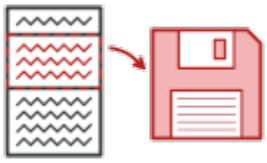
refactoring.guru

Memento em Go / Padrões de Projeto

3–4 minutes



O **Memento** é um padrão de projeto comportamental que permite tirar um “retrato” do estado de um objeto e restaurá-lo no futuro.

O Memento não compromete a estrutura interna do objeto com o qual trabalha, nem os dados mantidos dentro dos retratos.

Exemplo conceitual

O padrão Memento nos permite salvar instantâneos do estado de um objeto. Você pode usar esses instantâneos para reverter o objeto ao estado anterior. É útil quando você precisa implementar operações desfazer-refazer em um objeto.

originator.go: Originador

```
package main

type Originator struct {
    state string
}
```

```
func (e *Originator) createMemento() *Memento {
    return &Memento{state: e.state}
}

func (e *Originator) restoreMemento(m *Memento) {
    e.state = m.getSavedState()
}

func (e *Originator) setState(state string) {
    e.state = state
}

func (e *Originator) getState() string {
    return e.state
}
```

memento.go: Memento

```
package main

type Memento struct {
    state string
}

func (m *Memento) getSavedState() string {
    return m.state
}
```

caretaker.go: Cuidador

```
package main

type Caretaker struct {
    mementoArray []*Memento
}

func (c *Caretaker) addMemento(m *Memento) {
    c.mementoArray = append(c.mementoArray, m)
}

func (c *Caretaker) getMemento(index int) *Memento {
    return c.mementoArray[index]
}
```

main.go: Código cliente

```
package main

import "fmt"

func main() {

    caretaker := &Caretaker{
        mementoArray: make([]*Memento, 0),
    }

    originator := &Originator{
        state: "A",
    }
```

```
    fmt.Printf("Originator Current State: %s\n", originator.getState())
    caretaker.addMemento(originator.createMemento())

    originator.setState("B")
    fmt.Printf("Originator Current State: %s\n", originator.getState())
    caretaker.addMemento(originator.createMemento())

    originator.setState("C")
    fmt.Printf("Originator Current State: %s\n", originator.getState())
    caretaker.addMemento(originator.createMemento())

    originator.restoreMemento(caretaker.getMemento(1))
    fmt.Printf("Restored to State: %s\n", originator.getState())

    originator.restoreMemento(caretaker.getMemento(0))
    fmt.Printf("Restored to State: %s\n", originator.getState())

}
```

output.txt: Resultados da execução

```
originator Current State: A
originator Current State: B
originator Current State: C
Restored to State: B
Restored to State: A
```

Memento em outras linguagens


C#
C++
Java
php
python
Ruby
Rust
TS