



## Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

# Laboratorios de computación salas A y B

*Profesor:*

TISTA GARCÍA EDGAR

*Asignatura:*

PROGRAMACIÓN ORIENTADA A OBJETOS

*Grupo:*

3

*No de Práctica(s):*

PRACTICA #7-8. HERENCIA Y POLIMORFISMO

*Integrante(s):*

CARRILLO CERVANTES IVETTE ALEJANDRA

*No. de Equipo de  
cómputo empleado:*

TRABAJO EN CASA

*No. de Lista o Brigada:*

08

*Semestre:*

2022 - 1

*Fecha de entrega:*

10 NOVIEMBRE 2021

*Observaciones:*

**CALIFICACIÓN:** \_\_\_\_\_

## PRÁCTICA #7-8: Herencia y Polimorfismo

### Objetivo General:

- (7) Implementar los conceptos de herencia en un lenguaje de programación orientado a objetos.
- (8) Implementar el concepto de polimorfismo en un lenguaje de programación orientado a objetos.

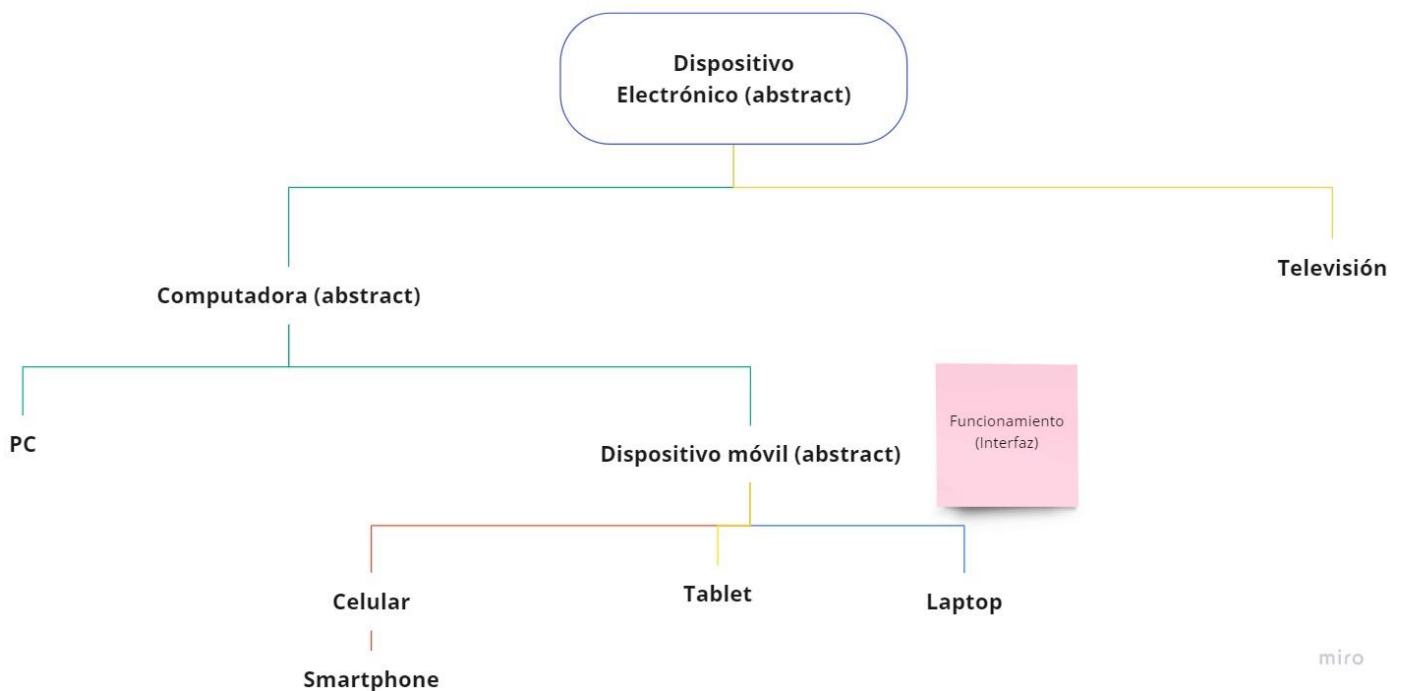
### Objetivo de clase:

Realizar aplicaciones sencillas donde el estudiante pueda visualizar de manera clara los conceptos relacionados con herencia como clases abstractas e interfaces así como la importancia del polimorfismo y la sobrescritura.

## ACTIVIDADES PARA EL DESARROLLO DE LA PRÁCTICA

### Ejercicio 1. Construyendo una jerarquía de Herencia.

Este programa tiene como función principal, administrar una tienda de tecnología, en la cual cuenta con los siguientes dispositivos, ordenados por la siguiente jerarquía:



### *Jerarquía clases para dispositivos electrónicos*

Para la creación de cada dispositivo mencionado anteriormente, se creó una clase padre "**Dispositivo electrónico**", la cual fue heredando cada uno de sus atributos y métodos correspondientes a las demás clases y posterior a ello estas clases heredaban a otras, así sucesivamente hasta tener todas las clases que se necesitaban.

Las clases que más se ocuparon, ya fuera de una forma directa o no, fueron las siguientes:

→ **Dispositivo electrónico.**

Una clase abstracta, la cual será la clase padre y tendrá como atributos la marca del producto y el precio (ya que todos los dispositivos que se venden en la tienda deben de tener esas características, ya sea televisor o computador).

→ **Computadora**

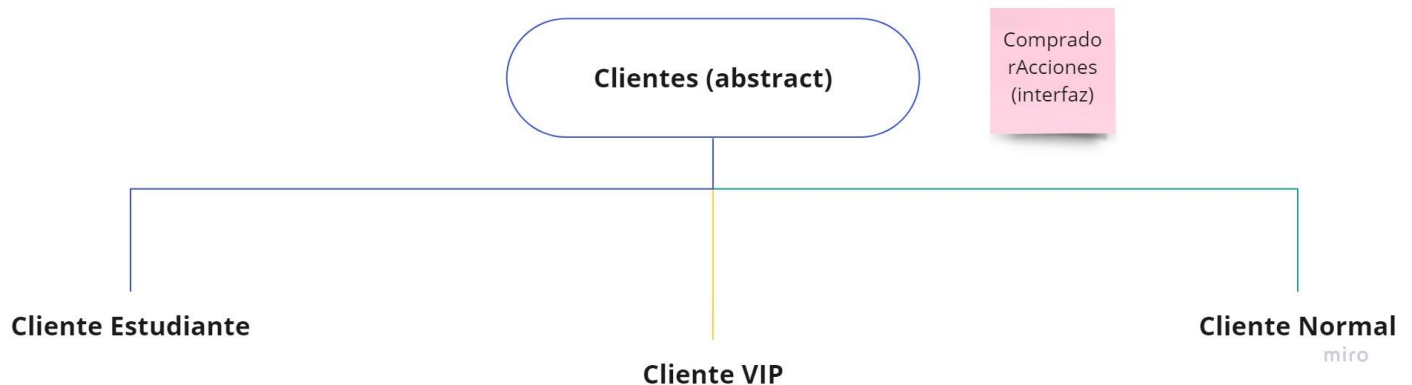
Esta clase hereda de Dispositivo electrónico y también es una clase abstracta, y de esta heredarán otras, las cuales serán los dispositivos que tienen los atributos del método descrito anteriormente y atributos nuevos que son memoria RAM y modelo.

→ **Dispositivo Móvil**

Esta clase hereda de Computadora, esta ya no es una clase abstracta, tiene como atributos, además de los atributos mencionados en la clase anterior, tiene duración de batería y Memoria GB. De esta se heredan, casi la mayoría de los dispositivos.

Para los demás dispositivos se crearon diferentes clases con diferentes atributos cada clase, con el objetivo de que el dispositivo que se vaya a comprar tuviera tantas funciones como sea necesario para que estuviera “completo” por así decirlo.

Mientras que, para que nuestra tienda tuviera clientes, se creó otra jerarquía con la finalidad de que todos los clientes tuvieran los mismos atributos y no se tuviera que tener la necesidad de estarlos inicializando en cada una de estas clases.



*Jerarquía de clases para los clientes que compran en la tienda de tecnología*

Los atributos que se mencionan anteriormente es el Nombre, edad, tipo de cliente y numero de cliente de este.

Para implementar ambas jerarquías se creó una clase principal, la cual contiene el método main, en esta se encuentra un menú con las siguientes opciones, con la finalidad de probar la herencia:

- ➔ Acceder a perfil
- ➔ Crear perfil nuevo
- ➔ Ver clientes
- ➔ Salir

Para ir almacenando los clientes, así como los dispositivos que han comprado cada uno de ellos, se creó una tabla Hash Map, la cual tiene como llave, el cliente con todos sus atributos y como valor, una lista de tipo Dispositivo electrónico, en la cual se guardarán los dispositivos que el cliente ha comprado.

\*NOTA: Al principio del programa se crearon algunos clientes con algunos artículos creados\*

La salida de cada opción del programa es la siguiente:

```
run:

-----

                        TIENDA DE TECNOLOGIA

1. Acceder a perfil
2. Crear perfil nuevo
3. Ver clientes
4. Salir
-> Elige una opción: 1

-----

                        ACCEDER A PERFIL DE CLIENTE EXISTENTE

Ingresa tu nombre con el que estas registrado para acceder: Draco
1. Comprar dispositivo
2. Salir
-> ¿Qué deseas hacer? 1
' _ ' _ ' _ ' _ ' _ ' _ ' _ ' _ ' _ '

                        COMPRAR DISPOSITIVO

1. Laptop
2. Tablet
3. Smartphone
4. PC
5. Television
-> ¿Qué deseas comprar? 3

Hola, soy un cliente VIP!!
Quisiera comprar un dispositivo electrónico
Listo. Aquí tiene la cantidad

-----

1. Comprar dispositivo
2. Salir
-> ¿Qué deseas hacer? 2
SALIR
```

---

TIENDA DE TECNOLOGIA

1. Acceder a perfil
  2. Crear perfil nuevo
  3. Ver clientes
  4. Salir
- > Elige una opción: 2

---

CREAR UN PERFIL NUEVO

1. Comprador
  2. Cliente VIP
  3. Estudiante
- > Elige el tipo de cliente: 3
- Ingrese sus datos:
- Nombre: Ron
- Edad: 17

---

TIENDA DE TECNOLOGIA

1. Acceder a perfil
  2. Crear perfil nuevo
  3. Ver clientes
  4. Salir
- > Elige una opción: 3

---

IMPRIMIR CLIENTES

-----

Cliente:

Nombre: Harry

Edad: 16

Dispositivos comprados:

Marca: APPLE

Precio: 35000.0

-----

Cliente:

Nombre: Ron

Edad: 17

Dispositivos comprados:

```

-----
Cliente:
Nombre: Hermionie
Edad: 16

Dispositivos comprados:
Marca: Huawei
Precio: 16000.0

-----
Cliente:
Nombre: Draco
Edad: 17

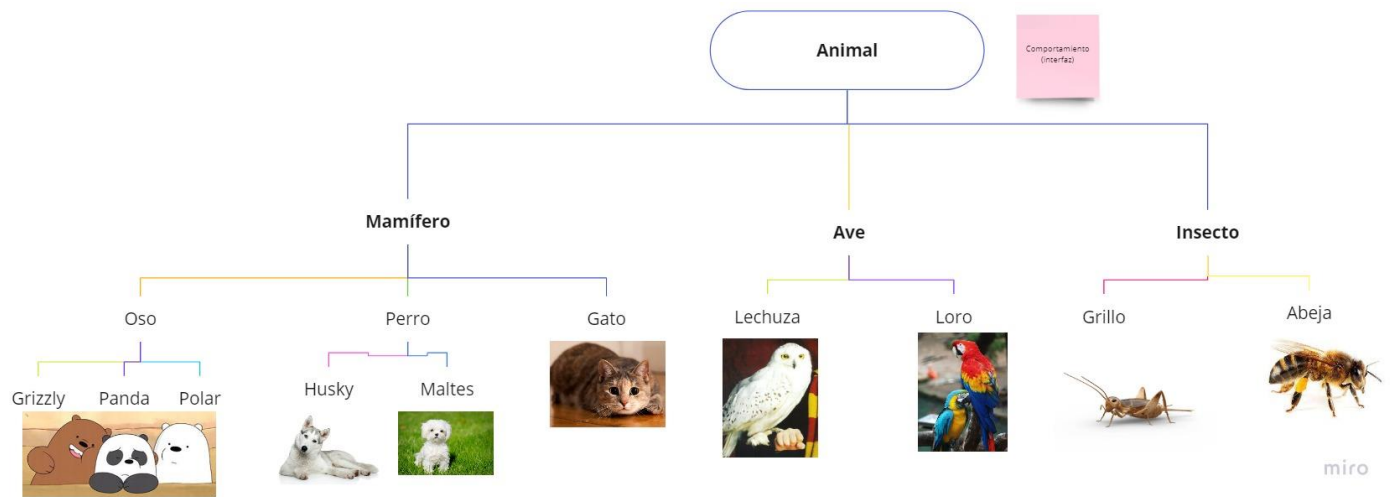
Dispositivos comprados:
Marca: Sony
Precio: 6000.0
Marca: Huawei
Precio: 7500.0

```

*Practica7-8[Carrillolvette] -> Package ejercicio1*

## Ejercicio 2. Polimorfismo – Creando una colección de objetos

Este programa tiene como función principal almacenar una colección de “Animales”, para ello se estableció la siguiente jerarquía:



*Jerarquía de Animales*

La idea principal del programa es realizar un menú el cual tiene como objetivo ir modificando la colección de animales, este menú cuenta con las siguientes opciones:

Antes de iniciar el programa, se creó una lista de "Animal", donde se irán almacenando cada uno de los animales creados según corresponda las clases de animales que se vieron en la jerarquía. También se creó un animal de cada tipo y se almaceno en esta para que no estuviera vacía al momento de que probará este menú el usuario.

➔ **Agregar Animal**

Para esta opción se le pregunta al usuario que animal quiere crear y con base a su respuesta, se instancia el objeto correspondiente a este animal y se agrega en la lista mencionada anteriormente.

➔ **Mostrar elemento de la colección**

Se le pregunta al usuario que indice de la lista quiere ver, con base a su respuesta se buscaba en la lista el indice y si este estaba ocupado (almacenaba un animal) se mostraba en pantalla su comportamiento. Este comportamiento se implementó por medio de los siguientes métodos (los cuales cuenta la clase padre) y se fue sobrescribiendo en cada clase según el animal correspondiente.

```
public void sonido() {  
    System.out.println("*Existen muchos sonidos de animales*");  
}  
  
public void pensando() {  
    System.out.println(" * * * * ");  
}  
  
public void trasladarse() {  
    System.out.println("*Depende el modo de trasladarse de cada animal*");  
}
```

➔ **Mostrar colección completa**

Recorría la lista mediante un ciclo de repetición for e imprimía cada uno de los atributos de cada animal.

➔ **Salir**

La salida del programa es la siguiente:

```
run:  
  
-----  
                                POLIMORFISMO  
  
1. Agregar animal  
2. Mostrar elemento de la colección  
3. Mostrar colección completa  
4. SALIR  
-> Elige una opción: 1  
  
-----  
                                AGREGAR ANIMAL
```

1. Oso 2. Perro 3. Gato 4. Lechuza  
5. Loro 6. Grillo 7. Abeja  
-> ¿Qué animal deseas agregar? 1

. . . . .

Ingresa los datos:

Nombre: Burbuja

Edad: 2

Habitad: Ártico

1. Grizzly 2. Panda 3. Polar  
-> ¿Qué tipo de oso quieres agregar?3

\*\*Animal agregado (:\*\*

---

## POLIMORFISMO

1. Agregar animal  
2. Mostrar elemento de la colección  
3. Mostrar colección completa  
4. SALIR  
-> Elige una opción: 2

— — — — —

## MOSTRAR ELEMENTO EN LA COLECCION

¿Qué índice de la colección quiere ver? 6

Animal en el índice 6:

Uh Uh Uh!!

\* \* Debo ir a Hogwars \* \*

\*Volando... \*



---

POLIMORFISMO

1. Agregar animal
  2. Mostrar elemento de la colección
  3. Mostrar colección completa
  4. SALIR
- > Elige una opción: 3

-----

MOSTRAR COLECCION DE ANIMALES COMPLETA

· \_ · \_ · \_ · \_ · \_ · \_ ·

Animal 0:  
Nombre: Pardo  
Edad: 3 años  
Habitad: Bosque  
Comportamiento:  
Grrrr!!!  
\* \* Quiero comer miel \* \*  
\*Caminando... Corriendo... Otra vez caminando...\*

· \_ · \_ · \_ · \_ · \_ · \_ ·

Animal 1:  
Nombre: Panda  
Edad: 5 años  
Habitad: Bosque de bambú  
Comportamiento:  
Grrrr!!!  
\* \* Quiero comer bambú \* \*  
\*Caminando... Corriendo... Otra vez caminando... Trepado en un bambú...\*

· \_ · \_ · \_ · \_ · \_ · \_ ·

Animal 2:  
Nombre: Polar  
Edad: 4 años  
Habitad: Ártico  
Comportamiento:  
Grrrr!!!  
\* \* Quiero comer pescado \* \*  
\*Caminando... Corriendo... Nadando... Otra vez caminando...\*

· \_ · \_ · \_ · \_ · \_ · \_ ·

Animal 3:  
Nombre: Kyra  
Edad: 1 años  
Habitad: Casa  
Comportamiento:  
Guau Guau!!!  
\* \* Quiero pollito \* \*  
\*Caminando... Corriendo... Caminando...\*

· \_ · \_ · \_ · \_ · \_ · \_ ·

Animal 4:  
Nombre: Kailani  
Edad: 1 años  
Habitad: Casa  
Comportamiento:  
Guau Guau!!!  
\* \* Quiero pollito \* \*  
\*Caminando... Corriendo... Caminando...\*

· \_ · \_ · \_ · \_ · \_ · \_ ·

Animal 5:  
Nombre: Michi  
Edad: 2 años  
Habitad: Casa  
Comportamiento:  
Miau Miau!!!  
\* \* Also: Esta dormido \* \*  
\*Corriendo... Tirando todo a su paso... Caminando... \*

· \_ · \_ · \_ · \_ · \_ · \_ ·

Animal 6:  
Nombre: Hedwig  
Edad: 6 años  
Habitad: Bosque  
Comportamiento:  
Uh Uh Uh!!  
\* \* Debo ir a Hogwars \* \*  
\*Volando... \*

· \_ · \_ · \_ · \_ · \_ · \_ ·

Animal 7:  
 Nombre: Lola  
 Edad: 10 años  
 Habitación: Selva  
 Comportamiento:  
 Hola HOLAAA, soy un LOROO!!  
 \* \* Quiero comer, debería de decirlo... \* \*  
 \*Volando... \*

· \_ · \_ · \_ · \_ · \_ · \_ ·

Animal 8:  
 Nombre: Pepe  
 Edad: 0 años  
 Habitación: Cualquier lugar que tenga Humedad  
 Comportamiento:  
 Cri cri cri...  
 \* \* No dejaré dormir hoy... \* \*  
 \*Caminando... Saltando...\*

· \_ · \_ · \_ · \_ · \_ · \_ ·

Animal 9:  
 Nombre: Queen  
 Edad: 0 años  
 Habitación: Flores  
 Comportamiento:  
 ZzzzzZZzzzzZzZ...  
 \* \* Que linda flor... \* \*  
 \*Volando... \*

· \_ · \_ · \_ · \_ · \_ · \_ ·

Animal 10:  
 Nombre: Burbuja  
 Edad: 2 años  
 Habitación: Ortico  
 Comportamiento:  
 Grrrr!!!  
 \* \* Quiero comer pescado \* \*  
 \*Caminando... Corriendo... Nadando... Otra vez caminando...\*

· \_ · \_ · \_ · \_ · \_ · \_ ·

*Practica7-8[Carrillolvette] -> Package ejercicio2*

## **Conclusiones**

Considero que se cumplieron los objetivos de esta práctica, ya que se aplicaron los conceptos de herencia en JAVA (un lenguaje de programación orientado a objetos) y se realizaron dos ejercicios relacionado con clases abstractas, interfaces, así como herencia y polimorfismo.

Personalmente, me costó un poco de trabajo comprender la diferencia de una clase abstracta y una interfaz, pero al ir desarrollando esta práctica, comprendí mejor el tema; también no entendía muy bien el concepto de polimorfismo, tuve que investigar un poco acerca de esto.

Considero que los ejercicios de esta práctica estuvieron muy bien, y aun que me faltaron algunas cosas como fue la implementación de la interfaz en el ejercicio 1 (si la hice pero se me olvido implementarla), considero que aprendí los conceptos de una manera adecuada, ya que en el ejercicio 2 tuve interfaz que si utilice. Con forme practique considero que haré los programas de una forma mejor.