

EXAMEN PARCIAL – UNIDADES 5 Y 6

TRIMESTRE: PRIMERO

Fecha:

CICLO: Desarrollo de Aplicaciones Web.

CURSO: 1º

CALIFICACIÓN:

MÓDULO: Programación

Turno: Mañana

Nombre:

Apellidos:

Instrucciones: Esta prueba tiene como finalidad evaluar los aprendizajes de Programación. Lee atentamente y responde escribiendo el código más adecuado.

Si las instrucciones no se siguen como se especifican el examen no será evaluado

PARTE PRÁCTICA. TIPO B.

- El examen práctico tiene una puntuación máxima de 10 puntos.
- Para superar la parte práctica se requiere alcanzar un mínimo de 5 puntos.

1. **(3 puntos)** Desarrolla un sistema de gestión de empleados en una empresa. Los empleados tienen un rol (ADMIN, GERENTE, EMPLEADO) y un salario. Debes calcular el salario final de cada empleado en función de su rol y un bono estático. La clase Empleado tiene métodos para calcular el salario global de todos los empleados, y una constante final para el bono base.

- Requisitos:
 - Crea un tipo enumerado Rol con los valores ADMIN, GERENTE, EMPLEADO.
 - La clase Empleado debe tener los atributos nombre, rol, salarioBase y un atributo BONO_BASE que se aplica a todos los empleados.
 - Si el rol del empleado es ADMIN, además del BONO BASE tendrá un BONO de 1000€.
 - Si el rol del empleado es GERENTE, además del BONO BASE tendrá un BONO de 500€.
 - Implementa un método en la clase Empresa que calcule el salario total de todos los empleados usando un array de Empleado como parámetro.
 - Los empleados deben ser gestionados en un array dentro de la clase Empresa.

Resultado de ejecución:

Juan : 4500,00€

Ana : 3000,00€

Carlos : 2000,00€

Lucía : 2100,00€

Total de salarios: 11600,00€

2. **(3 puntos)** Desarrolla un sistema para gestionar el inventario de una tienda online. Cada producto tiene un nombre, precio y cantidad en stock. Los productos pueden tener un descuento fijo. Los descuentos se aplican solo a productos disponibles en inventario. Además, la tienda tiene un descuento global (10%) para todos los productos, que se aplica al precio después de los descuentos individuales.

- Requisitos:
 - Crea un tipo enumerado `CategoriaProducto` con valores como `ELECTRONICA`, `ROPA`, `ALIMENTACION`.
 - Cada producto tendrá asociado una categoría.
 - Los descuentos para cada categoría son:
 - Electrónica – 5%
 - Ropa – 10%
 - Alimentación – 15%
 - Crea una clase `Producto` que tenga los atributos `nombre`, `precio`, `cantidadEnStock` y un atributo estático `DESCUENTO_GLOBAL`.
 - Crea una clase `Inventario` que contenga un array de productos.
 - Implementa un método que calcule el precio final de todos los productos considerando los descuentos y la cantidad en stock.

Resultado de ejecución:

Producto: Laptop - Precio unitario: 1026,00€ - Stock: 10 - Precio final: 10260,00€

Producto: Camiseta - Precio unitario: 16,20€ - Stock: 50 - Precio final: 810,00€

Producto: Manzanas - Precio unitario: 1,91€ - Stock: 200 - Precio final: 382,50€

Total del inventario: 11452,50€

3. **(4 puntos)** Diseña un sistema para gestionar los pedidos en un restaurante. El restaurante tiene un menú con diferentes tipos de platos. Los clientes pueden hacer pedidos, y dependiendo del número de platos pedidos, se les aplica un descuento en función de la categoría del plato (`ENTRANTE`, `PLATO_PRINCIPAL`, `POSTRE`). Además, si un cliente ha realizado más de 5 pedidos, se le aplica un descuento adicional de fidelidad del 15%. Los platos tienen un precio, y cada pedido está compuesto por una lista de platos. Los platos y los pedidos deben ser gestionados mediante arrays. El IVA será el 21%

- Requisitos:
 - Enumerados:
 - Crea un tipo enumerado `CategoriaPlato` con los valores `ENTRANTE`, `PLATO_PRINCIPAL`, `POSTRE`.
 - Clases:
 - Crea una clase `Plato` con los atributos `nombre`, `categoria` (tipo enumerado), `precio` y un método `calcularPrecioFinal` que aplica un descuento en función de la categoría. Los entrantes tienen un descuento del 10%. Los platos principales NO tienen descuento. Los postres tienen un descuento del 20%.
 - Crea una clase `Cliente` con los atributos `nombre`, `pedidos[]` (array de pedidos), un contador de pedidos realizados y un método que calcule el descuento por fidelidad si tiene más de 5 pedidos en el mes.
 - Crea una clase `Pedido` con los atributos `numeroPedido`, `platos[]` (array de platos), `cliente`, `fecha` (fecha del pedido), y un método que calcule el precio total del pedido considerando los descuentos por categorías de plato y fidelidad del cliente.
 - Métodos estáticos:
 - Crea un método estático en la clase `Restaurante` que calcule el precio final de un pedido dada una lista de platos y un cliente.
 - Métodos finales y atributos finales:
 - La clase `Cliente` debe ser final para evitar que sea heredada.
 - Los atributos `descuentoFidelidad` y `umbralPedidos` en la clase `Cliente` deben ser finales.

Resultado de ejecución:

Pedido #1

Cliente: Carlos Sánchez

Fecha: 10/01/2025

Precio total del pedido con IVA: 29,40€

Pedido #2

Cliente: Ana Martínez

Fecha: 14/01/2025

Precio total del pedido con IVA: 23,96€