

## Lecture 39 : Memento Design Pattern

classmate

Date \_\_\_\_\_

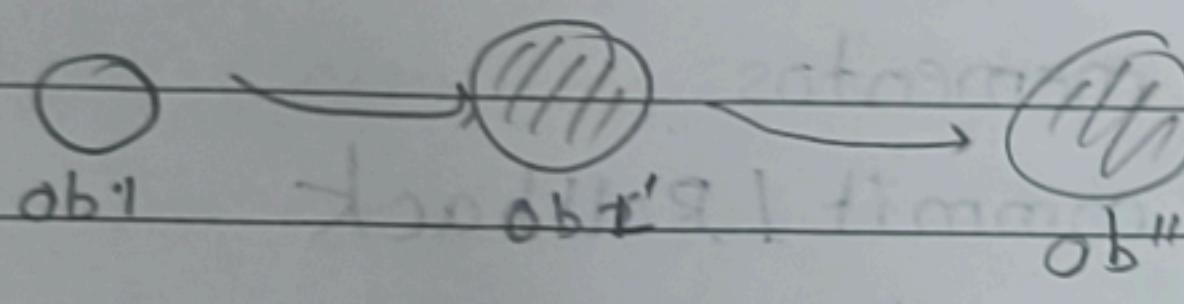
Page \_\_\_\_\_

### # What is Memento Design Pattern?

- It is used even when we want to save snapshots of object or state.

- Let's understand with an example -

• We have an object jiski state baar baar change ho rhi hai -



- Ab hum iss ob1 ka snapshot lena chahte jab bhi yeh new state par jaye.

- snapshots will help us to understand object ki previous state kya thi suppose ob1 was fine till ob' and jb it changed state to ob'' toh humein pks state yaane ob1 par jaane ke liye snapshot help krega / rekar jaayega

-

Inshort -

→ Save objects snapshots/states

→ Provides undo capability

→ Rollback to previous stable state

### # 3 Magic Words to understand Memento

#### 1. Originator

- Object whose state changes (ex-database)

- creates memento

- restores from memento

## 2. Memento:

- Snapshot of Originator's state
- Stores internal state
- Provides get state

## 3. Caretaker:

- Manages Mementos
- stores list of mementos
- initiates begin / commit / Rollback

## Example - DB Transaction Manager

Suppose, in SQL you are changing some values suppose in row 1 but due to certain issues it only changed roll-no to 102 but did not updated name, so to prevent this inconsistent state

Inshort, Problem = Database Inconsistency due to -

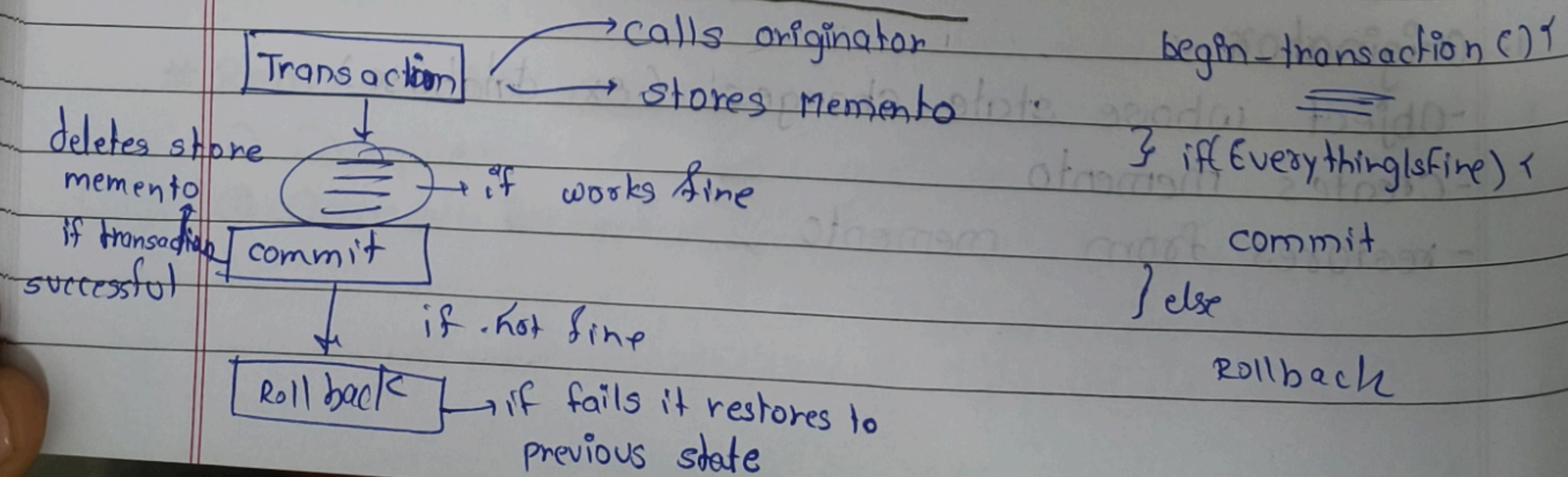
↳ Failed CRUD Operation

↳ Partial updates

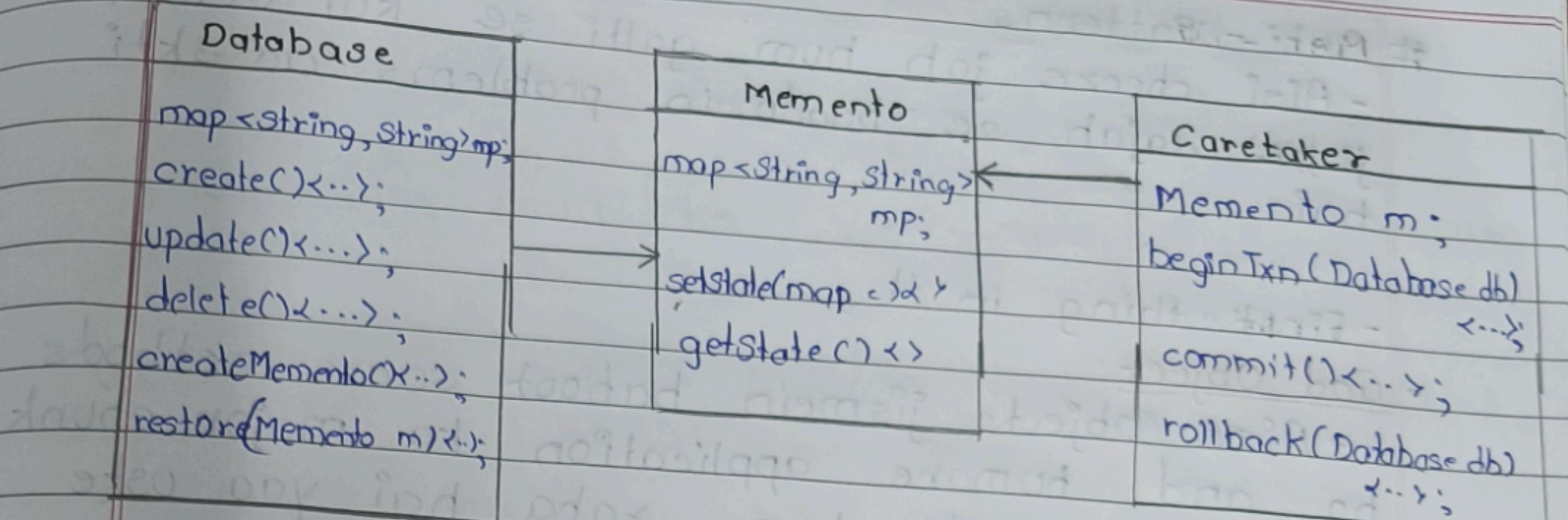
Solution : Rollback

↳ Revert to previous consistent state

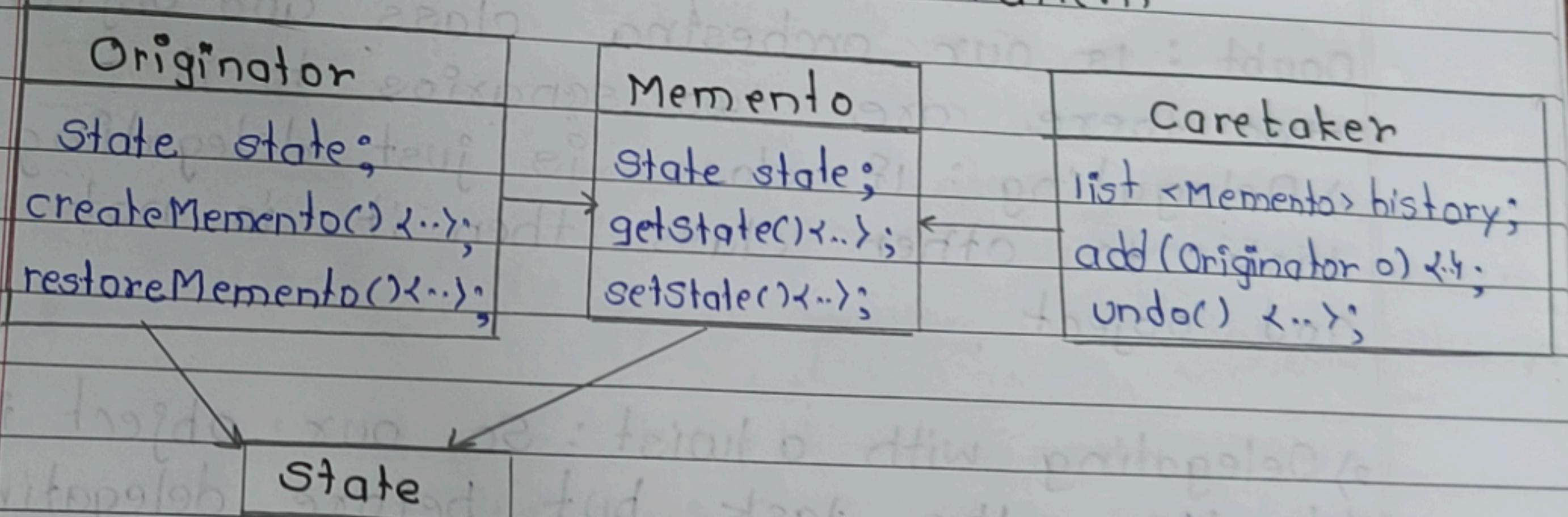
## Steps in Database Transaction



## # UML Diagram



## # Standard UML of Memento Pattern



## # Standard Definition

→ Provides an ability to take snapshot of an object at various point in time and provide undo capabilities to a previous state.

## # Real World Use Case

- ① Database Transaction Management
- ② Version control Systems
- ③ Any application needing Undo capabilities
- ④ Saving object state for Failure Recovery