Modelos y Sistemas.

Diagrama de clases.

1. ¿Qué es una clase? Elegí dos opciones:

* Un conjunto de objetos organizados en categorías.
* Un modelo que define atributos y métodos que son comunes a los objetos de un mismo tipo.
* La representación grafica de un objeto.
* En programación se define como un tipo de dato.

1. ¿Qué es un objeto? Elegí dos opciones:

* Se considera una instancia de una clase.
* Un diagrama que representa una clase.
* En programación se considera como una variable compleja.
* Es un método definido en una clase.

1. Explicar de forma breve los elementos que componen el diagrama de clases.

 El diagrama de clases está conformado por:

* Clase: es una plantilla para armar un objeto. Estas son detectadas, en la mayoría de los casos, como sustantivos en singular. Están formadas por atributos y métodos, y por convención, la primera letra debe estar en mayúsculas.
* Atributo: son las características que posee una clase, y determinan el estado que luego tendrá un objeto. Por convención, la primera letra debe estar en minúscula.
* Método: es el comportamiento que tiene una clase. Generalmente son nombrados con verbos.
* Clases Abstractas: son clases que representan un concepto abstracto, de carácter muy general, Por ejemplo, una institución, que tiene los atributos dirección y superficie, pero no es posible determinar qué tipo de institución es.
* Interfaz: A diferencia de la clase, la interfaz define únicamente un comportamiento, es decir un conjunto de métodos que no poseen implementación.

1. Indicar que representa los siguientes elementos de una clase.

* -saldo: double. representa el conjunto de números enteros que puede tener una variable.
* -nombre: string. Representa una secuencia de caracteres guardados en una variable.
* +calcularSaldo(): void. Representa una variable vacía.

1. Explicar brevemente (con tus palabras) cual es la diferencia entre clases concretas, clases abstractas en interfaces.

Las clases abstractas no puede contener un objeto dentro de la misma, se utiliza como base para crear otras clases en relación de generalización. En cambio, las clases concretas son todo lo contrario a las abstractas, pueden contener objetos, métodos, etc. Y las interfaces, solo definen un comportamiento, no puede contener atributos.

1. Indicar que tipo de clase o interfaz se quiere representar con la siguiente información:

* Persona. CONCRETA.
* EmpleadoAdministrativo. INTERFAZ.
* Auto. ABSTRACTA.
* Controlable. ABSTRACTA.

1. ¿Qué es la Herencia entre clases? Elegí dos opciones:

* Un método de construcción de clases, donde cada clase se concatena con otra.
* Indica que la clase que hereda toma los atributos y métodos de la super clase.
* Se puede decir que la herencia es una especialización de la super clase.
* Se puede decir que la herencia es una generalización de la clase subclase.

1. Indicar el tipo de relación:

* Un perro es un animal. ASOCIACION.
* Una empresa esta compuesta por varios departamentos. DEPENDENCIA.
* Un banco tiene varios clientes. DEPENDENCIA.
* alumno es una persona. ASOCIACION.

1. Decir que tipo de relaciones son las siguientes expresiones (agregación o composición)

* Departamentos de una empresa. COMPOSICION.
* Clientes de un banco. AGREGACION.
* Diccionario Palabra. COMPOSICION.
* Frase Palabra. COMPOSICION.

1. Indicar los estereotipos de clase que representan el patrón de diseño MVC.

* Modelo: representa la capa de datos o lógica de negocio de la aplicación. En general, el modelo no está asociado con un estereotipo de clase específico, ya que su estructura y funcionalidad dependerán de los requerimientos y características de la aplicación en cuestión.
* Vista: representa la capa de presentación de la aplicación, es decir, la interfaz de usuario o cualquier otro mecanismo utilizado para mostrar información al usuario final. En este caso, es común utilizar estereotipos de clase asociados con la interfaz de usuario, como Ventana, Panel, Botón, etc.
* Controlador: representa la capa de control o coordinación entre el modelo y la vista. El controlador se encarga de procesar las solicitudes del usuario, actualizar el modelo y reflejar los cambios en la vista correspondiente. En general, se utilizan estereotipos de clase como Controlador, Gestor o Manejador para identificar las clases responsables de esta tarea.