

CONTENIDOS

- Implementación de POO
- Asociación entre clases, manejo de composición, agregación
- Niveles de acceso, abstracción y ocultamiento de la información
- Herencia y polimorfismo
- Métodos, atributos y propiedades

1) En el mismo repositorio del TP1 (<https://classroom.github.com/a/yIIzTnUy>) cree una rama que se llame TP2 y trabaje en ella para el presente trabajo práctico

2) Refactorización del Sistema para una Cadeteria

El cliente presentó como nuevo requisito que los pedidos puedan no estar asignados a algún cadete. Esto evidenció una falla en el diseño de clases del sistema, por lo que se decidió realizar una refactorización del mismo.

Para poder cumplir con dicho requisito se propuso las siguientes modificaciones:

- Quitar el ListadoPedidos de la clase Cadete
- Agregar una referencia a Cadete dentro de la clase Pedido
- Agregar ListadoPedidos en la clase Cadeteria que contenga todo los pedidos que se vayan generando.
- Agregar el método JornalACobrar en la clase Cadeteria que recibe como parámetro el id del cadete y devuelve el monto a cobrar para dicho cadete
- Agregar el método AsignarCadeteAPedido en la clase Cadeteria que recibe como parámetro el id del cadete y el id del Pedido

- i) Implemente las modificaciones sugeridas más todas aquellas que crea necesarias para cumplir con los requerimientos.
- ii) Modifique la interfaz de usuario para cumplir con los nuevos requerimientos.

3) Se desea migrar además la carga de datos inicial a un archivo Json, sin perder la posibilidad de seguir accediendo a los datos guardados en el archivo csv. Para ello se propone un nuevo diseño de acceso a datos basado en los principios de herencia y polimorfismo.

Este nuevo diseño consta de una clase base llamada AccesoADatos y dos clases derivadas llamadas AccesoCSV y AccesoJSON, donde se implementaran

Refactorizar la clase AccesoADatos (que implementó en el práctico anterior) para que ésta cumpla con lo solicitado.

Modifique la interfaz de usuario para que al inicio del sistema se pida que tipo de acceso usar (CSV o JSON). y en función de esto instanciar el objeto a datos adecuado.