# CSS同时水平居中和垂直居中

1.水平居中：margin: 0 auto; text-align:center;

2.水平垂直居中

①已知子div高度宽度用position 父级：relative 子：absolute+left/top50%+margin(-子高/2 0 0 -子宽/2)

②未知子div高度宽度用position 父级：relative 子：absolute+left/top50%+transform：translate(-50%,-50%)

注：transform：none|*transform-functions（旋转div元素）其中translate（x,y）定义2D转换*

③父级：display：flex；

justify-content: center;//使子项目水平居中

align-items: center;//使子项目垂直居中

④table-cell布局（不推荐使用）

注：css3 Flex布局（伸缩布局盒模型）

其中容器的属性用的比较多：

* flex-direction、flex-wrap、flex-flow、justify-content、align-items、align-content
* **flex：**
* flex: *flex-grow* *flex-shrink* *flex-basis*|auto|initial|inherit;
* 相对于其他flex项目进行扩展的量；相对于..收缩的量；项目长度
* flex:1的子元素代表让所有弹性盒模型对象的子元素都有相同的长度，且忽略它们内部的内容

# audio

属性 autoplay 音频就绪后马上开始播放

属性 controls 向用户显示控件比如播放按钮

属性 src 要播放音频的URL

duration：返回当前音频的长度（s）

# 左侧固定，右侧自适应

|  |  |  |
| --- | --- | --- |
| 父容器 | 左侧div | 右侧div |
| display: flex; | flex:0 0 100px; | flex:1 |
| overflow: auto; | float:left; | overflow:auto; |
| overflow: hidden | float: left;width:200px; | margin-left:200px; |

flex: *flex-grow* *flex-shrink* *flex-basis*|auto|initial|inherit;

相对于其他flex项目进行扩展的量；相对于..收缩的量；项目长度

flex:1的子元素代表让所有弹性盒模型对象的子元素都有相同的长度，且忽略它们内部的内容

# BFC布局

特点：块级格式化上下文

1.BFC下内部的box垂直放置

2.垂直方向的距离由margin决定，且属于同一个bfc的两个相邻box的margin会重叠。解决：每个盒子都设置为BFC

3.BFC中每个盒子的margin-left会触碰到父容器的border-left

4.BFC不会与浮动元素重叠，计算高度时浮动元素也参与计算

触发条件

1.body根元素

2.浮动元素

3.绝对定位元素

4.overflow除了visible以外的值：hidden scroll auto

5. display为inline-block, table-cell, table-caption, flex, inline-flex

# CSS有哪些单位

px:像素：相对于显示器屏幕分辨率而言

%：相对于父元素的大小设定的比率

em:相对长度单位：基准是父节点字体大小

rem:相对单位。root em 相对根节点HTML的字体大小计算

vw: viewport width，视窗宽度，1vw等于视窗宽度的1%。

vh: viewport height，视口高度，1vh等于视窗高度的1%。

vmin：vw和vh中较小的那个。  
vmax：vw和vh中较大的那个。