Lab 13 - VaccineRecord

Learning Objective: demonstrate creation of a class.

To organize and keep track of individuals who got the COVID-19 vaccine we are creating the back-end of the system. The most fundamental part of the project is securely storing the data for each vaccination.

Part 1: Start your class definition. Name it *VaccineRecord* and create the __init__ method. Your method should assign parameter values for the following data:

first name
last name
date of birth
dose 1 manufacturer, batch number, date, and location
dose 2 manufacturer, batch number, date, and location

The identifier for each variable is given in the UML diagram in vaccine_record.py.
Test your method in main.py by creating a VaccineRecord object.

Part 2: Write the __str__ method which returns a string of all instance variables for a VaccineRecord object.
Test your method in main.py by printing your VaccineRecord to the console.

Part 3: Write an accessor method for each variable of the VaccineRecord class. Remember to follow naming conventions by starting each with get_ then the identifier for the variable.
Add the methods to the UML diagram.
Test each accessor method in main.py.

Part 4: Write a mutator method for each variable of the VaccineRecord class. Remember to follow naming conventions by starting each with set_ then the identifier for the variable.
Add the methods to the UML diagram.
Test each accessor method in main.py.

Part 5: Often we want additional functions in a class definition that will be helpful to use with objects of that type. Create a format_date function in vaccine_record.py that takes three inputs: month, day, year. The function should return a string with the date formatted as mm/dd/yyyy.
Add the function to the UML diagram.
Test the function in main.py.

Submit: There are automated tests for this lab so make sure they pass before submitting. As always, stop by student hours, send an email, check in with a peer, or stop by the STEM Center if you need any assistance.