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 de la Cátedra:**

* **DOCENTES:**

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

* 1. ¿Qué es un TAD?

**TAD significa (Tipo Abstracto de Datos). Es una abstracción que se encarga de “Esconder” los detalles de la implementación de un código, esto permite que estos sean más estructurados y ordenados, lo que facilita su entendimiento y futuras modificaciones por los usuarios.**

* 1. ¿Dónde se produce el encapsulamiento?

**El encapsulamiento se produce en las clases, cuando definimos los atributos, lo podemos hacer de manera public, private o protected, según la visibilidad y capacidad de modificación que queramos agregar de los datos en distintos paquetes o clases.**

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

**Las funciones, procedimientos y métodos son estructuras de programación que permiten agrupar y organizar código para realizar tareas específicas. La principal diferencia radica en su manejo del valor de retorno: las funciones devuelven un valor, los procedimientos no retornan nada y los métodos, específicamente en la programación orientada a objetos, pueden o no tener un valor de retorno. Además, los métodos están asociados a clases, mientras que funciones y procedimientos pueden existir fuera del contexto de la programación orientada a objetos.**

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

**El Lenguaje de Modelado Unificado (UML) se utiliza para representar gráficamente la estructura de un código de software a través de diagramas, estos incluyen clases, objetos, atributos, métodos etc.**

**Se representan con un rectángulo dividido en 3 partes, la superior con el nombre de la clase, la del medio con el tipo y el nombre de los atributos, y por último la parte de abajo donde se representan los métodos.**

* 1. 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** |  |

* 1. 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. **F**
       - Se puede sobreescribir. **F**
       - Se puede sobrecargar. **V**
       - Su nombre se escribe con mayúscula. **V**
       - Su calificador de acceso es static. **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. **F**
       - 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. **F**
  2. Calificadores de acceso. Completa.

1. Se necesita que cualquiera pueda acceder al color de un vehículo. Entonces, declaro color como: **Public**
2. 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**

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

**A CODIFICAR!**

* 1. 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.

* 1. 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.

* 1. 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.