

Respuestas y Explicaciones de las Preguntas

Pregunta 1: ¿Cómo se accede al tercer elemento de la lista `lista = ['x', 'y', 'z', 'w']`?

Respuesta correcta: d. `lista[2]`

Explicación: Los índices en Python comienzan en 0, así que el tercer elemento está en el índice 2.

Pregunta 2: Dada la lista `nums = [10, 20, 30, 40, 50]`, ¿qué devuelve `nums[1:4]`?

Respuesta correcta: a. `[20, 30, 40]`

Explicación: El corte incluye el índice 1, 2, y 3, pero no el índice 4.

Pregunta 3: En slicing, ¿qué índice NO se incluye al usar la sintaxis `lista[inicio:fin]`?

Respuesta correcta: c. El índice `fin`

Explicación: El índice `'fin'` no se incluye en el slicing.

Pregunta 4: Con la lista `a = ['p', 'q', 'r', 's']`, ¿qué obtiene `a[0:2]`?

Respuesta correcta: d. `['p', 'q']`

Explicación: El slicing incluye los elementos en los índices 0 y 1, pero no el 2.

Pregunta 5: Para añadir el elemento 100 al final de `mi_lista`, ¿qué método se usa?

Respuesta correcta: b. `mi_lista.append(100)`

Explicación: El método `'append'` agrega un elemento al final de la lista.

Pregunta 6: Si quiero eliminar el valor `'hola'` de la lista `saludos`, ¿cuál es la llamada correcta?

Respuesta correcta: b. `saludos.remove('hola')`

Explicación: El método `'remove'` elimina el primer valor que coincide con el valor especificado.

Pregunta 7: ¿Qué ocurre si se llama a `lista.remove(valor)` y `valor` no está en la lista?

Respuesta correcta: b. Devuelve un `ValueError`

Explicación: Si el valor no está en la lista, se lanza un `'ValueError'`.

Pregunta 8: Dada `datos = [1, 2, 3, 4]`, ¿cómo se reemplaza el segundo elemento por 99?

Respuesta correcta: a. `datos[1] = 99`

Explicación: Para reemplazar un elemento de la lista, se usa la asignación por índice.

Pregunta 9: ¿Cuál de estas operaciones demuestra la mutabilidad de las listas?

Respuesta correcta: d. Ejecutar `lista[0] = 'nuevo'`

Explicación: Las listas son mutables, lo que significa que puedes cambiar sus elementos.

Pregunta 10: ¿Qué sintaxis serviría para obtener desde el elemento de índice 2 hasta el final de `mi_lista`?

Respuesta correcta: c. `mi_lista[2:]`

Explicación: Al omitir el índice de fin, se obtiene desde el índice 2 hasta el final de la lista.