



IIC2115 – Programación como Herramienta para la Ingeniería (II/2025)

Ejercicio Formativo 1 Capítulo 1

Aspectos generales

- **Objetivos:** familiarizarse los elementos básicos de programación orientada a objetos y estructuras de datos.
- **Entrega:** lunes 11 de agosto a las 17:30 hrs. en **repositorio privado y ticket de salida**.
- **Formato de entrega:** archivo `E1.ipynb` con los solicitado, ubicado en la carpeta **C1** del repositorio.

Descripción del problema (versión normal)

En este ejercicio deberá elegir un tema (por ejemplo, alimentos, transporte, vivienda, medicamentos u organizaciones) y diseñar un modelo en programación orientada a objetos que lo represente, decidiendo por usted mismo cómo dividirlo en clases y relaciones. El modelo debe incluir todos los elementos de POO necesarios para generar una representación clara y coherente del problema, junto con un diagrama de clases. Además, deberá demostrar su funcionamiento mediante ejemplos de uso que evidencien cómo interactúan sus componentes. Para finalizar, deberá crear múltiples objetos con nombres únicos, almacenarlos en una lista y en un diccionario, y reflexionar —con justificación en código— sobre en cuál estructura es más eficiente buscar estos objetos por nombre.

Descripción del problema (versión guiada)

Siga las instrucciones que se entregan a continuación, recordando siempre hacer *commit* regularmente en el repositorio privado, usando comentarios descriptivos:

1. Cree una carpeta de nombre **C1** dentro del repositorio privado y agregue un nuevo **Python Notebook** con nombre **E1.ipynb**.
2. Cree una celda de texto dentro del documento e ingrese el título de la actividad “Ejercicio Formativo 1 Capítulo 1” junto con su nombre.
3. A continuación, escoja **uno** de los siguientes temas, con los que creará clases con ciertas condiciones. Registre el tema elegido en una nueva celda de texto.

- Alimentos
- Transporte
- Vivienda
- Medicamentos
- Organizaciones

4. Su objetivo será crear un modelo de clases que al menos posea:
 - Una clase base que tenga al menos un método sin implementación.
 - Dos clases que hereden de otra
 - Una clase que participe como atributo en otra
 - Una implementación del metodo `__str__()`
 - Dos sobrecargas de métodos (*override* - polimorfismo)
 - Todas las clases deben contener un campo **nombre**, que permita identificar individualmente a cada objeto creado.

Recuerde comentar su código y definir en celdas de texto qué es cada clase. HINT: Si el tema fueran Juguetes, algunas clases que se podrían crear serían: JuguetesElectronicos, JuguetesDeMadera, Switch, Pelota, etc.

5. Genere un diagrama de clases del modelo desarrollado e inclúyalo como figura en una celda antes de la definición de las clases.

6. Cree múltiples objetos de las distintas clases del modelo, asegurando que todos tengan nombres distintos. Almacénelos en 2 estructuras de datos distintas, de manera separada: lista y diccionario.
7. En una celda de texto, conteste la siguiente pregunta: si tuviera que buscar un objeto con un nombre específico en estas estructuras, ¿en cuál de las dos sería más rápido encontrarlo y por qué? Agregue a continuación de la celda de texto, una celda con código que justifique la respuesta entregada.