

:FreshWorks Engineering Assignment Submission :

Name - Arunabha Mondal

* The Class for Key-Value datastore operation is named as “**KeyValueStoage**”

*(All the utility functionality Classes and Exception classes have been kept in a single .java file(KeyValueStorage.java) for convenience in checking of the code.

*Singleton design pattern is used to keep the class thread safe.

Methods :

public synchronized KeyValueStorage getInstance():

- 1.If no object has been created for the class by the client then create a new object and return the object refernece and set a default storage path(by invoking setDefaultPath())
2. If already an object has been created by the client then the old object reference is returned.
3. Internally invoke deadKeyCleaner() method.

public synchronized KeyValueStorage getInstance(String path):

- 1.If no object has been created for the class by the client then create a new object and return the object refernece and set the storage path(by invoking setDefaultPath())
2. If already an object has been created by the client then the old object reference is returned.
3. Internally invoke deadKeyCleaner() method.

public synchronized void create(String key,JSONObject value,int timeToLive) :

1. To add a new Key-Value pair to the storage with providing Time-To-Live value(seconds)
2. Validations for Key Length(Not more than 32 characters),Null check for the key(Null keys are not allowed),Unique Key Check,JSONObject size (Not more than 16KB),File Size(not more than 1GB) has been done
3. Proper messages will be shown to the invoker(SOPs)

public synchronized void create(String key,JSONObject value) :

1. To add a new Key-Value pair to the storage without providing Time-To-Live value.(Default value will be given as *Integer.MAX_VALUE* seconds)
2. Validations for Key Length(Not more than 32 characters),Null check for the key(Null keys are not allowed),Unique Key Check,JSONObject size (Not more than 16KB),File Size(not more than 1GB) has been done
3. Proper messages will be shown to the invoker(SOPs)

public synchronized void read(String key) :

1. Print the corresponding value (JSONObject) for the given key.
2. Show error message to the invoker if the key doesn't exist in the file or if the Time-To-Live period is over for the invoked key for any other undesirable failure.

public synchronized void delete(String key) :

1. Delete the key-value pair from the storage for the given key.
2. Show error message to the invoker if the given key doesn't exist in the file.

private synchronized static void deadKeyCleaner() :

1. Internally called by the getInstance() and getInstance(String path) methods to delete the keys for which the time to live duration is over and show the list of dead keys to the invoker.
2. Show error message to the client for any undesirable failure.

private static String setDefaultPath():

1. return a default storage path .
2. Internally called by the getInstance() method when client has not given any specified path for storage.
3. Internally called by the getInstance(String path) when the user given path is invalid.

private static boolean validatePath(String path):

1. Validate a user given path and if validation succeeds return true; else return false.
2. Internally called by the getInstance(String path) method to validate the given path.

private static boolean validateTimeToLive(int timeToLive) :

1. return true if the given time to live value is >0 otherwise return false

THANK YOU

Date : 01/01/2021