EJERCICIO 1

PARTE A

Considera el lenguaje JavaScript acotado al paradigma de programación estructurada y analízalo en términos de [los cuatro componentes de un paradigma](https://www.notion.so/Paradigmas-lenguajes-y-programas-f36d432c55274b93913dc289446f424d?pvs=21) mencionados por Kuhn.

1. Generalización simbólica: ¿Cuáles son las reglas escritas del lenguaje?

1) En JavaScript, entendido solo desde la programación estructurada, las reglas formales son las que definen su sintaxis y semántica. Se pueden declarar variables con let, const o var, usar tipos de datos primitivos como números, cadenas de texto o booleanos, y aplicar operadores aritméticos, lógicos y relacionales. Para el control del flujo existen condicionales como if, else y switch, así como bucles for, while y do…while. También es posible definir funciones con la palabra clave function y organizar el código de forma modular. El alcance de las variables se rige por reglas de bloque, función y ámbito global. Por último, la entrada y salida básica se suele hacer con console.log o prompt en los navegadores.

2.Creencias de los profesionales: ¿Qué características particulares del lenguaje se

cree que sean "mejores" que en otros lenguajes?

2)Entre quienes usan JavaScript se suele creer que una de sus mayores ventajas es la facilidad de comenzar a programar en él, ya que no requiere compiladores ni instalaciones complejas, solo un navegador. También se valora su flexibilidad, que permite escribir programas sin tantas restricciones de tipos de datos como en otros lenguajes. Otra creencia extendida es que es un lenguaje universal y muy accesible, pues funciona en prácticamente cualquier dispositivo y sistema gracias al soporte en los navegadores. Además, se lo considera de aprendizaje rápido por la gran cantidad de recursos disponibles y se lo reconoce como portable, ya que los programas pueden ejecutarse en distintos entornos sin grandes adaptaciones

PARTE B

Considera el lenguaje JavaScript acotado al paradigma de programación estructurada y analízalo en términos de los ejes propuestos para la elección de un lenguaje de programación ([¿Cómo elegir un lenguaje?](https://www.notion.so/C-mo-elegir-un-lenguaje-818a7dd066514c119d8a97857443ed4c?pvs=21) ) y responde:

1. ¿Tiene una sintaxis y una semántica bien definida? ¿Existe documentación oficial?

1)Sí, JavaScript tiene una sintaxis y semántica bien definidas en el estándar ECMAScript, que está disponible de manera pública. Existe documentación oficial mantenida por ECMA y también por Mozilla (MDN), que es la fuente más consultada por programadores.

2.¿Es posible comprobar el código producido en ese lenguaje?

2)El código se puede comprobar directamente ejecutándolo en navegadores o entornos como Node.js. También existen herramientas que permiten verificar la calidad y posibles errores, como linters (por ejemplo, ESLint).

3.¿Es confiable?

3)En términos de confiabilidad, JavaScript es estable y ampliamente probado, aunque al ser un lenguaje de tipado débil pueden aparecer errores por falta de validación estricta de tipos. Sin embargo, para el uso estructurado, si se programa con disciplina, resulta confiable.

4.¿Es ortogonal?

4)No es un lenguaje completamente ortogonal. Hay casos donde combinaciones de características producen resultados inesperados (por ejemplo, la coerción de tipos en comparaciones). Aun así, en el nivel estructurado su comportamiento es razonablemente predecible.

5.¿Cuáles son sus características de consistencia y uniformidad?

5)Tiene cierta uniformidad en sus estructuras básicas (condicionales, bucles, funciones), pero también arrastra inconsistencias históricas como la diferencia entre == y === o el comportamiento de null y undefined. Estas particularidades afectan su consistencia.

6.¿Es extensible? ¿Hay subconjuntos de ese lenguaje?

6)Es extensible a través de librerías y frameworks. También es posible trabajar con subconjuntos, como limitarse al JavaScript estructurado (sin objetos ni programación funcional), o incluso usar versiones tipadas como TypeScript, que se basa en JavaScript.

7.El código producido, ¿es transportable?

7)El código JavaScript es altamente portable, ya que puede ejecutarse en cualquier navegador moderno y en distintos entornos como Node.js. Esto hace que el mismo programa funcione en diversos sistemas operativos y dispositivos sin modificaciones.