# Less语法

|  |
| --- |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |

## 注意:由于我们现在都是使用组件库的开发模式,组件的样式都已经数组好了,所以真正自己写样式的场景不多,less用到的机会也比较少,但是,作为学习总是可以的.其实less只是写代码的时候比较方便一点点而已.

## 注意上面的命令有问题,其实是: npm i less less-loader --save-dev

## 演练-less案例

## 1>利用前面的elementplus模板创建一个项目: lesson29-elementplus\_less,进入项目安装依赖,然后安装less: npm i less less-loader --save-dev,然后在App.vue中添加下面的代码

|  |
| --- |
| <**template**>  <**div class="outer"**>  <**div class="inner"**>  <**h1**>Hello</**h1**>  </**div**>  </**div**> </**template**>  <**script setup**>   </**script**>  <**style lang="less" scoped**>  .**outer**{  **width**: 300**px**;  **height**: 300**px**;  **background-color**: **antiquewhite**;  .**inner**{  **width**: 200**px**;  **height**: 200**px**;  **background-color**: **aqua**;  **h1**{  **background-color**: **gold**;  }  }  } </**style**> |

### 运行项目,效果如下

|  |
| --- |
|  |

### 注意:在less中可以使用变量,如我们可以修改一下代码

|  |
| --- |
| <**template**>  <**div class="outer"**>  <**div class="inner"**>  <**h1**>Hello</**h1**>  </**div**>  </**div**> </**template**>  <**script setup**>   </**script**>  <**style lang="less" scoped**>  **@gld**: **gold**;  .**outer**{  **width**: 300**px**;  **height**: 300**px**;  **background-color**: **antiquewhite**;  .**inner**{  **width**: 200**px**;  **height**: 200**px**;  **background-color**: **aqua**;  **h1**{  **background-color**: **@gld**;  }  }  } </**style**> |

### 效果是一样的

|  |
| --- |
|  |

## 2>我们在components文件夹里面新建一个LessChildComp组件,内容如下,这里用到另外一种less语法

|  |
| --- |
| <**template**>  <**div id="wrap"**>  <**div class="inner"**></**div**> *<!-- <div class="inner2"></div>-->* </**div**>  <**ul**>  <**li**></**li**>  </**ul**> </**template**>  <**script setup**>   </**script**>  <**style lang="less" scoped**>  .**juzhong**{  **position**: **absolute**;  **left**: 0;  **right**: 0;  **top**: 0;  **bottom**: 0;  **margin**: **auto**;  **background**: **deeppink**;  **width**: 100**px**;  **height**: 100**px**;  }   **#wrap**{  **position**: **relative**;  **width**: 300**px**;  **height**: 400**px**;  **border**: 1**px solid**;  **margin**: 0 **auto**;  .**inner**{  .**juzhong**;  }  .**inner2**{  .**juzhong**;  }  }  **li**{  .**juzhong**;  } </**style**> |

## 3>然后在App.vue里面导入这个组件并且使用

|  |
| --- |
|  |

### 效果如下

|  |
| --- |
|  |

## 4>然后我们又编写一个组件LessChildComp2,代码如下

|  |
| --- |
| <**template**>  <**div id="wrap"**>  <**div class="inner"**></**div**>  </**div**>  </**template**>  <**script setup**>   </**script**>  <**style lang="less" scoped**>  **@color**:**purple**;  **\***{  **margin**: 0;  **padding**: 0;  }   .**juzhong**{  **position**: **absolute**;  **left**: 0;  **right**: 0;  **top**: 0;  **bottom**: 0;  **margin**: **auto**;  **background**: **@color**;  **width**: 100**px**;  **height**: 100**px**;  }   **#wrap**{  **position**: **relative**;  **width**: 300**px**;  **height**: 400**px**;  **border**: 1**px solid**;  **margin**: 0 **auto**;  .**inner**{  .**juzhong**;  *//伪类的写法* **&**:**hover**{  **background-color**: **orange**;  }  }   }  </**style**> |

## 5>然后在App.vue里面把前面的代码注释了,只使用LessChildComp2组件

|  |
| --- |
|  |

### 效果如下

|  |
| --- |
|  |
|  |

## 6>我们可以在mixin中使用参数,我们来创建LessChildComp3组件,代码如下

|  |
| --- |
| <**template**>  <**div id="wrap"**>  <**div class="inner"**></**div**>  <**div class="inner2"**></**div**>  </**div**>  </**template**>  <**script setup**>   </**script**>  <**style lang="less" scoped**>  .**juzhong**(**@w**,**@h**,**@c**,**@t**,**@l**){  **position**: **absolute**;  **left**: **@l**;  **right**: 0;  **top**: **@t**;  **bottom**: 0;  **margin**: **auto**;  **background**: **@c**;  **width**: **@w**;  **height**: **@h**;  }   **#wrap**{  **position**: **relative**;  **width**: 300**px**;  **height**: 400**px**;  **border**: 1**px solid**;  **margin**: 0 **auto**;  .**inner**{  .**juzhong**(100**px**,100**px**,**deeppink**,20,50);  }  .**inner2**{  .**juzhong**(150**px**,150**px**,**orange**,0,0);  }  }  </**style**> |

## 7>然后App.vue中只使用LessChildComp3组件

|  |
| --- |
|  |

### 效果如下

|  |
| --- |
|  |

## Less就暂时学习到这里