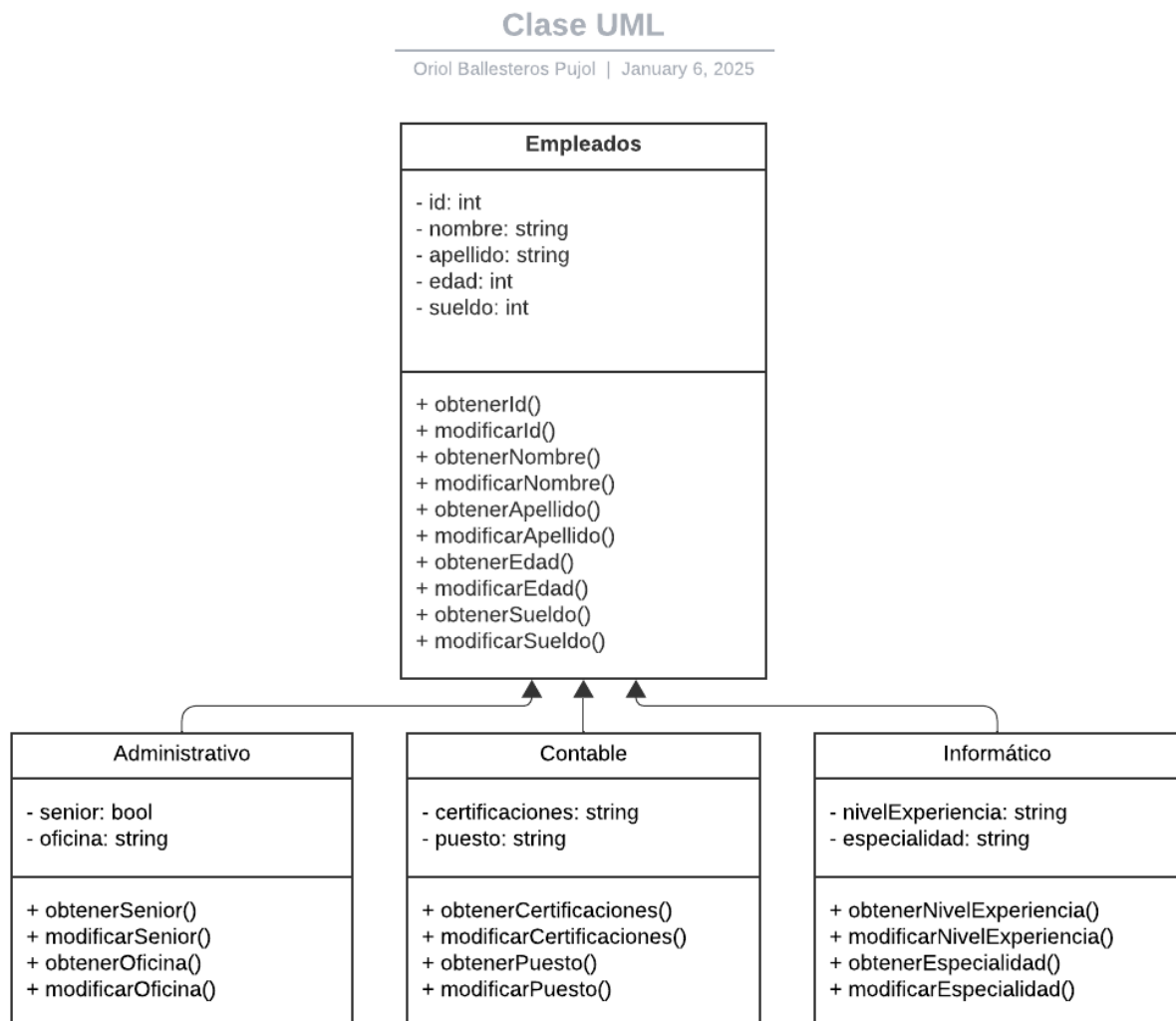


Teniendo en cuenta los puntos a seguir en el primer ejercicio, he decidido utilizar la herramienta Lucidchart (Lucid.app) para diseñar el diagrama UML y poder mostrar la relación entre la clase Empleado y sus subclases: Administrativo, Contable e Informático.



En este caso, la relación entre las clases está basada en el principio de herencia y es por eso que las subclases tienen una flecha hacia la clase Empleado. Esta clase Empleado es la superclase principal que define atributos y métodos comunes para todos los empleados. Contiene información básica como el Id de empleado, su Nombre y Apellido, la Edad y el Sueldo del empleado.

El resto de clases (Administrativo, Contable e Informático) son subclases que extienden a Empleado y heredan sus atributos y métodos. Estas tres subclases añaden atributos y métodos específicos:

1. **Administrativo:** Incluye atributos para determinar con un booleano si el empleado es senior o no, y para determinar en que oficina pertenece. Contiene también los métodos que permiten modificar y obtener (setters y getters) los atributos de cada objeto.
2. **Contable:** Incluye atributos para saber si el empleado tiene o no certificaciones, y para conocer el puesto de trabajo dentro del departamento contable. Contiene los setters y getters correspondientes. Contiene también los métodos que permiten modificar y obtener (setters y getters) los atributos de cada objeto.

3. Informático: Incluye atributos para saber el nivel de experiencia del empleado y su especialidad. Contiene los setters y getters correspondientes. Contiene también los métodos que permiten modificar y obtener (setters y getters) los atributos de cada objeto.

Comentar también que en cada clase he dejado un constructor con y sin parámetros para más tarde poder utilizar el archivo Main.java y poder usar cada clase.

Todos los atributos se han marcado como privados en las clases, aprovechando que la relación de tipo herencia entre clases permite reutilizar el código común en la superclase y extender funcionalidades específicas en cada subclase.