|  |  |  |
| --- | --- | --- |
| **Nombre del Alumno** | | **Fecha:** dd/mm/aaaa |
|  | | **Grupo** |
| **Número de Control:** |  | **Cabaña:** |

1. Objetivo

Aprender a utilizar var, let, const, y a comprender la inmutabilidad de constantes en JavaScript a través de ejercicios prácticos que simulan situaciones de programación reales.

2. Contexto de la Vida Real

Imagina que trabajas como desarrollador en una startup tecnológica. Estás encargado de mantener y mejorar una aplicación que gestiona inventarios en tiempo real. Las decisiones que tomes sobre cómo declarar y manejar variables afectarán directamente el rendimiento y la estabilidad del sistema.

3. Descripción de los Ejercicios

Ejercicio 1: Diferencias entre var y let

* **Escenario:** Durante una reunión con el equipo, te das cuenta de que el código tiene errores debido al uso incorrecto de var y let. Tu tarea es corregir estos errores.
* **Pregunta de Reflexión:** ¿Cómo podría impactar el alcance de una variable en la ejecución del código si es utilizada incorrectamente? -En el fallo o modificación de los resultados

Ejercicio 2: Inmutabilidad de const

* **Escenario:** Estás trabajando con una API externa que proporciona datos en forma de objetos. Necesitas asegurar que ciertos valores no cambien una vez asignados.
* **Pregunta de Reflexión:** ¿Cómo la inmutabilidad de const puede ayudarte a prevenir errores en la manipulación de datos? -En datos que siempre tendrán el mismo valor

Ejercicio 3: Bloques de Código y Alcance

* **Escenario:** Te enfrentas a una situación donde necesitas mantener la integridad de ciertas variables dentro de bloques específicos de código.
* **Pregunta de Reflexión:** ¿Qué impacto tiene el uso de let y const en la legibilidad y mantenimiento del código? -En que se mantienen ciertos valores en algunos bloques y otros no

Ejercicio 4: Ciclos y Alcance de Variables

* **Escenario:** Estás optimizando un proceso que involucra iteraciones sobre grandes conjuntos de datos y necesitas manejar adecuadamente las variables de control.
* **Pregunta de Reflexión:** ¿Cómo afecta la elección de var, let o const en la eficiencia y claridad del código en ciclos? -En el comportamiento de cada uno

Ejercicio 5: Comprendiendo Closure con let y var

* **Escenario:** Estás depurando un problema donde una función interna no está capturando los valores esperados de las variables.
* **Pregunta de Reflexión:** ¿Cómo influye el alcance de let y var en el comportamiento de las funciones anidadas y cierres? -En que el valor no puede variar entre cada una de estas

4. Resultados

Documenta los resultados obtenidos para cada ejercicio. Incluye capturas de pantalla del código y la salida en consola. Comenta sobre los hallazgos.

Ejercicio 1:

* **Código y Resultado:**
* **Reflexión:** ¿Qué diferencias observaste entre el uso de var y let en términos de alcance y redeclaración? -En la primera actividad no se notó diferencia

Ejercicio 2:

* **Código y Resultado:**
* **Reflexión:** ¿Qué sucedió al intentar modificar una constante? ¿Por qué es importante entender la inmutabilidad? -Hubo un error al intentar colocar dos valores dentro de un mismo bloque

Ejercicio 3:

* **Código y Resultado:**
* **Reflexión:** ¿Cómo afecta el alcance de bloque al comportamiento de tu código en situaciones complejas? -Que colocar este tipo de variables puede diferenciar uno de los otros mediante los bloques

Ejercicio 4:

* **Código y Resultado:**
* **Reflexión:** ¿Qué desafíos encontraste al trabajar con ciclos y cómo los resolviste? -En el proceso del código usando valores, var utiliza los datos necesarios poniendo limites clave, mientras que let es con limites estructurales, mientras que la constante no se puede por solo poder ser un valor fijo

Ejercicio 5:

* **Código y Resultado:**
* **Reflexión:** ¿Cómo cambia el comportamiento del código al usar let en lugar de var en un cierre? -En que los valores pueden ser diferentes mediante lo que se escribe

5. Cuadro CQA

|  |  |  |
| --- | --- | --- |
| Conocimientos Previos | Preguntas Surgidas | Aprendizajes Adquiridos |
| Ejemplo: Sabía que let tiene alcance de bloque. | Ejemplo: ¿Por qué no puedo usar const en un ciclo for? | Ejemplo: Aprendí cómo usar let y const para evitar errores de alcance. |