**Actividad**:

**Resolución a problemas algorítmicos aplicando estructuras de almacenamiento**

**GA3-220501093-AA3-EV01**

**Aprendiz**:

Wilmer Jair Espinosa Silva

CC: 1.095.910.391

**Instructor**:

ISRAEL ARBONA GUERRERO

Servicio Nacional de aprendizaje-SENA

Curso: TECNOLOGÍA EN ANÁLISIS Y DESARROLLO DE SOFTWARE

Ficha: 2455285

**Evidencia de conocimiento: GA3-220501093-AA3-EV01 bases teóricas de estructuras de almacenamiento en memoria**

Esta evidencia se centra en consolidar los conceptos básicos relacionados con los lenguajes de programación, entornos de codificación e instalación y la sintaxis del lenguaje de JavaScript.

Para su desarrollo es importante la lectura del componente formativo. Elaborar un documento en el cual se registren los siguientes elementos:

**Principales diferencias entre los lenguajes compilados e**

**interpretados**.

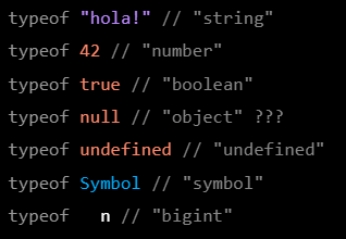
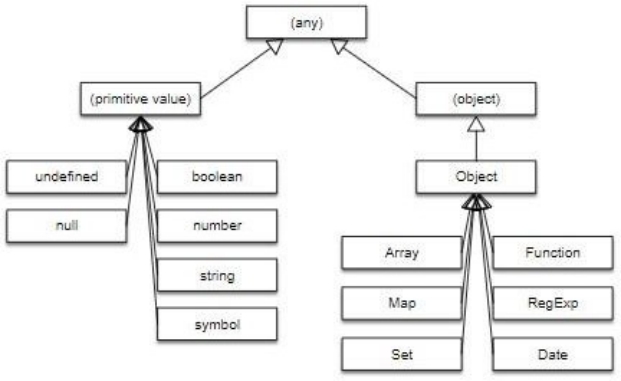
|  |  |
| --- | --- |
| **LENGUAJES COMPILADOS** | **LENGUAJES INTERPRETADOS** |
| Un lenguaje compilado genera una fila binario no modificable. | Un lenguaje interpretado es escrito en un lenguaje de programación definido y modificable en cada momento. |
| Las instrucciones vienen enviadas directamente al procesador. | Las instrucciones deben traducirse antes de llegar al procesador. |
| Se requieren dos pasos separados para ejecutar el programa desde el código fuente. | El código fuente se ejecuta a través de un solo comando. |
| Dado que el programa ya se ha traducido, la ejecución es más rápida. | El programa debe traducirse cada vez aumentando el tiempo de ejecución. |
| El programa solo se puede ejecutar en ciertas máquinas y sistemas operativos. | El programa funciona en todas las máquinas y sistemas. |
| Los errores de compilación impiden que se compile el código | Los errores de compilación son visibles solo si se inicia el programa. |
| Ejemplos de lenguajes compilados son C, C++, Delphi | Ejemplos de lenguajes interpretados son Python, JavaScript, Perl, PHP |

**Características principales de JavaScript**

Su sintaxis es similar a la de Java, débilmente tipado, es case sensitivo, no existen las constantes, basado no orientado a objetos. <Universalidad= de usos, seguridad de ejecución, páginas más ligeras de cargar.



**Tipos de datos primitivos y uso en JavaScript**

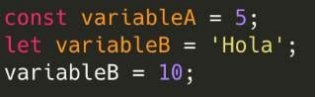


**Operadores en JavaScript**

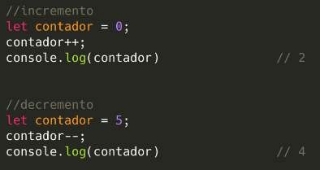
Los operadores permiten manipular las variables, realizar operaciones matemáticas, comparaciones lógicas o asignaciones.

Existen varios tipos de operadores

**Operador de Asignación**: Asigna un valor en nuestras variables.



**Operador de Incremento y Decremento:** Permite incrementar o decrementar en una unidad el valor de la variable.



**Operadores Lógicos**: Nos permite tomar decisiones sobre las instrucciones incluso nos permite negar una instrucción.

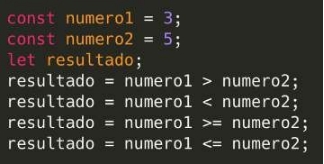
**NEGACIÓN(!) AND(&&)**



**OR(II)**



**Operadores Relacionales:** Las usamos para evaluar expresiones.



**Operadores de Igualdad:** Existe estricta se evalúa la expresión sea igual o diferente, el tipo y el valor y no estricta se evalúa solo el valor.



**Operadores Aritméticos:** Evalúan la expresión y devuelve un único resultado.

