Guia de practica - Estructuras de control: condicionales

Nombre: _____

Fecha: _____

Puntaje: _____

1.

¿Qué hace una estructura condicional if?

1. Repite un bloque de código varias veces
2. Ejecuta un bloque solo si se cumple una condición
3. Define una función

Respuesta correcta:
B.

Ejecuta un bloque solo si se cumple una condición

2.

La expresión booleana '5 > 3' evalúa a _____

Respuesta: _____

Respuesta correcta:

true

3.

¿Qué operador se usa para verificar si dos valores son iguales en la mayoría de lenguajes de programación?

1. ==
2. =
3. ===
4. !=

Respuesta correcta:
A.

==

4.

Dado el código: if (edad >= 18) { mensaje = 'Mayor de edad'; } Supón que mensaje no ha sido definida antes. Si edad = 17, ¿cuál es el valor de mensaje?

1. Mayor de edad
2. Menor de edad
3. undefined
4. null

Respuesta correcta:
C.

undefined

5.

¿Cuál es la salida de: if (x > 0) { print('Positivo'); } else { print('No positivo'); } con x = -5?

1. Positivo
2. No positivo
3. Ninguna

Respuesta correcta:
B.

No positivo

6.

Complete: El operador lógico && devuelve true solo si _____ operandos son true.

Respuesta: _____

Respuesta correcta:

ambos

7.

¿Qué valor tiene la expresión: (true && false) || true?

1. true
2. false
3. undefined
4. null

Respuesta correcta:
A.

true

8.

En un juego, si el jugador toca un enemigo (colision = true) y tiene un escudo activo (escudo = true), se reduce la vida. ¿Qué condición usarías?

1. if (colision) { ... }
2. if (colision || escudo) { ... }
3. if (colision && escudo) { ... }
4. if (!colision) { ... }

Respuesta correcta:
C.

if (colision && escudo) { ... }

9.

¿Qué ocurre si en un if-else anidado la primera condición es verdadera?

1. Se ejecuta el primer bloque y luego el else
2. Se ejecuta solo el primer bloque y se salta el resto
3. Se evalúan todas las condiciones

Respuesta correcta:
B.

Se ejecuta solo el primer bloque y se salta el resto

10.

La expresión (10 == '10') en algunos lenguajes (como JavaScript) devuelve ____ debido a la coerción de tipos.

Respuesta: _____

Respuesta correcta:

true

11.

¿Cuál es la diferencia entre = y ==?

1. = es asignación, == es comparación
2. = es comparación, == es asignación
3. Ambos son iguales
4. Ninguna

Respuesta correcta:
A.

= es asignación, == es comparación

12.

Si una variable 'edad' vale 15, ¿cuántas condiciones se evalúan en el siguiente código? if (edad > 18) { A } else if (edad > 12) { B } else { C }

1. 1
2. 2
3. 3
4. 0

Respuesta correcta:
B.

2

13.

El operador lógico NOT (!) invierte el valor booleano. Si a = true, entonces !a es ____.

Respuesta: _____

Respuesta correcta:

false

14.

¿Cuál es la sintaxis más común de if-else en lenguajes como C, Java y JavaScript?

1. if (condicion) { bloque } else { bloque }
2. if condicion then bloque else bloque
3. if (condicion): bloque else: bloque
4. Ninguna de las anteriores

Respuesta correcta:
A.

if (condicion) { bloque } else { bloque }

15.

En un programa, la condición (x > 0 && x < 100) es verdadera cuando x está en el rango:

1. x > 100
2. x < 0
3. 0 < x < 100
4. x = 0

Respuesta correcta:
C.

0 < x < 100

16.

¿Qué imprime el siguiente código? if (true) { print('A'); } print('B');

1. A
2. B
3. A y B

Respuesta correcta:
C.

A y B

17.

¿Cuál es un error común en condicionales?

1. Escribir if sin llaves cuando hay más de una instrucción
2. Usar else después de un if
3. Comparar con ==
4. Usar operadores lógicos

Respuesta correcta:
A.

Escribir if sin llaves cuando hay más de una instrucción

18.

Complete la condición para que el código imprima 'Par' solo si el número es divisible por 2: if (numero % 2 _____ 0) { print('Par'); }

Respuesta: _____

Respuesta correcta:

==

19.

En un semáforo, si la luz es verde, avanzar; si es amarilla, precaución; si es roja, detenerse. ¿Qué estructura es más adecuada?

1. if-else simple
2. switch-case
3. if-else anidado
4. condicional ternario

Respuesta correcta:
B.

switch-case

20.

¿Qué resultado da: (5 > 3) && !(4 == 4)?

1. true
2. error
3. false
4. ninguno

Respuesta correcta:
C.

false

Respuestas

1. **B.**

   Ejecuta un bloque solo si se cumple una condición

2. true
3. **A.**

   ==

4. **C.**

   undefined

5. **B.**

   No positivo

6. ambos
7. **A.**

   true

8. **C.**

   if (colision && escudo) { ... }

9. **B.**

   Se ejecuta solo el primer bloque y se salta el resto

10. true
11. **A.**

    = es asignación, == es comparación

12. **B.**

    2

13. false
14. **A.**

    if (condicion) { bloque } else { bloque }

15. **C.**

0 < x < 100

16. **C.**

    A y B

17. **A.**

    Escribir if sin llaves cuando hay más de una instrucción

18. ==

19. **B.**

    switch-case

20. **C.**

    false