**[ES Lint](https://eslint.org/)**

JavaScript linter that enables you to enforce a set of style, formatting, coding standards for your codebase.

**Prerequisites**

* A code editor installed (VS Code )
* An existing codebase (node js project + a typescript)

## Installation and setup

1. Run the following command

npm install --save-dev eslint @typescript-eslint/parser @typescript-eslint/eslint-plugin

1. Create an *eslintrc*file
2. In it, use the following starter config.

{

"root": true,

"parser": "@typescript-eslint/parser",

"plugins": [

"@typescript-eslint"

],

"extends": [

"eslint: recommended",

"plugin:@typescript-eslint/eslint-recommended",

"plugin:@typescript-eslint/recommended"

]

}

### Ignore files we don't want to lint in the *eslintignore* file

### Adding a lint script in package.json

"scripts": {

"lint": "eslint. --ext. ts",

}

1. Run the following command.

npm run lint

## Rules

There are three modes for a rule in eslint: off, warn and error.

"off" means 0 (turns the rule off completely)

"warn" means 1 (turns the rule on but won't make the linter fail)

"error" means 2 (turns the rule on and will make the linter fail)

**Adding a rule**

In *eslintrc*, add a new attribute to the JSON object called "*rules*".

**Rule 1**. Add the[***no-console***](https://eslint.org/docs/rules/no-console)rule, if it encounters a console.log statement with the no-console rule set to error.

"rules": {

"no-console": 2

}

we could turn the rule off by setting it to 0- off.

"rules": {

"no-console": 0

}

# **Rule 2.** Add the ***import/newline-after-import*** rule, ‘count’ which sets the number of newlines that are enforced after the last top-level import statement or require call. This option defaults to 1.

"rules": {

"import/newline-after-import": [“error”, {“count”: 2}]

}

For this, install the plugin

npm install - eslint-plugin-import

Add the following to the *eslintrc* file

"plugins": [

"@typescript-eslint", "eslint-plugin-import"

],

"extends": [

"plugin: eslint-plugin-import/typescript"

]

# **Rule 3.** Add the @typescript-eslint/type-annotation-spacing rule, which determines a space to put after ‘:’ in type annotations.

"rules": {

"@typescript-eslint/type-annotation-spacing": ["warn", {

          "before": false,

          "after": true

        }]

    }

**Rule 4.** Add the semicolon rule. It takes one or two options. The first one is a string, which could be "always" or "never". The default is "always". The second one is an object for more fine-grained configuration when the first option is "always".

When setting the first option as "always", an additional option can be added to omit the last semicolon in a one-line block, that is, a block in which its braces (and therefore the content of the block) are in the same line:

"rules": {

"semi”: [2, "always", { "omitLastInOneLineBlock": true}]

}