

08.PostCss 兼容性转换

学习要点：

1. 兼容处理
2. CSS4 处理

本节课我们来开始学习一下 CSS 对浏览器兼容处理并转换的方法：

一. 兼容处理

1. CSS 尤其是很多 CSS3，在部分浏览器是有兼容处理的，需要进行设置；
2. 可以使用 `postcss-loader` 来实现 CSS 语法的兼容性处理；

```
npm i postcss-loader -D
```

```
//放在最右边
use : [
  ...
  'postcss-loader'
]
```

3. 单有 `postcss-loader` 还不行，需要搭配创建 `autoprefixer` 插件转换；

```
npm i autoprefixer -D
```

4. 单有 `postcss-loader` 还不行，需要搭配创建 `autoprefixer` 插件转换；
5. 这个插件会在需要兼容性的样式加上 CSS 前缀，比如：-ms、-webkit 等；
6. 我们在 `base.css` 和 `font.css`(被`@import`导入)，增加需要兼容的 css；

```
display: flex;
transform: rotateY(130deg);
```

7. 在当前目录创建 `postcss.config.js`，用来配置兼容的浏览器；

```
const AutoPreFixer = require('autoprefixer');

module.exports = {
  plugins: [
    new AutoPreFixer({
      overrideBrowserslist : [
        '> 0.15% in CN' //可以写多个列表
      ]
    })
  ]
}
```

```
{  //由于采用了@import，被导入的css无法解析，需要设置importLoaders=1即可
  loader : 'css-loader',
  options: {
    importLoaders: 1
  }
}
```

8. 下图就是配置浏览器的语法，可以自行添加；

例子	说明
> 1%	全球超过1%人使用的浏览器
> 5% in US	指定国家使用率覆盖
last 2 versions	所有浏览器兼容到最后两个版本根据CanIUse.com追踪的版本
Firefox ESR	火狐最新版本
Firefox > 20	指定浏览器的版本范围
not ie <=8	方向排除部分版本
Firefox 12.1	指定浏览器的兼容到指定版本
unreleased versions	所有浏览器的beta测试版本
unreleased Chrome versions	指定浏览器的测试版本
since 2013	2013年之后发布的所有版本

二. CSS4 代码

1. 除了兼容浏览器之外，还可以通过 postcss-preset-env 实现 CSS4 代码兼容；

```
npm i postcss-preset-env -D
```

```
:root: {
  --green : green;
};
h1 {
  color : var(--green);
}
```

```
//请注意：如果使用 postcss-preset-env，就删除 autoprefixer
//因为 post-preset-env 集成了 autoprefixer，不需要再引入设置
const PostPresetEnv= require('postcss-preset-env);
```

```
module.exports = {
  plugins: [
    new PostPresetEnv({
      browsers : [
        '> 0.15% in CN'
      ]
    })
  ]
}
```

```
    }),  
  ]  
}
```