VMWARE Nested Transactions Code Review

By Aditya Kumar Sinha

Terminologies:

1. READ : To read and print the value of the associated key from memory.
2. WRITE : To write or overwrite the key - value data in the memory.
3. DELETE : Removes all the key - value from memory.
4. START : Initiate the Transition.
5. COMMIT : Saving the Transition as permanent memory.
6. ABORT : Cancel the Transition, and revert will the changes done during the transition.
7. QUIT : To exit the project and whole process.

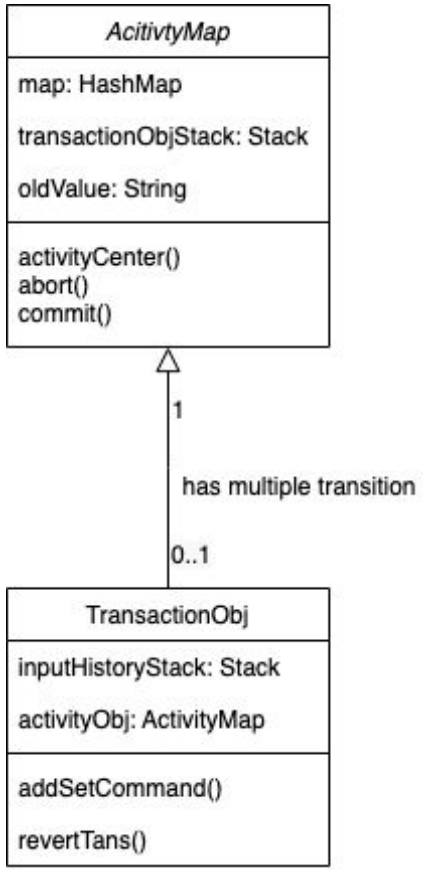
Drawback:

1. No Backup of COMMITED Transition.
2. Does not support multithreading.
3. Takes only one word key and one word value.

Class Diagram:

World's largest coral reef system

1. Class ActivityMap:
   a. activityCenter() - - TC: O(1)
   b. abort() - - TC: O(1)
   c. commit() - - TC: O(1)
2. Class TransactionObj:
   a. addSetCommand() - - TC: O(1)
   b. revertTrans() - - TC: O(commands in transition)



Output:

```
WELCOME TO KEY VALUE INMEMORY PROJECT
Tip: 'QUIT' for exiting the project.


----------------------------------------


WRITE a hello
READ a
hello
DELETE a
READ a
Key not found: a
WRITE a helloWorld
READ a
helloWorld
START

----------Transition Started---------------

WRITE b helloVmware
READ a
helloWorld
WRITE a justHello
READ a
justHello
COMMIT

----------Transition Saved---------------

READ a
justHello
READ b
helloVmware
START

----------Transition Started---------------

READ a
justHello
WRITE a faultText
READ a
faultText
READ c TESTING
Key not found: c
WRITE c TESTING
READ c
TESTING
ABORT

----------Transition Aborted---------------

READ a
justHello
READ c
Key not found: c
QUIT
Exiting...
```