

配置 ts eslint prettier

- * 首先要安装 eslint，eslint 默认使用 `Espree` 进行解析，无法识别 ts 的一些语法，所以需要安装一个 ts 的解析器 `@typescript-eslint/parser`，用它来代替默认的解析器，然后由 `@typescript-eslint/eslint-plugin` 来提供有关 ts 的规则补充。
- * 由于是 react 项目，所以还需要插件 `eslint-plugin-react` 来支持 `.tsx`。
- * eslint 规则众多，且有些原生规则在 ts 中不兼容，推荐 alloy 的这套配置 `eslint-config-alloy`，它提供了 ts + react 的版本，并且不包含代码格式的部分，与 Prettier 完全兼容。
- * 最后是提供给 webpack 的 `eslint-loader`，可以在使用 `webpack-dev-server` 开发中实时检查，越早发现错误越好解决。

```
# eslint tyepscript prettier
yarn add eslint -D
yarn add typescript -D
# eslint TS 解析器 和 TS 规则补充
yarn add @typescript-eslint/parser -D
yarn add @typescript-eslint/eslint-plugin -D
# eslint React 补丁 (用于支持 .tsx)
yarn add eslint-plugin-react -D
# eslint Config-alloy 配置补丁
yarn add eslint-config-alloy -D
# eslint loader 配置补丁
yarn add eslint-loader -D
# eslint prettier 补丁
yarn add prettier -D
yarn add eslint-plugin-prettier -D
# eslint prettier 配置补丁(解决插件冲突)
yarn add eslint-config-prettier -D
```

PowerShell ▾

添加 TS 配置 文件 .tsconfig.json

```
{
  "compilerOptions": {
    "module": "commonjs",
    "noImplicitAny": true,
    "sourceMap": false,
    "outDir": "dist",
    "baseUrl": ".",
    "paths": {
      "*": ["node_modules/*"]
    }
  },
  "include": [
    "src/**/*"
  ]
}
```

JSON ▾

添加 eslint 配置文件 .eslintrc.json

```
{
  "parser": "@typescript-eslint/parser",
  "plugins": [
    "@typescript-eslint/eslint-plugin"
  ],
  "extends": [
    "alloy",
    "alloy/react",
    "alloy/typescript",
    "plugin:prettier/recommended"
  ],
  "settings": {
    "react": {
      "version": "detect"
    }
  }
}
```

```
    },
    "env": {
      // 环境变量（包含多个预定义的全局变量）
      // Your environments (which contains several predefined global variables)
      //
      // browser: true,
      // node: true,
      // mocha: true,
      // jest: true,
      // jquery: true
    },
    "globals": {
      // 全局变量（设置为 false 表示它不允许被重新赋值）
      // Your global variables (setting to false means it's not allowed to be reassigned)
      //
      // myGlobal: false
    },
    "rules": {
      // 自定义规则
      // Customize your rules
    }
  }
}
```

JSON ▾

配置 .prettierrc.json

```
{
  "printWidth": 80,
  "tabWidth": 2,
  "singleQuote": true,
  "semi": false,
  "useTabs": true,
  "jsxSingleQuote": true,
  "arrowParens": "avoid"
}
```

JSON ▾

添加 script 指令

```
// package.json

{
  "scripts": {
    "lint": "eslint '{components,example}/**/*.ts,tsx,js,jsx'",
    "lint_fix": "eslint '{components,example}/**/*.ts,tsx,js,jsx' --fix"
  }
}
```

JSON ▾

给 webpack 添加 eslint-loader

```
// webpack

module: {
  rules: [
    {
      test: /\.jsx|js|ts|tsx$/,
      include: [
        path.resolve(__dirname, '../components'),
        path.resolve(__dirname, '../example')
      ],
      exclude: [/node_modules/],
      use: ['eslint-loader'],
      enforce: 'pre'
    }
  ]
}
```

JavaScript ▾