**我的方法：**

//div居中样式(margin-left:-250px的作用是让div居中)

<div style="width: 500px;height: 100px;background: blue;position: absolute;left: 50%;margin-left: -250px;"></div>

**方法一：**

该方法是将外部容器的显示模式设置成display:table，img标签外部再嵌套一个span标签，并设置span的显示模式为display:table-cell，这样就可以很方便的使用vertical-align象表格元素那样对齐了，当然这只是在标准浏览器下，IE6/IE7还得使用定位。  
**HTML代码**

|  |  |
| --- | --- |
| 1  2  3 | <div id="box">  <span><img src="images/demo.jpg" alt="" /></span>  </div> |

**CSS代码**

[复制代码](javascript:void(0);)

<style type="text/css">

#box{

width:500px;height:400px;

display:table;

text-align:center;

border:1px solid #d3d3d3;background:#fff;

}

#box span{

display:table-cell;

vertical-align:middle;

}

#box img{

border:1px solid #ccc;

}

</style>

<!--[if lte IE 7]>

<style type="text/css">

#box{

position:relative;

overflow:hidden;

}

#box span{

position:absolute;

left:50%;top:50%;

}

#box img{

position:relative;

left:-50%;top:-50%;

}

</style>

<![endif]-->

[复制代码](javascript:void(0);)

**方法二：**

方法二和方法一的实现的原理大同小异，结构也是相同的，方法一用的是条件注释，方法二就用的CSS Hack。  
**CSS代码**

[复制代码](javascript:void(0);)

<style type="text/css">

#box{

width:500px;height:400px;

overflow:hidden;

position:relative;

display:table-cell;

text-align:center;

vertical-align:middle;

border:1px solid #d3d3d3;background:#fff;

}

#box span{

position:static;

\*position:absolute; /\*针对IE6/7的Hack\*/

top:50%; /\*针对IE6/7的Hack\*/

}

#box img {

position:static;

\*position:relative; /\*针对IE6/7的Hack\*/

top:-50%;left:-50%; /\*针对IE6/7的Hack\*/

border:1px solid #ccc;

}

</style>

[复制代码](javascript:void(0);)

该方法有个弊端，在标准浏览器下由于外部容器#box的显示模式为display:table-cell，所以导致#box无法使用margin属性，并且在IE8下设置边框也无效。

**方法三：**

标准浏览器还是将外部容器#box的显示模式设置为display:table-cell，IE6/IE7是利用在img标签的前面插入一对空标签的办法。

**HTML代码**

|  |  |
| --- | --- |
| 1  2  3 | <div id="box">      <i></i><img src="images/demo.jpg" alt="" />  </div> |

**CSS代码**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24 | <style type="text/css">  #box{  width:500px;height:400px;  display:table-cell;  text-align:center;  vertical-align:middle;  border:1px solid #d3d3d3;background:#fff;  }  #box img{  border:1px solid #ccc;  }  </style>  <!--[if IE]>  <style type="text/css">  #box i {      display:inline-block;      height:100%;      vertical-align:middle      }  #box img {      vertical-align:middle      }  </style>  <![endif]--> |

**方法四：**

在img标签外包裹一个p标签，标准浏览器利用p标签的伪类属性:before来实现，IE6/IE7使用了CSS表达式来实现兼容。  
**HTML代码**

<div id="box">

<p><img src="images/demo.jpg" alt="" /></p>

</div>

**CSS代码**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22 | #box{      width:500px;height:400px;      text-align:center;      border:1px solid #d3d3d3;background:#fff;  }  #box p{      width:500px;height:400px;      line-height:400px; /\* 行高等于高度 \*/  }    /\* 兼容标准浏览器 \*/  #box p:before{      content:"."; /\* <a href="http://casinogreece.gr/">????????????</a> 具体的值与垂直居中无关，尽可能的节省字符 \*/      margin-left:-5px; font-size:10px; /\* 修复居中的小BUG \*/      visibility:hidden; /\*设置成隐藏元素\*/  }    #box p img{      \*margin-top:expression((400 - this.height )/2); /\* CSS表达式用来兼容IE6/IE7 \*/      vertical-align:middle;      border:1px solid #ccc;  } |

使用:beforr这个方法对于标准浏览器来说比较给力，也没发现有副作用，但是对于IE6/IE7，如果对性能要求较高的话CSS表达式的方法要慎用。

**方法五：**

该方法针对IE6/IE7，将图片外部容器的字体大小设置成高度的0.873倍就可以实现居中，标准浏览器还是使用上面的方法来实现兼容，并且结构也是比较优雅。  
**HTML代码**

<div id="box">

<img src="images/demo.jpg" alt="" />

</div>

**CSS代码**

[复制代码](javascript:void(0);)

#box{

width:500px;height:400px;

text-align:center;

border:1px solid #d3d3d3;background:#fff;

/\* 兼容标准浏览器 \*/

display: table-cell;

vertical-align:middle;

/\* 兼容IE6/IE7 \*/

\*display:block;

\*font-size:349px; /\* 字体大小约为容器高度的0.873倍 400\*0.873 = 349 \*/

\*font-family:Arial; /\* 防止非utf-8引起的hack失效问题，如gbk编码 \*/

}

#box img{

vertical-align:middle;

}

[复制代码](javascript:void(0);)

设置字体大小的方法感觉比较怪异，也没有看到一个合理的解释，只知道图片元素有一些不同于其他元素的特性，但是对于IE6/IE7来说，这个方法还是比较给力的。  
思考：很多方法都是依赖于将外部容器的显示模式设置成table才能实现垂直居中，也就是div来模拟table。