



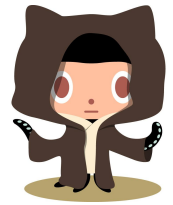
GUÍA TRABAJO PRÁCTICO N°3 JAVA JEDI

Les presento la guía de **trabajos prácticos opcionales**, todos estos los pueden realizar para obtener mayor práctica en el lenguaje.

UTILIZAR LO VISTO EN LA CLASE DE PROGRAMACIÓN ORIENTADA A OBJETOS CON JAVA PARA LOS EJERCICIOS.

Ejercicios nivel : Iniciado JAVA JEDI

1. Crear una clase alumno, el cual tenga atributos como :
 - a. Nombre
 - b. Apellido
 - c. Fecha de nacimiento
 - d. EdadLuego cree 3 objetos de la clase e imprima por pantalla el nombre, apellido y edad de cada uno.
2. Crear una clase curso, el cual tenga atributos como :
 - a. Nombre
 - b. Si está habilitado o no
 - c. Descripción
 - d. Alumnos.Cree uno o más cursos y asigne a la lista de alumnos, objetos del ejercicio 1. Luego imprima el nombre del curso, su descripción y luego los nombres de los alumnos de dicho curso.
3. Se desea modificar el ejercicio 1. Se requiere que ahora se carguen N alumnos con sus datos, se los guarde en una estructura de datos. Luego imprima por pantalla el nombre, apellido y edad de cada uno.
4. Se desea modificar el ejercicio 1. Se requiere que ahora se carguen N cursos con sus datos, se los guarde en una estructura de datos. Luego imprima el nombre del curso, su descripción y luego los nombres de los alumnos N de dicho curso.
5. Modificar los ejercicios anteriores para que los datos sean ingresados por consola. Usar clase Scanner.



Ejercicios nivel : Padawan JAVA JEDI

1. **Aplicación de lista de tareas simple.** Implementar una clase Tarea con atributos como título, descripción y fecha de vencimiento. Luego, usa un ArrayList para almacenar estas tareas. La aplicación debe permitir al usuario agregar, eliminar y listar tareas.
2. **Agenda de Contactos.** Implementar una clase Contacto con atributos como nombre, número de teléfono y correo electrónico. Utiliza una LinkedList para almacenar los contactos. La aplicación debe permitir agregar, eliminar, buscar y mostrar la lista de contactos.
3. **Aplicación de estudiantes y calificaciones.** Define una clase Estudiante con atributos como nombre, número de identificación y calificaciones. Cree una clase calificaciones y para cada estudiante cree un par de calificaciones, cada calificación tiene atributos como calificación (1 al 10) y fecha de creación. Utiliza un ArrayList para almacenar las calificaciones y los estudiantes.
 - a. La aplicación debe permitir :
 - i. Agregar estudiantes junto con sus calificaciones
 - ii. Calcular promedios de calificaciones
 - iii. Mostrar por pantalla identificación, nombre del estudiante y promedio de calificaciones.

Ejercicios nivel : Caballero JAVA JEDI

1. **Diccionario simple.** Crea una aplicación de diccionario que almacene palabras y sus definiciones utilizando un HashMap. Debes definir una clase Diccionario que tenga un atributo nombre de diccionario y un atributo (De tipo Hashmap) palabras. Se debe definir una clase Palabra que tenga una palabra y una definición, y luego almacenar instancias de esta clase en el HashMap. La aplicación debe permitir agregar, buscar y mostrar las definiciones de palabras.
2. **Gestión de inventario.** Desarrolla una aplicación de gestión de inventario utilizando un HashMap. Define una clase Producto que incluya propiedades como nombre, precio y cantidad en stock. Usa el HashMap para mantener un registro de productos y sus existencias.
 - a. Permite a los usuarios :
 - i. Agregar producto
 - ii. Actualizar producto
 - iii. Consultar productos.
3. **Libros de biblioteca.** Simula un sistema de registro de libros en una biblioteca utilizando un HashMap. Crea clases como Libro, Autor y Biblioteca. Los libros pueden tener autores y se almacenan en el HashMap junto con su disponibilidad. Una biblioteca puede tener N libros.
 - a. La aplicación debe permitir:



- i. Buscar libros por título o autor
 - ii. Registrar libros.
- b. Crear el diagrama de clases del escenario.

Ejercicios nivel : Maestro JAVA JEDI

Aplicación Gestión de una Tienda de Electrónica

Supongamos que estás desarrollando un programa para gestionar una tienda de electrónica. La tienda vende productos electrónicos como teléfonos, computadoras portátiles, tabletas y accesorios. Debes modelar las clases y relaciones para manejar este escenario.

Clases y Objetos:

- Producto Electrónico: Crea una clase Producto Electrónico que represente un producto en la tienda. Esta clase debe tener propiedades como nombre, descripción, precio y cantidad en stock.
- Cliente: Crea una clase Cliente que represente a un cliente de la tienda. Un cliente debe tener propiedades como nombre, dirección y una lista de productos que ha comprado.
- Tienda Electrónica: Crea una clase Tienda Electrónica que represente la tienda en sí. Esta clase debe tener una lista de productos disponibles, una lista de clientes y métodos para realizar ventas.

Relaciones:

- La clase Tienda Electrónica debe contener una lista de Productos Electrónicos disponibles para la venta.
- La clase Cliente debe tener una lista de productos que ha comprado.
- La clase Tienda Electrónica debe tener métodos para realizar ventas, que incluyan la actualización de las existencias de productos y el registro de ventas para los clientes.



Operaciones:

- La tienda debe ser capaz de agregar nuevos productos a su inventario y registrar nuevos clientes.
- Debe ser posible para los clientes comprar productos y para la tienda mantener un registro de las ventas.

Tareas:

1. Desarrolle el diagrama de clases que represente el escenario.
2. En base al diagrama de clases crear las clases para Producto Electrónico, Cliente y Tienda Electrónica con las propiedades, relaciones y métodos necesarios.
3. Implementa métodos para agregar productos al inventario y para registrar nuevos clientes en la tienda.
4. Desarrolla un método de venta que permita a los clientes comprar productos. Actualiza las existencias de productos y registra las ventas para los clientes.