

CSS

- description : CSS
- author :
- email : dhan@repia.com
- lastupdate : 2020-03-13

WIKI: [HTML JAVASCRIPT](#)

Case Study

FlexBox

```
float: {left, center, right}

container
  display
  flex-direction
  flex-wrap
  flex-flow
  justify-content
  align-items
  align-content

item
  order
  flex-grow
  flex-shrink
  flex
  align-self

main axis(    ), cross axis(    )

EMMET(    )
div.container>div.item.item${$}

.container {
```

```
background: beige;
height: 100vh; /* viewport height */
display: flex;
flex-direction: row
  row:
  row-reverse;
  column:
  column-reverse:
flex-wrap: nowrap;
  wrap
  wrap-reverse
/* flex-flow: column nowrap */ flex-direction, flex-wrap
justify-content: flex-start;
  flex-start:
  flex-end:
  center
  space-around, space-evenly, space-between /* main axis */
align-items: center; baseline(
  align-content: center, space-around, space-evenly, space-between

}

.item {
  width: 40px;
  height: 40px;
  border: 1px solid black;
}

.item1 {
  background: #ef9a9a;
  flex-grow: 2; /*
  flex-shrink: 2; /*
  flex-basis: 60%; /* , flex-grow, flxe-shrink
  align-self: center; /*
}

.item2 {
  background: #ef9a9a;
  flex-grow: 1;
  flex-shrink: 1;
  flex-basis: 60%;
}

.item3 {
  background: #ef9a9a;
  flex-grow: 1;
  flex-shrink: 1;
  flex-basis: 60%;
}
```

jsbin.com
[COLOR TOOL](#)
[Flexbox Froggy](#)

Grid

CSS Layout

flex

```
1
  flxe, flex(column)
  row, column
```

grid

```
2
  display: grid
```

grid-template-columns:

```
grid-template-columns: 100px 100px 100px 100px
grid-template-columns: repeat(5, 100px)
grid-template-columns: repeat(5, 20%)
grid-template-columns: 1fr 2fr 1fr
grid-template-columns: repeat(5, 1fr)
```

grid-template-rows

```
grid-auto-rows: 150px
  grid-auto-rows: minmax(150px, auto)
```

grid-template-areas: grid cell labeling

```
grid-gap: 10px
  grid-column-gap: 10px
  grid-row-gap: 10px
```

grid cell

```
grid-column-start, grid-column-end, grid-row-start, grid-row-end:
```

```
1 (0 )
  grid-column-start: 2;
  grid-column-end: 4;
  grid-row-start: 1;
  grid-row-end: 3
```

or

```
grid-column: 2 / 4;
grid-row: 1 / 3;
```

or

```
grid-column: 2 / span 2
grid-row: 1 / 3;
```

```
,
    grid-column: 2 / -1
    grid-row: 1 / 3;

    grid-area

body {
    padding: 5rem;
    background-color: black;
}

.container {
    display: grid;
    grid-template-columns: repeat(3, 1fr);
    grid-auto-rows: 150px;
    grid-gap: 1rem;
    grid-template-areas:
        'a a a'
        'b c c'
        'b d g'
        'e f g'
}

.image {
    width: 100%;
    height: 100%;
    object-fit: cover;
}

.image1 {
    grid-area: a
}

.image2 {
    grid-area: b
}

.image3 {
    grid-area: c
}

.image4 {
    grid-area: d
}

.image5 {
    grid-area: e
}
```

```
.image6 {  
  grid-area: f  
}
```

```
.image7 {  
  grid-area: g  
}
```

```
body {  
  width: 100vw;  
  height: 100vh;  
  margin: 0;  
  display: grid;  
  grid-template-columns: 3fr 1fr;  
  grid-template-rows: 100px auto 50px;  
  grid-template-areas:  
    'header header'  
    'main side'  
    'footer footer';  
}  
  
header {  
  grid-area: header;  
}  
  
main {  
  grid-area: main;  
}  
  
aside {  
  grid-area: side;  
}  
  
footer {  
  grid-area: footer;  
}
```

[CSS Flexbox](#)

[CSS Grid](#)

!!

[Emmet, HJTML](#)

[unsplash](#)

Plugin

font-awesome

```
<link href="/fonts/font-awesome/css/all.min.css" rel="stylesheet"/>  
<link rel="stylesheet" href="/components/font-awesome/css/font-
```

```
awesome.min.css">
```

bootstrap

```
<link rel="stylesheet" href="/components/bootstrap/dist/css/bootstrap.min.css">
```

Ionicons

, SVG

```
<link rel="stylesheet" href="/components/Ionicons/css/ionicons.min.css">
```

AdminLTE

Bootstrap , Adminlet3

```
<link rel="stylesheet" href="/dist/css/AdminLTE.min.css">
```

adminty

Bootstrap , Adminty

	Bootstrap
Adminty	.
가	.
Adminty	.
Adminty	가
	가

google font

```
<link rel="stylesheet"
href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,600,700,300italic,400italic,600italic">
```

overlayScrollbars

```
<link
th:href="${#HttpServletRequest.getContextPath()}+'/plugins/overlayScrollbars/css/OverlayScrollbars.min.css'" rel="stylesheet" />
```

daterangepicker

```
<link
th:href="${#HttpServletRequest.getContextPath()}+'/plugins/daterangepicker/daterangepicker.css'" rel="stylesheet" />
```

summernote-bs4.css

```
<link
th:href="${#HttpServletRequest.getContextPath()}+'/plugins/summernote/summernote-bs4.css'" rel="stylesheet" />
```

How To Add CSS

External CSS

```
<!DOCTYPE html>
<html>
<head>
  <link rel="stylesheet" type="text/css" href="mystyle.css" />
</head>
<body>
</body>
</html>
```

Internal CSS

```

<!DOCTYPE html>
<html>
<head>
<style>
body {
  background-color: linen;
}

h1 {
  color: maroon;
  margin-left: 40px;
}
</style>
</head>
<body>
</body>
</html>

```

[How To Add CSS](#)

Tag Cloud

<https://demo.chattymango.com/shuffle-box/>
<http://dimox.name/beautiful-tags-markup/>

Tip

CSS

```

span {
  /* all: unset !important; */
  all: unset;
}

```

CSS

- CSS
- property: value !important;

<code html>

```
color: red !important;
```

```
color: blue;
```



```
color: green !important;
```

</code CSS

Ref

- [COOL AND RESPONSIVE BUTTONS WITH A FEW LINES OF CSS](#)
- [Ultimate Blocks - Gutenberg Blocks Plugin](#)
- [CSS Selector](#)
- [A Complete Guide to Flexbox](#)
- [FLEXBOX FROGGY](#)

SCSS

Sass(Syntactically Awesome StyleSheets) 가


Scss CSS pre-processor CSS CSS

-
-
- Import
- Nesting
- Mixin
- Extend / Inheritance

가

Sass Sass 가 Sass(.scss) CSS

- [Sass](#)

 (Header Label), ~ (jhgong, 2020-05-29)

CSS Note

- [css note](#)

, , [button](#)

From:

<http://rwiki.repia.com/> -
2023.12

Permanent link:

<http://rwiki.repia.com/doku.php?id=wiki:css&rev=1665486794>

Last update: **2022/10/11 20:13**

