

<u>UTN – FR Mar del Plata – TSP</u> <u>Laboratorio III</u>	<i>Nombre y Apellido</i>	<i>Nota</i>
---	--------------------------	-------------

IMPORTANTE:

- Crear un proyecto con su nombre y apellido.
- Realizar todas las funciones que se indican. Añadir comentarios
- Desarrollar un método main() que invoque únicamente a las funciones que compilan correctamente. No se pide ingreso por teclado de ningún tipo.

Tabla de puntuación:

Obtenido	10	20	30	40	50	60	70	80	90	100
Nota	1	2	3	4	5	6	7	8	9	10

Un concesionario de vehículos nos pide un sistema para la gestión de su inventario. Este concesionario vende motos y autos. De ambos vehículos sabemos la patente, la marca, el modelo, consumo de combustible, cantidad de ruedas y precio. El auto tiene una lista de prestaciones y el modo de pasar los cambios. En cambio la moto tiene una cilindrada y si dispone de baúl debajo del asiento.

- Crear constructores vacíos, completos.
- Crear getters/setters según crea conveniente.
- Sobreescibir el método toString() en todas las clases.
- Sobreescibir el método equals()

Puntaje: 20 pts

Todos los vehículos deben almacenarse dentro de un ArrayList con las funciones de agregar, buscarPorPatente, contar, devolverListado y eliminarPorPatente.

- Crear la clase envoltorio/contenedora.
- Crear y codificar los métodos correspondientes.
- Invocar a dichos métodos para probar funcionamiento.

Puntaje: 30 pts.

Todos los vehículos tendrán un precio. Los autos salen 15 (no importa que son estos quince) y las motos cuestan 7. En una entrevista más profunda, y quien dice profunda dice en un bar a altas horas de la noche, el cliente nos confeso que venderá yates. Los yates NO compartiran nada con los vehiculos pero deben tener una forma de calcular su precio. Los yates tiene un valor de 100.

- Crear ?????? necesaries para soportar los yates.
- Crear la clase yate con atributos propios de un yate. Algún marinero o dueño de yate en la sala?

- Crear un método que nos permita calcular el total del inventario. Se puede hacer de MUCHAS formas pero hay una o varias que son las correctas en términos conceptuales con polimorfismo, herencia, etc)

Puntaje: 30 pts.

Para probar todo nuestro sistema haremos un lindo main con un switch que deberá estar modularizado (recuerden que los métodos son públicos y static y no deben ir variables de clase (globales) sino pasaje de parámetros). El menú deberá contar con todas las opciones necesarias para demostrar el correcto funcionamiento.

Puntaje: 10 pts.

Utilización de conceptos de la POO. Diseño de clases. Comentarios. Prolijidad. Modularidad, acoplamiento, cohesión. Entrega del parcial. Originalidad.

Puntaje: 10 pts.

¡Los mejores éxitos!

