HTML && CSS

## IE8

margin、padding负值可能会赘余

使用ES3语法

## 动画(animation)

动画停留在最后一帧的样子

animation-fill-mode：forwards (默认none)

动画执行一次

animation-iteration-count:**1**;

## input

去除H5 input默认样式：

input[type=number] {

-moz-appearance: textfield;

}

input[type=number]::-webkit-inner-spin-button,

input[type=number]::-webkit-outer-spin-button {

-webkit-appearance: none;

margin: 0;

}

## textarea

禁止拉伸：resize:none

## select

获取选择的option数值：查找此select元素获取value值

多选：select中添加multiple,用数组接受select value值

## float

作用：将元素设置为外部内联，内部块级的特性

外部内联：元素宽度为内部占据的最小空间宽度

内部块级：可设置宽高等

## Table

Tbody 超出大小滚动显示， 伪固定table头

thead{

display: block;

width: 100%;

background-color: orange;

}

tbody{

display: block;

height: 70px;

background-color: palegoldenrod;

overflow-y: auto;

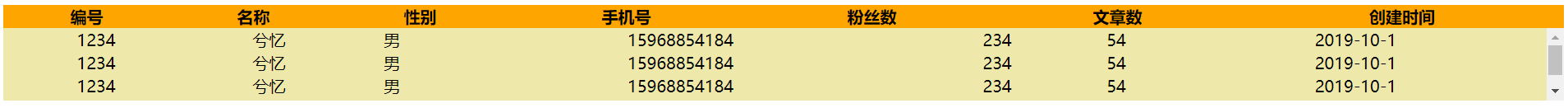
width: 100%;

}

此时table变形，失去display:table效果：

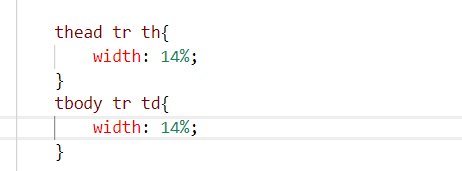


再将thead、tbody中的tr添加display:table可给其恢复各自的table特性：



但由于表头和body中使用了各自的table 自动分配空间，导致分布错位：解决方案：

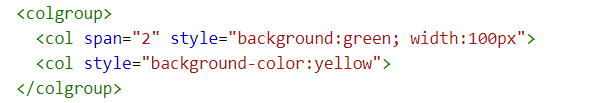
给thead与tbody中每个表格设置宽度，使其上下一致对齐

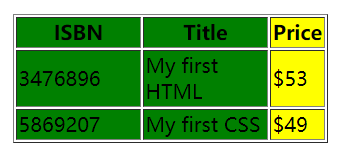




Colgroup：设置整个组（表）样式，目前仅支持背景颜色和宽度

Col：设置列样式 （背景色和宽度） 通常配合使用

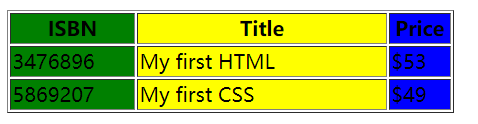




Span：col跨度，默认为1，如span=2, 设置两列。

一个table中仅支持一个colgroup，若出现多个，colgroup按照上下文顺序合并（拼接），按表格列数依次设置样式，若多与表格列数则无效：





## div布局

核心思想：一个盒子难以达成效果，可以套用盒子组合来完成效果

如：单独一个盒子无法使用flex居中时，可以在此盒子外添加有大小的盒子，对外部盒子添加flex属性达到目标盒子居中的效果，**总想着节约标签往往适得其反，事倍功半**

**手机端页面：使用100%高度控制页面满屏显示**

## 弹性布局

#### Flex

Flex : flex-grow | flex-shrink | flex-basis （父元素设置display:flex，并且只对直接子元素起伸缩效果，其中若设置margin、padding，此padding间隔距离无法被压缩）

{放大比例 默认0 | 缩小比例 默认1 | 计算主轴是否有多余空间 默认auto,原本大小}

【flex-grow: 当父元素宽度大于子元素宽度之和，其可让子元素分享父元素多余空间】

【flex-shrink: 当父元素宽度小于子元素宽度之和，其让子元素按比例收缩】

【flex-basis: 设置元素宽度，当元素同时设置width和flex-basis，width值被其覆盖】

【设置align-self可让此元素宽度为原本大小，取消auto中继承stretch 100%】

当无flex-direction: column; 时，此flex无效，设置align-self后同过min-width设置width

例：flex: 0 0 50%;

**注意点：flex-wrap: wrap ：空间不足时折行**

**坑点：折行后align-content默认值：stretch（各行伸屏以占用剩余的空间，若剩**

**余空间不足，等效于flex-start，其他情况下，剩余空间被所有行平分）**

## 盒模型

box-size: border-box; //以自身空间自动计算实际大小

如：father{width:200px} son{width:200px,padding:20px}

此时子元素实际大小为width:240px height:240px;

设置box-size:border-box后, 子元素实际大小：width:200px height:200px;

内部空间：160px,即从自身去除了padding值

总结：未设置时：以自身向外添加 | 减少大小

设置后： 以自身向内添加 | 减少大小

## 栅格布局-bootstrp

使用：类名添加栅格标签： class=”col-md-4”

class：row --让此标签占据整行

class：form-inline --表单水平排列

class：form-group --设置元素为inline-block

class：form-control --设置元素width:100%，阴影效果，颜色:近灰色,border,padding

class：btn btn-default --类似control，添加悬浮变色

## 选择器

#### nth-child与nth-of-type

nth-child: 从父元素内的全部子元素选择子元素，且选择对象要存在

例：p:nth-child(5) ,选择父元素中的第五个元素，若第五个元素不是p，则不起作用

nth-of-type: 从父元素内的选择元素中排列顺序进行选择，且选择对象要存在

例：a:nth-of-type(2): 选择第二个a元素

## 注意项

1. 行内元素无法包含块级元素，块级元素可包含行内元素，同级元素可互相包含：

高等级包含低等级，不可逆

* 1. Inline:1
  2. Inline-block:2
  3. block:3

#### 居中

1. flex居中【

display:flex; //设置为flex（

justify-content:center; //水平方向居中

align-items:center //垂直方向居中

】

2、文字居中【

Text-align: center; //继承 {父元素设置，子元素中的行内元素在

其之内水平居中 ，，只对行内元素有效}

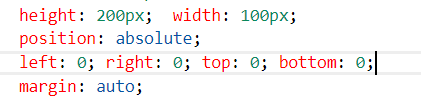
】

1. margin居中【

目标居中元素设置:margin: 0 auto; //水平居中

上下居中使用margin: auto;

//只能绝对定位后left,right,top,bottom=0(相对0位置并且本身与父容器有宽高的情况下再margin:auto时才有效)



】

1. transform:translate居中【

position: absolute;

top:50%;

left:50%;

transform:translate(-50%,-50%)

//优势： 不需知晓本身宽高即可自动居中

//缺点： 兼容性较差，ie10+才支持使用

】

1. table布局居中【

father: { width: 200px; height: 200px; display: table; text-aligin:center;

vertical-align:middel}

son: {width:10px; height:10px; display:inline-block;}

**display:table-cell：将子元素适应vertical-aligin:middle**

**子元素：display:inline-block; 让此元素适应text-aligin:center**

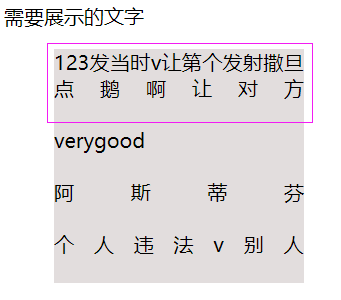
**以此达到：上下居中，左右居中**

】

### 文字两端对齐

父容器设置text-align: justify; --两端对齐

单独设置时只对多行文字的情况生效



单行文字需添加占位标签，目的：添加索引去设置文字宽度100%

【对数字字母无效，空格分开的数字字母按文字效果对齐】

使用标签占位后会导致多出垂直方向的间隔，取消间隔可以设置容器高度方式，如

li{height:30px; line-height:30px}





## 未解疑惑

#### 1、flex

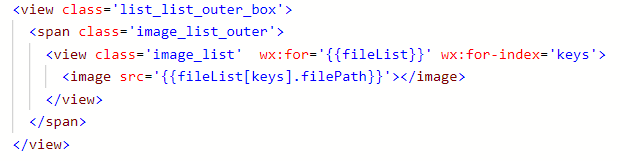
父元素设置display:flex 后，子元素中的元素都变为块级元素，重新设置display:inline无效。

## 已解疑惑

1. flex最后一行依照先前左对齐排列：

给整体元素添加一个盒子，对此盒子再添加对齐方式，盒子中的内容根据想要的方式排序（不成熟方案）







页面加载局部图片图标：

## 结论

1. 避免因padding、margin导致内容超出而出现滚动
   1. 设置父元素overflow:hidden (超出隐藏)
   2. 设置父元素flex，自动分配内容（但是margin与padding依旧占有空间，可能无法均分）
   3. 设置父元素box-sizing:border-box | content-box (默认) | inherit

border-box：怪异模式，border、padding均包含在盒子内

content-box：内容包含在盒子内 （默认值, border、margin、padding均向外撑）

inherit：继承父元素的box-sizing值

兼容：box-sizing:border-box;

-moz-box-sizing:border-box; /\* Firefox \*/

-webkit-box-sizing:border-box; /\* Safari \*/