



ESCUELA POLITÉCNICA NACIONAL  
ESCUELA DE FORMACIÓN DE TECNÓLOGOS  
DESARROLLO DE SOFTWARE



PERÍODO ACADÉMICO: 2025-A  
DOCENTE: MSc. Yadira Franco

ASIGNATURA: PROGRAMACIÓN ORIENTA A OBJETO  
PRUEBA 1 BIMESTRE

NOMBRE DEL ESTUDIANTE: Zapata González Felipe Javier

NOTA: /

### Instrucciones Generales:

- Resolver la presente emulación evaluación teórica
- No se aceptan tachones ni borrones, en caso de tenerlos la pregunta no será calificada.

### Objetivo:

- Medir el nivel de comprensión y aplicación de los conceptos enseñados en clase. La evaluación busca determinar la capacidad de los estudiantes para comprender los principios fundamentales de la POO

### Preguntas: (1 punto cada pregunta)

1. Determinar si hay un constructor en la clase Addition, seleccione la opción correcta.

<pre>1. public class Addition{ 2.   int a = 5; 3.   int b= 5; 4.   public void add(){ 5.     int c = a+b; 6.   } 7. }</pre>	<p>a. Sí, en la línea 1.</p> <p>b. Sí, en la línea 4.</p> <p>c. No hay constructor en esta clase.</p> <p>d. Sí, en la línea 6.</p>
---	--

2. Seleccione donde está ubicado el método constructor y cuál es el nombre?

<pre>1. public class Addition{ 2.   public static void main(String[] args){ 3.     Addition add = new Addition(); 4.   } 5. }</pre>	<p>a. Línea 1, nombre: _____</p> <p>b. Línea 2, nombre: _____</p> <p>c. Línea 3, nombre: Addition</p> <p>d. Línea 4, nombre: _____</p>
---	--

3. Defina ¿Qué son los getters y setters en Java y para qué se utilizan?

Los getters y setters son métodos públicos en una clase que se usan para poder acceder a atributos privados con el fin de proteger los atributos al no declararlos como públicos. Get para mostrarlo, y Set para acceder a él si necesitamos modificarlo.

4. ¿Qué significa el modificador de acceso public en Java?

- a. Accesible solo dentro de la misma clase.
- b. Accesible desde cualquier parte.
- c. Pertenece a la clase en lugar de a una instancia específica de la clase.
- d. Pertenece solo al mismo paquete.

5. ¿Qué implica el modificador de acceso private en Java?

- a. Accesible desde cualquier parte.
- b. Accesible solo dentro de la misma clase.
- c. Pertenece a la clase en lugar de a una instancia específica de la clase.
- d. Pertenece solo al mismo paquete.



6. ¿Cuál es la función principal de un constructor en Java?
- a. Inicializar los atributos de una clase con valores predeterminados.
  - b. Crear nuevas instancias de una clase.
  - c. Definir métodos y comportamientos de una clase.
  - d. Inicializar variables locales dentro de un método.
7. ¿Qué significa "instanciar una clase" en Java?
- a. Crear un método en una clase existente.
  - b. Definir un nuevo paquete en el proyecto.
  - c. Inicializar un atributo estático en una clase.
  - d. Crear un objeto de esa clase utilizando el operador new.
8. ¿Cuál es la diferencia entre un constructor vacío y un constructor con parámetros en Java?
- a. El constructor vacío no acepta ningún parámetro, mientras que el constructor con parámetros sí.
  - b. El constructor vacío inicializa todos los atributos de la clase, mientras que el constructor con parámetros solo inicializa algunos.
  - c. El constructor vacío siempre debe estar presente en una clase, mientras que el constructor con parámetros es opcional.
  - d. El constructor vacío es más eficiente en términos de rendimiento que el constructor con parámetros.
9. ¿Cuál es la estructura correcta de un getter:
- a. 

```
public TipoDato getNombreAtributo()
{
}
```
  - b. 

```
public TipoDato getNombreAtributo() {
    return nombreAtributo;
}
```
  - c. 

```
public TipoDato getNombreAtributo(String nombre) {
    return nombreAtributo;
}
```
  - d. 

```
public TipoDato getNombreAtributo(String nombre) {
}
```
10. ¿Cuál es la estructura correcta de un setter en Java:
- a. 

```
public void setNombreAtributo() {
    this.nombreAtributo = nombreAtributo;
}
```
  - b. 

```
public void setNombreAtributo() {
    return nombreAtributo;
}
```
  - c. 

```
public void setNombreAtributo(TipoDato nombreAtributo) {
    this.nombreAtributo = nombreAtributo;
}
```
  - d. 

```
public void setNombreAtributo(TipoDato nombreAtributo) {
    return nombreAtributo;
}
```