

# Reporte de Evaluación - Fork de GitHub

## Información General

Estudiante: Roque Aldana  
Repositorio: Julio28012020/act\_web1\_s3  
Fecha de evaluación: 21/8/2025, 16:07:55  
Evaluado por: Sistema de Evaluación

## Resumen de Calificaciones

Calificación general: 4.6/5.0  
Actividades completadas: 20/20  
Porcentaje de completitud: 100.0%

## Detalle de Actividades

#	Descripción	Archivo	Encontrado	Calificación
1	Crea un programa que use un ciclo while ...	ejercicios-while/ejercicio_while_01.js	Sí	5.0
2	Utiliza un ciclo while para calcular la ...	ejercicios-while/ejercicio_while_02.js	Sí	5.0
3	Implementa un sistema que use un ciclo w...	ejercicios-while/ejercicio_while_03.js	Sí	2.0
4	Crea un programa que use un ciclo while ...	ejercicios-while/ejercicio_while_04.js	Sí	5.0
5	Desarrolla una función que use un ciclo ...	ejercicios-while/ejercicio_while_05.js	Sí	5.0
6	Dado un array de nombres, usa un ciclo w...	ejercicios-while/ejercicio_while_06.js	Sí	4.0
7	Implementa un programa que use un ciclo ...	ejercicios-while/ejercicio_while_07.js	Sí	5.0
8	Utiliza un ciclo while para generar los ...	ejercicios-while/ejercicio_while_08.js	Sí	5.0
9	Crea un juego que use un ciclo while don...	ejercicios-while/ejercicio_while_09.js	Sí	2.0
10	Desarrolla un programa que use un ciclo ...	ejercicios-while/ejercicio_while_10.js	Sí	5.0
11	Usa un ciclo for anidado para crear las ...	ejercicios-for/ejercicio_for_01.js	Sí	5.0
12	Implementa un programa que use ciclos fo...	ejercicios-for/ejercicio_for_02.js	Sí	5.0
13	Dado un array de 20 números aleatorios, ...	ejercicios-for/ejercicio_for_03.js	Sí	4.0
14	Utiliza ciclos for anidados para encontr...	ejercicios-for/ejercicio_for_04.js	Sí	5.0
15	Crea un programa que use un ciclo for pa...	ejercicios-for/ejercicio_for_05.js	Sí	5.0
16	Implementa una función que use un ciclo ...	ejercicios-for/ejercicio_for_06.js	Sí	5.0
17	Usa ciclos for anidados para crear una m...	ejercicios-for/ejercicio_for_07.js	Sí	5.0
18	Desarrolla un programa que use un ciclo ...	ejercicios-for/ejercicio_for_08.js	Sí	5.0
19	Utiliza un ciclo for para generar la sig...	ejercicios-for/ejercicio_for_09.js	Sí	5.0
20	Implementa un programa que use un ciclo ...	ejercicios-for/ejercicio_for_10.js	Sí	5.0

## Retroalimentación Detallada

**Actividad 1: Crea un programa que use un ciclo while para mostrar una cuenta regresiva desde 20 hasta 1, mostrando cada número en la consola. Al final debe mostrar '¡Despegue!'.**

Archivo esperado: ejercicios-while/ejercicio\_while\_01.js

Estado: Archivo encontrado

Calificación: 5.0/5.0

Retroalimentación:

La solución es correcta y concisa. Cumple con todos los requisitos de la actividad, utilizando un ciclo while de manera eficiente y mostrando el mensaje final correctamente.

**Actividad 2: Utiliza un ciclo while para calcular la suma de todos los números pares entre 1 y 50. Muestra el resultado final y cuántos números pares se sumaron.**

Archivo esperado: ejercicios-while/ejercicio\_while\_02.js

Estado: Archivo encontrado

Calificación: 5.0/5.0

Retroalimentación:

La solución es correcta y eficiente. El código es limpio, legible y cumple con el objetivo planteado. ¡Excelente trabajo!

**Actividad 3: Implementa un sistema que use un ciclo while para solicitar una contraseña al usuario hasta que ingrese 'admin123'. Debe mostrar cuántos intentos fallidos hubo antes del acceso correcto.**

Archivo esperado: ejercicios-while/ejercicio\_while\_03.js

Estado: Archivo encontrado

Calificación: 2.0/5.0

Retroalimentación:

El código no solicita la contraseña al usuario; asigna directamente '123456', generando un bucle infinito. Además, la condición del `while` está mal implementada y el contador de intentos no funciona correctamente.

**Actividad 4: Crea un programa que use un ciclo while para generar números aleatorios entre 1 y 100 hasta que salga un número mayor a 95. Muestra cada número generado y al final indica cuántos números se generaron.**

Archivo esperado: ejercicios-while/ejercicio\_while\_04.js

Estado: Archivo encontrado

Calificación: 5.0/5.0

Retroalimentación:

Excelente solución. El código es claro, funcional y cumple con todos los requisitos de la actividad. Bien hecho!

**Actividad 5: Desarrolla una función que use un ciclo while para calcular el factorial de un número dado. Debe mostrar paso a paso cómo se va calculando el factorial.**

Archivo esperado: ejercicios-while/ejercicio\_while\_05.js

Estado: Archivo encontrado

Calificación: 5.0/5.0

Retroalimentación:

Excelente solución. El código es claro, funcional y cumple con todos los requisitos, mostrando el proceso de cálculo paso a paso y aplicando buenas prácticas.

**Actividad 6: Dado un array de nombres, usa un ciclo while para buscar un nombre específico. El programa debe mostrar en qué posición se encontró el nombre o indicar si no existe.**

Archivo esperado: ejercicios-while/ejercicio\_while\_06.js

Estado: Archivo encontrado

Calificación: 4.0/5.0

Retroalimentación:

La solución es correcta y funcional. Se podría mejorar la legibilidad utilizando `===` en lugar de `==` y considerar el uso de `const` o `let` según la mutabilidad esperada de las variables.

**Actividad 7: Implementa un programa que use un ciclo while para contar cuántos dígitos tiene un número entero positivo. Por ejemplo, 12345 tiene 5 dígitos.**

Archivo esperado: ejercicios-while/ejercicio\_while\_07.js

Estado: Archivo encontrado

Calificación: 5.0/5.0

Retroalimentación:

La solución es correcta y el código es claro, incluyendo logs que facilitan el entendimiento del proceso. La lógica es impecable y cumple con el objetivo.

**Actividad 8: Utiliza un ciclo while para generar los primeros 15 números de la secuencia de Fibonacci. Muestra cada número de la secuencia.**

Archivo esperado: ejercicios-while/ejercicio\_while\_08.js

Estado: Archivo encontrado

Calificación: 5.0/5.0

Retroalimentación:

Excelente solución. El código es claro, conciso y genera correctamente los primeros 15 números de la secuencia de Fibonacci. Buena presentación con mensajes informativos.

**Actividad 9: Crea un juego que use un ciclo while donde el usuario debe adivinar un número entre 1 y 50. Proporciona pistas ('muy alto', 'muy bajo', 'cerca') y limita a 7 intentos máximo.**

Archivo esperado: ejercicios-while/ejercicio\_while\_09.js

Estado: Archivo encontrado

Calificación: 2.0/5.0

Retroalimentación:

La lógica general es correcta pero la adivinanza siempre es 19, impidiendo que el juego funcione correctamente. Se debe solicitar la entrada del usuario para que pueda adivinar.

**Actividad 10: Desarrolla un programa que use un ciclo while para procesar calificaciones de estudiantes. Debe continuar pidiendo calificaciones hasta que se ingrese -1, luego calcular y mostrar el promedio, la calificación más alta y más baja.**

Archivo esperado: ejercicios-while/ejercicio\_while\_10.js

Estado: Archivo encontrado

Calificación: 5.0/5.0

Retroalimentación:

La solución es correcta y completa. El código es legible y sigue buenas prácticas al validar las calificaciones y manejar el caso de no ingresar datos. ¡Excelente trabajo!

**Actividad 11: Usa un ciclo for anidado para crear las tablas de multiplicar del 1 al 10. Cada tabla debe mostrarse claramente separada y formateada.**

Archivo esperado: ejercicios-for/ejercicio\_for\_01.js

Estado: Archivo encontrado

Calificación: 5.0/5.0

Retroalimentación:

Excelente solución. El código es limpio, funcional y cumple con todos los requisitos. Podrías agregar comentarios para mejorar aún más la legibilidad, pero en general, ¡muy bien!

**Actividad 12: Implementa un programa que use ciclos for anidados para crear un patrón de asteriscos en forma de diamante: \*** **\*\*** **\*\*\*** **\*\*\*\*** **\*\*\*\*\*** **\*\*\*\*** **\*\*\*** **\*\*** **\***

Archivo esperado: ejercicios-for/ejercicio\_for\_02.js

Estado: Archivo encontrado

Calificación: 5.0/5.0

Retroalimentación:

¡Excelente solución! El código es claro, conciso y resuelve correctamente el problema del diamante de asteriscos. Bien hecho.

**Actividad 13: Dado un array de 20 números aleatorios, usa un ciclo for para encontrar: el número mayor, el menor, la suma total, el promedio, y cuántos números son pares e impares.**

Archivo esperado: ejercicios-for/ejercicio\_for\_03.js

Estado: Archivo encontrado

Calificación: 4.0/5.0

Retroalimentación:

La solución es correcta y funcional. Se podría mejorar la legibilidad inicializando las variables en el momento de la declaración.

**Actividad 14: Utiliza ciclos for anidados para encontrar y mostrar todos los números primos entre 1 y 100. Debe indicar cuántos números primos se encontraron.**

Archivo esperado: ejercicios-for/ejercicio\_for\_04.js

Estado: Archivo encontrado

Calificación: 5.0/5.0

Retroalimentación:

La solución es correcta y eficiente para encontrar números primos. El código es limpio, bien estructurado y fácil de entender, cumpliendo con el objetivo de la actividad.

**Actividad 15: Crea un programa que use un ciclo for para invertir una cadena de texto carácter por carácter. Muestra tanto la cadena original como la invertida.**

Archivo esperado: ejercicios-for/ejercicio\_for\_05.js

Estado: Archivo encontrado

Calificación: 5.0/5.0

Retroalimentación:

La solución es correcta y eficiente. El código es claro y fácil de entender, cumple con el objetivo planteado.

**Actividad 16: Implementa una función que use un ciclo for para calcular la potencia de un número sin usar Math.pow(). Debe mostrar el proceso paso a paso.**

Archivo esperado: ejercicios-for/ejercicio\_for\_06.js

Estado: Archivo encontrado

Calificación: 5.0/5.0

Retroalimentación:

Excelente solución. El código es claro, funcional y cumple con todos los requisitos, incluyendo la demostración paso a paso. Demuestra un buen entendimiento del ciclo for.

**Actividad 17: Usa ciclos for anidados para crear una matriz 5x5 donde cada elemento sea la suma de sus índices (fila + columna). Muestra la matriz formateada.**

Archivo esperado: ejercicios-for/ejercicio\_for\_07.js

Estado: Archivo encontrado

Calificación: 5.0/5.0

Retroalimentación:

Excelente solución. El código es claro, funcional y resuelve el problema planteado utilizando ciclos for anidados de manera eficiente. Bien hecho!

**Actividad 18: Desarrolla un programa que use un ciclo for para contar cuántas vocales (a, e, i, o, u) hay en una frase dada. Debe mostrar el conteo de cada vocal por separado.**

Archivo esperado: ejercicios-for/ejercicio\_for\_08.js

Estado: Archivo encontrado

Calificación: 5.0/5.0

Retroalimentación:

La solución es correcta y eficiente. El código es legible y cumple con los requisitos del problema planteado. ¡Buen trabajo!

**Actividad 19: Utiliza un ciclo for para generar la siguiente secuencia: 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024. Debe mostrar cada término y la suma total de la secuencia.**

Archivo esperado: ejercicios-for/ejercicio\_for\_09.js

Estado: Archivo encontrado

Calificación: 5.0/5.0

Retroalimentación:

La solución es correcta y eficiente. El código es claro, legible y cumple con los requisitos de la actividad.

**Actividad 20: Implementa un programa que use un ciclo for para verificar si una palabra o frase es un palíndromo (se lee igual de izquierda a derecha que de derecha a izquierda). Ignora espacios y mayúsculas.**

Archivo esperado: ejercicios-for/ejercicio\_for\_10.js

Estado: Archivo encontrado

Calificación: 5.0/5.0

Retroalimentación:

La solución es correcta y eficiente. El código es legible y cumple con las buenas prácticas, abordando correctamente la verificación de palíndromos.

## Resumen General

Excelente trabajo. Completó 20/20 actividades (100%) con una calificación promedio de 4.6/5. Demuestra buen dominio de los conceptos.

## Recomendaciones

- Revisar y mejorar las actividades con calificación baja