Configure Prettier and ESLint for Angular

Table of Contents

Install and configure ESLint

1. Add `eslint` using angular schematics.

   ```
   $ ng add @angular-eslint/schematics
   ```

2. To find out lint issues run below command

   ```
   $ ng lint
   ```

3. To fix all the fixable bugs in the project.

   ```
   $ ng lint --fix
   ```

Install and configure Prettier

1. Install Prettier

   ```
   $ yarn add prettier -D
   ```

2. Add `.prettierignore` file with following content.

   .prettierignore

   ```
   package.json
   CHANGELOG.md
   README.md
   ```

TIP

Inside `.prettierignore` it’s better to add whatever we have inside `.gitignore` file.

3. Add `.prettierrc` file with following content.

.prettierrc

```
{
  "arrow-body-style": "off",
  "prefer-arrow-callback": "off",
  "printWidth": 120,
  "singleQuote": true,
  "trailingComma": "none",
  "endOfLine": "auto",
  "useTabs": false,
  "bracketSameLine": true,
  "arrowParens": "avoid",
  "bracketSpacing": true
}
```

4. Run the below command to format project.

```
$ npx prettier --write
```

Integrate Prettier with ESLint plugin.

1. Install prettier and eslint plugin.

```
$ yarn add prettier-eslint eslint-config-prettier eslint-plugin-prettier -D
```

2. Edit the .eslintrc.json file to include the prettier plugin.

```
{
  "root": true,
  "ignorePatterns": [
    "projects/**/*"
  ],
  "overrides": [
    {
      "files": [
        "*.ts"
      ],
      "parserOptions": {
        "project": [
          "tsconfig.json"
        ],
        "createDefaultProgram": true
      },
      "extends": [
        "plugin:@angular-eslint/recommended",
        "plugin:@angular-eslint/template/process-inline-templates",
        "plugin:prettier/recommended" ①
      ],
      "rules": {
        "@angular-eslint/directive-selector": [
          "error",
          {
            "type": "attribute",
            "prefix": "k",
            "style": "camelCase"
          }
        ],
        "@angular-eslint/component-selector": [
          "error",
          {
            "type": "element",
            "prefix": "k",
            "style": "kebab-case"
          }
        ],
        ②
        "sort-imports": [
          "error",
          {
            "ignoreCase": false,
            "ignoreDeclarationSort": false,
            "ignoreMemberSort": false,
            "memberSyntaxSortOrder": ["none", "all", "multiple", "single"],
            "allowSeparatedGroups": true
          }
        ]
        ③
      }
    },
    {
      "files": [
        "*.html"
      ],
      "extends": [
        "plugin:@angular-eslint/template/recommended"
      ],
      "rules": {}
    },
    ④
    {
      "files": ["*.html"],
      "excludedFiles": ["*inline-template-*.component.html"],
      "extends": ["plugin:prettier/recommended"],
      "rules": {
        "prettier/prettier": ["error", { "parser": "angular" }]
      }
    }
    ⑤
  ]
}
```

① Add plugin

② Add additional import rule start.

③ Add additional import rule end.

④ Start of html rule configuration.

⑤ Enad of html rule configuration.

Enable Prettier for HTML and typescript in `VS Code`.

- Create/update `.vscode/settings.json` file to use `prettier` formatter for html and typescripts.

```
{
  "[html]": {
    "editor.defaultFormatter": "esbenp.prettier-vscode",
    "editor.codeActionsOnSave": {
      "source.fixAll.eslint": true
    },
    "editor.formatOnSave": false
  },
  "[typescript]": {
    "editor.defaultFormatter": "esbenp.prettier-vscode",
    "editor.codeActionsOnSave": {
      "source.fixAll.eslint": true
    },
    "editor.formatOnSave": false
  },
}
```