

[笔记][LIKE-JS][3-HTML+CSS-高级篇]

前端

[笔记][LIKE-JS][3-HTML+CSS-高级篇]

077. 伸缩布局(flex)-主轴方向(掌握)

078. 伸缩布局(flex)-主轴和侧轴对齐方式(掌握)

077. 伸缩布局(flex)-主轴方向(掌握)

flex 布局

`display: inline-flex;` 将对象作为弹性伸缩盒展示，用于行内元素

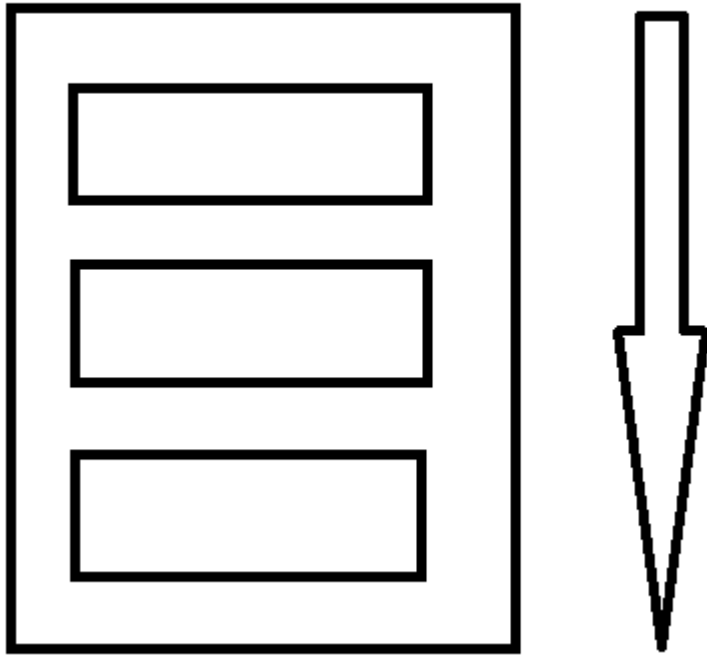
`display: flex;` 将对象作为弹性伸缩盒展示，用于块级元素

一般针对块级元素

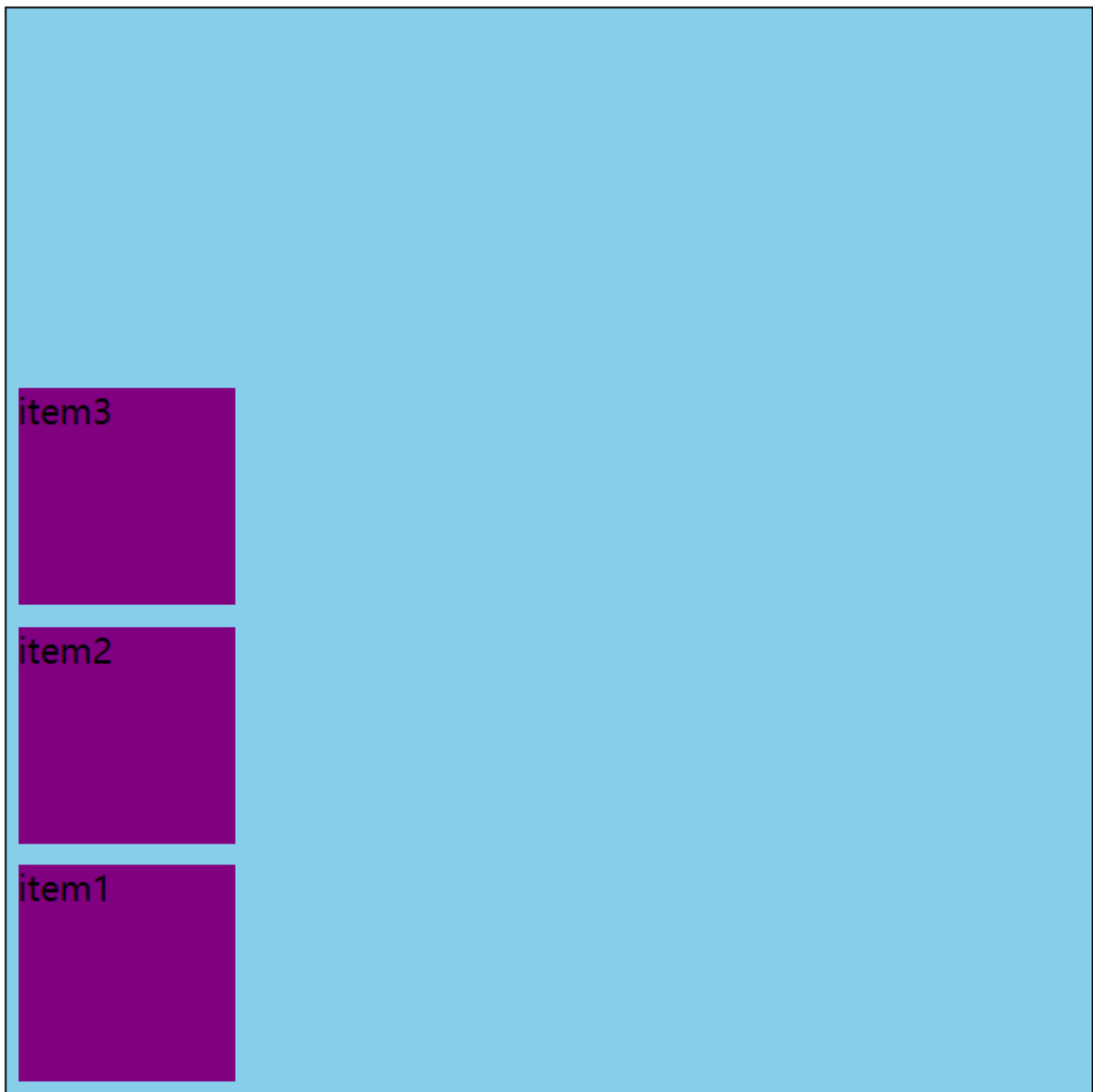
常用属性

`flex-direction`

主轴方向：从上到下



- 用于指定 `flex` 主轴的方向，继而决定 `flex` 子项在 `flex` 容器中的位置
- 取值
 - `row` 默认值，表示水平方向从左到右排列，此时水平方向轴线为主轴
 - `row-reverse` 与 `row` 相反
 - `column` 表示垂直方向从上到下排列，此时垂直方向轴线为主轴。
 - `column-reverse` 与 `column` 相反



```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Title</title>
  <style>
    html{
      font-size: 10px;
    }

    .item{
      width: 10rem;
      height: 10rem;
      background-color: purple;
      margin: 5px;
    }

    body{
      font-size: 1.6rem;
    }
  </style>
</head>
<body>
  <div>
    <div>
      <div>item3</div>
      <div>item2</div>
      <div>item1</div>
    </div>
  </div>
</body>
</html>
```

```
}

#box1{
  border: 1px solid #000;
  width: 50rem;
  height: 50rem;
  background-color: skyblue;

  /* 伸缩布局 */
  display: flex;
  /* 主轴方向: 默认从左到右, 即 row */
  /*flex-direction: row;*/
  /*flex-direction: row-reverse;*/
  /*flex-direction: column;*/
  flex-direction: column-reverse;
}

</style>
</head>
<body>
<div id="box1">
  <div class="item">item1</div>
  <div class="item">item2</div>
  <div class="item">item3</div>
</div>
</body>
</html>
```

078. 伸缩布局(flex)-主轴和侧轴对齐方式(掌握)

`justify-content`

`justify` 有整理版面的意思，所以直译过来是整理内容

- 用于指定主轴上子项的对齐方式

或许翻译成交叉轴更好？

- 取值：
 - `flex-start`：默认值，表示与行的起始位置对齐
 - `flex-end`：表示与行的结束位置对齐
 - `center`：表示与行的中间位置对齐
 - `space-between`：表示两端对齐，中间间距相等。（当剩余空间为负数或者只有一个项时，效果等同于 `flex-start`）

- **space-around** : 间距相等，中间间距是两端间距的两倍。（当剩余空间为负数或只有一个项时，效果等同于 **center**）

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Title</title>
  <style>
    html{
      font-size: 10px;
    }

    .item{
      width: 10rem;
      height: 10rem;
      background-color: purple;
      margin: 5px;
    }

    body{
      font-size: 1.6rem;
    }

    #box1{
      border: 1px solid #000;
      width: 50rem;
      height: 50rem;
      background-color: skyblue;

      /* 伸缩布局 */
      display: flex;

      /* 主轴上子项对齐方式 */
      /* 默认：开始对齐 */
      /*justify-content: flex-start;*/

      /* 尾部对齐 */
      /*justify-content: flex-end;*/

      /* 居中对齐 */
      /*justify-content: center;*/

      /* 两端对齐 */
      /*justify-content: space-between;*/

      /* 两端的空间是中间空间的二分之一 */
      /* 个人理解：可以把它当成一个环 */
      /* 个人理解：把 space 空间平均分布在一个环上 */
      /* 个人理解：前面的 space-between 就是把空间平均分配在元素之间 */
      justify-content: space-around;
    }
  </style>

```

```
    }  
    </style>  
</head>  
<body>  
<div id="box1">  
    <div class="item">item1</div>  
    <div class="item">item2</div>  
    <div class="item">item3</div>  
</div>  
</body>  
</html>
```

align-items：用于指定侧轴（垂直方向）上 **flex** 子项的对齐方式

- **stretch** 默认值，当 **flex** 子项未设置高度或者高度为 **auto** 时，**stretch** 起作用，将 **flex** 子项高度设置为行高度。在只有一行的情况下，行的高度为容器的高度，即 **flex** 子项高度为容器的高度。
 - **flex-start** 表示与侧轴开始位置对齐
 - **flex-end** 表示与侧轴结束位置对齐
 - **center** 表示与侧轴中间对齐
 - **baseline** 表示基线对齐，当行内轴与侧轴在同一直线上，即所有 **flex** 子项的基线在同一直线上时，效果等同于 **flex-start**。
-