# 2019.4.20

### <!--代替div的结构元素-->

<section></section> <!-- 内容块，章节，页眉，页脚，页面中其它部分 -->

<article></article> <!-- 上下文不相关的独立内容 -->

<aside></aside> <!-- 与article相关的辅助信息 -->

<header></header> <!-- 整个页面标题，我感觉放banner合适 -->

<footer></footer> <!-- 页面底部 -->

<nav></nav> <!-- 导航栏 -->

<main></main> <!-- 网页主要内容区域 -->

<figure> <!-- 独立的流内容,一般表示主体流内容的一个独立单元 -->

<figcaption>GRC</figcaption>

<p>Tje asd asdsadas asdasd asdasdas</p>

</figure>

### <!--具有布尔类型的属性标签,设定真假方式-->

#### <!--true-->

<input type="checkbox" checked > <!-- 只写属性为真 -->

<input type="checkbox" checked="checked"> <!-- 属性="属性"为真 -->

<input type="checkbox" checked=""> <!-- 属性=""空为真 -->

#### <!--false-->

<input type="checkbox"> <!-- 不写为假 -->

## HTML5新增标签

### <!-- HTML5中新增元素 -->

#### <!-- 1.建立播放视频/音频流文件 -->

<!—video,audio>

<video src="E:/迅雷下载/西虹市首富.mp4" controls style=" width:1440px; height:500px; background:#000; " >test</video> <!—video中controls显示控制栏,src索引视频文件-->

<br />

<audio src="music/GILL.mp3" controls >audio元素</audio><!—音频索引-->

<br />

#### <!-- 2.插入多媒体文件,自动播放多媒体文件,视频无法全屏 -->

<!--embed-->

<embed src="E:/迅雷下载/西虹市首富.mp4" /> <!—可以使用controls属性加入控制栏 -->

#### <!-- 3.突显高亮文字,mark代替span -->

<!--mark-->

<mark>我卡啊</mark>

<br />

#### <!-- 4.显示javascript耗费时间的函数进程 -->

<!--progress-->

<progress>

<script type="text/javascript"> var a,b; a=0; b=1; a+b; </script>

</progress>

<br />

#### <!-- 5.规定数值范围 -->

<!--meter-->

<meter><p style="width:200px; height:100px;" ></p></meter>

<br />

#### <!-- 6.时间 -->

<!—time-->

<time></time>

<br />

#### <!-- 7.ruby创建有发音符的字符串,rt发音 -->

<!—ruby,rt,rp-->

<ruby>你<rt>1</rt></ruby>

<rp></rp> <!-- 放在ruby里,定义不支持ruby的浏览器所显示的内容 -->

<br />

#### <!-- 8.软换行,宽度足够时不换行，宽度不够时进行换行,对中文用处不大 -->

<!--wbr-->

<p style="width:50px;" >

123213123<wbr>asdasdasdasd

</p>

<br />

#### <!-- 9.canvas本身没有行为,给javascript提供一片画布，进行绘制图形 -->

<!--canvas-->

<canvas id="myCanvas" width="200" height="100px" >1</canvas>

<br />

#### <!-- 10.建立单选复选框，目前大多浏览器不支持,暂且不谈 -->

<command></command>

<br />

#### <!-- 11.折叠显示信息 -->

<!—details,summary-->

<details>

<summary>HTML5</summary> <!-- 标题 -->

ASDASDASDASDASDSADASDASDASDASDAS ASDASDASDSA ASDASD . <!-- 折叠信息 -->

</details>

<br />

#### <!-- 12.内容框,下拉选项创建 -->

<!—input,datalist,option-->

<form>

<input list="abc" name="abc">

<datalist id="abc" > <!-- 创建下拉选项 -->

<option value="1"> <!-- 选项内容 -->

<option value="2">

</datalist>

<input type="submit">

</form>

<br />

#### <!-- 13.HTML5中已经删除 -->

<datagrid>

　　<ol>

　　　　<li> (datagrid row 0) </li>

　　　　<li> (datagrid row 1)

　　　　　　<ol style="list-style-type:lower-alpha;">

　　　　　　　　<li> (datagrid row 1,0) </li>

　　　　　　　　<li> (datagrid row 1,1) </li>

　　　　　　</ol>

　　　　</li>

　　　　<li> (datagrid row 2) </li>

　　</ol>

</datagrid>

<br />

#### <!-- 14.keygen生成密码,HTML5中已经删除 -->

<keygen>

<br />

#### <!-- 15.output输出计算结果 -->

<!--<form oninput=””>,output-->

<form oninput=" x.value = parseInt(a.value) \* parseInt(b.value) " > <!-- oninput与output联用,parseInt()转换为整数类型,id.value能索引出目标的value值 -->

<input id="a" type="range" value="10" >

<input id="b" type="number" value="10" style="width:50px;" > =

<output name="x" for="a b"></output> <!-- 输出x值,for属性描述计算关系,用空格隔开 -->

</form>

<br />

#### <!-- 16.source设置多个播放源,可以使用到流媒体播放标签,如: audio,video,embed -->

<!-- source -->

<audio controls>

<source src="music/GILL.ogg" type="audio/ogg">

<source src="music/GILL.mp3" type="audio/mp3">

</audio>

<br />

#### <!-- 17.已经删除 -->

<menu>

<li><input type="checkbox">Red</li>

<li><input type="checkbox">Blue</li>

</menu>

<br />

#### <!-- 18.dialog弹窗交互,目前只要Chrome和Safari支持 -->

<!--dialog-->

<dialog open style=" width:90%; height:70%; " > <!-- open属性表示显示弹窗,默认不显示 -->

<div style="width:100px; height:100px; background:#000;" ></div>

</dialog>

<div style="width:100px; height:100px; background:red;" ></div> <!-- dialog弹窗不影响排序 -->

# 2019.4.21

<form>

## <!-- 新增input元素 -->

**<input type="email">**

**<input type="url">**

**<input type="number">**

**<input type="range">**

**<input type="Date Pickers">**

**<br/>**

### <!-- 新增属性 -->

**<input type="text" autofocus> <!-- autofocus打开页面自动获取焦点 --> <br/>**

**<input type="text" placeholder="请输入数字"> <!-- placeholder在文本框显示提示字符串 --> <br/>**

#### <!-- fieldset指向表单，fieldset内容也属于指向的表单，大部分浏览器不支持该属性 -->

**<form id="name1">**

**<input type="text">**

**</form>**

**<br/>**

**<fieldset form="name1">**

**<input type="text">**

**</fieldset>**

#### <!-- required强制填写属性 -->

**<br/>**

**<form action="">**

**<input type="text" required> <!—如果不填写,则无法进行提交表单 -->**

**<textarea rows="4" cols="50" name="comment" required> </textarea><!—文本框-->**

**<input type="submit">**

**</form>**

## <p> ----------- </p>

## <!-- input详细属性解析 -->

### <!--type属性-->

#### <!-- 按钮类型 -->

**<input type="button"> <br/> <!-- 定义可单击按钮 -->**

**<input type="chekbox"> <br/> <!-- 复选框 -->**

**<input type="file"> <br/> <!-- 上传文件 -->**

**<!--**

**accept: 规定上传文件类型**

**accept="audio/\*" 音频**

**accept="image/\*" 图像**

**accept="video/\*" 视频**

**accept="MIMI\_type"**

**multiple: 设定允许多个元素上传**

**-->**

**<input type="image" alt="123"> <br/> <!-- 定义图像为提交按钮 -->**

**<!--**

**alt="无法加载图片时显示"**

**src="索引图像位置"**

**width="设定宽度"**

**-->**

**<input type="radio"> <br/> <!-- 单选框 -->**

**<input type="range"> <br/> <!-- 滑动条控制数字 -->**

**<input type="reset"> <br/> <!-- 重置按钮 -->**

**<input type="submit"> <br/> <!-- 提交按钮 -->**

#### <!-- 时间类型 -->

**<input type="datetime"> <br/> <!-- 年月日时间,基于时区 -->**

**<input type="date"> <br/> <!-- 年月日 -->**

**<input type="datetime-local"> <br/> <!-- 年月日时间,基于本地 -->**

**<input type="mouth"> <br/>**

**<input type="time"> <br/> <!-- 定义输入时间 -->**

**<input type="week"> <br/> <!-- 定义周期和年 -->**

#### <!-- 输入类型 -->

**<input type="email"> <br/>**

**<input type="number"> <br/> <!-- 定义输入数字字段 -->**

**<input type="password"> <br/> <!-- 密码 -->**

**<input type="search"> <br/> <!-- 搜索字符串 -->**

**<input type="tel"> <br/> <!-- 定义电话号码输入的字段 -->**

**<input type="text"> <br/> <!-- 定义单行文本字段 -->**

**<input type="url"> <br/> <!-- 输入链接 -->**

#### <!-- 其它类型 -->

**<input type="hidden"> <br/> <!-- 定义隐藏类型，浏览器不渲染，且不占用布局空间 -->**

**<input type="color"> <br/> <!-- 拾色器 -->**

### <!-- input适用多属性 -->

<!--

#### 1. autocomplete自动补全开启/关闭

**适用: 输入的type类型**

**autocomplete="off"/"on"**

#### 2. autofocus打开页面直接获取焦点

适用: 所有input类型

#### 3. checked预选

适用: type="checkbox"/"radio"

#### 4. disabled禁用input属性

适用: 所有input类型

#### 5. form指定表单id，设定input元素归属

适用: 所有input类型

大多浏览器不支持

#### 6. formaction设定表单提交链接

**formmethod设定提交方法( 会覆盖form的method属性值 )**

**formenctype设定传输数据编码形式**

**适用: type="submit"/"image"**

**formaction="URL": 注意formaction会覆盖掉<form action="URL">action属性值**

**formmethod="get"/"post" 二种提交表单时的传输方式推荐使用post**

#### 7. formnovalidate不验证提交

**适用: type="submit"/"image"**

**<form>**

**<input type="email">**

**<input type="submit"> 验证提交**

**<input type="submit" formNoValidate> 不验证提交**

**</form>**

#### 8. list属性,与datalist配合形成下拉选项

**适用: 输入类型属性**

**<input list="name1">**

**<datalist id="name1">**

**<option value="1">**

**<option value="2">**

**</datalist>**

#### 9. min/max限定数值大小

适用: 时间类型,数子类型

<input type="range" min="1" max="5">

#### 10. maxlength规定字符串长度

适用: 输入类型

<input type="text" maxlength="10"> 设定输入字符串数量不超过10个

#### 11. name设定input元素名称

**有name可以在javascript中引用input值,并且可以提交表单**

**如果没有name则input内容不会被提交**

**<form action="xx.php" method="post">**

**<input type="text" name="name1">**

**<input type="submit">**

**</form>**

-->

<br/>

#### <!-- 12. pattern:内容进行正则验证 -->

**<!-- 适用: 输入类型 -->**

**<!-- title全局属性,当鼠标移动到元素标签时，进行信息提示 -->**

**<form action="xx.php" method="post">**

**<input type="text" required name="test1" pattern="[A-Za-z]{5}" title="5个字母" >**

**<input type="submit">**

**</form>**

**<br/>**

#### <!-- 13. placeholder: 在文本框内显示提示字符串 -->

**<!-- 适用: 输入类型-->**

**<form action="xx.php" method="post" >**

**<input type="text" required placeholder="TEST\_233" >**

**<input type="submit">**

**</form>**

<!--

#### 14. readonly: 设定字段为只读状态

适用: 字段类型的input

#### 15. size: 设置input宽度

适用: 大部分input类型

#### 16. step: 设置数字间隔

适用: 输入类型

<input type="text" step="3"> step为3则能输入的为-3,0,3,6依次类推

-->

<br/>

#### <!-- 17. 关于value属性 -->

**<form action="xx.php" method="post">**

**<!-- value使用在按钮类型，按钮显示的文本如: button,submit,reset -->**

**<input type="button" value="111" >**

**<!-- value使用在输入类型上,为默认值如: text,password,hidden -->**

**<input type="text" value="233">**

**<!-- value在checkbox,radio,image的值会通过action发送到URL中 -->**

**<!-- 注意,checkbox,radio是必须写入value属性的 -->**

**<input type="checkbox" value="男">**

**<input type="radio" value="女">**

**</form>**

<br/>

#### <!-- 关于form属性 -->

<form action="xx.php" method="post"></form> <!-- action="表单提交目标"，method="提交方式" -->

### <!-- 关于a标签与属性 -->

**<!--**

**href="链接的URL"**

**download="指定下载文件路径"**

**rel="写明链接到的URL与本页的关系,方便搜索引擎录取"**

**target="打开方式"**

**media="规定链接的URL媒体类型"**

**type="规定URL,MIME类型"**

**-->**

#### <!-- 下载图片链接 -->

<a href="images/1.jpg" download="images/1.jpg" > image\_download</a>

<br/>

#### <!-- 测试target -->

**<!--**

**\_blank: 新窗口打开**

**\_self: 同窗口打开**

**\_parent/\_top: 不太了解,发现于\_self没有区别**

**-->**

**<a href="http://www.oo7.fun" target="\_blank">塔式甜点\_blank</a>**

**<a href="http://www.oo7.fun" target="\_self">塔式甜点\_self</a>**

**<a href="http://www.oo7.fun" target="\_parent">塔式甜点\_parent</a>**

**<a href="http://www.oo7.fun" target="\_top">塔式甜点\_top</a>**

**111**

## **<!-- 全局属性-任何元素都可以使用的属性 -->**

#### **<!-- 1. contentEditable="true/false"变为可编译模式 -->**

**<!--**

**他可以使标签变为直接可以编辑的模式，但是并没有保存功能**

**并且他具有,继承性**

**-->**

**<p contentEditable="true" >**

**啊实打实的**

**</p>**

#### **<!-- 2. designMode="on/off" 使全页面开启/关闭可编辑模式 -->**

**<!--**

**注意: 只能在js中使用**

**document.designMode="on/off"**

**-->**

**<script type="text/javascript"> document.designMode="off" // 开启可编辑模式 </script>**

**<p> 1111 </p>**

#### **<!-- 3. hidden隐藏标签-->**

**<!--**

**hidden告诉浏览器不进行渲染此标签,在未设定显示时,不占用布局空间**

**通常可以配合js来显示或者不显示**

**-->**

**<p hidden > 隐藏起来 </p>**

**<p> 正常显示 </p>**

#### **<!-- 4. spellcheck专门为input的text,textarea类型设定的,拼写语法错误检查 -->**

**<textarea rows="10" cols="10" spellcheck="true" ></textarea>**

**经测试在chrome不行**

#### **<!-- 5. tabindex获取焦点 -->**

**<!--**

**tabindex="顺序数字",在使用tab键逐个切换焦点顺序**

**通常情况下:**

**在HTML5中tabindex可以在任何标签中使用,并使标签为焦点**

**tabindex="-1": tab键将无法锁定,但是可以通过javascript的focus()语句来控制焦点锁定**

**-->**

**<br/>**

**<input type="text" tabindex="1" >**

**<p tabindex="2" >tabindex\_test</p>**

# **2019.4.26**

### text-decoration控制字体下划线

Text-decoration:none; <!--去除下划线-->

#### Font-weight控制字体粗细

Font-weight:lighter; <!—特别细-->

### CSS3检测屏幕大小变换更换样式

// 注意空格,一定要按照标准写

//当检测到屏幕宽度小于800分辨率时,启动样式

@media screen and (max-device-width:800px){

nav ul li a{

font-size:20px;

font-weight:bold;

}

}

### 设置标签最小宽度

Min-width：620px； // 最小宽度为620px

## Flex弹性布局

**<!doctype html>**

**<html>**

**<head>**

**<title>test</title>**

**<meta charset="utf-8">**

**<!--**

### **Flex布局研究总结( 弹性布局 ):**

#### **1. 开启弹性布局 ( display;flex; | display:-webkit-flex; )**

**display:flex;**

**display:-webkit-flex;**

**display:inline-flex;**

#### **2. 控制x/y排序方向 ( flex-direction:; | -webkit-flex-direction:; )**

**flex-direction:row | row-reverse | column | column-reverse;**

#### **3. x对齐 ( justif-content:; | -webkit-justif-content:; )**

**justify-content:flex-start | flex-end | flex-center | space-between | space-around;**

#### **4. y对齐 ( align-items:; | -webkit-align-items:; )**

**align-items:flex-start | flex-end | center | space-between | space-around | stretch;**

#### **5. 换行模式 ( flex-wrap:; | -webkit-flex-wrap:; )**

**flex-wrap:nowrap | wrap | wrap-reverse;**

#### **6. 换行排序 ( align-content:; | -webkit-align-content:; )**

**align-content:flex-start | flex-end | center | space-between | space-around | stretch;**

#### **7. 自定义排序顺序( order )**

**order:number;**

**-webkit-order:number;**

#### **8. 完美剧中( flex弹性布局下的auto作用 )**

**margin:auto;**

**核心属性: flex模式下auto会沾满容器**

#### **9. y对齐单独自身调控 ( align-self:; | -webkit-align-self:; )**

**align-self: flex-start | flex-end | center | space-between | space-around ;**

#### **10. 常用属性 ( initial | inherit )**

**initial: 默认值**

**inherit: 继承父类属性**

#### **11. 等比放大 ( flex-grow:; | -webkit-flex-grow:; )**

**flex-grow: number | initial | inherit ;( 默认为0 )**

#### **12. 等比缩小 ( flex-shrink:; | -webkit-flex-shrink:; )**

**flex-shrink:number;( 默认为1 )**

#### **13. 固定宽度 ( flex-basis:; | -webkit-flex-basis:; )**

**flex-basis: numberPX | inherit ;**

#### **14. 便捷式-控制缩放( flex:; | -webkit-flex:; )**

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

**-->**

**<style type="text/css" >**

**body{**

**margin:0px;**

**padding:0px;**

**}**

**/\*\***

### **0. 行内元素开启弹性布局:**

**.box{**

**display:inline-flex;**

**}**

**\*\*/**

#### **/\*\* 1. 块级元素开启弹性布局 \*\*/**

**.flexTest{**

**display:flex; /\*\* 开启弹性布局 \*\*/**

**display:-webkit-flex; /\*\* 开启弹性布局,兼任其它浏览器 \*\*/**

**/\*\*direction:rtl; 使内部元素反方向排序\*\*/**

#### **/\*\* 2. flex-direction属性,控制x/y排序方向 \*\*/**

**flex-direction:row;**

**/\*\***

**flex-direction:row | row-reverse | column | column-reverse ;**

**row: 默认重左到右**

**row-reverse: 反方向**

**column: 垂直方向排序**

**column-reverse: 反方向**

**\*\*/**

#### **/\*\* 3. justify-content沿主轴线(main-aix,水平轴)水平对齐 \*\*/**

**justify-content:space-around;**

**-webkit-justify-content:space-around; /\*\* 为兼容性 \*\*/**

**/\*\***

**justify-content: flex-start | flex-end | center | space-between | space-around ;**

**flex-start: 默认重左排序**

**flex-end: 重右侧排序**

**center: 剧中排序**

**space-between:平均分布排序,左右二边不留空隙;**

**space-around:平均分布排序,左右二边留出空袭;**

**\*\*/**

#### **/\*\* 4. align-items属性,控制垂直排序方向\*\*/**

**/\*\***

**align-items: flex-start | flex-end | center | baseline | stretch ;**

**flex-start: 顶部对齐**

**flex-end: 底部对齐**

**center: 上下剧中对齐**

**baseline: 基于第一行文字对齐**

**stretch: 如果高度为auto,则沾满容器高度,相当于height:100%;**

**\*\*/**

**align-items:baseline; /\*\* 基于第一行文字对齐 \*\*/**

**-webkit-align-items:baseline; /\*\* 为兼容性 \*\*/**

**width:1200px;**

**height:500px;**

**background:pink;**

**}**

**.pic{**

**width:100px;**

**height:100px;**

**margin:10px;**

**background:green;**

**}**

**.flexTest2{**

**width:1200px;**

**height:500px;**

**background:#000;**

**display:flex;**

**display:-webkit-flex-;**

#### /\*\* 5. flex-wrap控制换行模式\*\*/

**/\*\***

**flex-wrap: nowrap | wrap | wrap-reverse ;**

**nowrap: 不换行,大于容器会溢出**

**wrap: 自动换行**

**wrap-reverse: 反转换行,这个挺有意思的**

**\*\*/**

**flex-wrap:wrap; /\*\* 自动换行 \*\*/**

**-webkit-flex-wrap:wrap; /\*\* 为兼容性 \*\*/**

#### **/\*\*6. align-content控制flex-wrap行为,控制"多行垂直"的排布 \*\*/**

**/\*\***

**aligin-content:stretch | flex-start | flex-end | space-between | space-around;**

**stretch: 高度设置为auto时,将沾满容器的高度;**

**flex-start: 上方起始位置堆叠**

**flex-end: 下方底部位置开始堆叠**

**center: 高度中间位置开始堆叠**

**space-around: 上下二边有空隙( 空隙大小,为子元素之间高度空隙的一半 )，平均发布**

**space-between: 没有空隙的平均分布**

**\*\*/**

**align-content:flex-start; /\*\* 上方起始开始堆叠 \*\*/**

**-webkit-align-content:flex-start; /\*\* 为兼容性 \*\*/**

**}**

**.pic2{**

**width:300px;**

**height:100px;**

**background:red;**

**margin:10px;**

**}**

#### **/\*\*7. order设置元素排序顺序,小值在前,大值在后 \*\*/**

**/\*\***

**默认排序初始值,为0**

**order也可以设置负数**

**\*\*/**

**.firstTest{**

**order:-1; /\*\* 设置排序顺序 \*\*/**

**-webkit-order:-1; /\*\* 为兼容性 \*\*/**

**}**

#### **/\*\* 8. 一行代码搞定完美剧中,及底层原理解析 \*\*/**

**/\*\***

**在flex模式下"margin:auto;"直接使元素剧中**

**因为flex状态下,auto会自动沾满容器**

**\*\*/**

**.preCenter{**

**width:1200px;**

**height:500px;**

**background:pink;**

**display:flex;**

**-webkit-display:flex;**

**}**

**.pre{**

**width:500px;**

**height:100px;**

**background:#fff;**

**margin:auto; /\*\* 核心代码,完美剧中 \*\*/**

**}**

#### **/\*\*设置自身排序,控制语句将放在子元素中\*\*/**

**.flexTest3{**

**width:1200px;**

**height:500px;**

**background:yellow;**

**display:flex;**

**-webkit-display:flex;**

**align-items:center;**

**-webkit-align-items:center;**

**}**

**.pic3{**

**width:300px;**

**height:300px;**

**background:red;**

**margin:10px;**

**}**

#### **/\*\* 9. align-self设置自身垂直排序 \*\*/**

**/\*\***

**align-self:flex-start | flex-end | center | baseline | stretch;**

**flex-start: y轴顶部**

**flex-end: y轴底部**

**center: y轴中间**

**baseline: y轴方向文字对齐,默认值其实为flex-start**

**stretch: y轴方向沾满容器**

**\*\*/**

**.selfTest{**

**align-self:flex-start;**

**-webkit-align-self:flex-start;**

**}**

**</style>**

**</head>**

**<body>**

**<div class="flexTest" >**

**<div class="pic" ><p style="margin-top:30px;" >1</p></div>**

**<div class="pic" >2</div>**

**<div class="pic" >3</div>**

**<div class="pic" >4</div>**

**</div>**

**<div class="flexTest2" >**

**<div class="pic2">1</div>**

**<div class="pic2">2</div>**

### **<!-- 多个class简洁写法 -->**

**<!--**

**.firstTest{**

**即可调用**

**}**

**-->**

**<div class="pic2 firstTest">3</div>**

**<div class="pic2">4</div>**

**<input class="pic2" style="border:0px ;" type="text"> <!-- 测试行内元素排序 -->**

**</div>**

**<div class="preCenter" >**

**<div class="pre" >111</div>**

**</div>**

**<div class="flexTest3" >**

**<div class="pic3 selfTest"></div>**

**<div class="pic3"></div>**

**</div>**

**</body>**

**</html>**

**<!doctype html>**

**<html>**

**<head>**

**<title>test</title>**

**<meta charset="utf-8">**

**<style type="text/css">**

**.flexTest{**

**width:1200px;**

**height:500px;**

**display:flex;**

**display:-webkit-flex;**

**background:pink;**

**}**

**.items{**

**flex-basis:500px;**

**height:200px;**

**background:#000;**

**margin:10px;**

**}**

#### **/\*\*10. 常用的属性值解析 \*\*/**

**/\*\***

**initial( 默认值的意思 )**

**inherit( 继承指定父类的属性,**

**如父元素文字大小为font-size:15px;**

**子元素可以直接font-size:inherit;**

**)**

**\*\*/**

#### **/\*\*11. flex-grow: 等比放大,值至占满容器**

**number: 等比放大**

**initial: 默认值( 默认值为0 )**

**inherit: 继承父类属性设置**

**.it1{**

**flex-grow:1;**

**}**

**.it2{**

**flex-grow:2;**

**}**

**.it3{**

**flex-grow:1;**

**}**

**.it4{**

**flex-grow:1;**

**}**

**\*\*/**

#### **/\*\*12. flex-basis设置固定值宽度大小\*\*/**

**/\*\***

**flex-basis:固定宽度值,但是依然受flex-grow/flex-shrink影响;**

### **分类指定选择: div:nth-of-type(2){}**

**选择第二个DIV标签**

**.flexTest div:nth-of-type(2){**

**flex-basis:300px;**

**}**

**\*\*/**

#### **/\*\*13. flex-shrink空间不足时进行项目缩小 \*\*/**

**/\*\***

**flex-shrink默认值为1**

**当容器变小时,子元素也会跟着缩小**

**子元素,flex-shrink设置的值越大则缩小值越大( 指宽度 )**

**\*\*/**

**.it1{**

**flex-shrink:1;**

**}**

**.it2{**

**flex-shrink:1;**

**}**

**.it3{**

**flex-shrink:3;**

**}**

**.it4{**

**flex-shrink:3;**

**}**

#### **/\*\*14. flex便捷式控制缩放问题 \*\*/**

**/\*\***

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

**flex:auto; == flex:1 1 auto;**

**flex:none: == flex:0 0 auto;**

**flex:initial; == flex:0 1 auto; ( initial默认值的意思 )**

**flex:inherit;( 非常好用 )**

**\*\*/**

**</style>**

**</head>**

**<body>**

**<div class="flexTest">**

**<div class="items it1">1</div>**

**<div class="items it2">2</div>**

**<div class="items it3">3</div>**

**<div class="items it4">4</div>**

**</div>**

**</body>**

**</html**