Task(13.1)

Page object class

Requirements:

In this task, you will work with a base class PageObject and two derived classes: Page and Copy. The goal is to create a basic structure for handling page-related objects and their operations, specifically for adding, removing, and deleting items from both Page and Copy objects.

Key Points:

- **Base Class** (PageObject): This class has three virtual methods: addItem(), removeItem(), and deleteItem(), which provide a common interface for derived classes. It does not have any pure virtual methods, so it is not an abstract class, and you can instantiate it directly.
- **Derived Classes** (Page and Copy):
  - Page: This class overrides the addItem(), removeItem(), and deleteItem() methods to handle actions related to individual pages. Since Page provides concrete implementations of these methods, it is not an abstract class, meaning you can instantiate Page objects.
  - Copy: This class manages a collection of PageObject items in a vector. It also implements the same virtual methods, allowing for operations related to multiple pages (or other PageObject items) in a copied context.
- **Task Goal** (main):
  - Create instances of Page, PageObject, and use the addItem() function to add a PageObject to a page.
  - Use the provided methods in both classes to manipulate PageObject items and print relevant messages to the console when adding, removing, or deleting items.
- **Class diagram:** use a UML tool (such as UMLet or draw.io) to create a class diagram for this structure. The class diagram should visually represent the inheritance hierarchy between the PageObject, Page, and Copy classes, including their methods.
- **Link to make the diagram:** https://app.diagrams.net/

Thank You