



Nº matrícula: \_\_\_\_\_ Grupo: \_\_\_\_\_ Nombre: \_\_\_\_\_

Apellidos: \_\_\_\_\_

**NOTAS:** Las preguntas respondidas correctamente suman 1 punto y las respondidas de forma errónea restan 0.25 puntos. La duración del examen es de 50 minutos.

1. Indica cuál de las siguientes afirmaciones sobre una interfaz es **cierta**:
  - a) Pueden implementar clases.
  - b) Pueden extender clases.
  - c) Pueden extender interfaces.
  - d) Pueden extender interfaces y ser implementados por clases.
  - e) Pueden implementar interfaces y clases.
  - f) Pueden implementar interfaces y ser implementados por clases.
  - g) Pueden extender clases e interfaces
  - h) Ninguna de las anteriores
2. Indica cuál de las siguientes afirmaciones sobre una interfaz es **falsa**:
  - a) Una interfaz es una colección de métodos abstractos.
  - b) Una interfaz puede contener miembros privados.
  - c) Una interfaz puede contener atributos públicos estáticos finales (*public static final*).
  - d) Si una clase implementa una interfaz, debe reescribir todos sus métodos.
3. Indica cuál de las siguientes afirmaciones sobre un método constructor es **cierta**:
  - a) Un constructor no puede recibir argumentos.
  - b) Un constructor puede devolver cualquier tipo de datos.
  - c) Sólo puede haber un método constructor en una clase.
  - d) Desde un constructor se puede invocar a otro constructor de la misma clase.
4. Marcar la respuesta **correcta**. Desde una clase hija se puede invocar al constructor de la clase padre si:
  - a) Lo hacemos desde el constructor de la clase hija.
  - b) Lo hacemos desde un método privado desde la clase hijas.
  - c) Si la clase hija implementa la misma interfaz que la clase padre.
  - d) Sólo si lo hacemos como la primera instrucción de un constructor de la clase hija.
5. Marcar la respuesta **correcta**. ¿Se pueden instanciar objetos de clases abstractas?
  - a) Siempre.
  - b) No es posible.
  - c) Siempre que la clase abstracta tenga métodos estáticos.
  - d) Siempre la clase abstracta tenga un constructor público.
  - e) Siempre que la clase abstracta implemente una interfaz.
  - f) Siempre que la clase abstracta extienda una interfaz.

6. Marcar la respuesta **correcta**. Para poder llamar a un método de una clase sin instanciarla, es necesario que:
- a) La clase esté declarada como estática.
  - b) El método sea estático.
  - c) Se utilice el constructor por defecto.
  - d) El método sea estático y público.
7. Completa la frase con la respuesta **correcta**. El uso de clases anidadas (clases internas) \_\_\_\_\_ la encapsulación.
- a) Incrementa
  - b) Disminuye
  - c) No afecta
  - d) Disminuye ligeramente
8. Se pide implementar el método `setEdad` que recibe un argumento de tipo entero y que lanza la excepción `ExEdad` si dicho argumento es mayor que el valor 150.

```
public void setEdad(int edad) throws ExEdad {  
    if (edad>150)  
        throw new ExEdad();  
}
```

9. Se pide implementar el método `toString` que permite generar una cadena (`String`) con los contenidos de una propiedad `agenda` declarada como: `HashMap<String, Persona>`

```
public String toString() {  
    Set<String> claves = agenda.keySet();  
    String aux = "[";  
    for (String clave : claves)  
        aux = aux+"("+clave+", " +agenda.get(clave) +")";  
    return aux+"]";  
}
```

10. Dado el siguiente interfaz: `public interface IFlota <E extends Vehiculo>.` Se pide, escribir la cabecera de la clase pública `Flota` que hace uso de ese interfaz y escribir una instrucción que declare la propiedad `camiones` como una flota de objetos de tipo `Camion`. Usar el constructor vacío.

```
public class Flota <E extends Vehiculo> implements IFlota<E>  
  
/* respuestas válidas */  
IFlota<Camion> camiones = new Flota<Camion>();  
Flota<Camion> camiones = new Flota<Camion>();  
IFlota<Camion> camiones = new Flota<>();  
Flota<Camion> camiones = new Flota<>();
```