



天平 Lv2

2021年07月31日 阅读 1071

关注

【建议收藏】全网最全的讲清eslint和prettier的npm包和vscode插件的文章

一直在研究eslint和prettier，网上看了很多文章，发现要么是互相抄袭，要么就是讲得不够通俗易懂，于是乎我就自己写一篇文章来讲清它们的关系。

大家可以认真的看一下，全程精品，毫无废话。

我们都知道它们是用来做代码格式化的，但是又分为npm包和vscode插件，到底要怎么使用它们？怎么配置呢？

顺序是——先有npm包，再有vscode插件。

先从eslint讲起，eslint不仅作代码规范，还有检查变量声明了是否调用，是否有console.log语句等。

废话不多说，直接上手来演示一下。

(如果有安装格式化工具的同学，先把工具禁用了，否则会影响演示效果)

eslint的npm包

简单的用法，先 `npm init -y`，初始化包。再建一个index.js文件，写点东西。

js 复制代码

//index.js文件

```
var a = 1
```

然后全局安装eslint `npm i eslint -g`（如果之前全局安装过了就不用了），再 `npm run eslint --init`，接着会出现几个选择，因为是测试，所以随便选，一直回车就行了。

选完之后会自动下载eslint包，并在项目的根目录自动创建 `.eslintrc.js` 文件。

此时准备工作已经做好了，我们可以开始演示了。先执行 `eslint index.js`

这时，终端会报错，说定义了几个变量，但没有使用它们。这就是最基本的用法。

```
E:\test\prettier-eslint\index.js
1:5  error  'a' is assigned a value but never used  no-unused-vars
2:8  error  'str' is assigned a value but never used no-unused-vars

✖ 3 problems (3 errors, 0 warnings) @掘金技术社区
```

但我上面书写的格式风格都不同，eslint并没有检测出来。所以我们要去配置 `.eslintrc.js` 文件

因为里面已经生成好了，所以这里只放出rules部分

0是忽略，1是警告，2是报错

js 复制代码

```
//.eslintrc.js

"rules": {
  "quotes": 2,
  "semi": 1,
  "no-console": 1,
  "space-before-function-paren": 0
}
```

```
E:\test\prettier-eslint\index.js
1:5   error    'a' is assigned a value but never used    no-unused-vars
1:10  warning   Missing semicolon                        semi
2:8   error    'str' is assigned a value but never used  no-unused-vars
2:18  warning   Missing semicolon                        semi
3:2   warning   Unexpected console statement             no-console
3:14  error    Strings must use doublequote              @juejin技术社区
```

格式的问题也被检测出来了。

那么我要怎么修复它们呢？执行 `eslint index --fix`

这样js文件里面的内容的格式就被修复好了，比如结尾要不要分号，用单引号还是双引号。

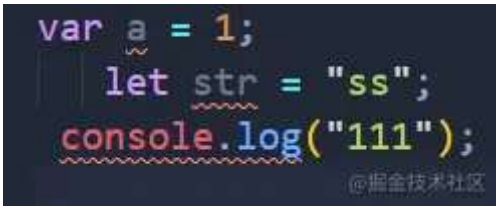
```
var a = 1;
let str = "ss";
console.log("111");
@juejin技术社区
```

eslint的vscode插件

我们每次做项目的时候，都要先 `eslint . --fix` 统一格式，再push提交代码。或者是把指令添加到package.json文件中，执行 `npm run lint`，再push。这样就很麻烦，并且也不够直观，因为错误问题只有在你执行了指令之后才能看到。

那么我想在我一边写代码的时候就能直接看到错误，然后就能直接随手改正错误，这要怎么做呢？ 这时eslint的vscode插件诞生了。





接着去配置自动保存，到vscode的 `setting.json` 下增加以下语句：

//setting.json文件

```
"editor.formatOnType": true,
"editor.formatOnSave": true,
"eslint.codeAction.showDocumentation": {
  "enable": true
},
"editor.codeActionsOnSave": {
  "source.fixAll.eslint": true,
},
"eslint.validate": ["javascript", "javascriptreact", "html"]
```

js 复制代码

这样每次保存文件的时候，就能看到错误信息了，并且自动改正了。（这里只是演示，后期因为和prettier的效果冲突了，要删除）

既然有vscode插件，那么还装eslint的npm包吗？

要装。虽然vscode插件也可以单独配置格式，但是如果项目中有 `.eslintrc.js` 文件，那么eslint插件会优先执行 `.eslintrc.js` 文件的配置。

并且不是每个人都会装eslint的vscode插件。此时eslint的npm包就作为一个保

prettier的npm包

既然eslint能做格式化，那为什么还要prettier呢？

聪明的小伙伴肯定发现了，执行了 `eslint index.js --fix` 之后，js文件只是增加了分号，双引号。但是代码的格式该怎么乱还是怎么乱，即不美观，也不工整。并且eslint只能作用于js文件，像html, css, json, vue文件，eslint都处理不了他们的代码格式的问题。此时prettier认为代码的格式也很重要，于是prettier诞生了。

下载prettier `npm i prettier -D`，然后执行 `npx prettier --write index.js`，我们可以发现刚才不工整的代码，全部都变得工工整整了。

```
var a = 1;
var str = "ss";
console.log(a, str);
```

@掘金技术社区

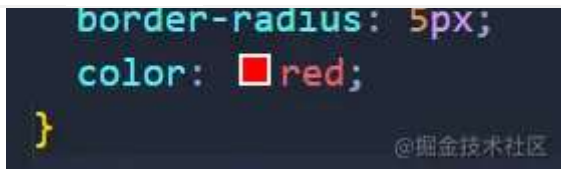
并且我们新建一个 `index.css` 文件，格式要多丑有多丑。

```
a {
    font-size: 15px;
    border-radius: 5px;

    color: red;
}
```

@掘金技术社区

执行完 `npx prettier --write index.css`，css文件也变得工整了，解决了eslint解决不了的问题。



```
border-radius: 5px;  
color: ■ red;  
}
```

@掘金技术社区

接着可以自定义代码格式配置，在项目的根目录创建 `.prettierrc.js` 文件。
(如果是 `.prettierrc` 文件，则是json类型，要按json格式来配置)

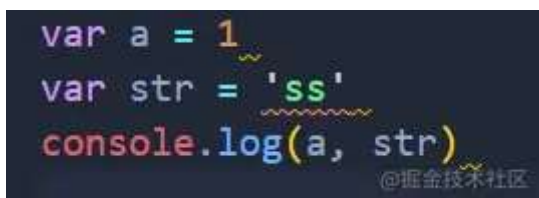
js 复制代码

//.prettierrc.js文件

```
module.exports = {  
  semi: false,  
  singleQuote: true,  
}
```

再执行 `npx prettier --write index.js`，我们可以发现格式改成了单引号，并且结尾没有了分号。

(这里的波浪线是刚才eslint插件的效果，因为prettier的配置与eslint的配置冲突了。但是一Ctrl+S保存，又会改成eslint的配置格式。**所以eslint和prettier的配置要统一，不能冲突。**)



```
var a = 1  
var str = 'ss'  
console.log(a, str)
```

@掘金技术社区

prettier的vscode插件

还是和eslint一样的老问题，我需要每次执行 `npx prettier --write .` 才会修改格式。我想要在我写代码的时候，一Ctrl+S保存就能自动格式化代码，需要怎么做？这时prettier的vscode插件诞生了。



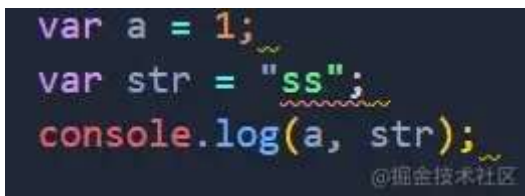
安装完后再配置一下prettier自动保存：

js 复制代码

//setting.json文件

```
"[javascript]": {  
  "editor.defaultFormatter": "esbenp.prettier-vscode"  
},  
"[html]": {  
  "editor.defaultFormatter": "esbenp.prettier-vscode"  
},
```

然后就会出现一个非常奇怪的现象：



一Ctrl+S保存，先是自动加上分号和双引号，然后又没了。这还是因为eslint插件自动保存的效果和prettier插件自动保存的效果冲突了。

一保存，先是执行了eslint的配置，然后再执行prettier的配置。完蛋，不管前面的eslint设置得多好，只要和prettier的设置冲突了，就直接被prettier覆盖了。

解决办法：

动保存就行了。不过eslint和prettier的配置也要保持一致。
不想关闭eslint的自动保存也行，只要 `.eslintrc.js` 的配置和 `.prettierrc.js` 的配置严格一致就行了。

配置

关于 `.eslintrc.js` 和 `.prettierrc.js` 的配置，按照团队的需求自定义，或者在网上找一份配置就行了。因为大家的配置都不一样，我也不展示了。

稍微提一下，eslint默认是要双引号，加分号的。而prettier默认是单引号，不加分号的。所以这两个默认配置会冲突。我个人也是喜欢单引号和不加分号，所以我一般会把eslint的这两个配置关了，这样eslint的自动保存也会和prettier的自动保存一致。

并且eslint不加上node的话，关于module.exports的，老是显示报错，但实际上写得没有问题，只是eslint的问题。



js 复制代码

```
//.eslintrc.js

module.exports = {
  env:{
    browser:true,
    es2021:true,
    node:true
  },
  rules:{
    quotes:0,
    semi:0
  },
}
```


总结

eslint是做js的代码格式化的，但是扩展性很强，包括变量是否使用，是否要console.log。而prettier是做所有代码的格式化，并且只专注于格式化，范围更广。

vscode的插件可以单独配置，单独起作用。但是如果项目根目录中有`.eslintrc.js`和`.prettierrc.js`文件，则以`.eslintrc.js`和`.prettierrc.js`的配置为标准，执行它们的配置。

prettier的效果在eslint的效果之后，如果配置冲突了，则prettier会覆盖eslint的配置。

问题来了

package.json中的依赖有eslint和prettier，那么做项目`npm i`的时候，每个人都是会自动下载安装eslint和prettier的，这是必须的。但vscode插件可不一定，也就是说vscode插件不是必须的，可装可不装，只是方便我们自己开发而已。

那么假设一个人不喜欢这些vscode插件扩展，就喜欢简简单单的开发，那么我们要统一代码规范，要怎么配置使项目一跑起来就能自动运行eslint和prettier达到不依靠vscode插件自动格式化的效果呢？

下期再研究。

我在b站录制了视频给大家补充观看：[🔗【建议收藏】全网最全的讲清eslint和prettier关系的视频!!! 前端开发的利器!!!](#)

[前往安装](#)

多内容聚合浏览、多引擎快捷搜索、多工具便捷提效、多模式随心畅享，你想要的，这里都有！

输入评论 (Enter换行, Ctrl + Enter发送)

发表评论

热门评论



用户568677783153 1月前

用vue-cli 创建选择的lint模式是 eslint + prettier，用的是vscode编辑器，新增了prettier.config.js 设置无分号和单引号 模式，保存的时候自动按照prettier规则保存了，但是代码还是提示用双引号和分号结尾

点赞 1

天平 Lv2 (作者) 1月前

prettier只会自动格式化，不会报错。报错的是eslint，说明eslint的配置和prettier冲突了。去修改eslint的配置。

点赞 回复

全部评论 (6)

最新

最热



啥都学的交互 UI设计师/前端工程师 23天前

得出结论，关闭eslint就世界和平

点赞 回复



用户568677783153 1月前

package.json配置

```
testDependencies: {
  "@vue/cli-plugin-babel": "~4.5.0",
  "@vue/cli-plugin-eslint": "~4.5.0",
  "@vue/cli-plugin-router": "~4.5.0",
  "@vue/cli-plugin-vuex": "~4.5.0",
  "@vue/cli-service": "~4.5.0",
  "@vue/eslint-config-prettier": "^6.0.0",
  "babel-eslint": "^10.1.0",
  "eslint": "^6.7.2",
  "eslint-plugin-vue": "^7.0.1",
  "eslint-plugin-prettier": "^6.2.2",
  "prettier": "^2.1.3",
  "vue-template-compiler": "^2.6.11"
}
```

👍 点赞 🗨 回复

用vue-cli 创建选择的lint模式是 eslint + pretter , 用的是vscode编辑器, 新增了prettier.config.js 设置无分号和单引号 模式, 保存的时候自动按照prettier规则保存了, 但是代码还是提示用双引号和分号结尾

👍 点赞 🗨 1



天平 Lv2 (作者) 1月前

prettier只会自动格式化, 不会报错。报错的是eslint, 说明eslint的配置和prettier冲突了。去修改eslint的配置。

👍 点赞 🗨 回复



PROMISE 1月前

捉

👍 点赞 🗨 1



天平 Lv2 (作者) 1月前



👍 点赞 🗨 回复

相关推荐

小兀666 2年前 ESLint

最全的Eslint配置模板, 从此统一团队的编程习惯

随着项目的不断增加, 急切需要统一每个项目的代码规范, 将一些低级错误在萌芽状态下掐死。所以特此结合当前项...

1.7w 191 8

蔡小真 9月前 ESLint

不以规矩, 不能成方圆-彻底搞懂 ESLint 和 Prettier

1. 为什么写这篇文章? 2020 年 3 月, 懵懂的蔡小真怀揣着对技术的向往无比乐观地进入...

4297 149 16



FrankCheung 3年前 Vue.js 前端 图片资源

Vue项目使用eslint + prettier规范代码风格

团队开发的项目, 如果没有对代码风格作要求, 有多少团队成员, 就当然会出现多少种不同的代码风格。因此, 我们...

1.6w 154 2

杨村长 9月前 前端

3.3w 1173 128

ikoala 2年前 Vue.js HTML

vue中8种组件通信方式, 值得收藏!

作者简介: koala, 专注完整的 Node.js 技术栈分享, 从 JavaScript 到 Node.js,再到后端...

7.9w 1683 76



Ethan_Zhou 2年前 ESLint

使用 ESLint & standard 规范前端代码

JavaScript的动态语言类型, 给它带来了独特的魅力, 产生了风格多样的开发范式, 同时也带来了一些问题, 从运行...

5126 10 评论

卖油条的。 10月前 代码规范

前端代码规范工具原理和最佳实践: eslint+prettier+gitHooks

阻止不规范代码进入代码库, 规范提交信息。 本文会从依据各工具官方文档等分别对其原理和使用进行解读, 并给...

498 7 评论

wfz 2年前 代码规范

一步一步, 统一项目中的编码规范 (vue, vscode, vetur, prettier, eslint)

团队开发中, 多人开发同一个项目, 由于个人编码习惯不同, 一个项目中最终的代码风格可能差别很大, 所以需要通...

1.3w 162 28

YDJFE 3年前 SCSS TypeScript Visual Studio Code

vscode + prettier 专治代码洁癖 (一)

相信很多小伙伴都纠结过自己代码的格式化问题, 特别是团队协作的时候, 加上前端语言种类繁多, 各种语法糖+团...

6.1w 354 28

joking_zhang 2年前 React.js

如何在 React 项目中整合 ESLint 和 Prettier

当我们来到一个新的团队, 或者接手一个新的项目时, 一定会遇到代码规范的问题。随着前端应用越来越复杂, 代码...

7239 42 19

云影sky 2年前 ESLint

使用 ESLint + Prettier + husky + lint-staged 提高前端项目质量、统...

前端代码风格因人而异, 一个项目参与的人多了, 不加强控制可能就是一个大杂烩, 对开发...

夏末海棠 12天前 前端

vscode+eslint+prettier自动格式化配置

vscode基本上时每一个前端工作者的必备了 不用vscode请略过这句话 ~ ~ ~ 下面就vscode根据eslint+prettier 自...

141 1 评论

xcatliu 3年前 ESLint TypeScript React.js

如何用 ESLint 检查 TypeScript 代码

ESLint 是一个代码检查工具，主要用来发现代码错误、统一代码风格，目前已被广泛的应用于各种 JavaScript 项目...

2.1w 125 16

神三元 1年前 JavaScript 面试

(建议收藏)TCP协议灵魂之问，巩固你的网路底层基础

TCP 作为传输层的协议，是一个软件工程师素养的体现，也是面试中经常被问到的知识点。在此，我将 TCP 核心的...

5.7w 1500 95

神三元 1年前 JavaScript 前端

(建议收藏)原生JS灵魂之问, 请问你能接得住几个? (上)

笔者最近在对原生JS的知识做系统梳理，因为我觉得JS作为前端工程师的根本技术，学再多遍都不为过。打算来做一...

12.8w 3049 226

王仕军 3年前 前端 开源 VIM

能让你开发效率翻倍的 VSCode 插件配置 (中)

如果说美食需要美器的衬托，优雅美观的代码编辑器会大大提升工程师的编码体验、工作幸...

4.1w 1399 32

向日葵同志44330 2年前 Visual Studio Code

vscode-settings.json配置 (prettier + eslint)

1、设置全局settings.json文件command+p搜索到settings.json文件，文件配置及注释如下2、设置项目自己的规...

7352 11 评论

上沅同学 2年前 前端

ES6、ES7、ES8、ES9、ES10新特性一览

ECMA规范最终由TC39敲定。TC39由包括浏览器厂商在内的各方组成，他们开会推动JavaScript提案沿着一条严格...

18.7w 2824 132

从项目规范（eslint + prettier）到自动化配置

前言：为什么需要项目规范 我们学习编程的方式各有不同，对于知识的理解也各有不同，在一天天的编程过后，每...

1423

28

11

JowayYoung

1年前

JavaScript

ECMAScript 6

1.5万字概括ES6全部特性(已更新ES2020)

第三次阅读阮一峰老师的《ES6标准入门》了，以前阅读时不细心，很多地方都是一目十行...

17.6w

4767

217