

Práctico 5 – Objetos

Ejercicio 1

Crear un objeto llamado Punto. Contendrá dos coordenadas x, y. Un método constructor, además de los get y los set.

Ejercicio 2

Crear un objeto Zapatilla, que tendrá los campos **marca, modelo, numero, industria**. Hacer sus set y sus get. Pedir en otro programa, al usuario que ingrese un par de zapatillas de un amigo, y otro suyo. Determinar si son de la misma marca.

Ejercicio 3

Crear un objeto **Auto**, que tendrá los campos **auto, marca, modelo, patente, km, ultcambioAceite**. Hacer sus Set y sus Get y dos métodos constructores. Los métodos constructores serán: uno completo y otro que será para un auto 0km recién adquirido donde solo se le darán los campos para auto, marca y patente, llenando como corresponde en el constructor modelo y km.

Además agregar un método **cambioAceite** que permita determinar si hay que hacerle cambio de aceite o no.

(Considerar que se hace cada 5.000km)

** Importar java.util.Calendar para poder obtener el año actual usando la función

`Calendar.getInstance().get(Calendar.YEAR)`

Realizar otro programita que permita crear un auto de vuestra familia, cargar los datos, y ver si hace falta cambiarle el aceite o no.

Ejercicio 4

Crear un objeto llamado **Rectangulo**. Tendrá los campos base, altura. Crear sus métodos set y get, además de un método constructor. Y por último crear métodos **getPerimetro, getArea**.

Crear otro programita donde se le pida al usuario los datos de base y altura para 5 rectángulos. Utilizando variables del tipo de datos Rectangulo que acaba de crear, determinar sus perímetros y áreas.

Ejercicio 6

Suponer que tengo un local de venta de zapatillas. Donde trabajo con zapatillas fabricadas en Taiwán y en Canadá. Llenar, pidiéndole al usuario, un array llamado modelosVendo, con 6 modelos de zapatillas diferentes, cuya industria sea alguna de esas dos mencionadas. Determinar si tengo más modelos de industria Taiwanesa o Canadiense.

Ejercicio 10

Crear un tipo de datos llamado **Racional**. Un objeto del tipo Racional, tendrá dos campos de números enteros, llamados **numerador** y **denominador**. **Este tipo de datos será para representar fracciones.*

Crearle dos constructores, uno que sin tomar argumentos, inicializará el numerador en 0 y el denominador en 1.

El otro constructor, tomará como argumentos dos números enteros, *p* y *q*, que le darán forma a la fracción *p/q*.

Crear entre los métodos extras:

- Un método llamado **show** que imprimirá en pantalla el racional en una forma "*p/q*" donde *p* es el numerador y *q* es el denominador.
- Un método llamado **invert** que invertirá la fracción, intercambiando los valores entre el numerador y el denominador.
- Un método **toDouble** que devolverá el número decimal de dividir el numerador por el denominador.
- Un método llamado **sum** que tomará como argumento otro racional, y devolverá la fracción resultado de la suma de las dos fracciones. **Tener en cuenta que hay dos casos, uno cuando ambas fracciones tienen mismo denominador, y otro caso cuando tienen diferente denominador.*
- Un método llamado **resta**, que es equivalente al caso anterior, pero la segunda fracción será restada a la primera.
- Un método llamado **mult** que tomará otro racional, y devolverá la fracción resultado de multiplicar ambos racionales.
- Un método llamado **div** que tomará otro racional y devolverá la fracción resultado de dividir ambos racionales.

Material elaborado por Carina Povarchik.