



Creating Custom ESLint Shareable Configs

One of [ESLint](#)'s strongest points is its plugin system. You can use existing shareable configurations or create your own rules.

I've been meaning to research the process and implement a few rules and a plugin to go with them and think now it's the time to do it.

Creating a Shareable Configuration

I will model my shareable configuration after Google's ([eslint-config-google](#)) since it's the one I use the most and the one that has all the rules I follow when writing Javascript.

The idea behind a shareable configuration is to create a Node module, write the rules you want to use and publish it to NPM so others members of the community can use it.

The first step is to initialize the Node module:

```
npm init --yes
```

This will prefill the `package.json` file with information from your Git settings. You can always edit it later.

The next step is to write the rules that you want to use and how to handle them (error or warning) in your default entry point, in this case, `index.js`.

All the rules are wrapped in a `rules` object that lives `module.exports`, this way you can export multiple rules.

```
module.exports = {  
  rules: {}  
};
```

Inside the rules object you can add as many rules as you want.

```
module.exports = {  
  rules: {  
    'curly': ['error', 'multi-line'],  
    'guard-for-in': 'error',  
    'no-caller': 'error',  
    'no-extend-native': 'error',  
    'no-extra-bind': 'error',  
    'no-invalid-this': 'error',  
    'no-multi-spaces': 'error',  
    'no-multi-str': 'error',  
    'no-new-wrappers': 'error',  
    'no-throw-literal': 'error',  
    'no-with': 'error',  
    'prefer-promise-reject-errors': 'error',  
  }  
}
```

If you've created your own plugin, you can add it to the rules object, assuming you've loaded it in your `.eslintrc` configuration file.

Links and References

- [Shareable Configs](#)
- [Working with Rules](#)
- [Selectors](#)
- [AST selectors rule](#)
- [How To Write Your First ESLint Plugin](#)
- [Yeoman](#)
- [Yeoman ESLint generator](#)