




# Husky, lint-staged를 활용한 pre-commit Validation

🕒 작성일	@2023년 3월 13일 오후 1:52
🕒 수정일	@2023년 5월 31일 오후 1:27
👤 작성자	 행크

## 목적

코드를 작성하고 commit 시에 해당 코드가 eslint의 규칙이 지켜졌는지 체크하고 지켜지지 않은 경우

해당 commit을 받아들이지 않고 에러를 출력하여 규칙을 지키도록 강제

## 설치

```
npm install husky lint-staged prettier
npm init @eslint/config
```

## lint-staged

```
npx lint-staged
```

git add를 통해서 stage된 상태의 파일들에 대해서 linting을 진행한다.

## 설정

package.json

```
"lint-staged": {
  "**/*.js,ts,jsx,tsx,html,css": "prettier --write",
  "**/*.js,ts,jsx,tsx": "eslint"
},
```

npx lint-staged를 실행하면 package.json에 설정해둔 대로 작업을 실행한다

prettier —write는 prettier rule대로 파일 내용 변경 옵션

js,ts 파일들에만 eslint 실행

## Husky

git hook 사용을 도와주는 라이브러리다.

Git도 다른 버전 관리 시스템처럼 어떤 이벤트가 생겼을 때 자동으로 특정 스크립트를 실행하도록 할 수 있다. 이 혹은 클라이언트 혹은 서버 혹은 나눌 수 있다. 클라이언트 혹은 커밋이나 Merge 할 때 실행되고 서버 혹은 Push 할 때 서버에서 실행된다.

husky 설정을 통해서 pre-commit 이벤트 발생 시 lint-staged를 실행하여 stage된 파일들을 검사하도록 한다.

```
npx husky-init && npm install
npx husky add .husky/pre-commit "npm test"
```

husky폴더 안에 pre-commit hook을 생성하고 내용은 npm test로 하도록 생성

.husky/pre-commit

```
#!/usr/bin/env sh
. "$(dirname -- "$0")/_/husky.sh"

npx lint-staged
```

npm test로 되어있는 내용을 npx lint-staged로 변경

```

husky - updated .husky/pre-commit
• → rn-husky git:(main) ✖ git add .
• → rn-husky git:(main) ✖ git commit -m"checking"
✓ Preparing lint-staged...
> Running tasks for staged files...
  > package.json - 5 files
    ✓ ./**/*. {js,ts,jsx,tsx,html,css} - 1 file
    > ./**/*. {js,ts,jsx,tsx} - 1 file
      ✖ eslint [FAILED]
  ↓ Skipped because of errors from tasks. [SKIPPED]
  ✓ Reverting to original state because of errors...
  ✓ Cleaning up temporary files...

✖ eslint:
Warning: React version not specified in eslint-plugin-react settings. See https://github.com/jsx-esl
int/eslint-plugin-react#configuration .

/Users/seednpc03/rn-husky/App.tsx
5:3  error    Unexpected var, use let or const instead  no-var
5:7  warning  'a' is defined but never used             @typescript-eslint/no-unused-vars
7:5  error    'React' must be in scope when using JSX  react/react-in-jsx-scope
8:7  error    'React' must be in scope when using JSX  react/react-in-jsx-scope
9:7  error    'React' must be in scope when using JSX  react/react-in-jsx-scope

✖ 5 problems (4 errors, 1 warning)
  1 error and 0 warnings potentially fixable with the `--fix` option.

husky - pre-commit hook exited with code 1 (error)

```

커밋 시에 나오는 에러

error와 warning이 있는 경우 commit이 안되도록 막는다.

에러 해결 후 commit하면 성공적으로 commit 된다

```

• → rn-husky git:(main) ✖ git add .
• → rn-husky git:(main) ✖ git commit -m"error fixed"
✓ Preparing lint-staged...
✓ Running tasks for staged files...
✓ Applying modifications from tasks...
✓ Cleaning up temporary files...
[main 483733a] error fixed
6 files changed, 24598 insertions(+), 6915 deletions(-)
create mode 100644 .eslinttrc.json
create mode 100755 .husky/pre-commit
create mode 100644 package-lock.json
delete mode 100644 yarn.lock

```

eslint-plugin-react settings warning 해결 및 기타 에러 예방

.eslinttrc.json 에 추가

```

"settings": {
  "react": {
    "version": "detect"
  }
}

"rules": {
  "react/react-in-jsx-scope": ["off"],
  "react/jsx-props-no-spreading": ["off"],

```

```
"react/jsx-uses-react": ["off"]
},
```

## ESLint, Typescript, Prettier설정

### prettier

.prettierrc파일 생성 없이 기본 extension 설정

### ESLint

```
{
  "env": {
    "browser": true,
    "es2021": true
  },
  "extends": [
    "eslint:recommended",
    "plugin:react/recommended",
    "plugin:@typescript-eslint/recommended"
  ],
  "overrides": [],
  "parser": "@typescript-eslint/parser",
  "parserOptions": {
    "ecmaVersion": "latest",
    "sourceType": "module"
  },
  "plugins": ["react", "@typescript-eslint"],
  "rules": {
    "react/react-in-jsx-scope": ["off"],
    "react/jsx-props-no-spreading": ["off"],
    "react/jsx-uses-react": ["off"]
  },
  "settings": {
    "react": {
      "version": "detect"
    }
  }
}
```

settings 및 rules 변경

expo init 시에 blank typescript 앱 생성으로 typescript plugin기본 생성

prettier format on save, lint staged —write 설정으로 prettier plugin추가는 필수적이지 않습니다.

# TypeScript

```
{
  "extends": "expo/tsconfig.base",
  "compilerOptions": {
    "strict": true
  }
}
```

expo/tsconfig 기본 설정