Trabajo práctico Nro. 1

|  |  |  |
| --- | --- | --- |
| Logo Departamento 2.jpg | Asignatura: Programación II | |
|  | |
| Cursado: Segundo semestre | Horas semanales: |
|  | Horas semestrales:  Cantidad estimada de horas  semestrales/anuales. |
| Carrera: Tecnicatura Universitaria en  Programación | Nivel (Año):  1°  2°  3° |
| Ciclo Lectivo: 2024 |

Integrantes del grupo: Luciano Aguilera

DOCENTES:

|  |  |  |
| --- | --- | --- |
| Nombre del Profesor | Periodo | Cantidad horas materia |
| Cinthia Rigoni |  | 6 horas |

¿Qué es un TAD?

Un Tipo de Dato Abstracto (TDA) es un modelo que define los valores y las operaciones que pueden realizarse sobre ellos. Se llama abstracto porque su objetivo es que el usuario no necesite conocer los detalles de la representación interna ni cómo se implementan las operaciones. Esta práctica nos proporciona un nivel de abstracción que permite desacoplar el código que utiliza un TDA del que lo implementa.

¿Dónde se produce el encapsulamiento?

El encapsulamiento puede ocurrir tanto en los datos como en los métodos de una clase. En el caso de los datos, se utiliza para proteger la integridad de la información, permitiendo el acceso a ellos únicamente a través de métodos específicos como getters y setters. En cuanto a los métodos, pueden ser diseñados como públicos, permitiendo su acceso desde fuera de la clase, o como privados, restringiendo su uso únicamente al interior de la clase.

¿Cuáles son las semejanzas y las diferencias entre funciones, procedimientos y métodos?

Los métodos y las funciones en Java pueden realizar tareas similares y son funcionalmente idénticos. Sin embargo, la diferencia clave radica en su asociación con objetos. Un método siempre está vinculado a un objeto o clase específica, requiriendo una instancia para ser utilizado. Por otro lado, una función puede existir independientemente y ser invocada sin necesidad de un objeto previo.

Los procedimientos son básicamente un conjunto de instrucciones que se ejecutan sin retornar ningún valor. En el contexto de Java un procedimiento es básicamente un método cuyo tipo de retorno es void.

¿Qué es UML? ¿Y cómo se representa una clase en ese lenguaje?

UML (Unified Modeling Language) es un lenguaje estándar utilizado en el diseño y la documentación de sistemas de software. Se utiliza para visualizar, especificar, construir y documentar artefactos de un sistema. Proporciona diagramas gráficos que permiten a los desarrolladores y analistas representar diversos aspectos de un sistema de software de manera clara y comprensible.

Marcar con cruz.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | No se aplica a clases | Solo se aplica a atributos | Solo se aplica a clases | Se aplica a atributos, métodos y clases |
| Public |  |  |  | X |
| Private |  | X |  |  |
| Protected |  | X |  |  |
| Static |  |  |  | X |
| Final |  |  |  | x |
| Primera letra en  minúscula |  | X |  |  |
| Primera letra en  mayúscula |  |  | X |  |

Verdadero o falso

Un constructor…

Es el método principal para ejecutar un programa. F

Crea instancias. V

Devuelve el valor de un atributo privado. F

Tiene sentencia return. F

Siempre existe uno por defecto, sin parámetros ni inicializaciones de atributos. V

Se puede sobreescribir. V

Se puede sobrecargar. V

Su nombre se escribe con mayúscula. V

Su calificador de acceso es statu. F

Su tipo de devolución no se indica y corresponde a la clase. V

Un método…

Puede tener múltiples parámetros con el mismo nombre, siempre y cuando tengan tipos diferentes

Puede sobrecargarse. V

Puede sobreescribirse. V

Puede ser static. V

Puede ser tanto public como protected, pero no private. F

Un método puede tener un modificador de acceso final. V

Calificadores de acceso. Completa.

Se necesita que cualquiera pueda acceder al color de un vehículo. Entonces, declaro color como: public



Se necesita que color se pueda acceder a través no sólo de vehículo, sí no ahora también de Buses, y como todos sabemos un bus es un tipo de vehículo, entonces también deberá tener acceso a color. Entonces, declaro color como: protected

Se necesita que color se pueda acceder solamente para vehículo. Entonces, declaro color como: private

¡A CODIFICAR!

Se desea desarrollar un sistema de gestión de empleados para una empresa. El sistema debe permitir registrar empleados de dos tipos diferentes: gerentes y trabajadores. Cada empleado debe tener un nombre, una edad y un salario.

Los gerentes tienen la capacidad de organizar actividades dentro de un departamento específico, mientras que los trabajadores están encargados de producir en un área determinada.

Implementa un sistema que modele esta situación utilizando herencia en Java. Define una clase base llamada Empleado que contenga los atributos y métodos comunes para todos los empleados, como el nombre, la edad, el salario y la capacidad de trabajar.

Luego, crea dos subclases: Gerente y Trabajador, que hereden de Empleado. Los gerentes deben tener un atributo adicional para almacenar el departamento en el que trabajan, así como un método para organizar actividades dentro de ese departamento.

Por otro lado, los trabajadores deben tener un atributo para indicar el área en la que trabajan y un método para producir en esa área.

Finalmente, en el programa principal, crea instancias de ambas subclases y muestra cómo se utilizan los métodos específicos de cada tipo de empleado, así como sus atributos.

Se desea implementar un programa en Java para modelar diferentes figuras geométricas, como círculos y rectángulos. Cada figura geométrica debe tener la capacidad de calcular su área y su perímetro.

Define una clase base llamada FiguraGeometrica que contenga métodos abstractos para calcular el área y el perímetro de la figura. Luego, crea subclases para representar diferentes tipos de figuras geométricas, como Circulo y Rectángulo, que hereden de la clase base FiguraGeometrica.

En la subclase Círculo, implementa métodos para calcular el área y el perímetro de un círculo, utilizando el radio como atributo de la clase. En la subclase Rectángulo, implementa métodos para calcular el área y el perímetro de un rectángulo, utilizando la longitud y la anchura como atributos de la clase.

En el programa principal, crea instancias de diferentes figuras geométricas (al menos un círculo y un rectángulo) y muestra sus áreas y perímetros.

Definir una clase Libro para manejar la información asociada a un libro. La información de interés para un libro es: el título, el autor y el precio. Los métodos de interés son:

Un constructor para crear un objeto libro, con título y autor como parámetros.

Imprimir en pantalla el título, los autores y el precio del libro.

Métodos get y set para cada atributo de un libro.

Se debe extender la clase Libro definiendo las siguientes clases:

Libros de texto con un nuevo atributo que especifica el curso al cual está asociado el libro.

Libros de texto de la Universidad Nacional de Colombia: subclase de la clase anterior. Esta subclase tiene un atributo que especifica cuál facultad lo publicó.

Novelas: pueden ser de diferente tipo, histórica, romántica, policíaca, realista, ciencia ficción o aventuras.

Para cada una de las clases anteriores se debe definir su constructor y redefinir adecuadamente el método para visualizar del objeto.

Repositorio con los ejercicios de código:

https://github.com/LuchoAZ/Programacion2\_LucianoAguilera/tree/main/TP1