**Abstract Syntax Tree en Práctica**

‌

Vamos a usar el **AST** para crear una regla de **eslint**, este analizará estéticamente nuestro código a ver si hay que levantar un *warning* por violar la sintaxis. Muchas de estas reglas ya viene con e *eslint*, pero podemos agregar nuestras propias reglas. Vamos a usar la herramienta [AST | Explorer](https://astexplorer.net/#/gist/16fc27fc420f705455f2b42b6c804aa1/d9cc7988c2c743d7edfbb3c3b1abed866c975ee4) para experimentar. Usaremos la configuración por defecto, veremos en la parte superior izquierda el código que vamos a ingresar, a la derecha el *tree* creado, en la parte inferior izquierda las funciones de las reglas y a la derecha de eso la salida de nuestro código.

‌

**Test**

‌

En el **link** de **AST Explorer** ya tenemos un código escrito. Donde el la primera entrada tenemos las tareas que debe cumplir nuestro **fixer**.

const pi = 3.1415;

const half\_pi = 1.57075;

// variable constantes

// variables que guarden un numero

// El nombre de la variable tiene que estar en UPPERCASE

‌

A la derecha tenemos el árbol completo de todas estas declaraciones y gracias a el podemos manipular, detectar errores o interpretar lo que escribamos. Luego implementamos una función que recibe la declaración de la variable y accedemos a los datos que nos ofrece el AST para lograr cumplir con los requerimientos de nuestro solucionador.

export default function(context) {

return {

VariableDeclaration(node) {

// tipo de variable const

if (node.kind === "const") {

const declaration = node.declarations[0];

// asegurarnos que el valor es un numero

if (typeof declaration.init.value === "number") {

if (declaration.id.name !== declaration.id.name.toUpperCase()) {

context.report({

node: declaration.id,

message: "El nombre de la constante debe estar en mayúsculas",

fix: function(fixer) {

return fixer.replaceText(declaration.id, declaration.id.name.toUpperCase())

}

})

}

}

}

}

};

};

‌

Con context.report() podemos mandar un *warning* y además podemos solucionar el problema que se haya presentado.