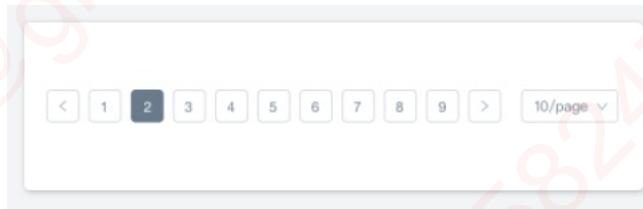
Tabs



Vertical



Navigation



Pagination





VS Code Extension

Auto Close Tag
Auto Rename Tag
HTML Snippets
Babel ES6/ES7
Debugger for Chrome
ESLint
JavaScript (ES6) code snippets
Code Spell Checker
Node.js Modules Intellisense
React-Native/React/Redux snippets for ES6/ES7
TSLint
vscode-styled-components

Color Highlight
Custom CSS and JS Loader
EditorConfig for VS Code
File Templates
Git History (git log)
Material Icon Theme
Prettier
Theme—Seti-Monokai



个人项目

- <https://hades217.github.io/>

<https://medium.freecodecamp.org/css-naming-conventions-that-will-save-you-hours-of-debugging-35cea737d849>

联系方式



1. 微信联系求职顾问



匠人小花



悉尼匠人小嘿



墨尔本匠人小萌



阿德匠人小德

2. 匠人官网报名

Jiangren.com.au



澳洲匠人学院

电话 : 0421-672-555
微信 : jracademy
邮箱 : hello@jiangren.com.au
网址 : jiangren.com.au



匠人找工

免费发雇主招聘信息 , 免费发招
聘者求职信息

微信 : jiangren_jobs