Trabajo Practico Nº1

Maciel Vigneau

Comision 2

# 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?

RPT:JavaScript tiene una **sintaxis definida por la especificación **ECMAScript**. Puede reordenar sus f**unciones como **un**idades principales de estructura** y **modularidad****.En paradigma estructurado, se privilegia la secuencia, selección y repetición sobre el uso de objetos o funciones de orden superior.Tambien algunas caracteristicas de sus tipados:

\*Control de flujo: **if, else, while, for, switch.**

\*Declaración de variables: **var, let, const.**

\*delimitados por llaves **{}** y punto(**.)** y coma (**;**) (aunque este puede ser opcional).

\*Uso de **tipos de datos dinámicos** (**number, string, boolean, object, null, undefined, symbol, bigint**).

1. Creencias de los profesionales: ¿Qué características particulares del lenguaje se cree que sean "mejores" que en otros lenguajes?

RPT: Es fácil y rápido de aprender, e incluso fácil de usar ya que casi todos los navegadores lo tienen ya incorporado. A la hora de usarla, es bastante dinámica,ya que no tiene que declarar variables estrictas y eso permite resultados visibles en el navegador casi sin preparación, lo cual resulta ser mas facil que C y Java que son mas estrictos en su sintaxis.

## 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?

RPT:Sí. La sintaxis y semántica están estandarizadas en **ECMAScript** (mantenido por ECMA International) y toda su documentacion se encuentra aqui:https://tc39.es/ecma262/.

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

RPT:Sí, se puede ejecutar en navegadores o entornos como Node.js.(Aunque puede tener errores y aun asi ejecutarlo debido a su dinamismo)

1. ¿Es confiable?

RPT: Si, debido a que es usado globalmente, sigue teniendo un gran soporte.

1. ¿Es ortogonal?

RPT:· No del todo. Algunas combinaciones de características no funcionan igual en todos los contextos (ejemplo: diferencias entre **var, let y const**; comportamiento raro de **this**).Hay inconsistencias históricas (por ejemplo, == vs. **===**).

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

RPT:Por asi decirlo, sigue evolucionando. Hay algunas inconsistencias como el null y undefined son raros en su forma de actuar... al igual que los arrays y objetos pueden ser similares pero tan raros como distintos al actuar.

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

RPT: Si,mediante librerías y frameworks (ej. React, Angular, jQuery) o tambien subconjuntos conocidos, como **TypeScript** (superset tipado de JavaScript) y el subset "strict mode".

1. El código producido, ¿es transportable?

RPT:Sí. Un programa en JavaScript puede ejecutarse en cualquier navegador moderno y en entornos como Node.js, sin importar el sistema operativo.(obvio pueden que hayan excepciones especiales con navegadores, pero la mayoria ya son viejos o no se actualizan).