**Trabajo Práctico de entrada en Calor – Objetos 2**

**Evaluación de protocolos de una clase.**

La opción 2 es la correcta ya que en esta opción se utiliza un punto y dos valores de referencia (alto y ancho) para asegurarnos de definir un rectángulo. En la opción 1 sólo se definen 4 puntos que pueden ser al azar y no forman un rectángulo.

**Delegación**

La opción 2 es la correcta ya que en esa implementación se realiza correctamente la delegación entre los objetos de la oficina (secretario y jefe). El jefe no tiene por qué conocer la forma en la que el secretario realiza la búsqueda del fichero solicitado. Cada uno debe recibir su respectivo mensaje y realizar la acción que le corresponde.

**Polimorfismo**

Opción 1: Los defectos de esta opción son: No se hace un uso correcto del Polimorfismo ya que cada objeto sabe a qué clase pertenece y puede responder y evaluar cada caso por sí mismo mediante mensajes, en lugar de hacer uso del IF. Es responsabilidad de cada objeto conocer su estado y responder a la solicitud mediante mensajes propios.

Opción 2: Los defectos de esta opción son: Ocurre lo mismo que en el caso de la opción 1 sólo que en este caso se le pide al objeto su tipo de Cuenta para poder compararlo y luego realizar una extracción si corresponde o no, cuando debería ser el mismo objeto quien responda a esta cuestión de si se puede realizar una extracción o no.

Opción 3: El defecto de esta opción es que se repite la implementación del método de Extracción. Tanto la cuenta corriente como la caja de ahorro tienen el mismo método para el mensaje de extraer:UnMonto. Cuando se lo puede definir una sola vez y que ambos tipos de cuenta utilicen el mismo mensaje.

Opción 4: En este caso es correcta la resolución ya que cumple con el propósito de realizar una extracción a una cuenta bancaria pero no sería correcto el uso del ifTrue para evaluar si se puede realizar la extracción o no. Los mismos objetos son los que deberían responder realizando la extracción o no según el estado en el que se encuentre la cuenta bancaria en cuestión.

**Actividad de Lectura #1:**

1. El acceso directo a las variables significa acceder a ellas de manera directa sin utilizar getters o setters de por medio.

Ejemplo: pepita salud.

En este caso, “salud” es el Colaborador Interno (Atributo) del Objeto Pepita.

1. El acceso indirecto a las variables significa acceder a ellas mediante métodos de setters y getters y no obtener dicha variable directamente del objeto. Este caso permite aportar mas legibilidad al código a comparación con el acceso directo y además simplicidad, ya que con los setters y los getters se puede manipular la variable antes de ser utilizada y hacer un análisis previo al momento de utilizarla.

Ejemplo: pepita salud

Setter: pepita aumentarSalud.

Getter: pepita getSalud.

1. Las ventajas de las estrategias del uso de setters y getters son: El correcto uso de la herencia y encapsulamiento de cada uno de los objetos. Esto permite asegurar que la información de un objeto está oculta del mundo exterior, es decir que es información Privada de cada uno de ellos. El encapsulamiento nos permite agrupar en una Clase las características (atributos) con un acceso privado y los comportamientos (métodos) con un acceso público, accediendo o modificando los miembros de una clase a través de sus getters y setters de una manera más práctica y ágil.

Sin embargo, la implementación de getters y setters puede resultar un poco tediosa tanta para su lectura, como para su escritura. Lleva mucho mas tiempo escribir código extra y es más código para leer. Aunque en la mayoría de los casos se agradece su uso ya que se puede encontrar fácilmente la parte del código que se quiere visualizar o modificar.

**Actividad de Lectura #2:**

El “Creation Parameter Method” es conveniente usarlo cuando necesitamos crear una instancia a partir de una clase pero necesitamos que contemple variables al momento de su creación. De esta manera, podemos redefinir el mensaje de creación (NEW) que las clases comprenden por defecto y en ese nuevo METHOD podemos definir las variables que va a tener esa instancia y que valor va a recibir cada una de ellas, dejando explícito, el tipo de cada una.

**Actividad de Lectura #3:**

La forma correcta de proporcionar acceso a las variables que referencian a una colección es mantener las variables privadas para el exterior pero definir métodos que sean fáciles de comprender y que sus nombres sean bien descriptos para que el público haga uso de los mismos y pueda manipular dichas variables sin tener acceso directo a ellas.

**Actividad de Lectura #4:**

Es necesario la creación de dos métodos para asignar un estado a una propiedad booleana ya que si se utiliza uno solo se expone la representación del estado de la clase a los clientes y además es difícil determinar quién está interrogando al objeto o quién le esta pidiendo realizar alguna acción. En cambio, si se utilizan dos métodos distintos para los dos estados hace que la comunicación del código resultante sea mucho mejor.