

一、渐变

- CSS中可以通过背景来设置渐变效果
- 关于渐变你需要注意的第一个问题就是渐变是图片，而不是颜色
- 我们需要通过background-image或background简写属性来设置渐变效果
- 线性渐变

○ linear-gradient(角度, 颜色 位置, 颜色 位置, 颜色 位置)

○ 例子：

- background-image: linear-gradient(red, orange, yellow);



- background-image: linear-gradient(to right, red, orange, yellow);



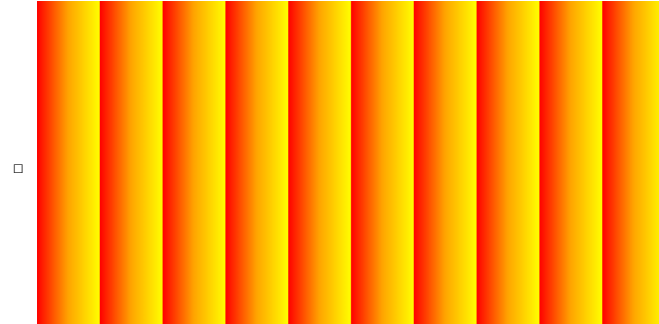
- background-image: linear-gradient(90deg, red, orange, yellow);



- background-image: linear-gradient(90deg, red 33px, orange 33px 66px, yellow 66px);



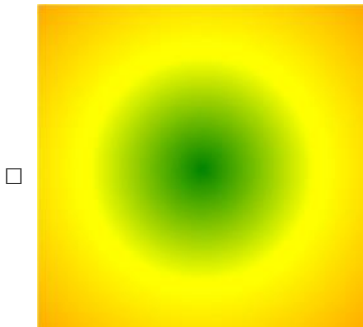
- repeating-linear-gradient(90deg, red, orange 25px, yellow 50px);



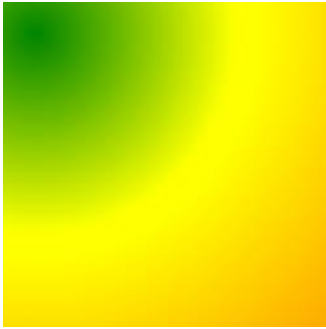
- 径向渐变

○ radial-gradient(形状 at 位置, 颜色, 颜色, 颜色)

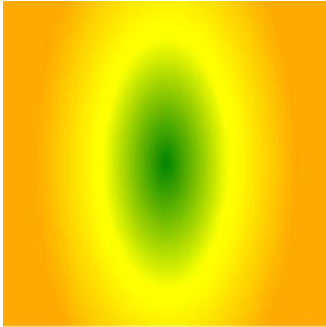
- background-image: radial-gradient(green, yellow, orange);



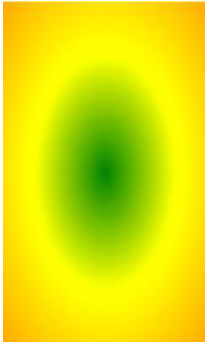
- `background-image: radial-gradient(at 50px 50px, green, yellow, orange);`



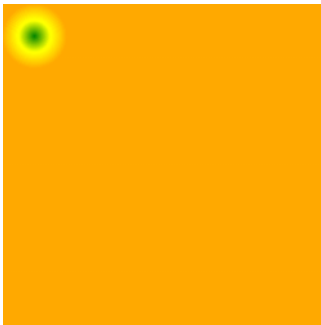
- `background-image: radial-gradient(200px 400px, green, yellow, orange);`



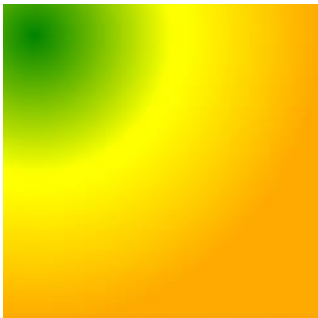
- `background-image: radial-gradient(green, yellow, orange);`



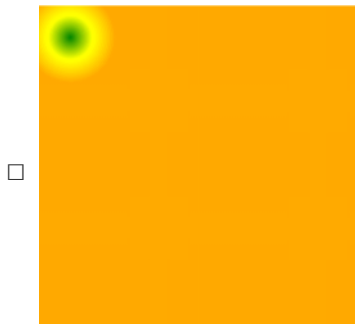
- `background-image: radial-gradient(closest-side at 50px 50px, green, yellow, orange);`



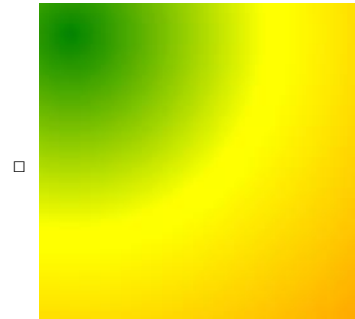
- `background-image: radial-gradient(farthest-side at 50px 50px, green, yellow, orange);`



- `background-image: radial-gradient(closest-corner at 50px 50px, green, yellow, orange);`



- `background-image: radial-gradient(farthest-corner at 50px 50px, green, yellow, orange);`



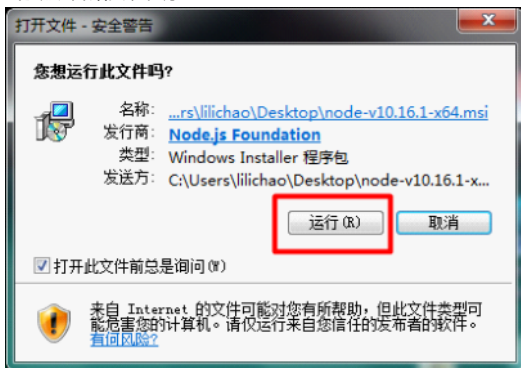
二、LESS

- 安装Nodejs

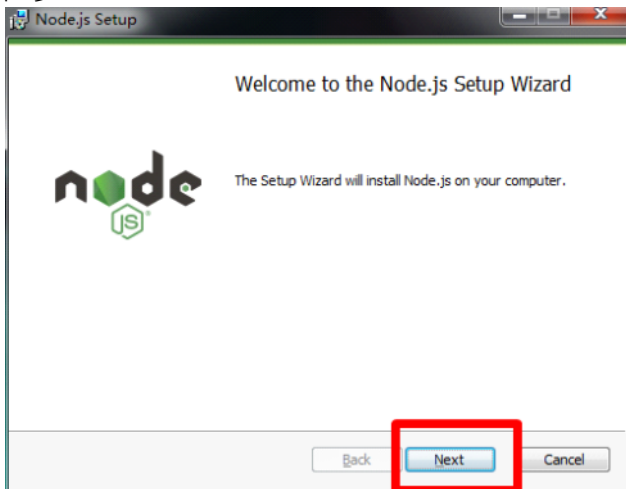
- 双击安装图标



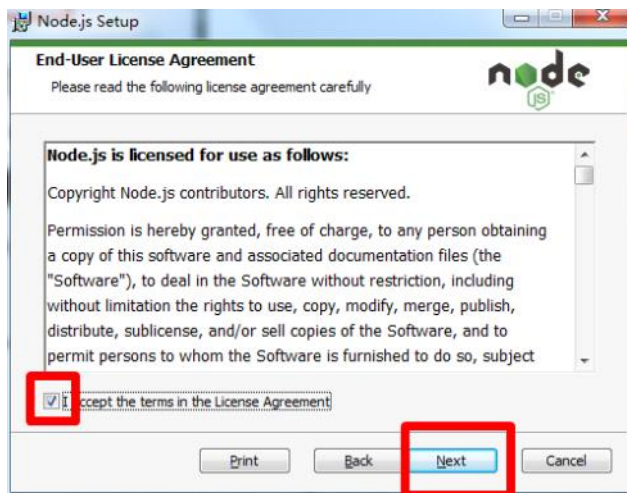
- 点击运行确认开始安装程序



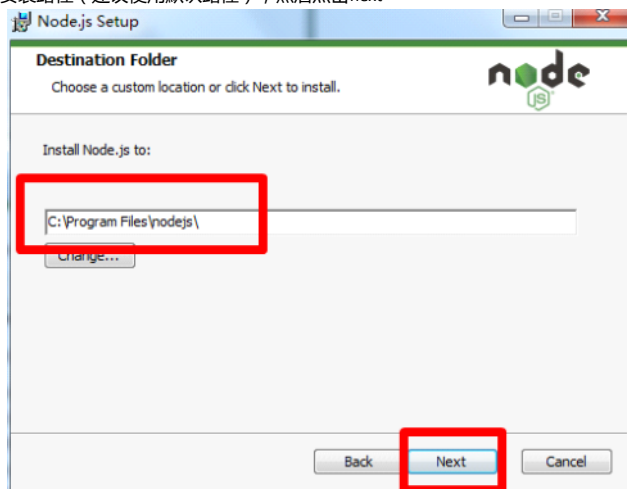
- 点击下一步



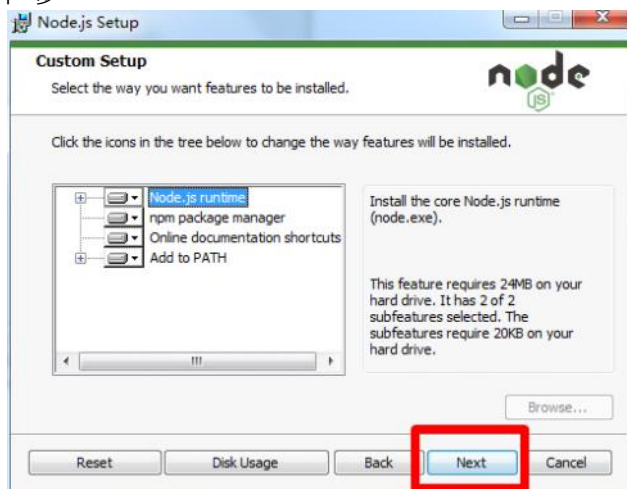
- 选择同一，然后点击next



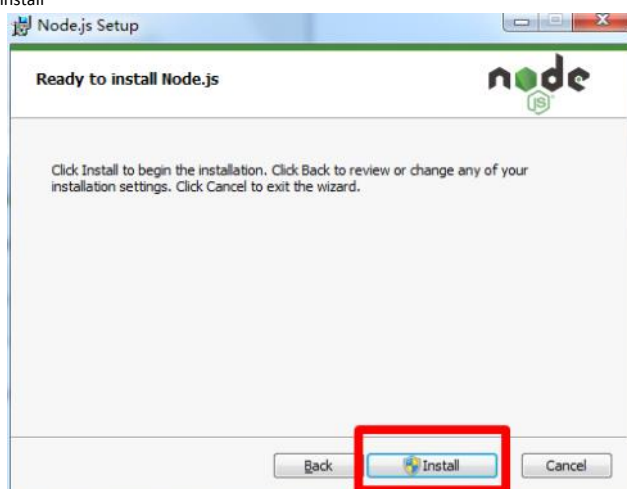
- 选择安装路径（建议使用默认路径），然后点击next



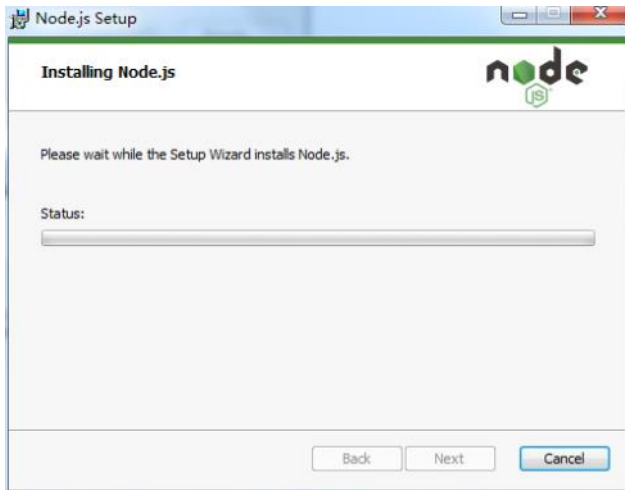
- 点击下一步



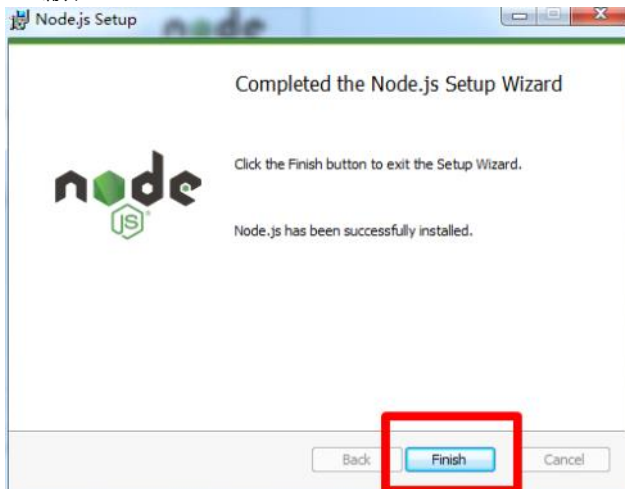
- 点击install



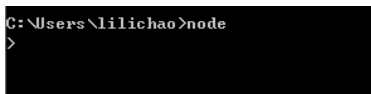
- 等待安装结束



- 点击Finish确认



- 打开命令行，输入node，显示如下表示安装成功



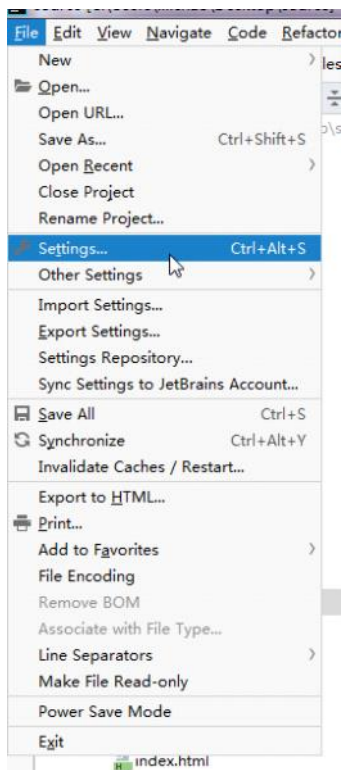
- NPM的基本使用

- NPM是Node的包管理器，通过NPM可以下载js所编写的各种工具，从而在计算机中使用这些工具
- 由于NPM的服务器在国外，所以在国内访问的速度的是比较慢的，所以一般我们可以通过配置镜像服务器来加快下载速度
- 安装Cnpm

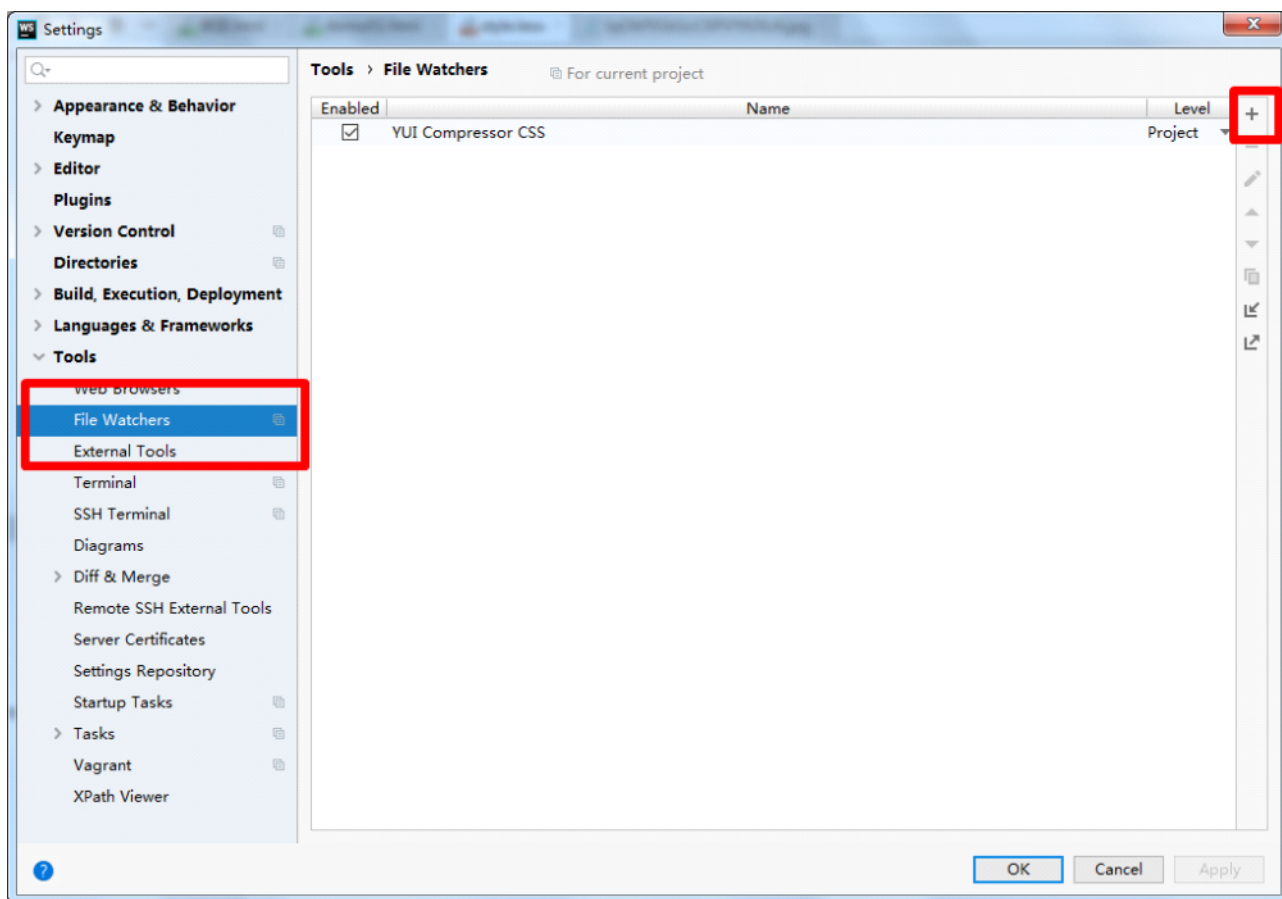
```
npm install -g cnpm --registry=https://registry.npm.taobao.org
```

- Less的简介

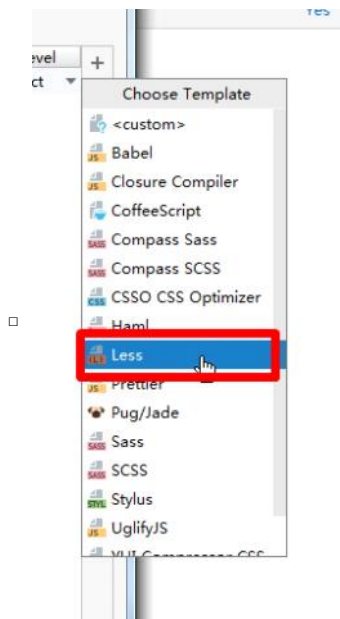
- Less 是一门 CSS 预处理语言，它扩展了 CSS 语言，增加了变量、Mixin、函数等特性，使 CSS 更易维护和扩展。
- 使用Less
 - 安装less的编译器
 - 在命令行中执行 `npm install -g less`
 - 使用方法
 - `lessc xxx.less xxx.css` 将xxx.less 编译为 xxx.css
- 整合Webstrom
 - 打开WebStorm --> File --> Settings



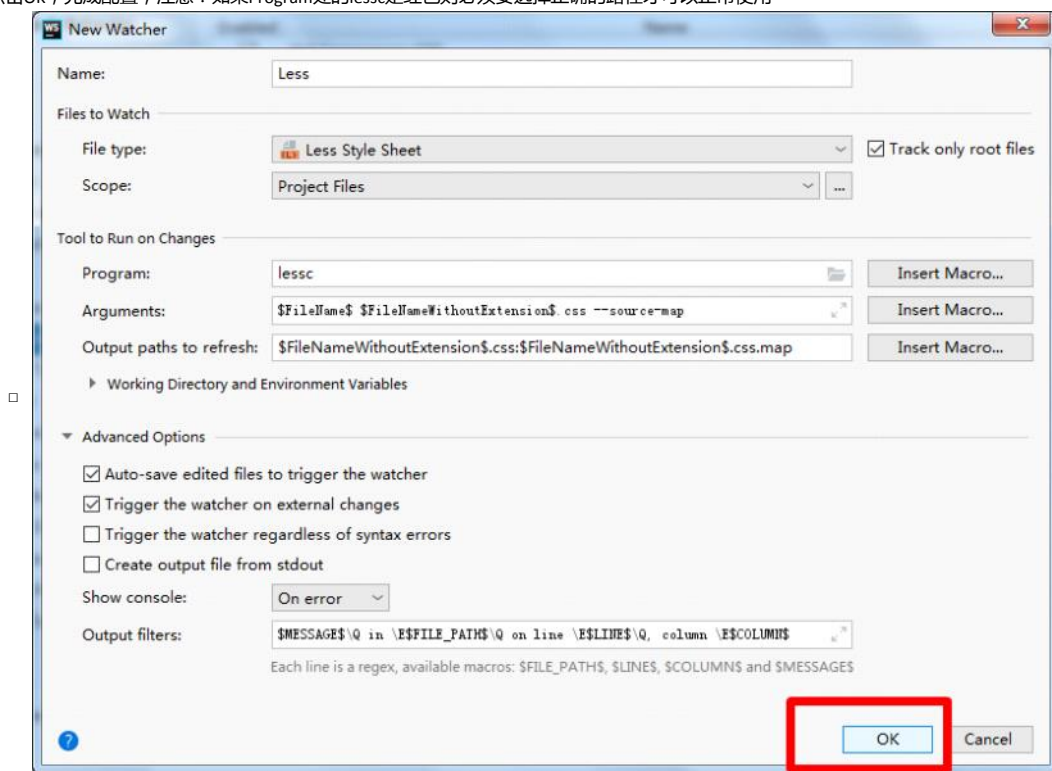
- 选择File Watchers，并点击右侧加号



- 选择less



- 点击OK，完成配置，注意！如果Program处的lessc是红色则必须要选择正确的路径才可以正常使用



- Less的语法

○ 注释

- 在Less中，使用//作为单行注释，使用/**/作为多行注释

○ 变量

- 在Less中可以创建并使用变量
- 语法：

□ @变量名: 变量值;

- 例子：

□ 源码

```
1 //声明一个变量
2 @hello : #bfa;
3 @w : 100px;
4 @h : 200px;
5
6 .box1{
7   width: @w;
8   height: @h;
9   background-color: @hello;
10 }
```

□ 编译后

```
1 .box1 {
2   width: 100px;
3   height: 200px;
4 }
```

```
4     background-color: #bfa;
5 }
```

□ 在其他位置使用变量

- ◆ 语法：@{变量名}
- ◆ 例子：

◇ 源码：

```
1 @name:hello;
2
3 @{name} {
4     width: 100px;
5     background: url("aa/{name}/aa.png");
6 }
```

◇ 编译后：

```
1 .hello {
2     width: 100px;
3     background: url("aa/hello/aa.png");
4 }
```

■ 变量引用变量

□ 语法：@@变量名

□ 例子：

◆ 源码：

```
1 @a: green;
2 @b: blue;
3
4 .section {
5     @color: a;
6
7     .element {
8         color: @@color;
9     }
10 }
```

◆ 编译后：

```
1 .section .element {
2     color: green;
3 }
```

■ 将其他属性作为变量使用：

□ 语法：\$属性名

□ 例子：

◆ 源码：

```
1 .box1 {
2     width: 100px;
3     height: $width;
4 }
```

◆ 编译后：

```
1 .box1 {
2     width: 100px;
3     height: 100px;
4 }
```

○ 父级选择器

■ 语法：&

■ 例子：

□ 源码：

```
1 .box1 {
2     width: 100px;
3     height: $width;
4     &.box2 {
5         color: red;
6
7         &.box3 {
8             background-color: #bfa;
9         }
10     }
11 }
```

□ 编译后：

```
1 .box1 {
2     width: 100px;
3     height: 100px;
4 }
5 .box1.box2 {
6     color: red;
7 }
8 .box1.box2.box3 {
9     background-color: #bfa;
10 }
```

○ 扩展

■ 语法：

- :extend()
- 通过extend()可以使用其他样式来扩展当前样式

- 例子：

- 源码：

```
1 .box1{
2   width: 100px;
3   height: 100px;
4 }
5
6 .box2{
7   &:extend(.box1);
8 }
```

- 编译后：

```
1 .box1,
2 .box2 {
3   width: 100px;
4   height: 100px;
5 }
```

- 源码：

```
1 .box1{
2   width: 100px;
3   height: 100px;
4 }
5
6 .box2:extend(.box1){
7 }
```

- 编译后：

```
1 .box1,
2 .box2 {
3   width: 100px;
4   height: 100px;
5 }
```

- Mixins

- 通过Mixins可以对样式进行混用

- Mixins的使用有两种方式

- 第一种：

- ◆ 源码：

```
1 .box1{
2   width: 100px;
3   height: 100px;
4 }
5
6 .box2{
7   .box1()
8 }
```

- ◆ 编译后：

```
1 .box1 {
2   width: 100px;
3   height: 100px;
4 }
5 .box2 {
6   width: 100px;
7   height: 100px;
8 }
```

- 第二种：

- ◆ 源码：

```
1 .box1(){
2   width: 100px;
3   height: 100px;
4 }
5
6 .box2{
7   .box1()
8 }
```

- ◆ 编译后：

```
1 .box2 {
2   width: 100px;
3   height: 100px;
4 }
```