

# 真正了解CSS3背景下的@font face规则

这篇文章发布于 2017年03月30日, 星期四, 23:28, 归类于 [CSS相关](#)。阅读 28681 次, 今日 15 次 [18 条评论](#)

by zhangxinxu from <http://www.zhangxinxu.com/wordpress/?p=6063>

本文可全文转载, 但需得到原作者书面许可, 同时保留原作者和出处, 摘要引流则随意。

很多人只要一提到 `@font face` 规则, 心中就会不由自主“哦~~”的一声: “这个我知道, 可以用来生成自定义字符小图标!”话是没错, 问题在于很多人就以为生成字符小图标就是 `@font face` 规则的全部, 实际上只是功能之一, 如果真正了解 `@font face` 规则, 你会发现, `@font face` 规则可以做的事情其实非常多, 尤其我们不考虑IE7, IE8浏览器的情况下。

## 一、@font face的本质是变量

虽然说CSS3新世界才出现真正意义上的变量 `var` (参见此文: [“了解CSS/CSS3原生变量var”](#)), 但实际上, CSS世界中, 就已经出现了本质上就是变量的东西, `@font face` 规则就是其中之一, `@font face` 的本质就是一个定义字体或字体集的变量, 这个变量不仅仅是简单的自定义字体, 还包括字体重命名, 默认字体样式设置等等。

这个“变量”的意识很重要, 有助于我们发挥 `@font face` 的潜力, 可以让我们的CSS代码更加的精简, 更方便的维护等。

在进入正题之前, 我们先要对 `@font face` 规则有个大致的面面观。

`@font face` 规则支持的CSS属性有: `font-family`, `src`, `font-weight`, `font-style`, `unicode-range`, `font-variant`, `font-stretch`, `font-feature-settings`。例如:

```
@font-face {
  font-family: 'example';
  src: url(example.ttf);
  font-style: normal;
  font-weight: normal;
  unicode-range: U+0025-00FF;
  font-variant: small-caps;
  font-stretch: expanded;
  font-feature-settings: "liga" on;
}
```

属性还是挺多的, 而且有些属性, 估计是他认识你, 你不认识他。但是从实用角度来讲, 有些属性其实可以不用去深究, 例如 `font-variant`, `font-stretch`, `font-feature-settings` 这 3 个属性。为什么呢? 因为按照我的经验理解, 这 3 个属性给我感觉更像是专为英文设计的, 所以如果不是有业务需要, 可以先放一放。

好, 现在, 是不是感觉压力一下子小了很多, 我们需要在意的可以自定义的属性就只剩下下面这些:

```
@font-face {
  font-family: 'example';
  src: url(example.ttf);
  font-style: normal;
```

```
font-weight: normal;
unicode-range: U+0025-00FF;
}
```

估计会有小伙伴疑惑 `@font-face` 规则中的 `font-style`，`font-weight` 以及 `unicode-range` 这些属性有什么用，尤其 `font-style`，`font-weight`，好像就是专门过来打酱油的。实际上，这里的每个属性都不是泛泛之辈，都是有故事的人。

在介绍每个属性之前，有必要郑重声明一下，接下来有大量的案例，全是基于 `local` 本地字体做示意的，而IE7，IE8浏览器是不支持 `local` 本地字体的，所以，本文的标题是“CSS3背景下的.....”，就是这么个原因，大家注意明辨，如果你的项目还需要兼容IE8，本文介绍，这些好东西可能就需要斟酌斟酌了。

## 二、@font face常用CSS属性详细介绍

我们一个一个来介绍：

### 1. font-family

这里的 `font-family` 可以看成是一个字体变量，名称可以非常随意，例如直接一个美元符号 `'$'`，例如：

```
@font-face {
  font-family: '$';
  src: local("Microsoft Yahei");
}
```

这时候，对普通HTML元素，你设置其 `font-family` 属性值为 `'$'`，则其字体表现就变成了“微软雅黑”（如果本地有这个字体）。

甚至非IE浏览器下可以直接使用纯空格 `' '`，不过有一点需要注意，就是使用这些稀奇古怪的字符或者有空格的时候，一定要加引号。

虽然说自己变量名可以很随意，但是有一类名称，不能随便设置，就是原本系统就有的字体名称，例如下面的代码：

```
@font-face {
  font-family: 'Microsoft Yahei';
  src: local(SimSun);
}
```

从此“微软雅黑”字体就变成了“宋体”。当然，有时候我们说不定就需要这种覆盖，比如说新来了一个设计主管，平生最看不惯“微软雅黑”，希望换成其他字体，这个时候我们就可以使用这个变量覆盖轻轻松松完成整站的字体变更。

### 2. src

`src` 表示调用字体文件，可以是本地字体文件（IE9+支持），也可以是线上地址（可能有跨域限制）。

本文主要着重介绍本地字体文件的应用。

**作用1：字体文件名简写**

现在很多网站会使用“微软雅黑”字体，但是，“微软雅黑”的名称有点长：

```
.font {
  font-family: 'Microsoft Yahei';
}
```

小手一抖，说不定就拼错了，还要加引号，还要加空格，好麻烦的勒。此时我们就可以利用 `@font face` 规则简化，这样就容易记忆了，书写更快了：

```
@font-face {
  font-family: YH;
  src: local("Microsoft Yahei");
}
```

使用的时候直接：

```
.font {
  font-family: YH;
}
```

多么干净，多么清爽，心情多么舒畅！

`src` 还支持多个 `local` 字体地址同时出现，嘿，这简化的CSS代码可不是一个字体名了，而是一大批字体名称了，例如某网站使用了大量类似下面的 `font-family` 属性值：

```
body {
  font-family: PingFangSC-Regular,HelveticaNeue-Light,'Helvetica Neue Light','Microsoft YaHei',sans-serif;
}
.xxx {
  font-family: PingFangSC-Regular,HelveticaNeue-Light,'Helvetica Neue Light','Microsoft YaHei',sans-serif;
}
```

虽然说，实现的时候，借助了Sass, Less之类工具让字体名称们成为了变量，写代码的时候好像也没怎么吃力，但是最终生成的CSS其实是比较啰嗦，如果我们走本质上就是变量的 `@font face`，是不是不仅开发简单，代码也简单了，如下处理：

```
@font-face {
  font-family: BASE;
  src: local("HelveticaNeue-Light"), local("Helvetica Neue Light"), local("PingFang SC"), local("Microsoft YaHei"), local(sans-serif);
}
```

于是我们用字体的时候直接：

```
body {
  font-family: BASE;
}
.xxx {
  font-family: BASE;
}
```

看语义又好，代码又简单，实用又方便，超级使用的小技巧。

当然，`local` 本地作用还不止这些。

业界有个名为“[字蛛](#)”的中文字体处理工具，可以提取页面中使用的特殊中文字体并生成新的体积更小的自定义字体。我们可以在这个基础上更进一步，举个例子：

一些特殊的标题设计师就是喜欢用“方正粗雅宋”这个字体（已经购得版权），但是很显然，这么帅气的字体，几乎很少有用户安装的，所以实际开发的时候，全都是通过工具而生成一个全新的精简版的“方正粗雅宋”字体文件，再通过 `src` 属性 `url()` 方法外链这个字体文件。

很棒的方案，但还不够完美，为什么呢？虽然安装“方正粗雅宋”这个字体的用户并不多，但并不表示没有用户使用之。试想一下，假如有用户安装了这个字体，结果你在网页呈现的时候，还要再去外链一个额外的文件地址，岂不是白白的浪费？既浪费流量，体验又不一定好，如果字体文件加载慢，会看到文字“变形”的过程，显然是不友好的。

我们可以借助 `@font face` 规则中的 `local` 实现代码和体验上的进一步的提升，如下：

```
@font-face {
  font-family: FZCYS;
  src: local("FZYaSongS-B-GB"),
       url("FZCYS.woff2"),
       url("FZCYS.woff"),
       url("FZCYS.ttf");
}
```

于是乎，那些安装了“方正粗雅宋”这个字体的用户，就没有任何字体文件请求。加载更快了，用户体验上升了，还省了流量，如此百益无一害的事情，还不要妥妥的用起来。

### 3. font-style

在Chrome浏览器下，`@font face` 规则设置 `font-style:italic` 可以让文字倾斜，但是这并不是其作用所在。

`@font face` 规则中的 `font-style` 和 `font-weight` 类似，都是用来设置对应字体样式或自重下该使用什么字体。因为有些字体可能会有专门的斜体字，注意这个斜体字，并不是让文字的形状倾斜，而是专门设计的倾斜的字体，很多细节会跟物理上的请求不一样。于是，我在CSS代码中使用 `font-style:italic` 的时候，就会调用这个对应字体，如下示意：

```
@font-face {
  font-family: 'I';
  font-style: normal;
  src: local('FZYaoti');
}
@font-face {
  font-family: 'I';
  font-style: italic;
  src: local('FZShuTi');
}
```

由于专门的斜体字不太好找，所以我使用“方正姚体”和“方正舒体”代替做示意。上面CSS代码我解读一下：制定一个字体，名叫 `'I'`，当文字样式正常的时候，字体表现使用“方正姚体”，当文字设置了 `font-style:italic` 的时候，字体表现为“方正舒体”。

好，现在假设有下面这样的CSS和HTML：

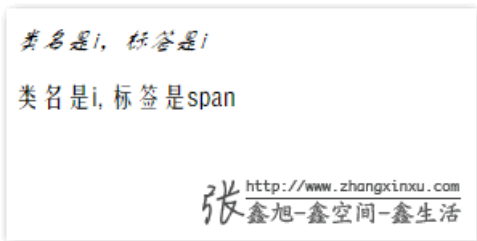
```
.i {
  font-family: I;
}
```

```
<p><i class="i">类名是i, 标签是i</i></p>
<p><span class="i">类名是i, 标签是span</span></p>
```

请问最终的表现是怎样的?

由于 `<i>` 标签天然 `font-style:italic` , 因此理论上, 上面一行文字表现为“方正舒体”, 下面一行为“方正姚体”, 最终结果是如何呢?

当当当当, 完全符合, 如下截图:



这下大家应该明白, `@font face` 规则中的 `font-style` 是干嘛用的了吧。

#### 4. font-weight

`font-weight` 和 `font-style` 类似, 不过是定义了不同字重使用不同字体, 对于中文而言, 这个要比 `font-style` 适用性强很多。我这里就有一个非常具有代表性的例子, 版权字体“汉仪旗黑”自重非常丰富, 但是这个字体不像“思源黑体”, 天然可以根据 `font-weight` 属性值加载对应的字体文件, 怎么办呢? 很简单, 使用 `@font face` 规则重新定义一下即可, 例如下面的CSS代码:

```
@font-face {
  font-family: 'QH';
  font-weight: 400;
  src: local('HYQihei 40S');
}
@font-face {
  font-family: 'QH';
  font-weight: 500;
  src: local('HYQihei 50S');
}
@font-face {
  font-family: 'QH';
  font-weight: 600;
  src: local('HYQihei 60S');
}
```

解读一下就是, 是一个全新的字体, 名为 `'QH'` , 当字重 `font-weight` 为 `400` 的时候, 使用“汉仪旗黑 40S”字重字体, 为 `500` 的时候, 使用“汉仪旗黑 50S”字重字体, 为 `600` 的时候, 使用“汉仪旗黑 60S”字重字体。

于是乎, 当我们应用如下的CSS和HTML代码的时候:

```
.hy-40s,
.hy-50s,
.hy-60s {
  font-family: 'QH';
}
.hy-40s {
  font-weight: 400;
}
```

```
.hy-50s {
  font-weight: 500;
}
.hy-60s {
  font-weight: 600;
}
```

```
<ul>
  <li class="hy-40s">汉仪旗黑40s</li>
  <li class="hy-50s">汉仪旗黑50s</li>
  <li class="hy-60s">汉仪旗黑60s</li>
</ul>
```

我们就可以看到，文字粗细错落有致的效果，如下截图：



如果用在网页开发中，必定会让我们的界面更加的细腻，设计更加的精致，视觉更加的愉悦。

## 5. unicode-range

`unicode-range` 的作用是可以让特定的字符或者字符片段使用特定的字体。举个例子，下图是文字应用了“微软雅黑”字体后的效果：

我是“微软雅黑”或者是“苹方字体”

但是，这两个“引号”左右间隙不均，方向不明，实在是看着不爽，此时我们就专门指定这两个“引号”使用其他字体，如下CSS：

```
@font-face {
  font-family: BASE;
  src: local("Microsoft Yahei");
}
@font-face {
  font-family: quote;
  src: local('SimSun');
  unicode-range: U+201c, U+201d;
}
.font {
  font-family: quote, BASE;
}
```

然后效果就变成这样子了：



嗯，一下子变得舒服多了。

关于 `unicode-range` 更多内容，可参考我之前写的文章[“CSS unicode-range特定字符使用font-face自定义字体”](#)。

## 三、结束语

根据我个人的感觉，`@font-face` 的为人所知要得益于icon fonts小图标技术。因为自定义的小图标字体几乎一定是外链的，而恰恰好，低版本IE7，IE8浏览器的 `src` 只能是 `url()` 外部字体文件。外加矢量、颜色可控等特性，于是成为了兼容时代最好的小图标解决方案，可谓功不可没，但也带来了另外一个弊端，让很多同学误以为 `@font-face` 除了自定义一些小图标之外就一无是处了。于是，当越来越多产品和项目不需要兼容IE7，IE8浏览器的时候，由于有更好的SVG图标解决方案，`@font-face` 似乎又很少被提及了。

通过本文的一些介绍，大家应该感觉到 `@font-face` 规则的作用被严重低估了，除了小图标外，`@font-face` 还能做的事情非常多，而这些事情，才更像是 `@font-face` 应该做的事情。

就算你的项目需要兼容IE8，本文介绍的一些特性也是可以渐进增强使用的，例如特殊中文字体优先使用本地字体，或者 `unicode-range` 特定字符使用特定字体等。

希望本文的内容可以唤起大家对 `@font-face` 真正的认识！

感谢阅读，欢迎交流！

《CSS世界》签名版独家发售，包邮，可指定寄语，点击显示购买码

(本篇完) // 想要打赏？点击[这里](#)。有话要说？点击[这里](#)。



« CSS font-family常见中文字体对应的英文名称

CSS :focus伪类JS focus事件提高网站键盘可访问性 »

### 猜你喜欢

- CSS font-family常见中文字体对应的英文名称
- CSS font-feature-settings 50+关键字属性值完整介绍
- 了解CSS属性font-kerning,font-smoothing,font-variant
- CSS unicode-range特定字符使用font-face自定义字体
- 页面可用性之浏览器默认字体与CSS中文字体
- CSS font关键字属性值的简单研究
- 等宽字体在web布局中应用以及CSS3 ch单位嘿嘿
- 请使用千位分隔符(逗号)表示web网页中的大数字
- CSS蛋疼应用之：数据上报和HTML验证
- 翻译 - CSS继承详解
- 实现兼容性的CSS粗虚线边框(dashed)效果

分享到： 1

标签： css3, font, font-face, font-family, font-feature-settings, font-stretch, font-variant, unicode-range, 中文字体, 思源黑体, 汉仪旗黑

发表评论（目前18条评论）

<input type="text"/>	名称 (必须)
<input type="text"/>	邮件地址(不会被公开) (必须)
<input type="text"/>	网站
<div></div>	
<div>提交评论</div>	

1. **gagarling**说道:

2018年09月18日 15:03



我想问一下，项目的字体要兼容到ie8的话，是不是不能用本地字体的相对路径（就是引用的时候不能用相对的路径？还是说只要上传到服务器后用相对的路径也是可以的？）

[回复](#)

张 鑫旭说道:

2018年09月18日 16:53



和CSS文件同源使用相对路径也是可以的。

[回复](#)

2. **mo**说道:

2017年09月6日 10:08



有个需求，中英文混排的情况下，怎么让中文和英文的字体大小设置成不一样？用纯css 能否实现？

[回复](#)

张 鑫旭说道:

2017年09月6日 10:52



引入一个自定义英文字体即可，这个英文字体默认规格不一样就可以了。

[回复](#)

3. **joseph**说道:

2017年06月7日 18:15



针对于@font-face 中src加载一个字体文件,加载太慢有什么好的解决办法吗

[回复](#)

4. **吉吉**说道:

2017年06月6日 18:02



大神，您好，在介绍第四点font-weight的时候，有句话是“对于中午而言，这个要比font-style适用性强很多”，这个 中午 是啥。。个人有些轻微强迫症。。。。望解释

[回复](#)



lingcarzy说道：  
2017年06月29日 10:45

应该是 中文 的笔误。

[回复](#)



口合口合说道：  
2017年07月1日 10:42

应该是中文吧

[回复](#)



5. KTP说道：  
2017年06月6日 11:59

非常牛掰..

[回复](#)



6. 哈哈说道：  
2017年05月25日 13:30

无意中发见你的网站，受益匪浅，十分感谢，希望你能坚持更新下去。

[回复](#)



7. 墨、水瓶说道：  
2017年05月15日 16:13

博主是用的阿里巴巴矢量图标库吧？

[回复](#)



8. 十八子说道：  
2017年05月3日 09:51

膜拜大神！

[回复](#)



9. woderitian说道：  
2017年04月19日 17:15

local的兼容性怎么查，我在caniuse上没搜出来。。。

[回复](#)



10. Web前端之家说道：  
2017年04月12日 15:29

不错不错

[回复](#)



11. yyhaxx说道：  
2017年04月10日 08:54

4.font-weight  
font-weight和font-style类似，不过是定义了不同字重使用不同字体，对于中午而言,...  
这里好像有个错别字，中“午”，应该是中“文”

[回复](#)



12.

jzz7280说道:

2017年03月31日 23:18

不错

回复


13.

saber说道:

2017年03月31日 15:03

很强大

回复


14.

GIN说道:

2017年03月31日 09:55

忘记是不是在大大你这里看到的了，<http://icomoon.io/> 这个网站可以将svg生成font-icon字体，坑爹的是清缓存了会找不到之前存的自定义字体库以及乱改名字可能使用有问题

回复



#### 最新文章

- » 常见的CSS图形绘制合集
- » 粉丝群第1期CSS小测点评与答疑
- » 分享三个纯CSS实现26个英文字母的案例
- » 小tips: 纯CSS实现打字动画效果
- » CSS/CSS3 box-decoration-break属性简介
- » CSS :placeholder-shown伪类实现Material Design占位符交互效果
- » 从天猫某活动视频不必要的3次请求说起
- » CSS vector-effect与SVG stroke描边缩放
- » CSS ::backdrop伪元素是干嘛用的?
- » 周知: CSS -webkit-伪元素选择器不再导致整行无效

#### 今日热门

- » 常见的CSS图形绘制合集 (190)
- » 未来必热: SVG Sprite技术介绍 (119)
- » 粉丝群第1期CSS小测点评与答疑 (115)
- » HTML5终极备忘大全 (图片版+文字版) (93)
- » 让所有浏览器支持HTML5 video视频标签 (86)
- » Selectivizr-让IE6~8支持CSS3伪类和属性选择器 (82)
- » CSS3下的147个颜色名称及对应颜色值 (79)
- » 视区相关单位vw, vh.简介以及可实际应用场景 (76)
- » 写给自己看的display: flex布局教程 (76)
- » 小tips: 纯CSS实现打字动画效果 (76)

#### 今年热议

- » 《CSS世界》女主角诚寻靠谱一起奋斗之人 (76)

- » [不借助Echarts等图形框架原生JS快速实现折线图效果](#) <sup>(64)</sup>
- » [看，for..in和for..of在那里吵架！](#) <sup>(60)</sup>
- » [是时候好好安利下LuLu UI框架了！](#) <sup>(47)</sup>
- » [原来浏览器原生支持JS Base64编码解码](#) <sup>(35)</sup>
- » [妙法攻略：渐变虚框及边框滚动动画的纯CSS实现](#) <sup>(33)</sup>
- » [炫酷H5中序列图片视频化播放的高性能实现](#) <sup>(31)</sup>
- » [CSS scroll-behavior和JS scrollIntoView让页面滚动平滑](#) <sup>(30)</sup>
- » [windows系统下批量删除OS X系统.DS\\_Store文件](#) <sup>(26)</sup>
- » [写给自己看的display: flex布局教程](#) <sup>(26)</sup>

## 猜你喜欢

- [CSS font-family常见中文字体对应的英文名称](#)
- [CSS font-feature-settings 50+关键字属性值完整介绍](#)
- [了解CSS属性font-kerning,font-smoothing,font-variant](#)
- [CSS unicode-range特定字符使用font-face自定义字体](#)
- [页面可用性之浏览器默认字体与CSS中文字体](#)
- [CSS font关键字属性值的简单研究](#)
- [等宽字体在web布局中应用以及CSS3 ch单位嘿嘿](#)
- [请使用千位分隔符\(逗号\)表示web网页中的大数字](#)
- [CSS蛋疼应用之：数据上报和HTML验证](#)
- [翻译 - CSS继承详解](#)
- [实现兼容性的CSS粗虚线边框\(dashed\)效果](#)