



# ESLint

outil d'analyse



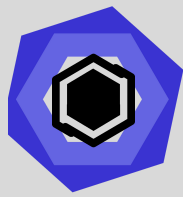
EVITE LES BEUGS AU RUNTIME



UNIFORMISE LA SYNTAXE ENTRE PLUSIEURS  
DEVELOPPEURS



GAIN DE TEMPS, IMPOSE LE RESPECT DE  
CERTAINES REGLES. FORMATE LE CODE



# ESLINT

UNE ERREUR C EST GLISSEE DANS CE CODE  
SAURAI-TU LA RETROUVER?

```
for (const i=0; i<3; i++){  
  console.log(i)  
}
```

```
let reader = "Rayed"  
if (reader = "Samy") {  
  console.log('Hello Samy')  
} else {  
  console.log('Hello ${reader}')}
```

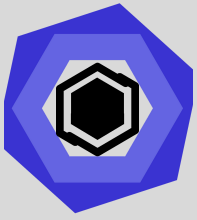
AVANT/APRES



```
// Avant ESLint  
function sum (a, b) {  
  return a+b  
}
```

```
// Après ESLint  
function sum = (a,b)=> a+b
```





## INSTALLATION

À CHAQUE  
NOUVEAU PROJET

**WARNING : A Installer une seule fois par PC :**  
**NODE.JS** (qui installera automatiquement le module 'npm')  
et le module **ESLint** sur VS Code

### 01 Créer le fichier **PACKAGE.JSON** grâce à la commande ci dessous

```
npm init
```

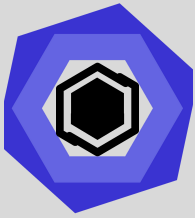
### 02 Ouvrir le fichier **PACKAGE.JSON** et supprimer la ligne contenant : <"main" : "index.js"

**Dans le terminal :**

### 03 `npm install eslint --save-dev` */\*\* Pour installation locale*



### 04 `npx eslint --init` */\*\* Lancer la configuration*



## CONFIGURATION

### Le Fichier .eslintrc.json est créé :

```
{
  "extends": "eslint:recommended",
  "rules": {
    "semi": ["error", "always"],
    "quotes": ["error", "double"]
  }
}
```

CONFIGURATION STANDARD

04

Rgles definies par google, airbnb...

Dans ce cas , les "rules" vont signaler une erreur :  
pour absence de point virgule ,  
et si une string est déclarée en utilisant des simples guillemets .

```
{
  "extends": "eslint:recommended",
  "rules": {
    "semi": ["error", "always"],
    "quotes": ["error", "double"]
  },
  "env": {
    "browser": true,
    "es2021": true
  },
  "rules": {
    "no-console": "off",
    "no-invalid-regexp": "warn"
  }
}
```

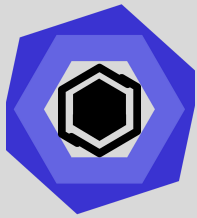
#### 3 VALEURS POSSIBLES

OFF (OU 0) , WARN (OU 1), ERROR (OU 2)

"OFF" : DE DÉSACTIVE

"WARN" : AVERTISSEMENT

"ERROR" : ERREUR BLOQUANTE, SOULIGNÉE EN ROUGE



## EN FONCTION DE NOS BESOINS: 05

une liste de plugins

Retrouvez la liste "Rules" officielles [ici](#)

### RÈGLES PUREMENT SYNTAXIQUES :

**CAMELCASE** : CONFIGURÉE AVEC {"PROPERTIES": "ALWAYS"}, ELLE PERMET DE VÉRIFIER QUE VOS VARIABLES SONT TOUTES NOMMÉES EN CAMELCASE

**QUOTES** : PERMET DE CHOISIR DÉFINITIVEMENT ENTRE LES SINGLES ET DOUBLES

(**COMMA-SPACING, COMMA-STYLE, KEY-SPACING...**), POUR DÉTECTER LES « ; » OU « , » OUBLIÉS, ÉVITER LES ESPACES OU SAUT DE LIGNES INUTILES.

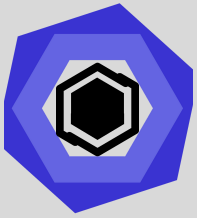
### RÈGLES POUR GARDER UN CODE PROPRE :

**NO-UNUSED-VARS** : PERMET D'ÉVITER LES VARIABLES DÉCLARÉES MAIS NON UTILISÉES QUI PRENNENT DE LA PLACE POUR RIEN DANS LE CODE.

**NO-SHADOW** : ÉVITE DE DÉCLARER UNE VARIABLE AVEC LE MÊME NOM QU'UNE AUTRE DE PORTÉE SUPÉRIEURE.

### EXCEPTIONS, MAIS EXCEPTIONNELLEMENT :

IL EST POSSIBLE QU'À CERTAINS ENDROITS DE VOTRE CODE VOUS SOUHAITIEZ PERMETTRE UNE ENTORSE À LA RÈGLE QU'ESLINT TENTE D'IMPOSER. AJOUTEZ EN PREMIÈRE LIGNE DE VOTRE FICHIER LE CODE /\* **ESLINT-DISABLE NOM-DE-LA-REGLE** \*/ POUR DÉSACTIVER UNE RÈGLE SUR TOUT UN FICHIER OU /\* **ESLINT-DISABLE-NEXT-LINE** \*/ JUSTE AU-DESSUS DE L'ENDROIT OÙ VOUS VOULEZ FAIRE TAIRE ESLINT SI VOUS SOUHAITEZ UN CIBLAGE PLUS PRÉCIS.



LINTER !

**Bon Bah Voilà !**

**Commande à faire uniquement si le module ESLint n'est pas installé sur VS.  
Sinon le "lint" se fait automatiquement**

06

**attention ne pas git le dossier node\_modules (tres lourd  
et inutile**

```
eslint --ext .js
```

Vous pouvez demander à ESLint de fixer les problèmes en ajoutant le drapeau `--fix` :

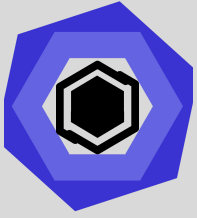
```
eslint --fix --ext .js
```

Si la commande ne vous plaît pas, il est possible de la lancer avec NPM en configurant celui-ci. Dans le fichier `package.json`, ajouter la ligne suivante dans l'objet `scripts` :

```
"scripts": {  
  "lint" : "eslint --fix --ext .js"  
}
```

Vous pouvez exécuter ESLint avec la commande suivante :

```
npm run lint
```



LIENS

07

<https://writecode.fr/tutoriel/eslint-coder-proprement-en-javascript>

<https://practicalprogramming.fr/eslint>

<https://jolicode.com/blog/donne-du-style-a-ton-javascript>

<https://www.synbioz.com/blog/tech/un-code-js-impeccable-grace-a-eslint>

<https://jeremiechazelle.dev/utiliser-eslint-et-prettier-sous-visual-studio-code-webstorm-phpstorm-pour-un-code-de-qualite/>





**Bisou!**