COMPOSITION

You will develop a smart door system which is called e-door. E-door should be capable of working with one of three lock systems. These lock systems are iris recognizer, fingerprint recognizer and card recognizer. Each type of lock requires different type of data as a key. There are three different types of key data. These are iris data , fingerprint data and card data.

1. Develop an abstract type which can refer to all Lock types. Its name will be Lock
    1. This abstract class will have two methods lock and unlock with the following signatures
        1. boolean lock()
        2. boolean unlock(Key key)

2. Develop three concrete types which extends the Lock . Their names will be IrisLock, FingerprintLock, CardLock
    1. These classes will implement the below methods and print the below messages according to the impleneting class.
        1. boolean lock()
            1. print “Card Locked ” returns true
            2. print “Iris Locked ” returns true
            3. print “Fingerprint Locked ” returns true
        2. boolean unlock(Key key)
            1. print “Card Unlocked ” returns true
            2. print “Iris Unlocked ” returns true
            3. print “Fingerprint Unlocked ” returns true

3. Develop an abstract type which can refer to all Key types. Its name will be Key
    1. This abstract class will only one method that calculates a numeric key according to the key type with the following signature
    2. long getData()

4. Develop three concrete types which extends the Key . Their names will be IrisKey, FingerprintKey, CardKey. Each type of key can only be accepted by unlock method of an appropriate lock type . For example unlock(Key irisKey) method only accepts IrisKey type , otherwise returns false
    1. These classes will implement the below method
    2. long getData() , for Card returns 1000, for Iris returns 2000, for Fingerprint returns 3000
5. Develop a Door class that has a Lock
6. Develop a test class, TestDoor , and add code for below jobs
    1.
        1. Create a Door object, flatDoor
        2. Create a CardLock object, cardLock
        3. flatDoor has cardLock
        4. Create CardKey object, cardKey
        5. Call lock() and unlock(cardKey) method
    2.
        1. Create a Door object, officeDoor
        2. Create a IrisLock object, irisLock
        3. officeDoor has irisLock
        4. Create IrisKey object, irisKey
        5. Call lock() and unlock(irisKey) method

3.
    1. Create a Door object, mainDoor
    2. Create a FingerprintLock object, fingerprintLock
    3. mainDoor has fingerprintLock
    4. Create FingerprintKey object, fingerprintKey
    5. Call lock() and unlock(fingerprintKey) method

7. All fields should be private.