

Memento Pattern

behavioral design pattern

Links

This link talks creating a text editor with the ability to undo a text. And is the main link we will use

- <https://refactoring.guru/design-patterns/memento>

You can also check out the wikipedia

- https://en.wikipedia.org/wiki/Memento_pattern

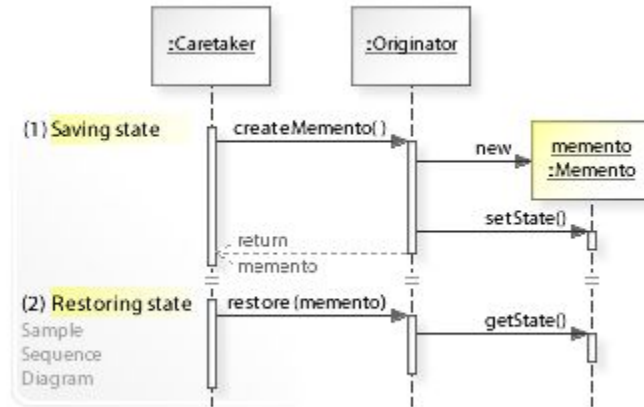
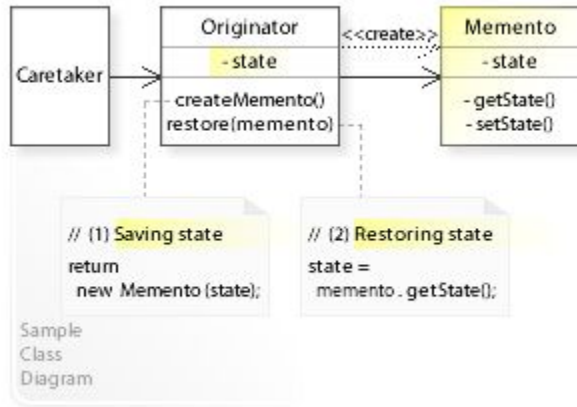
What Does it Do?

The Memento pattern lets you undo something. But a better term than undo is to restore or rollback to a previous state.

It maintains the state history of an object without revealing its implementation detail.

The basics of how it works is by saving mementos of an object's internal state, and those mementos can later be restored, thus the idea of rolling back to a previous state.

https://en.wikipedia.org/wiki/Memento_pattern



1. The Originator - the object whose state needs to be saved and restored
2. The Memento - the saved state object that stores the Originator's internal state
3. The Caretaker - responsible for storing and managing the Memento objects

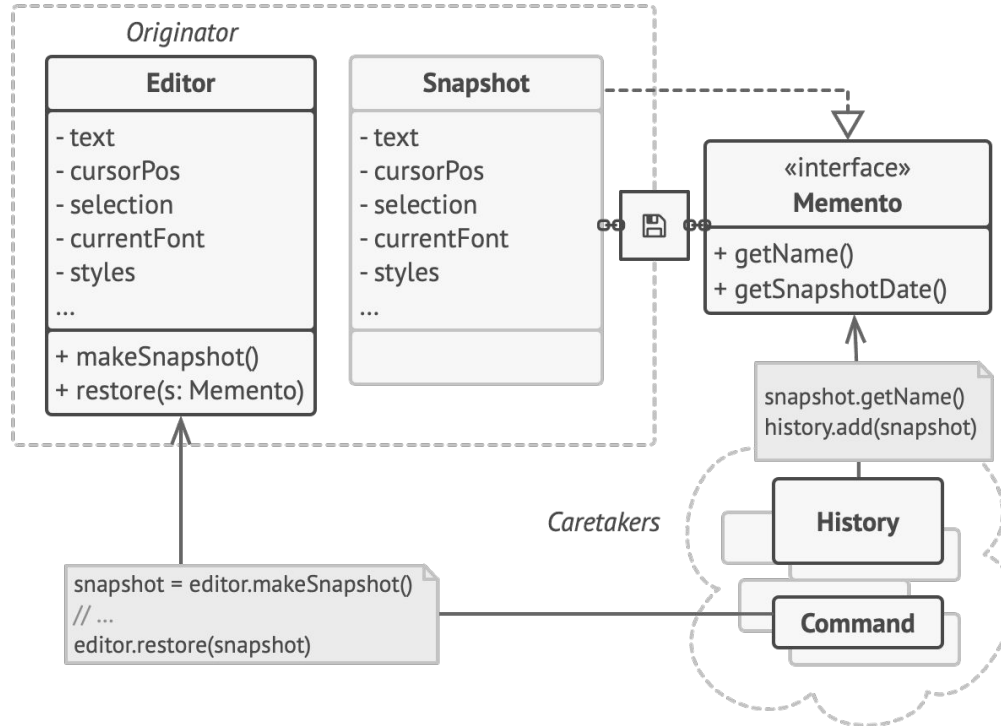
When To Use

Any time you want to restore an object state.

- such as in text editors
 - Think ctrl + z
- graphics applications
 - I mean ctrl + z again but instead of removing text its remove graphics from the screen
- games
 - A list of save files is a list of mementos

The UML we will implement and add to

<https://refactoring.guru/design-patterns/memento>



Link to code the code: <https://github.com/Quinn-T-Nash/Presentation>