# Actividad

1. Cuantos lenguajes de programación existen y sus versiones y mejoras con relación a los demás
2. Explique mediante un ejemplo si la instalación de los lenguajes de programación son de la misma manera para cada uno de ellos
3. Explore que es **MEAN** y **MERN** que programas lo componen
4. Cuantas clases de Windows existen haga un cuadro comparativo entre ellos

# Solución

1. **VS Code**

Primera versión: 1.0 (2015). Soporte básico para JavaScript/TypeScript.

Versión intermedia: 1.50 (2020). Mejor soporte para Python, depurador remoto.

Versión actual: 1.89 (2024). Copilot integrado, terminal AI.

Mejoras clave: De editor simple a IDE ligero con IA.

**Sublime Text**

Primera versión: Sublime Text 2 (2011). Multi-cursor, rendimiento rápido.

Versión intermedia: Sublime Text 3 (2017). Soporte para Python 3.

Versión actual: Sublime Text 4 (2021). GPU rendering, Apple Silicon.

Mejoras clave: Velocidad y estabilidad.

**Vim / Neovim**

Primera versión: Vim 1.0 (1991). Modal (comandos clásicos).

Versión intermedia: Vim 8.0 (2016). Soporte para async.

Versión actual: Neovim 0.9 (2023). Integración con Lua, LSP moderno.

Mejoras clave: De editor terminal a IDE programable.

**Emacs**

Primera versión: GNU Emacs 1.0 (1985). Lenguaje Elisp.

Versión intermedia: Emacs 24 (2012). Gestor de paquetes.

Versión actual: Emacs 28 (2022). Soporte para JSON nativo.

Mejoras clave: De editor a "ecosistema".

**Atom (Descontinuado)**

Primera versión: 1.0 (2015). Editor hackeable (HTML/JS/CSS).

Versión intermedia: 1.40 (2019). Teletype (colaboración).

Última versión: 1.60 (2022). Mejoras de rendimiento.

Mejoras clave: Comunidad fuerte, pero discontinuado.

1. La instalación para cada lenguaje de programación es diferente ya que cada uno tiene sus condiciones por ejemplo

Si tu quieres programas en HTML basta con el procesador de texto que viene el Windows que es el bloc de notas pero si quieres programar con algo mas complejo como lo es JAVA tendrías que descargar ya visual code que es un programador de texto mas universal, ya si quieres con phyton que es mas sofisticado ya que deberías instalar extensiones y demás

1. **MEAN y MERN** son dos *stacks* tecnológicos populares para el desarrollo de aplicaciones web full-stack basadas en JavaScript. Ambos permiten construir aplicaciones completas (frontend y backend) usando un mismo lenguaje de programación (JavaScript/TypeScript).

**1. MEAN**

Es un stack que utiliza las siguientes tecnologías:

**MongoDB**: Base de datos NoSQL orientada a documentos.

**Express.js**: Framework backend para Node.js, usado para construir APIs.

**Angular**: Framework frontend para aplicaciones web dinámicas (TypeScript).

**Node.js**: Entorno de ejecución de JavaScript en el servidor.

**Ventajas de MEAN**:

Ideal para aplicaciones empresariales complejas gracias a Angular.

Estructura modular y bien definida.

**2. MERN**

Es una variante de MEAN que reemplaza Angular con React:

**MongoDB**: Misma base de datos NoSQL.

**Express.js**: Igual que en MEAN.

**React**: Biblioteca frontend para interfaces dinámicas (JavaScript/TypeScript).

**Node.js**: Mismo entorno de servidor.

**Ventajas de MERN**:

Mayor flexibilidad en el frontend (React es más liviano que Angular).

Comunidad muy grande y ecosistema de librerías (Redux, Next.js, etc.).

1. 