**作 业**

1. **简答题**
2. 什么是弹性布局？容器属性有哪些？项目属性有哪些？

弹性布局是一种用于布局用户界面元素的 CSS 布局模型，旨在提供一种更加高效、简洁和一致的方法来排列和分布容器内的子元素。

容器属性

display: 定义 flex 容器。

display: flex; 表示将元素设为块级 flex 容器。

display: inline-flex; 表示将元素设为行内级 flex 容器。

flex-direction: 定义主轴的方向（即项目的排列方向）。

row (默认): 从左到右排列项目。

row-reverse: 从右到左排列项目。

column: 从上到下排列项目。

column-reverse: 从下到上排列项目。

flex-wrap: 定义项目是否换行。

nowrap (默认): 不换行，所有项目在一行内排列。

wrap: 换行，项目在必要时会换行。

wrap-reverse: 换行，但行的排列方向为反向。

flex-flow: 是 flex-direction 和 flex-wrap 的简写。

例如：flex-flow: row wrap;

justify-content: 定义主轴上的对齐方式。

flex-start (默认): 项目从主轴的起点开始。

flex-end: 项目从主轴的终点开始。

center: 项目在主轴上居中。

space-between: 项目之间的间隔相等。

space-around: 项目周围的间隔相等。

align-items: 定义交叉轴上的对齐方式。

stretch (默认): 项目被拉伸以适应容器。

flex-start: 项目在交叉轴的起点对齐。

flex-end: 项目在交叉轴的终点对齐。

center: 项目在交叉轴上居中对齐。

baseline: 项目的基线对齐。

align-content: 定义多根轴线的对齐方式。仅在有多根轴线时使用。

stretch (默认): 轴线被拉伸以适应容器。

flex-start: 轴线在交叉轴的起点对齐。

flex-end: 轴线在交叉轴的终点对齐。

center: 轴线在交叉轴上居中对齐。

space-between: 轴线之间的间隔相等。

space-around: 轴线周围的间隔相等。

**项目是 flex 容器的直接子元素。**

order: 定义项目的排列顺序。数值越小，项目排列越靠前。默认值为 0。

例如：order: 1;

flex-grow: 定义项目的放大比例。默认值为 0 (即如果存在剩余空间，也不放大)。

例如：flex-grow: 1; 表示项目会占据剩余空间。

flex-shrink: 定义项目的缩小比例。默认值为 1 (即如果空间不足，项目会缩小)。

例如：flex-shrink: 0; 表示项目不会缩小。

flex-basis: 定义项目的初始大小。可以是长度单位（如 px, em, %）或关键字 auto。默认值为 auto。

例如：flex-basis: 100px;

flex: 是 flex-grow, flex-shrink 和 flex-basis 的简写。

例如：flex: 1 1 auto;

align-self: 定义单个项目在交叉轴上的对齐方式，覆盖 align-items 属性。

auto (默认): 继承父容器的 align-items 属性。

flex-start: 项目在交叉轴的起点对齐。

flex-end: 项目在交叉轴的终点对齐。

center: 项目在交叉轴上居中对齐。

baseline: 项目的基线对齐。

stretch: 项目被拉伸以适应容器。

1. 在一个父元素div中，如何用flex布局让一个子元素div水平居中，垂直居中？

父元素 div 的 CSS 设置：

将 display 属性设置为 flex，使父元素成为一个 flex 容器。

使用 justify-content 属性将子元素水平居中。

使用 align-items 属性将子元素垂直居中。

子元素 div 的 CSS 设置：

应用父元素的 flex 布局属性。

比如：

.parent {

display: flex; /\* 将父元素设为flex容器 \*/

justify-content: center;/\* 水平居中 \*/

align-items: center; /\* 垂直居中 \*/

height: 100vh; /\* 父元素高度设为视口高度，便于观察效果 \*/

}

.child {

/\* 子元素可以有自己的样式设置 \*/

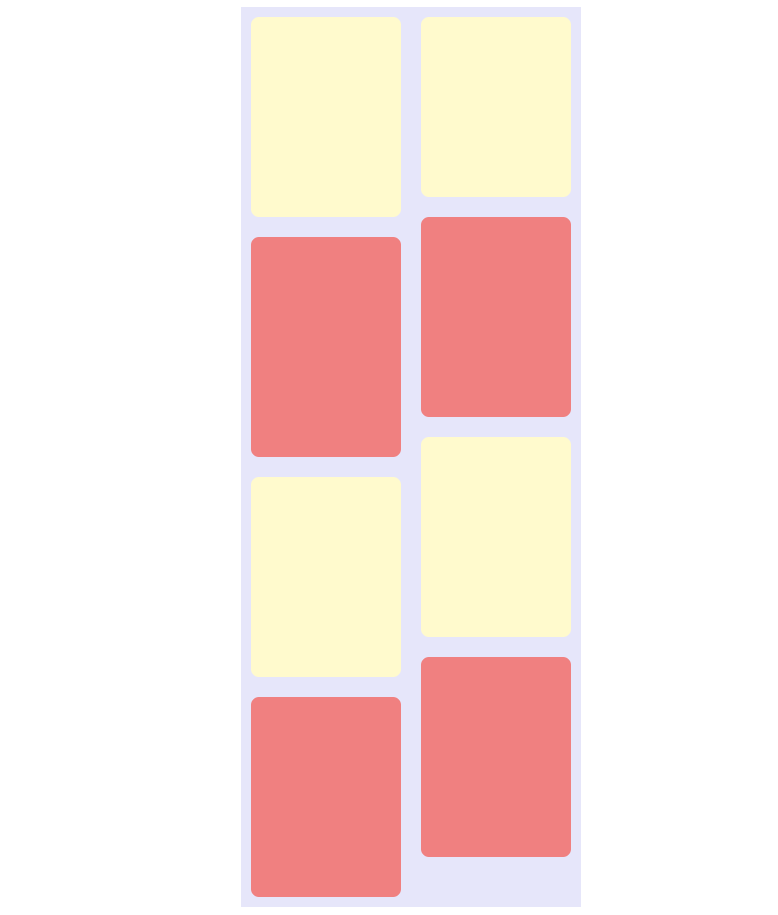
width: 100px;

height: 100px;

background-color: lightblue;

}

1. **编程题**
2. 使用弹性布局实现以下效果。



<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<style>

body {

display: flex;

justify-content: center;

align-items: center;

height: 100vh;

background-color: #E6E6FA;

margin: 0;

}

.container {

display: grid;

grid-template-columns: repeat(2, 1fr);

grid-auto-rows: 100px;

gap: 10px;

width: 250px;

}

.item {

background-color: #FFFACD;

border-radius: 5%;

}

.item:nth-child(even) {

background-color: #F08080;

}

.item:nth-child(3),

.item:nth-child(6) {

grid-row: span 2;

}

</style>

</head>

<body>

<div class="container">

<div class="item"></div>

<div class="item"></div>

<div class="item"></div>

<div class="item"></div>

<div class="item"></div>

<div class="item"></div>

<div class="item"></div>

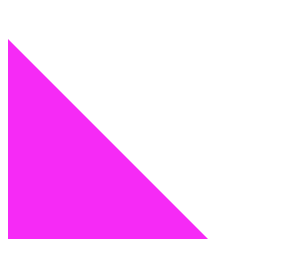
<div class="item"></div>

</div>

</body>

</html>

1. 使用border属性，绘制出一个直角三角形。



<!--

\* @Author: 陈佳敏 1685940181@qq.com

\* @Date: 2024-07-08 21:40:11

\* @LastEditors: 陈佳敏 1685940181@qq.com

\* @LastEditTime: 2024-07-08 23:06:50

\* @FilePath: \实训\7.8作业\T2.html

\* @Description: 这是默认设置,请设置`customMade`, 打开koroFileHeader查看配置 进行设置: https://github.com/OBKoro1/koro1FileHeader/wiki/%E9%85%8D%E7%BD%AE

-->

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>T2</title>

<style>

.right-angle-triangle {

width: 0;

height: 0;

border-bottom: 100px solid #F62AF6;

border-right: 100px solid transparent;

}

</style>

</head>

<body>

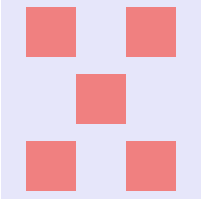
<div class="right-angle-triangle"></div>

</body>

</html>

1. 使用flex布局，实现骰子的六个面。（1-6）

例如： 5



<!--

\* @Author: 陈佳敏 1685940181@qq.com

\* @Date: 2024-07-08 21:40:11

\* @LastEditors: 陈佳敏 1685940181@qq.com

\* @LastEditTime: 2024-07-08 23:06:50

\* @FilePath: \实训\7.8作业\T3.html

\* @Description: 这是默认设置,请设置`customMade`, 打开koroFileHeader查看配置 进行设置: https://github.com/OBKoro1/koro1FileHeader/wiki/%E9%85%8D%E7%BD%AE

-->

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>骰子</title>

<style>

body {

display: flex;

flex-direction: column;

align-items: center;

justify-content: center;

height: 100vh;

background-color: #E6E6FA;

}

.dice {

display: flex;

flex-wrap: wrap;

width: 200px;

height: 200px;

gap: 20px;

}

.dot {

width: 50px;

height: 50px;

background-color: #F08080;

border-radius: 0%;

}

.hidden {

visibility: hidden;

}

.slider {

margin-top: 30px;

}

</style>

</head>

<body>

<header>这是一个骰子</header>

<div class="dice">

<div class="dot hidden"></div>

<div class="dot hidden"></div>

<div class="dot hidden"></div>

<div class="dot hidden"></div>

<div class="dot hidden"></div>

<div class="dot hidden"></div>

<div class="dot hidden"></div>

<div class="dot hidden"></div>

<div class="dot hidden"></div>

</div>

<a>1<input type="range" min="1" max="6" value="1" class="slider" id="diceSlider">6</a>

<script>

const slider = document.getElementById('diceSlider');

const dots = document.querySelectorAll('.dot');

const patterns = [

[4], // 1

[0, 8], // 2

[0, 4, 8], // 3

[0, 2, 6, 8], // 4

[0, 2, 4, 6, 8], // 5

[0, 1, 2, 6, 7, 8] // 6

];

slider.addEventListener('input', () => {

const value = slider.value;

dots.forEach(dot => dot.classList.add('hidden'));

patterns[value - 1].forEach(index => dots[index].classList.remove('hidden'));

});

// 初始化

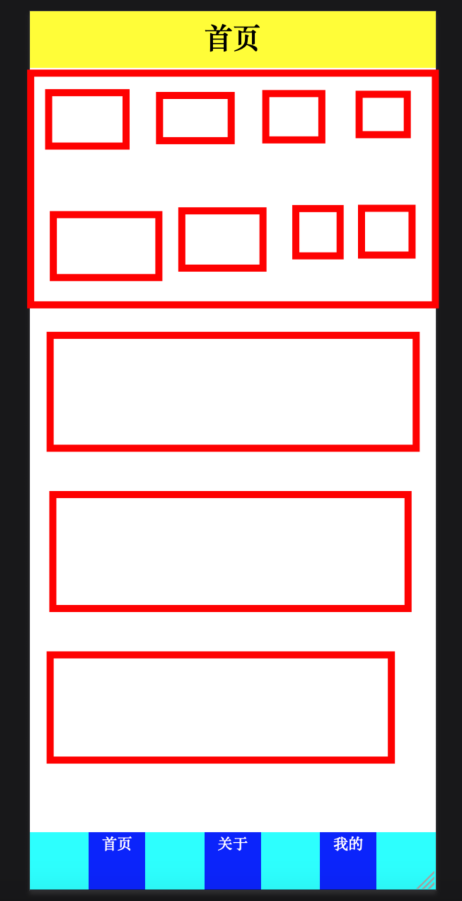
slider.dispatchEvent(new Event('input'));

</script>

</body>

</html>

1. （优先）实现手机app首页布局，如下图。



<!--

\* @Author: 陈佳敏 1685940181@qq.com

\* @Date: 2024-07-08 21:33:29

\* @LastEditors: 陈佳敏 1685940181@qq.com

\* @LastEditTime: 2024-07-08 21:34:59

\* @FilePath: \实训\7.8作业\index.html

\* @Description: 这是默认设置,请设置`customMade`, 打开koroFileHeader查看配置 进行设置: https://github.com/OBKoro1/koro1FileHeader/wiki/%E9%85%8D%E7%BD%AE

-->

<!DOCTYPE html>

<html lang="zh-CN">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>首页</title>

<style>

body {

margin: 0;

font-family: Arial, sans-serif;

}

.header {

background-color: yellow;

text-align: center;

padding: 10px;

font-size: 24px;

}

.container {

padding: 10px;

}

.grid-container {

display: grid;

grid-template-columns: repeat(5, 1fr);

gap: 10px;

}

.grid-container .item {

background-color: white;

border: 2px solid red;

height: 80px;

}

.grid-row {

display: grid;

grid-template-columns: repeat(3, 1fr);

gap: 10px;

margin-top: 10px;

}

.grid-row .item {

background-color: white;

border: 2px solid red;

height: 100px;

}

.box {

background-color: white;

border: 2px solid red;

height: 150px;

margin-top: 10px;

}

.footer {

display: flex;

justify-content: space-around;

background-color: cyan;

padding: 10px 0;

position: fixed;

width: 100%;

bottom: 0;

}

.footer div {

flex: 1;

text-align: center;

padding: 10px 0;

background-color: blue;

color: white;

}

</style>

</head>

<body>

<div class="header">首页</div>

<div class="container">

<div class="grid-container">

<div class="item"></div>

<div class="item"></div>

<div class="item"></div>

<div class="item"></div>

<div class="item"></div>

</div>

<div class="grid-row">

<div class="item"></div>

<div class="item"></div>

<div class="item"></div>

</div>

<div class="box"></div>

<div class="box"></div>

<div class="box"></div>

</div>

<div class="footer">

<div>首页</div>

<div>关于</div>

<div>我的</div>

</div>

</body>

</html>