

Both Memento and Command patterns are fairly easy implementations for undo-redo functionality in an SVG emoji generator.

The Memento Pattern requires creating a Memento Class that captures the current state of an object, in this case the Memento captured an SVG shape. By capturing the current state, it allows for it to be restored at a later time. In the case of my SVG Emoji Generator, the Memento was to store an SVG shape and the Caretaker was responsible for Adding, Removing, Moving (Up, Down, Left, Right), Rotating around own centre and Styling along with Undoing, Redoing and storing the redoList(list of undone shapes) and Canvas. By implementing these functions, I was capable of no problem fulfilling the criteria of the brief.

The Command Pattern requires treating actions as Classes. For example: I needed a separate class for each of the following; Storing Shape, Undoing, Redoing, Printing the SVG, Creating the SVG, Removing Shapes, Moving up, Moving down, Moving left, Moving right, Rotating(clockwise and anti clockwise) and Styling of the shapes. I then stored them in a List and was capable of using undo-redo functionality no problem by storing them in a RedoList(list of undone shapes).

In my experience, the Command Pattern worked better due to the fact that I was able to control the undo-redo functionality in much more detail. This along with the fact that it was easier to troubleshoot the program with the advantage of separate classes was a big advantage. With the Memento pattern, restoring a previous state would undo all changes made to the object since the last memento, which can cause unexpected results. The Command pattern allowed for more precise control over which specific actions were undone or redone.

In terms of scalability I believe that both the Command and Memento patterns would scale well in terms of adding more reasons but with that said, I believe that the implementation would be much simpler using the Command Pattern due to the fact that it poses a lot more precision with the undo-redo functionality as well as being significantly easier to troubleshoot and test. With the Memento pattern we would have to create new methods within the Caretaker class and with the Command Pattern we would make additional Command classes. I believe it is worth noting that with the Command Pattern, it is possible that it would require more effort to implement as each new Command would need to be a new class.

In summary, while both the Memento pattern and Command pattern can be used to implement undo-redo functionality, in my personal opinion the Command pattern was much easier to implement and proposes many advantages over the Memento pattern. Both patterns would Scale well but if I was to choose between the two, the only reason why I would pick the Memento pattern would be due to the fact that in short and simple Applications it takes much less to implement from my experience.