

# Programación Avanzada

## Parcial 3

19 de Mayo 2023

### Conceptos

1. La herencia permite que una clase adquiera los atributos y métodos de otra clase.
  - a. Verdadero
  - b. Falso
2. El polimorfismo permite que un objeto se comporte de diferentes formas según el contexto.
  - a. Verdadero
  - b. Falso
3. En Java, una clase puede heredar de múltiples clases al mismo tiempo.
  - a. Verdadero
  - b. Falso
4. El método “super()” se utiliza en una clase hija para llamar al constructor de la clase padre.
  - a. Verdadero
  - b. Falso
5. En el polimorfismo, se puede utilizar una referencia de la clase padre para referirse a un objeto de la clase hija.
  - a. Verdadero
  - b. Falso
6. ¿Cuál de las siguientes opciones describe correctamente la herencia en programación orientada a objetos?
  - A) La capacidad de una clase de heredar propiedades y métodos de otra clase.
  - B) El proceso de encapsular datos y comportamientos en un solo objeto.
  - C) La técnica de ocultar detalles internos de un objeto y mostrar solo la interfaz.
  - D) La capacidad de un objeto de ser tratado como uno de varios tipos posibles.

7. ¿Cuál de las siguientes opciones describe mejor el polimorfismo en Java?
- A) La capacidad de una clase de heredar propiedades y métodos de otra clase.
  - B) El proceso de encapsular datos y comportamientos en un solo objeto.
  - C) La técnica de ocultar detalles internos de un objeto y mostrar solo la interfaz.
  - D) La capacidad de un objeto de ser tratado como uno de varios tipos posibles.
8. ¿Cuál de las siguientes afirmaciones sobre las clases abstractas es correcta?
- A) Se pueden instanciar directamente.
  - B) Pueden heredar de múltiples clases al mismo tiempo.
  - C) Pueden contener implementaciones completas de todos sus métodos.
  - D) Se utilizan como plantillas para crear subclases concretas.
9. ¿Cuál de las siguientes opciones describe mejor el concepto de sobrescritura de métodos en Java?
- A) La capacidad de una subclase de proporcionar una implementación diferente para un método heredado de la clase padre.
  - B) La capacidad de una clase de heredar propiedades y métodos de otra clase.
  - C) La técnica de ocultar detalles internos de un objeto y mostrar solo la interfaz.
  - D) La capacidad de un objeto de ser tratado como uno de varios tipos posibles.
10. ¿Cuál de las siguientes afirmaciones sobre las interfaces en Java es correcta?
- A) Pueden tener atributos y métodos con implementaciones predeterminadas.
  - B) Pueden instanciarse directamente.
  - C) Pueden heredar de otras interfaces y clases.
  - D) Se utilizan para definir un contrato que una clase debe cumplir.

## 2. Sistema de manejo de un Conjunto de Datos

Considere una aplicación de análisis de datos que procesa diferentes tipos de conjuntos de datos.

### 1. Clase `ConjuntoDeDatos`

Defina una clase abstracta `ConjuntoDeDatos` con los siguientes atributos: `nombre` (String) y `tamaño` (int, representa el número de elementos en el conjunto de datos). Esta clase debe tener un constructor que inicialice estos atributos y un método abstracto `describir()` que no tome argumentos y devuelva un String.

### 2. Clase `ConjuntoDeDatosTabular`

Defina una clase `ConjuntoDeDatosTabular` que herede de `ConjuntoDeDatos` y tenga los siguientes atributos adicionales: `numeroDeColumnas` (int) y `numeroDeFilas` (int). El constructor de esta clase debe inicializar todos los atributos de la clase. Sobrescriba el método `describir()` para que devuelva un String con el siguiente formato:

Nombre: <nombre>  
Tamaño: <tamaño>  
Tipo: Tabular  
Filas: <numeroDeFilas>  
Columnas: <numeroDeColumnas>

### 3. Clase ConjuntoDeDatosImagen

Defina una clase ConjuntoDeDatosImagen que herede de ConjuntoDeDatos y tenga los siguientes atributos adicionales: ancho (int) y alto (int). El constructor de esta clase debe inicializar todos los atributos de la clase. Sobrescriba el método describir() para que devuelva un String con el siguiente formato:

Nombre: <nombre>  
Tamaño: <tamaño>  
Tipo: Imagen  
Ancho: <ancho>  
Alto: <alto>

#### 4. Clase AnalizadorDeDatos

Defina una clase `AnalizadorDeDatos` con un atributo `conjuntosDeDatos` (`ArrayList` de `ConjuntoDeDatos`). Esta clase debe tener los siguientes métodos:

- \* `añadirConjuntoDeDatos(ConjuntoDeDatos conjuntoDeDatos)`: añade un `ConjuntoDeDatos` a `conjuntosDeDatos`.
- \* `eliminarConjuntoDeDatos(String nombre)`: elimina un `ConjuntoDeDatos` de `conjuntosDeDatos` basándose en su nombre.
- \* `describirConjuntosDeDatos()`: devuelve un `ArrayList` de `String` donde cada elemento es el resultado de llamar al método `describir()` de cada `ConjuntoDeDatos` en `conjuntosDeDatos`.

5. Opcional: Main para validar el funcionamiento de las clases anteriores.

### Salida

Nombre: Datos de estudiantes  
Tamaño: 1000  
Tipo: Tabular  
Filas: 200  
Columnas: 5

---

Nombre: Imágenes de satélite  
Tamaño: 2000  
Tipo: Imagen  
Resolución Horizontal: 1080  
Resolución Vertical: 720

---

Nombre: Imágenes de satélite  
Tamaño: 2000  
Tipo: Imagen  
Resolución Horizontal: 1080  
Resolución Vertical: 720

---

