

Herencia y Colaboración, Fundamentos de Programación

Herencia y Colaboración, Fundamentos de Programación	1
¿Qué aprenderás?	2
Introducción	2
Colaboración y Herencia	3
Colaboración	3
Herencia	4
Ejercicio guiado: Identificar las clases y sus relaciones	6



¡Comencemos!

¿Qué aprenderás?

- Comprender conceptos de Herencia y Colaboración para realizar diagramas de clases de alto nivel.
- Aplicar conceptos de Herencia y Colaboración para realizar diagramas de clases de alto nivel.

Introducción

Al estudiar “Herencia y Colaboración” aprenderemos nuevos conceptos de programación orientados a objetos, sus relaciones y el uso práctico en el mundo moderno. Estos conceptos son de uso cotidiano en nuestras vidas, además de pilares fundamentales de JAVA porque son la base para entender el paradigma.

A partir de su aplicación en la vida cotidiana lograremos comprender la herencia y la colaboración a cabalidad, pero para lograr este objetivo necesitaremos que la primera unidad de UML esté aprendida ya que es la base de los conocimientos globales de Java para codificar y reconocer los distintos conceptos.

¡Vamos con todo!



Colaboración y Herencia

Colaboración

Anteriormente hablamos sobre propiedades y acción u operación de un objeto. La colaboración nos permite unir uno o más objetos distintos entre sí para crear uno nuevo, esto nace por los conceptos técnicos como, por ejemplo, “requiere de”, “necesita a” en las acciones de los mismos objetos.

Por ejemplo, describiremos una relación de colaboración de dos objetos distintos:

Se necesita crear un nuevo objeto, este objeto será llamado **AutoConRuedas**. Para crear el objeto **AutoConRuedas** se necesitan 4 objetos de tipo **Rueda**. El objeto **Rueda** ahora será una propiedad o atributo del objeto **AutoConRuedas**.

Esto es un claro ejemplo de colaboración de objetos distintos, aquí el término a utilizar es “el objeto **AutoConRuedas** requiere de cuatro objetos de tipo **Rueda** para ser un auto con ruedas”.

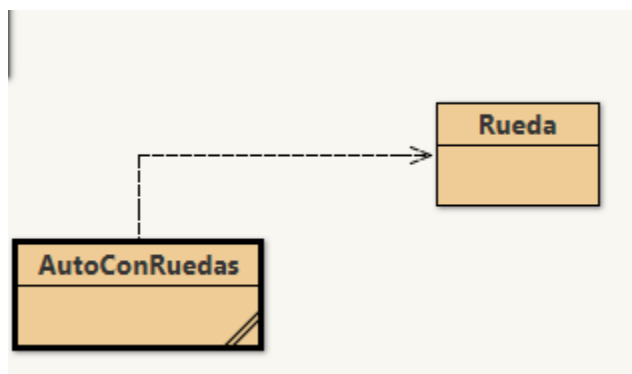


Imagen 1. Dos Clases Java colaborando entre ellas.
Fuente: Desafío Latam.

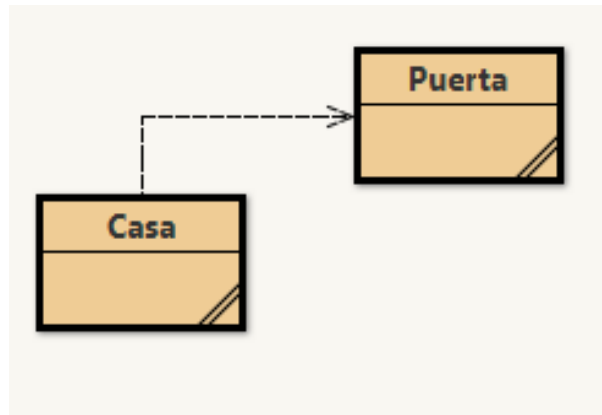


Imagen 2. Una Puerta colabora con el Objeto Casa.
Fuente: Desafío Latam.

Herencia

La herencia en Java se define como objetos con jerarquía relacionadas entre ellas, es distinta a la colaboración porque en la colaboración no existe jerarquía.

Java utiliza este pilar para seguir con el concepto de no repetir código, propiedades u operaciones, esto es para utilizar lo menos posible las líneas de código y hacer sus programas más livianos y rápidos.

Con la herencia esto se viene a reafirmar, ya que nace el concepto de superclase, donde todas las propiedades y operaciones comunes entre sí, siempre que sean públicas, se heredarán a las clases hijas o de jerarquía menor.

Ejemplo:

Tenemos el objeto **Abogado** con sus atributos nombre, rut y título, también tenemos el objeto llamado **Juez** con sus atributos nombre, rut y dirección.

Estos objetos tienen atributos similares y relación entre sí, y para no crear dos objetos donde se repiten las mismas operaciones y atributos, se crea una superclase donde se identifica la relación directa entre **Juez** y **Abogado**, esta relación es que ambos son personas.

Con esta información podemos crear el objeto Persona con los atributos rut y nombre, y bajo la definición de lo que es herencia, se dirá que la clase Persona le heredará todos sus atributos y operaciones públicas a las clases **Juez** y **Abogado**.

La clase **Juez** solo tendrá el atributo dirección y **Abogado** solo el atributo título, sin embargo, Java interpretará que la clase **Persona** y **Abogado** es solo una, así como **Persona** y **Juez**, ya que al heredar y tener jerarquía Java lo interpreta como una sola clase, para la **Java Virtual Machine (JVM)** solo existen dos clases **Juez** y **Abogados** con atributos y operaciones de la clase **Persona**.

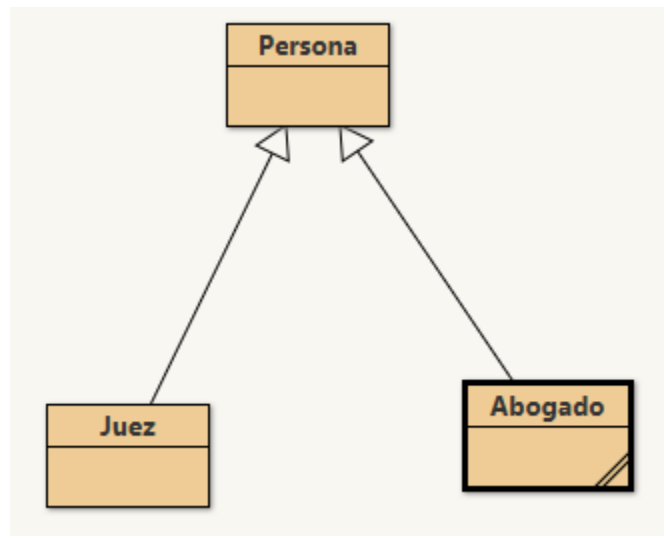


Imagen 3. Representación de Herencia en Java de las clases Persona-Juez-Abogado.
Fuente: Desafío Latam.

Ejercicio guiado: Identificar las clases y sus relaciones

En un zoológico hay un zorro, un león y una ciudad.

El zorro tiene:

- Edad.
- Tipo de zorro.
- Origen.

El león tiene:

- Edad.
- Origen.
- Sexo.

Identificar y describir las clases y sus relaciones.

Solución Guiada

Se debe identificar lo siguiente:

1. Si hay relación entre un león y un zorro.
2. Dónde están ubicados el león y el zorro.
3. Características de cada objeto.
4. Identificar la relación entre zoológico, león y zorro.

Luego de identificar las clases y sus relaciones, quedaría de la siguiente manera:

Animal:

- **edad:** int.
- **origen:** String.

Zorro es un Animal:

- **tipoDeZorro:** String.

León es un Animal:

- **sexo:** String.

Zoológico:

- **zorro:** Zorro.
- **león:** León.
- **ciudad:** String.