¿Qué es el análisis de código?

Es una metodología que establece, mediante herramientas de análisis, si nuestro código está bien formado y estructurado. En caso contrario, se nos reportará una serie de errores a corregir.

Existen dos tipos de análisis:

**Análisis de código estático:**

Se realiza sin ejecutar el código fuente, esto ayuda a detectar prácticas peligrosas, como fugas de memoria. Existen tres tipos; basado en patrones, en flujos y de correctores de estilo. Se debe realizar cuando se crea nuevo código y cuando se publiquen modificaciones.

**Análisis de código dinámico:**

Se hacen con el programa ejecutándose, evaluando su fiabilidad, tiempos de respuesta y consumo de recursos. Existen dos tipos: de caja negra, que verifica y solo verifica si las salidas sean correctas, y las de caja blanca, que se enfoca en los fallos de procedimiento, probando todas las entradas posibles.

**Existen varias herramientas populares para el análisis de código, nos centraremos en las de JavaScript:**

**JSLint:** Herramienta de análisis estático creado por Douglas Crockford, se provee principalmente como una aplicación web, siguiendo un estricto conjunto de reglas para JavaScript.

**SonarQube:** Plataforma de análisis estático de código abierto, el cual usa más herramientas de análisis estático, como Checkstyle, PMD o FindBugs.

**ESLint:** Herramienta de análisis estático, con reglas customizables y configurables, siguiendo el standard de escritura ECMA Script para asegurar la interoperabilidad de páginas web a través de Internet.

[¿Cuándo hacer un análisis de código estático o dinámico? - IT Analytics](https://www.itanalytics.com.mx/2022/12/02/cuando-hacer-un-analisis-de-codigo-estatico-o-dinamico/)