样式化复选框(Styling Checkbox)

时间 2014-10-13 13:16:00  [博客园精华区](http://www.tuicool.com/sites/bAzeYv)

原文  [http://www.cnblogs.com/xyzhanjiang/p/3989836.html](http://www.cnblogs.com/xyzhanjiang/p/3989836.html?utm_source=tuicool&utm_medium=referral)

主题 [CSS](http://www.tuicool.com/topics/11060002)

复选框 Checkbox 是 Web 应用常用控件，随处可见，原生的复选框控件一般就像下面这样：

这取决于操作系统和浏览器，有些时候，这种样子并不能满足设计要求，这时需要更为精致的复选框样式。以往只有少数浏览器才支持对这类控件应用样式，比如拿到这样一张设计图：

blue.png



在过去，想要通过简单地修改样式达成这种设计效果根本不行，不过，现在借助强大的 CSS3 属性 appearance 可以对该类控件有空前的样式控制能力：

**input**[type="checkbox"] {

**-webkit-appearance**: none;

}

这样设置该属性值为 **none** 就去掉了复选框原有的呈现方式，变成了一个普普通通的元素，然后就可以为之应用样式了，添加如下样式：

**input**[type="checkbox"] {

**-webkit-appearance**: none;

**background**: #fff url(i/blue.png);

**height**: 22px;

**vertical-align**: middle;

**width**: 22px;

}

通过结合使用状态伪类选择器 **:checked** 可以为选中状态下的 checkbox 设置不同的样式，用以从视觉上区别：

**input**[type="checkbox"]:checked {

**background-position**: -48px 0;

}

此时点击复选框，可以看到复选框样式的变化效果，另外，根据那张设计图片所示还得加上获取焦点，禁用等状态的样式：

**input**[type="checkbox"]:focus,

**input**[type="checkbox"]:hover {

**background-position**: -24px 0;

**outline**: none;

}

**input**[type="checkbox"]:checked {

**background-position**: -48px 0;

}

**input**[type="checkbox"][disabled] {

**background-position**: -72px 0;

}

**input**[type="checkbox"][disabled]:checked {

**background-position**: -96px 0;

}

因为图片已经事先合并成一张了，简单修改一下 **background-position** 就可以了，同时前面几个选择器的优先级（权重）一样，所以书写顺序很重要。

**兼容性**

目前只兼容 Webkit 系列浏览器；虽然 Firefox 也实现了替代的 **-moz-appearance** 属性，不过控件原有的背景颜色、边框样式无法修改，暂时也不大好用；IE 系列暂时不支持该属性，更详细的兼容情况查看 [***Caniuse/appearance***](http://caniuse.com/#feat=css-appearance) 。因此需要为 IE 浏览器清除掉背景图片的影响：

**input**[type="checkbox"] {

**background**: #fff url(i/blue.png);

**background**: none\0;

\***background**: none;

...

}

为了兼容更多的主流浏览器，需要寻求另外的解决方案，在所有主流浏览器里，点击关联某个复选框的 label 时，产生的效果和点击元素自身相同，会切换复选框控件的选中状态。浏览器的这种行为给了我们一个至关重要的挂钩，既然能依靠 label 元素来控制原生复选框的状态，那么就可以不必直接操作实际的复选框元素，而把操作和样式都转移到与之关联的 label 元素上去：

<input id="example" **type**="checkbox">

<label **for**="example"></label>

确保 label 元素的 **for** 属性的值和复选框 input 的 **id** 值一致，同时将 label 元素放置于 input 之后，这样 CSS 可以通过相邻兄弟选择器（Adjacent sibling selector）定位到这个 label 元素并为之应用样式：

**input**[type="checkbox"] + **label**:before {

**background**: #fff url(i/blue.png);

**content**: " ";

**height**: 22px;

**left**: 0;

**position**: absolute;

**width**: 22px;

}

有了样式化的 label 元素来提供交互，原生的 checkbox 控件就显得有点多余了，虽然可以用 **display: none** 把它隐藏掉，不过隐藏后的表单元素是不能获得焦点的，所以最好的方式还是用 label 元素把它遮住，这样既能支持键盘交互，同时当图片加载失败的时候，又能保证原生的控件可用：

**input**[type="checkbox"] {

**box-sizing**: border-box;

**left**: 4px;

**margin**: 0;

**padding**: 0;

**position**: absolute;

**top**: 3px;

}

图片要足够大能将原生的 checkbox 控件完全遮挡住，因为这里用到了绝对定位，所以需要增加一个定位参照：

<!-- HTML -->

<**div** class="checkbox">

<**input** id="exampleCheckbox" type="checkbox">

<**label** for="exampleCheckbox"></**label**>

</**div**>

/\* CSS \*/

.checkbox {

min-height: 24px;

padding-left: 24px;

position: relative;

}

左边预留内边距是为了排版更美观，同时，和之前一样，搭配上其它状态下的样式：

**input**[type="checkbox"]:focus + **label**:before,

**input**[type="checkbox"] + **label**:hover:before {

**background-position**: -24px 0;

}

**input**[type="checkbox"]:checked + **label**:before {

**background-position**: -48px 0;

}

**input**[type="checkbox"][disabled] + **label**:before {

**background-position**: -72px 0;

}

**input**[type="checkbox"][disabled]:checked + **label**:before {

**background-position**: -96px 0;

}

**兼容性**

只要支持 [***CSS3 selectors***](http://caniuse.com/#feat=css-sel3) 的浏览器基本上都能兼容，同时具备原生控件的绝大多数交互特性。IE 8 不支持 **:checked** 伪类选择器，将伪元素 **:before** 修改为双冒号 **::before** 可以去掉对 IE 8 的影响：

**input**[**type**="checkbox"] + label::**before** { ... }

关于伪元素生成内容的兼容性见 [***CSS Generated content for pseudo-elements***](http://caniuse.com/#feat=css-gencontent) 。诚然，上面的方法假设了支持 **:checked** 伪类选择器的浏览器同时也支持双冒号伪元素写法，而不支持的浏览器则都不支持，这是一种不太好的方式，这种假设事实上也是不正确的，造成了部分老旧浏览器不可用的问题，如果使用 **:not()** 选择器则更为合理，使用 **:not(:checked)** 来为未选中的控件添加样式， **:not()** 和 **:checked** 同属一个规范 [***css3-selectors***](http://www.w3.org/TR/css3-selectors/) ，兼容性应该一致 [***CSS3 selectors***](http://caniuse.com/#feat=css-sel3) 。不过写法有点变化， **:checked** 和 **:not(:checked)** 都需要添加上基本的样式：

**input**[type="checkbox"]:checked + **label**:before,

**input**[type="checkbox"]:not(:checked) + **label**:before {

**background**: #fff url(i/blue.png);

**content**: " ";

**height**: 22px;

**left**: 0;

**position**: absolute;

**width**: 22px;

}

**input**[type="checkbox"]:not(:checked):focus + **label**:before,

**input**[type="checkbox"]:not(:checked) + **label**:hover:before {

**background-position**: -24px 0;

}

**input**[type="checkbox"]:checked + **label**:before {

**background-position**: -48px 0;

}

**input**[type="checkbox"][disabled]:not(:checked) + **label**:before {

**background-position**: -72px 0;

}

**input**[type="checkbox"][disabled]:checked + **label**:before {

**background-position**: -96px 0;

}

[***查看简单示例***](http://jsfiddle.net/xey8r08a/) ，对于那些并不支持 **:checked** 伪类选择器的浏览器（比如 IE 8），则可以借助 javaScript 来根据控件状态修改真正的 class 属性达到区分不同状态的目的，比如根据是否被选中来添加或删除一个 **checked** 的 class：

// jQuery

$('input[type="checkbox"]').on('change', **function**() {

$(**this**)[$(**this**).prop('checked') ? 'addClass' : 'removeClass']('checked');

});

/\* CSS \*/

input[type="checkbox"].checked + label:before { ... }

有了前面的基础，要制作类似于开关按钮的控件也是非常简单的了，还是熟悉的结构：

<div **class**="checkbox">

<**input** id="example" type="checkbox">

<**label** for="example">Check</**label**>

</**div**>

首先勾勒出开关的形状，一个圆角矩形中间放一个圆形按钮：

**input**[type="checkbox"]:checked + **label**,

**input**[type="checkbox"]:not(:checked) + **label** {

**background-color**: #e0e0e0;

**border-radius**: 24px;

**cursor**: pointer;

**display**: inline-block;

**height**: 24px;

**position**: relative;

**text-indent**: -9999px;

**width**: 48px;

}

**input**[type="checkbox"]:checked + **label**:after,

**input**[type="checkbox"]:not(:checked) + **label**:after {

**background-color**: #fff;

**border-radius**: 20px;

**content**: " ";

**height**: 20px;

**left**: 2px;

**position**: absolute;

**top**: 2px;

**width**: 20px;

}



选中的效果只要简单修改下外框的背景色和中间按钮的位置就行：

**input**[type="checkbox"]:checked + **label** {

**background-color**: #8c8;

}

**input**[type="checkbox"]:checked + **label**:after {

**left**: 26px;

}



不过这种跳跃式变化实在是太生硬了，添加点过渡效果，看上去更平滑：

**input**[type="checkbox"]:checked + **label**,

**input**[type="checkbox"]:not(:checked) + **label** {

**-webkit-transition**: background-color 0.3s;

**transition**: background-color 0.3s;

}

**input**[type="checkbox"]:checked + **label**:after,

**input**[type="checkbox"]:not(:checked) + **label**:after {

**-webkit-transition**: left 0.3s;

**transition**: left 0.3s;

}

点击就能看到效果，对于中间的按钮部分使用 **CSS3 Transforms** 来替代 **left** 效果更好，速度更快：

**input**[type="checkbox"]:checked + **label**:after,

**input**[type="checkbox"]:not(:checked) + **label**:after {

**-webkit-transition**:

left-webkit-transform 0.3s; -o-transition: -o-transform 0.3s; transition: transform 0.3s; } input[type="checkbox"]:checked + label:after { -webkit-transform: translateX(24px); -ms-transform: translateX(24px); -o-transform: translateX(24px); transform: translateX(24px); }

不支持 **CSS3 Transforms** 的浏览器仍然可以看到背景色的变化，不影响可用性，详见 [***CSS3 Transforms***](http://caniuse.com/#feat=transforms2d) 。关于性能问题，请参考 [***High Performance Animations***](http://www.html5rocks.com/en/tutorials/speed/high-performance-animations/) 。快速点击“控件”会因选中效果造成不能切换状态的情况，所以去掉“控件”可以被选中的能力：

input[type="checkbox"]:checked + **label**,

input[type="checkbox"]:not(:checked) + **label** {

(-prefix-)**user**-select: none;

}

这里的浏览器厂商前缀根据需要替换成相应的， [***查看简单示例***](http://jsfiddle.net/ov785vcm/) 。当然还需要提供聚焦以及禁用等状态的样式，就不在这里重复了。以上所有技术可同时适用于单选框(radio)。