**Vue瀑布流踩坑记 (vue-waterfall-easy)**

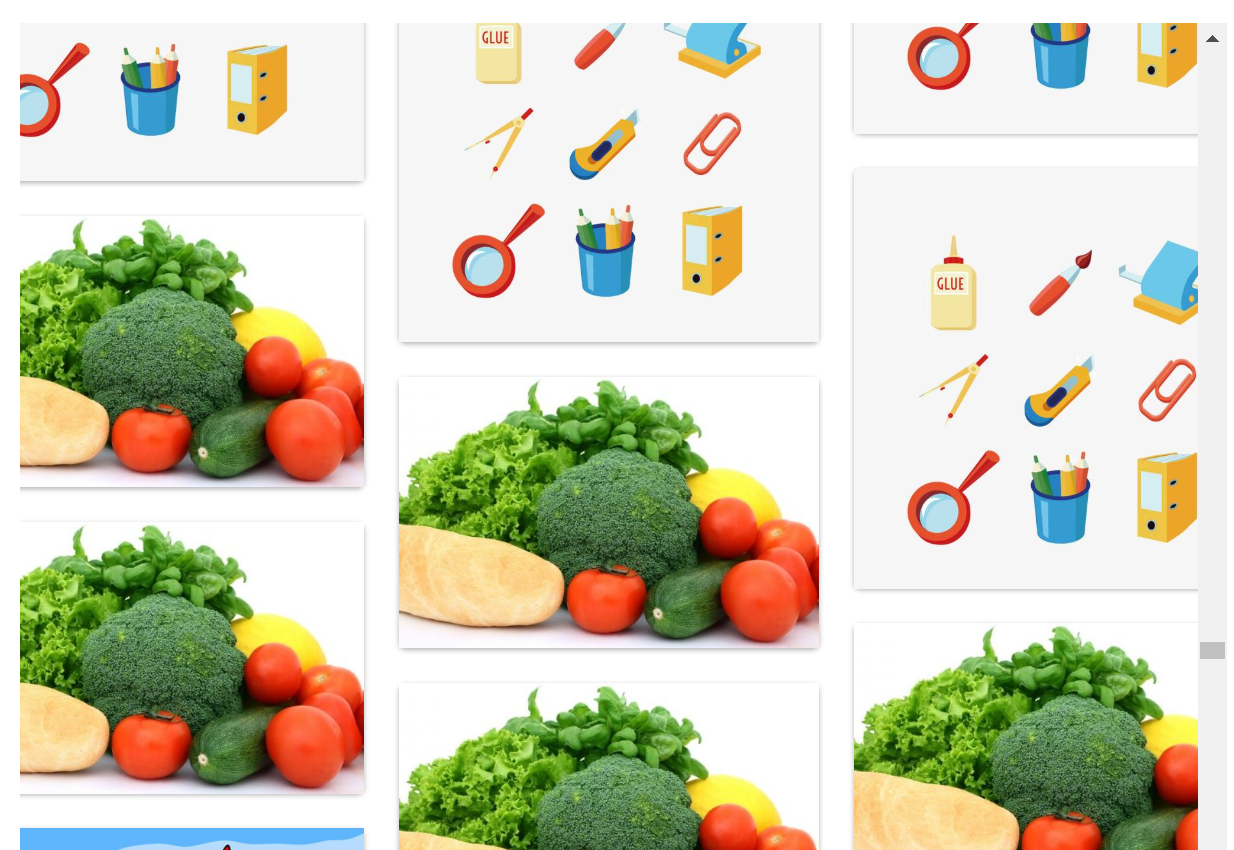
**皓月**

项目中间要展示一系列图片，做成瀑布流展示比较美观，且不用过多考虑图片的大小排列这个问题。通过网上查询得知了vue-waterfall-easy这个很棒的插件，部署简单，自动排列。

但尽管部署“简单”，但中间也碰到了一系列的小坑，在这里记录以帮助后面的人避坑。

插件github地址 <https://github.com/lfyfly/vue-waterfall-easy>

以下是实现的效果



**安装**

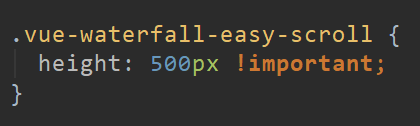


**遇到的坑点:**

1. **确保后端的数据正确情况下，图片在前端显示不出来**

点开inspect发现整个块的高度是0px，应当是在还没有拉取到数据时这个模块就已经渲染了，既然没有数据那么就是0px了，而又因为是0px，所以拉取数据后也无法展示，形成了一个死锁。

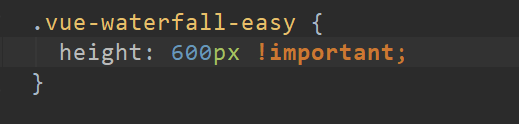
因此将它的高度强制改为500px



1. **有两个滚动条**

其中一个是插件自带的，另一个是浏览器提供的。

而想想浏览器出现滚动条的原因在于当前页面超过了显示器的范围，所以只要让当前所排布的模块高度和小于能显示的范围，就可以把浏览器自带滚动条去除了。



**部署技巧**

在完成一个功能时，一步一步来，大概可以分为这三步

1. 前端硬编码瀑布流展示

2. 从后端硬编码获取数据

3. 后端从数据哭获取数据

这样的话方便定位错误

**后记**

发现基础扎实才是硬道理，大家碰到的问题都不一样，所以直接采用其它人的方法而不理解的话会导致后续一系列问题。

比如在最初碰到图片无法显示的时候，有个博主给出的是将position改为fixed的方法，这个方法确实可以让图片显示，原因在于这时候它的展示不依赖于父组件了。

但是这样改的直接后果导致了没有滚动条，因为滚动条是在界面展示不下的时候产生的，但是fixed这个属性它的展示是固定的。这个地方卡了我很久，在github的issue上有人说这是因为@scrollReachBottom这个事件大写，但是html大小写不敏感，因此v-on 事件监听器在 DOM 模板中会被自动转换为全小写，无法激活事件。以为是这个问题，转而看插件github样例，发现引入了pug，也跟着尝试pug，发现依然有问题，总而言之，绕来绕去的。

最后是突然意识到这个fixed很奇怪，思考一下发觉问题在哪里了。

两个滚动条也是按照这个思路很快解决了。

所以，每一步代码都要理解了再往下完成。前面逻辑不清楚，只会导致后面一团糟。

**总的代码（只有五十来行）**

