**Informe de Trabajo Práctico 2: Aplicación de Conceptos del Paradigma de Objetos**

En el desarrollo del Trabajo Práctico 2 (TP2), nuestro grupo se centró en la aplicación de los conceptos fundamentales del paradigma de programación orientada a objetos (POO). A continuación, se presenta un análisis detallado de cómo estos conceptos se reflejan en los ejercicios realizados.

1. Abstracción

La abstracción es el proceso de simplificar la complejidad del mundo real al enfocarse en los aspectos esenciales de un objeto y ocultar los detalles no relevantes. En el TP2, la clase Persona ejemplifica la abstracción al modelar una persona con atributos esenciales como nombre, apellido, nroDni, y anioNacimiento. Esta representación permite al usuario interactuar con una versión simplificada de una persona, sin preocuparse por los detalles internos del cálculo de la edad o la concatenación del nombre completo.

**2. Encapsulamiento**

El **encapsulamiento** implica agrupar los datos (atributos) y los métodos que operan sobre esos datos dentro de una misma clase, restringiendo el acceso directo a algunos de los componentes del objeto. En la clase Empleado, los atributos nombre y sueldoBasico son privados, y su acceso se controla mediante métodos públicos (getters y setters). Este enfoque asegura que los datos sean manipulados de manera controlada, evitando modificaciones externas indeseadas.

**3. Ocultamiento de la Información**

La **ocultación de la información** es una extensión del encapsulamiento que se enfoca en ocultar los detalles internos de un objeto, exponiendo solo lo necesario. En la clase Punto, los atributos x e y están ocultos, y el usuario interactúa con el objeto a través de métodos como desplazar y coordenadas, sin conocer cómo se gestionan internamente las coordenadas.

**4. Visibilidad**

La **visibilidad** se refiere al alcance de acceso a los miembros de una clase, controlado por modificadores de acceso como private, protected, y public. En la clase Alumno, el atributo nombre es privado y solo se accede a través de métodos públicos, como getNombre y setNombre. Este control de acceso asegura que los datos se manejen de manera segura y coherente.

**5. Objeto y Clase**

Una **clase** es una plantilla para crear objetos, definiendo los atributos y métodos que estos objetos pueden tener. Un **objeto** es una instancia concreta de una clase. La clase Cliente define atributos como nombre y saldo, y un objeto de Cliente representa una instancia específica con valores concretos para estos atributos.

**6. Doble Encapsulamiento**

El **doble encapsulamiento** se refiere a encapsular tanto los datos como el comportamiento, asegurando que las operaciones sobre los datos se realicen a través de métodos públicos. En la clase Alumno, el atributo nota1 está encapsulado y solo puede ser modificado si cumple con una condición establecida en el método setNota1.

**7. Mensaje, Método, Protocolo, Firma, Comportamiento**

* **Mensaje**: Una solicitud para que un objeto realice una acción a través de un método.
* **Método**: Una función definida en una clase que realiza una acción específica.
* **Protocolo**: El conjunto de métodos que un objeto debe implementar para interactuar con otros objetos.
* **Firma**: La declaración de un método, incluyendo su nombre, tipo de retorno y parámetros.
* **Comportamiento**: La acción que un objeto realiza cuando recibe un mensaje.