

Evaluación Parcial N°2

Sigla	Nombre Asignatura	Tiempo Asignado	% Ponderación
DSY1102	Desarrollo Orientado a Objetos	120 minutos	40%

1. Situación evaluativa

<input checked="" type="checkbox"/>	Ejecución práctica	<input type="checkbox"/>	Entrega de encargo	<input type="checkbox"/>	Presentación
-------------------------------------	--------------------	--------------------------	--------------------	--------------------------	--------------

2. Agente evaluativo

<input checked="" type="checkbox"/>	Heteroevaluación	<input type="checkbox"/>	Coevaluación	<input type="checkbox"/>	Autoevaluación
-------------------------------------	------------------	--------------------------	--------------	--------------------------	----------------

3. Tabla de Especificaciones

Resultado de Aprendizaje	Indicador de Logro (IL)	Ponderación Indicador Logro
RA2 Aplica conceptos avanzados del paradigma orientado a objetos en componentes de software, para dar respuesta a requerimientos de clientes.	IL 2.1 Aplica las sentencias de ciclos, para ser representadas en la solución según el requerimiento del usuario en un caso de negocios.	15%
	IL 2.2 Aplica conceptos de encapsulación avanzada permitiendo cambios internos sin afectar otros componentes del sistema del software.	15%
	IL 2.3 Configura una colección para almacenar información temporal en el programa según lo solicitado por el usuario.	30%

IL 2.4 Aplica los conceptos de herencia en la programación orientada a objetos que permitan dar solución a un problema planteado un caso de negocios.	20%
IL 2.5 Utiliza abstracción y polimorfismo de manera efectiva, para permitir la creación de interfaces flexibles y genéricas que puedan adaptarse a diferentes situaciones.	20%
Total	100%

4. Instrucciones generales para el/la estudiante

Esta es una evaluación que corresponde a una ejecución práctica y tiene un 40% de ponderación sobre la nota final de la asignatura. El tiempo para desarrollar esta evaluación es de **120 minutos** y se realiza de manera **individual** en laboratorio PC avanzado.

Esta evaluación consiste en la resolución de un caso donde el estudiante debe aplicar colecciones y herencia. Por otro lado, realizar métodos abstractos, polimorfismos, sobrecarga e implementación de interface, entre otros.

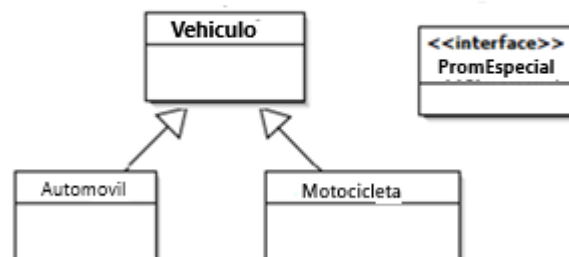
Para dejar registro deberá realizar los siguientes puntos:

- Programar en el lenguaje JAVA utilizando el IDE NetBeans el diagrama de clases generado.
- Deberá dejar registro del trabajo realizado por medio de la plataforma de Blackboard, para ello debe comprimir la carpeta de su proyecto y subirla a la actividad correspondiente en .ZIP

5. Evaluación

Concesionaria de venta de vehículos “La tuerca loca”

1. Se pide implementar la siguiente jerarquía de clases:



Se requiere modelar un sistema para gestionar los vehículos de una concesionaria. La información por considerar es la siguiente:

- Los **Automóviles**: tienen un tipo de carrocería (sedán o SUV), cantidad de puertas, código, una marca, un modelo, un precio base y un año de fabricación.
- Las **Motocicletas**: código, una marca, un modelo, un precio base y un año de fabricación, tipo (deportiva o clásica) y cilindrada (cc).
- La interfaz **PromEspecial** contiene el atributo DTO (12.3%) y el método descuentoFeriado. Este método aplicará el descuento DTO, solo si la marca es **Ford** y el tipo de carrocería es **SUV**. En cualquier otro caso, el descuento será cero.

Además, se necesitan los siguientes métodos:

- **ajustePrecio**: disminuirá el precio base en un 15% si el tipo es “clásica”.
 - **aplicarImpuesto**: aumentará el precio base en un X% si el tipo de carrocería es SUV o si la cilindrada es mayor a 600cc.
- Crear una clase manejadora, que se llamará **Concesionaria**, deberá permitir gestionar una colección de Vehiculo. Los métodos que esta clase debe tener son:
- **ingresarVehiculo()**: agrega un vehículo a la colección. Debe validar que no esté duplicado por su código.

- **buscarVehiculo():** busca y devuelve un vehículo por su código.
 - **aplicarAjusteATodos():** recorre la colección y aplica el método `ajustePrecio` a todos los vehículos que cumplan con la condición (Devolverá el total de vehículos a los que se les aplicó el ajuste).
 - **calcularDescuentoTotal():** recorre la colección y calcula el total de dinero ahorrado por concepto de descuentoFeriado.
- Construya las clases que represente la situación anterior en Java utilizando NetBeans, implementando además constructores, accesadores, mutadores.

En la aplicación, implemente el siguiente menú:

LA TUERCA LOCA

1. **Ingresar Vehículo:** permite agregar un Automóvil o una Motocicleta a la colección.
2. **Mostrar Información:** muestra toda la de un vehículo dado su código.
3. **Aplicar Ajuste de Precio:** ejecuta el método `aplicarAjusteATodos`, mostrando mensaje adecuado.
4. **Aplicar Impuesto:** ejecuta el método `calcularDescuentoTotal`.
5. **Salir**

Notas para el estudiante:

- Considere el uso de clase y métodos **abstractos** donde sea necesario para modelar la jerarquía de manera correcta.