

Eliminado: <object>

PROMETEO

Test Programación Unidad 4

PREGUNTAS

Pregunta 1

¿Cuál de las siguientes afirmaciones describe mejor un array unidimensional en Java?

- A) Una colección de elementos de distintos tipos almacenados en posiciones aleatorias
- B) Una secuencia ordenada y contigua de memoria con elementos del mismo tipo
- C) Una estructura dinámica que cambia de tamaño automáticamente
- D) Un conjunto de variables sin índice asociado

Pregunta 2

Si declaramos int[] numeros = new int[5];, ¿cuál es el índice del último elemento?

- A) 4
- B) 5
- C) 6
- D) Depende de los valores almacenados

Pregunta 3

¿Qué ocurre si intentamos acceder a numeros[5] en el array del ejercicio anterior?

- A) Devuelve siempre 0
- B) El compilador muestra un error y no deja ejecutar el programa
- C) Se produce una excepción en tiempo de ejecución (ArrayIndexOutOfBoundsException)
- D) El acceso se redirige automáticamente a la última posición válida

Pregunta 4

Eliminado: <object>

PROMETO

Dado el array int[][] matriz = new int[3][4], ¿qué representa la expresión matriz[1][2]?

- A) La tercera fila y la segunda columna
- B) La segunda fila y la tercera columna
- C) La primera fila y la tercera columna
- D) La segunda fila completa

Pregunta 5

¿Cuál de las siguientes sentencias obtiene correctamente la longitud de un array unidimensional llamado datos?

- A) datos.length()
- B) length(datos)
- C) datos.length
- D) size(datos)

Pregunta 6

¿Cuál es la forma correcta de declarar e inicializar un array de enteros con los valores 1, 2, 3, 4 y 5?

- A) int[] numeros = (1, 2, 3, 4, 5);
- B) int[] numeros = new int(1, 2, 3, 4, 5);
- C) int[] numeros = {1, 2, 3, 4, 5};
- D) int numeros[] = [1, 2, 3, 4, 5];

Pregunta 7

¿Qué significa que los objetos de tipo String en Java sean inmutables?

- A) Que no se pueden comparar entre sí
- B) Que su contenido no puede cambiar una vez creado; cualquier "cambio" genera un nuevo objeto
- C) Que solo se pueden crear dentro del método main
- D) Que ocupan siempre la misma cantidad de memoria

Pregunta 8

Dadas las sentencias:

Eliminado: <object>

PROMETO

```
String a = "Hola";
String b = new String("Hola");
```

¿Qué instrucción compara correctamente el contenido de ambas cadenas?

- A) a == b
- B) a.equals(b)
- C) a = b
- D) compare(a, b)

Pregunta 9

¿Qué método utilizarías para eliminar espacios en blanco al principio y al final de una cadena llamada texto?

- A) texto.clean()
- B) texto.trim()
- C) texto.stripSpaces()
- D) texto.remove(" ")

Pregunta 10

¿Por qué se recomienda usar StringBuilder cuando se realizan muchas concatenaciones de texto dentro de un bucle?

- A) Porque convierte automáticamente todas las cadenas a mayúsculas
- B) Porque evita la inmutabilidad de String y mejora el rendimiento en concatenaciones repetidas
- C) Porque obliga a que el código sea thread-safe en todos los casos
- D) Porque solo así se puede imprimir por consola

HOJA DE RESPUESTAS

Pregunta	Respuesta Correcta	Justificación
1	B	Un array unidimensional es una secuencia contigua de elementos homogéneos en memoria

Eliminado: <object>

PROMETO

2	A	Con tamaño 5, los índices van de 0 a 4; el último índice es 4
3	C	Acceder fuera de rango lanza ArrayIndexOutOfBoundsException en tiempo de ejecución
4	B	El primer índice es la fila (1 → segunda), el segundo es la columna (2 → tercera)
5	C	En Java la longitud del array se obtiene con la propiedad length, sin paréntesis
6	C	La sintaxis <code>{1, 2, 3, 4, 5}</code> es la forma correcta de inicialización literal
7	B	Cualquier operación que "modifica" un String crea un nuevo objeto en memoria
8	B	<code>equals()</code> compara el contenido de las cadenas; <code>==</code> compara referencias
9	B	<code>trim()</code> elimina espacios en blanco al inicio y al final de la cadena
10	B	<code>StringBuilder</code> permite modificar el texto sin crear nuevos objetos, mejorando el rendimiento