Question

* Create a class UserName, which contains two fields first name and last name of type String. UserName class must be properly encapsulated. This class will contain a method fullName, which will return full name by combining first name and last name. This class doesn't have any constructor.
* Sample
* first name = Ramesh, last name = Sharma, then full name is Ramesh Sharma
* Create a utility class MyUtility. This class will have one static method countVowel, which accept one parameter of type UserName and return the count of vowels in the parameter. This class also contains a private method isVowel, that will accept one parameter of type char and return true if the parameter is a vowel, otherwise it will return false. This class doesn't have any constructor.
* Create a class MyMain, which contains a main method. And perform the following tasks :
  + From main method accept first name and last name from the user and store it in UserName object.
  + Print the full name of the user
  + Print the number of vowels in the full name of the user, using the countVowel method.

**Design the application using the following specification**

* Create an Address class, which contains following field int flat number, String building name, String locality, String city, String district, String state, String country and int pin code. Address class must be properly encapsulated. This class will contain a method fullAddress() that will not accept any parameter, but return full address as per the below format. This class have parametrized and non parametrized constructors.
* Sample
* Flat number - 410
* Building name - Speaking Tree Tower
* Locality - Gandhidham
* City - Ramghar
* District - Sippy
* State - Madhya Pradesh
* Country - India
* Pincode - 462058
* fullAddress method should return
* 410, Speaking Tree Tower, Gandhidham, Ramghar, District - Sippy, Madhya Pradesh, India, Pincode - 462058
* Create an Employee class, which contains following fields of given types String name, String email, String pan card number, String mobile, Address permanent address, Address present address and Address communication address. Employee class must be properly encapsulated. This class have parametrized and non parametrized constructors.
* Create a utility class design.one.MyUtility, this class will have two static methods as defined below:
* public static boolean isValidAddress(Address address)
  + this method will check if the address is valid or not, if valid it will return true otherwise false. Validation rule for address are as follows
    - a valid address have 6 digits in pin code
    - a valid address have positive flat number
* public static boolean isValidPanCard(Employee employee)
  + this method will check if the pan card is valid or not, if valid it will return true otherwise false Validation rule for pan card are as follows:
    - a pan card is valid if the 4th letter of pan card is 'p' or 'P'
    - a pan card is valid if the 5th letter of pan card is Surname initial
    - Sample
    - Name - Ramesh Sharma
    - pan card number - AMFPS1234K
    - this is valid as 4th letter is P and 5th Letter S is surname initial.
* Create a class design.one.MyMain, which contains the main method and perform the following tasks:
  + Create Address objects by accepting details from user and validate them.
  + Create Employee object by accepting employee details from use and by supplying validated address objects.
  + print the Employee details, with address.