Examen Módulo II - Propuesta 4/4

1.- Teniendo definido el siguiente código:

abstract class Persona { private String nombre; public Persona (String n) { this.nombre = n; } public abstract void hablar (); } class Empleado extends Persona { private String puesto; public Empleado (String n, String pu) { super(n); this.puesto = pu; } } Tras la siguiente ejecución:

public static void main(String[] args) { Empleado emp = new Empleado("Pepe","Vendedor"); emp.hablar(); } Si el código está bien, ¿qué se muestra en la consola? Si no lo está, ¿cuál es el error?

2.- Complete la siguiente afirmación: “La siguiente palabra reservada nada tiene que ver con el concepto de herencia en la programación orientada a objetos...”

a. static. b. extends. c. implements. d. super.

3.- El método convertir de la clase Conversor se encarga de recibir un valor en dólares y devolverlo en pesos. Se sabe que el siguiente código genera una excepción de tipo RuntimeException. Explique con sus palabras cómo lo corregiría (sin usar try – catch).

public class Conversor { public double VALOR\_DOLAR; public int convertir (int dolares) {return dolares \* this.VALOR\_DOLAR; } } 4.- Escribe una aplicación que:

Un banco desea enviar a sus clientes una carta, mensaje de correo electrónico o mensaje al móvil (según los datos y preferencias de cada cliente) de agradecimiento por cada uno de ljos productos financieros que ha contratado el cliente. Cada cliente puede haber contratado varios productos (tarjetas de crédito, débito, plan de pensiones, seguro, fondos de inversión, etc.) y cada tipo de producto requiere una carta/mensaje diferente. Se quiere usar una jerarquía de clases para representar los diferentes tipos de producto y otra para los diferentes tipos de mensajes. Para el programa hay que usar un Arraylist del tipo más adecuado más un iterador para gestionar el envío de mensajes conjuntamente. a) Identificar la estructura y las relaciones de herencia y de uso de las clases necesarias para la aplicación que realice el trabajo descrito. b) Dibujar un diagrama UML de estas clases. c) Implementar el método "main" del programa mostrando cómo se gestiona el envío de los mensajes. d) ¿Qué cambios serían necesarios en el diseño y programa para mandar otro tipo de mensaje, algo de publicidad sobre nuevos productos, conjuntamente con el mensaje de agradecimiento?

5. - Indique cuál de las siguientes afirmaciones es correcta:

a. La interfaz de una clase describe lo que hace la clase y cómo puede usarse pudiendo mostrar parte de su implementación. b. Un mapa es una colección que almacena entradas de ternas de valores llave/valor/posición. c. La documentación de una clase debe ser suficientemente detallada como para que otros programadores puedan usar la clase sin necesidad de leer su implementación. d. Los modificadores de acceso definen las restricciones de uso de un objeto para determinados métodos, constructores o campos.