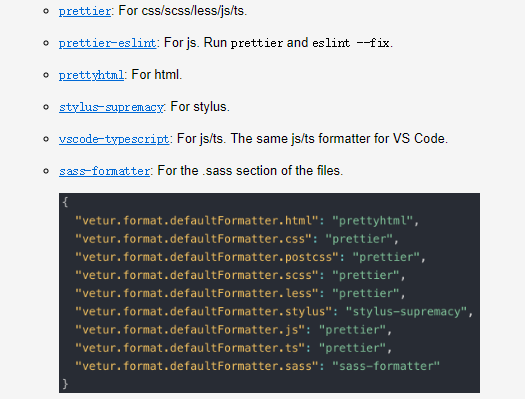
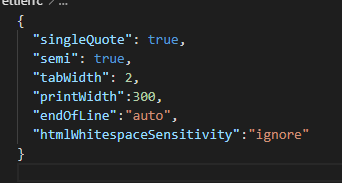
Vscode 配置prettier

1. 前置软件: vscode,
2. 前置插件: Vetur ,prettier（可以在vscode中直接搜索名字下载）
3. ，vetur 默认内置了 如下格式化工具 如果本地安装了相应版本 则会使用本地安装版本、默认配置的 如果在vscode的settings.json 里面没有配置没有关系，下面的配置都是默认的，自行添加后会覆盖原有配置



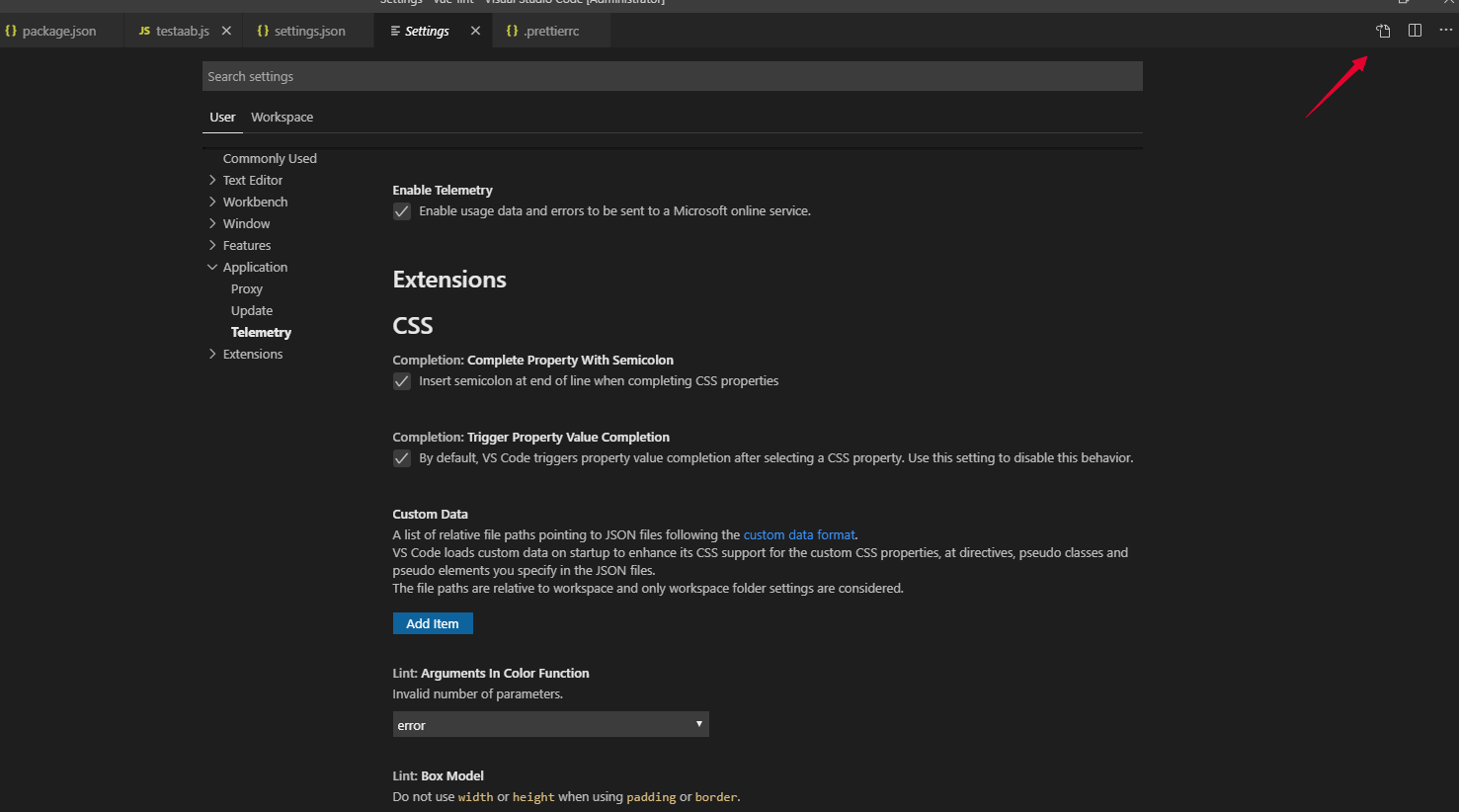
1. 作为项目的整体代码规范，如果依赖本地配置，显然不够科学，所以需要搞一个文件配置 直接引入配置文件就可以做到自动格式化代码。 所以创建 .prettierrc 文件放在根目录上（prettierrc有多种配置格式 目前我采用的是使用json形式的），下图是对.prettierrc 的配置截图. (部分配置都是默认的 配置文件目前只写入覆盖的配置)



注意： vetur 有个问题 就是只对vue文件格式化 js支持不够好 需要用prettier插件了 这里就是用到一开始安装的prettier插件

5.关于保存自动formatter的方法

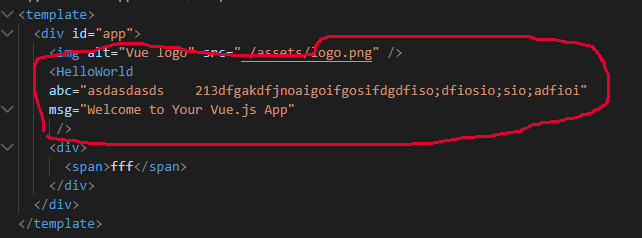
Vscode -> preferences -> settings



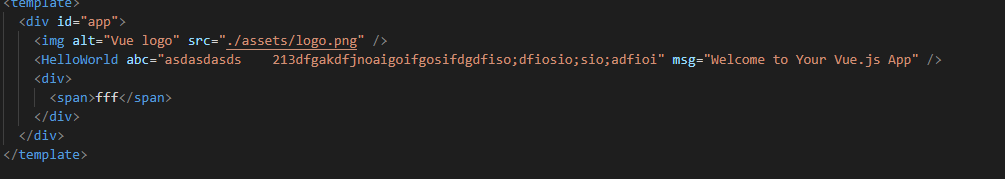


重点关注这两条 没有的话添加这两条配置 即可保存时自动formater， 如果不喜欢自动formatter 可以将editor.formatonsave 设置为false 使用 shift + alt +f 自己手动formater

注：vetur 关于html标签校验默认使用的是 内置插件 prettyhtml，使用这个插件可能会有一个不太舒服的标签格式化格式,在shift+alt+f 美化后即变成这个样子:



当然这样格式化的方式意在突出标签属性的重要性 使其看起来更为舒服。如果不习惯这种方式的话 可以改变成我们常见的格式即:

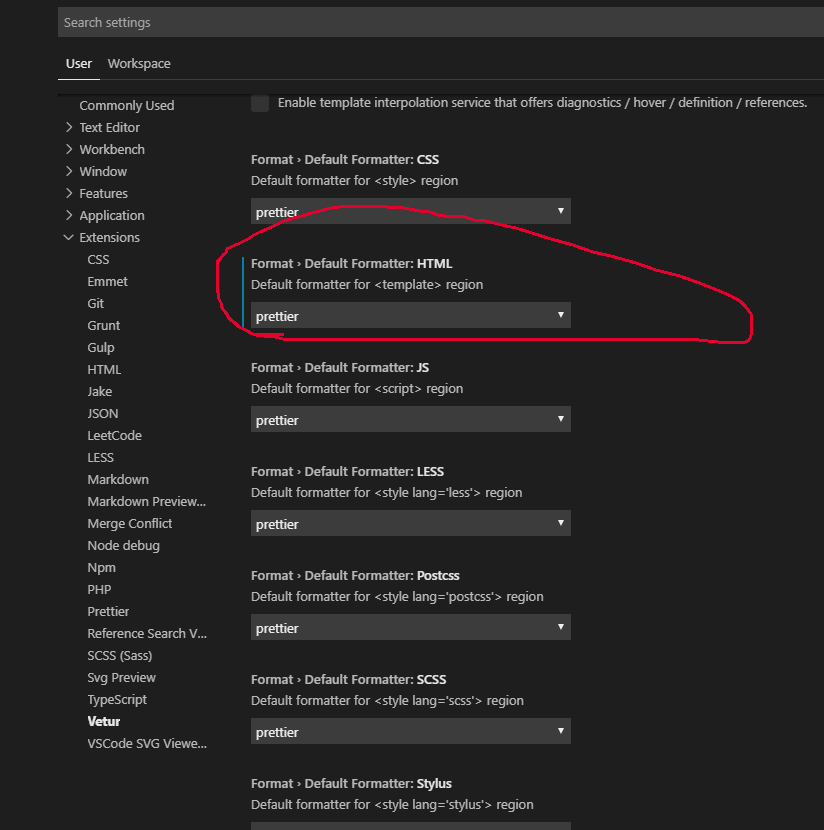


这里需要如下步骤: 1. File-> preferences - > setting -> extensions -> vetur

打开之后 找到

Format> default formatter:HTML

将默认的校验插件 prettyhtml 改为prettier 即可完成



Prettier格式化配置 <https://prettier.io/docs/en/options.html>

Vetur 格式化及相关配置 <https://vuejs.github.io/vetur/>