



## Trabajo Práctico N° 2: Trabajando con Objetos

Para la realización de los ejercicios de esta práctica vas a necesitar tener las siguientes clases compiladas. Asegurate de solicitárselas a los docentes para llevar a cabo la misma:

Persona.class

Ascensor.class

Cuenta.class

En la sección “Anexo” vas a encontrar una descripción de estas clases, la cual te ayudará a entender cómo resolver estos ejercicios.

1. Utilizando la clase Persona, desarrolla un programa a través de una clase llamada EjemploDeVida, teniendo en cuenta lo siguiente:
  - a. Definí un objeto persona a través de una variable llamada sofia.
  - b. Instancia dicho objeto. Para eso debes saber que el constructor de la clase Persona espera los siguientes parámetros:
    - i. Nombre (Asigne el valor “Sofía”).
    - ii. Genero (Asigne el valor “Mujer”).
    - iii. Dni (Asigne el valor 38432153).
    - iv. Peso (Asigne el valor 2.8).
    - v. Altura (Asigne el valor 0.50)
  - c. Invocá el método toString del objeto Sofia, e imprimí por pantalla su contenido. ¿Qué se observa?
  - d. Envíale los siguientes mensajes al objeto sofia:
    - i. alimentar. Pasale el valor 1.0 por parámetro (Es la cantidad de kilos de comida).
    - ii. crecer. Pasale el valor 0.5 por parámetro (Es la cantidad de metros que crece en altura).
    - iii. cumplirAnos.
  - e. Repetí el punto c. ¿Qué diferencias encontrás con respecto a la primera ejecución? ¿Tiene sentido?
2. Tomando como base lo realizado en el ejercicio 1, te invitamos a crear una clase llamada OtrasVidasNoTanEjemplares:
  - a. mauricio: Se espera visualizar el siguiente resultado por pantalla: “Mauricio es un hombre de 50 años de edad, pesa 70 kilos, mide 1.70 y su DNI es 33999776”. ¿Qué instrucciones ejecutarías para lograr el objetivo?
  - b. marcos. Se espera visualizar el siguiente resultado por pantalla: “Marcos es un hombre de 30 años de edad, pesa 80 kilos, mide 1.75 y su DNI es 2315321”. ¿Qué instrucciones ejecutarías para lograr el objetivo?
  - c. eugenia. Se espera visualizar el siguiente resultado por pantalla: “Eugenia es una mujer de 40 años de edad, pesa 60 kilos, mide 1.69 cm y su DNI es 2154215. ¿Qué instrucciones ejecutarías para lograr el objetivo?



- d. Elisa. Se espera visualizar el siguiente resultado por pantalla: “Elisa es una mujer de 60 años de edad, pesa 150 kilos, mide 1.60 cm y su DNI es 63215478. ¿Qué instrucciones ejecutarías para lograr el objetivo?
    - e. Para el primer objeto creado (mauricio) invocá el método morir, luego mostrá por pantalla nuevamente el método toString(). ¿Qué se observa?
3. Utilizando la clase Ascensor, desarrolla un programa a través de una clase llamada “ManejoAscensor” que realice lo siguiente:
  - a. Crea un objeto de tipo ascensor a partir de una variable llamada consorcioCallePeron. El constructor de dicho objeto espera que le informes la capacidad máxima y el peso máximo que soporta. Envíale los parámetros 4 y 150.0.
  - b. Imprimí por pantalla el piso actual en donde se encuentra el ascensor, invocando el método getPisoActual(). ¿En qué piso se encuentra?
  - c. Envíale 3 mensajes para que el ascensor suba. ¿En qué piso se encuentra? ¿Porqué?
  - d. ¿Qué debería hacerse antes para lograr llegar al piso 3? Hacerlo.
  - e. Utilizando la clase Persona, crea 3 objetos que tengan los siguientes pesos: 60, 80, 90.
  - f. Envíale el mensaje adecuado al objeto consorcioCallePeron para subir a cada uno de estos objetos (Recordá que vas a necesitar abrir la puerta del ascensor para que la gente suba). ¿Qué se observa?
  - g. ¿Podrías subir de piso en el estado actual del objeto ascensor? ¿Porqué?
  - h. ¿Qué debes hacer para poder subir de piso? Hacerlo.
4. Utilizando la clase Cuenta, desarrolla un programa a través de una clase llamada “CajeroAutomatico” que realice lo siguiente:
  - a. Crea un objeto de tipo Cuenta (podrías usar una variable llamada miCuenta). Uno de los constructores de esta clase sólo necesita el parámetro titular (Envíale tu nombre).
  - b. Invoca el método toString e imprimí su contenido por pantalla. ¿Qué se observa?
  - c. Deposita 100 pesos. Luego invoca el método toString nuevamente. ¿Qué cambio? ¿Podrías decir que está funcionando bien la función depositar?
  - d. Retira 50 pesos. Luego invoca el método toString nuevamente. ¿Qué cambio? ¿Podrías decir que está funcionando bien la función extraer?
  - e. ¿Qué otras operaciones realizarías para verificar el correcto funcionamiento de la clase Cuenta? Realízalas.
5. Como sabes, para poder utilizar una clase en Java, es necesario instanciar un objeto de dicha clase. Sin embargo, existen casos en los que esto no se cumple. Tomando como base los ejemplos vistos hasta el momento, ¿Te animás a identificar en qué caso no estamos instanciando la clase para poder utilizarla? ¿Cómo explicarías esta situación?
6. Si bien las clases base de Java las vamos a ver más adelante, te invitamos a desarrollar, compilar y ejecutar un programa que, en una variable llamada radio, almacene el radio de un círculo, luego calcule el perímetro del mismo y lo muestre por pantalla. Como ayuda te



recomendamos utilizar la variable Math.PI para realizar la operación. Al finalizar, te pedimos describas qué características y particularidades observas de esta variable PI utilizada.

7. Dada la clase Televisor:

```
public class Televisor {  
  
    private int volumen;  
    private int canal;  
  
    public Televisor() {  
        this.volumen = 10;  
        this.canal = 2;  
    }  
  
    public Televisor(int volumen, int canal) {  
        this.volumen = volumen;  
        this.canal = canal;  
    }  
  
    public int getVolumen() {  
        return this.volumen;  
    }  
  
    public void ajustarVolumen(String modo) {  
        if(modo.equals("subir")) {  
            volumen++;  
        }  
        else {  
            volumen--;  
        }  
    }  
  
    public void cambiarCanal(String modo) {  
        if(modo.equals("subir")) {  
            canal++;  
        }  
        else {  
            canal--;  
        }  
    }  
  
    public int getCanal() {  
        return this.canal;  
    }  
}
```

- ¿Dónde se observa el encapsulamiento?
  - En el constructor de la clase, identificar variables y atributos.
- ¿Qué mensajes debería poder enviarle un objeto de la clase Docente a un objeto de la clase Alumno? ¿Y viceversa? Ejemplificar cómo se enviarían dichos mensajes.
  - Desarrolla la clase Coche (fíjate el diseño que debe cumplir la misma en la sección Anexo).
  - Desarrolla la clase Calculadora.
  - Desarrolla la clase Usuario.
  - Desarrolla una clase en Java que realice la abstracción de un sistema planetario, debiendo tener en cuenta para cada cuerpo del sistema aspectos como: masa, densidad, diámetro, distancia al sol (supongamos órbitas circulares), un número de identificador único y un nombre textual. Incluyamos métodos que calculen la atracción gravitatoria entre dos cuerpos cualesquiera del sistema. (Recordar que  $F = G \frac{Mn}{r^2}$ ; donde F: Fuerza de atracción  $[N] = [\frac{m^2}{kg^2}]$ ; G: Constante de gravitación universal  $= 6.67 \times 10^{-11} [N \frac{m^2}{kg^2}]$ ; M y m: Masas  $[kg]$ ; r: Distancia entre los centros  $[m]$  de los cuerpos).
  - Sabiendo que un móvil se mueve en un movimiento MRU, cuya velocidad está expresada en m/seg. y durante número de horas. Para ello generar una clase Automóvil con los atributos velocidad y horas, y crear los métodos para conocer la distancia que recorrió el



mismo, informar la velocidad en km/h y la distancia recorrida. Luego, desarrollar una clase de prueba llamada `MovimientoRectilineoUniforme`, que valide lo siguiente:

- a. `miniCopper` que va a 120 km/h durante 20 minutos
  - b. `ferrari` que va a 200 km/h durante 40 minutos
  - c. `fitito` que va a 40 km/h durante 2, 25 horas
14. Crear la clase `DiscoRigido` que contenga un atributo tamaño y un método que informe cuantos cuántos dvd (4.5GB) equivale dicha capacidad y otro método que informe a cuántos blue ray (15GB) equivalen. Crear la clase `PruebaDiscoRigido` que verifique el correcto funcionamiento para los siguientes ejemplos:
  - a. `seagate` de 10 TB
  - b. `hitachi` de 20 TB
  - c. `westernDigital` de 32 TB
  - d. `toshiba` de 48 TB
15. ¿A qué método se invoca cuando utilizamos la palabra reservada `new`? ¿Qué se obtiene como resultado?  
`alfredoAlcon = new Actor();`