

PROGRAMACIÓN ORIENTADA A OBJETOS

PRACTICA 5-6: Abstracción y Encapsulamiento & Organización de clases

Objetivo General:

(P5) Aplicar el concepto de abstracción para el diseño de clases que integran una solución, utilizando el encapsulamiento para proteger la información y ocultar la implementación.

(P6) Organizar adecuadamente las clases según su funcionalidad o propósito bajo un namespace o paquete.

Objetivo de clase:

Realizar un programa orientado a objetos que permita aplicar los conceptos más importantes de los modificadores de acceso, uso de paquetes y composición de clases

ACTIVIDADES PARA EL DESARROLLO DE LA PRÁCTICA

Ejercicio 1.- Composición de clases

Se requiere diseñar un programa para la administración de un ejército, para ello se deben modelar los tipos de dato abstracto: Militar, Batallón, División

Los modelos de esos tipos de datos son libres, deberán contar con al menos 3 atributos y métodos (constructor, de acceso y otros)

- Define las clases para cada uno de los tipos de dato que se indican
- Militar y Batallón deberán tener una variable estática para tener control de cuantos se han creado en la ejecución del programa
- Agrega por lo menos un constructor en cada una de las clases, para que al crearlas se le pidan al usuario los valores de inicio
- Agrega los métodos correspondientes a cada clase para la interacción entre los objetos que se van a crear
- Ejemplos de interacciones entre objetos

AsignarMilitaraDivision(), AsignarDivisionaBatallon()

Mostrar_información()

- El diseño de otros métodos y otras funciones para el desarrollo del programa es libre
 - Se recomienda el uso de paquetes y modificadores de acceso, si el alumno los utiliza lo deberá hacer correctamente
 - Al ejecutar el programa se realizará la creación de una estructura militar (División → Batallón -> Militar(es) siguiendo la metodología Top-Down y solicitando al usuario la información
 - Deberás agregar métodos de acceso a las clases realizadas (getters y setters), a través de ellos se deberán agregar las restricciones apropiadas para que solamente se puedan asignar valores adecuados de acuerdo con los atributos correspondientes.
- ✓ El usuario final deberá contar con un menú de usuario para seleccionar las diferentes actividades del programa y deberá ser capaz de crear dos divisiones con dos batallones y 3 militares por batallón respectivamente, y mostrar la información de cada uno de ellos en pantalla.
- ✓ Recuerda que debes hacer un análisis adecuado de los elementos necesarios para la elaboración del programa, dificultades, problemas, errores etc.

CONTINÚA

Ejercicio 2.- Modificadores de acceso

Revisa el proyecto proporcionado que se trata de un programa para los registros del personal de una empresa. Para ello deberás revisar los siguientes aspectos

- Deberás revisar las clases Empleado, Directivo, Gerente, y Operativo. Explica los elementos que tienen dichas clases, atributos, métodos, constructores etc.
- Dibuja la jerarquía de herencia proporcionada y explica cómo están organizados los paquetes
- Explica cómo se utilizan los modificadores de acceso para encapsular los atributos y métodos de la siguiente manera:
- ¿Quién puede establecer salarios?
- ¿Qué salarios puede ver cada elemento del programa?
- Revisa el programa principal (main) e indica qué ocurre en él
- Indica qué ocurre si la clase principal se mueve a otro paquete, qué problemas se generan y cómo se podrían solucionar

Adicional

Realiza cambios en el programa, agrega o quita atributos, cambia modificadores y haz pruebas para entender de mejor manera los modificadores de acceso, agrega estos comentarios al reporte de la práctica

CONCLUSIONES

Escribe las conclusiones de tu práctica explicando tu avance en la comprensión de los temas vistos en esta práctica (composición de clases, modificadores de acceso y uso de paquetes)