**Nombre Alumno/s:**

**Escalante Marcelo DNI: 43824662**

**Serial Geronimo DNI: 42603578**

**Nombre Profesor: Ringa Monica**

**Grupo Laboratorio: 1 TP: 2 Fecha de entrega 25/08/2024**

1. **Introducción:**

El propósito del Trabajo Practico N° 2 es principalmente la interpretación de diagramas UML, para la creación de clases con sus constructores y métodos, incluye además, conceptos referidos a la visibilidad de los elementos y las buenas practicas de documentación.

1. **Objetivos**:

* Interpretar diagramas de clases representados en UML
* Aplicación de conceptos teóricos de POO.
* Aprender a definir clases, con sus constructores y métodos
* Interpretar y definir la visibilidad de los elementos (sus interfaces)
* Definir métodos de acceso a los atributos (get/set)
* Aprender a instanciar objetos, y que estos interactúen mediante mensajes.
* Adquirir buenas practicas de desarrollo, como lo es la documentación (Javadoc)
* Ejercitar el idioma inglés.

1. **Desarrollo:**

El objetivo de interpretar diagramas de clases se practica y se logra mediante cada uno de los puntos del practico realizado.

De la misma manera, se aplican los conceptos de teóricos de POO, como, por ejemplo:

* 1. **Abstracción**

En este practico se hizo hincapié principalmente en el concepto abstracción, debido a que en cada punto se requiere el desarrollo de una clase, con sus propiedades (atributos) y sus comportamientos (métodos)

Texto

Descripción generada automáticamente

* 1. **Atributos:**

Cuil, apellido, nombre, sueldoBasico y anioIngreso son atributos. Estos representan las características específicas del objeto Empleado

* 1. **Comportamientos:**

Los metodos setCuil, setApellido, setNombre, setSueldo y setAnio son métodos privados que se utilizan para establecer los valores de los atributos

* 1. **Encapsulación:**

La encapsulación se logra al declarar los atributos como privados y utilizar accessors para manipularlos. Esto protege los datos internos de la clase.

Con los métodos GET y SET se logra la **doble encapsulación.**

También se implementó la práctica de desarrollo de software conocida como documentación del código  
(Javadoc).

Aprendimos a instanciar objetos y permitir que interactúen mediante mensajes

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

Además, en uno de los puntos a desarrollar, estaba completamente escrito en INGLES, por lo que nos permitió ejercitar este idioma

1. **Conclusión:**

En resumen, se cumplieron todos los objetivos mencionados al comienzo del Trabajo Practico N°2 debido a que en cada punto se implementaron los conceptos desarrollados en la teoría de POO.

Nos encontramos además con funciones nuevas y propias de JAVA.

Se profundizo en la importancia de controlar la visibilidad de los elementos y en como aplicar buenas practicas para la mantenibilidad del código, la correcta implementación de los métodos y las diferentes lógicas para el desarrollo de cada punto.