Загальна мета заняття - розробити основні елементи сайту, що надає сервіс конструктора резюме для користувачів. Ідея полягає в тому, що користувач може додати свій обліковий запис, додавати/змінювати/видаляти контакти, навички та досвід роботи, а система згенерує сторінку з його резюме.

Для реалізації задуму треба:

1. Реалізувати класи, які будуть виконувати роль моделей даних.

- class Skill - описує одну за навичок користувача. Навички можуть бути трьох категорій (category):

технології (technologies),

методології (methodologies)

та мови (languages).

Кожна навичка характеризується такими параметрами:

назва (name),

досвід (experience) - кількість років використання цієї технології/методології/мови,

рівень володіння навичкою (level) - вибір з п'яти можливих варіантів: beginner, junior, middle, senior, expert.

- class Contact - описує контактні дані користувача. Описується полями тип (contact\_type) - вибір з варіантів 'phone' та 'email'; та значення (value) - конкретна мейл-адреса або номер телефону користувача.

- class JobExperience - описує досвід роботи користувача. Характеризується атрибутами: дата початку роботи (start\_date), дата завершення роботи (end\_date), компанія (company), посада (position).

- class Person - описує особу самого користувача. Має атрибути ім'я (first\_name), прізвище (last\_name), дата народження (birth\_date), а також списки контактів (об'єкти класу Contact), навичок (об'єкти класу Skill) та досвіду роботи (об'єкти класу JobExperience). Кожен об'єкт класу має також атрибут id - унікальний ідентифікатор користувача в системі.

2. Реалізувати відповідні методи для класу Person:

- Для кожного зі списків (контакти, навички, досвід роботи) мають бути реалізовані методи додавання (add), видалення (delete) та оновлення (update) елементів списку. Для реалізації цих методів можливо буде необхідне додавання вспоміжних атрибутів для кожного класу.

- Реалізувати методи збереження інформації про об'єкт класу Person разом з усіма вкладеними об'єктами у JSON файл та завантаження JSON файлу із створенням всіх вкладених об'єктів.

- Реалізувати метод, який представляє список skills персони, розбитий за категоріями. Метод має повертати словник, де ключами є категорії навичок, а значеннями - списки об'єктів навичок персони, що належать до цієї категорії, відсортовані за зменшенням досвіду (навичка з найбільшим значенням досвіду у цій категорії йде перша).

- Реалізувати метод, який сортує досвід роботи персони від найбільш актуального до найбільш давнього (останній досвід роботи йде першим у відсортованому списку, найбільш давній - останнім)

3. За допомогою фрейм-ворку Flask реалізувати простий сервер, який буде мати два url:

- "/" - повертає список повних імен всіх персон (first\_name + last\_name), у текстовому представленні.

- "/person/<int:person\_id>" - повертає текстове представлення інформації про одного користувача.

За необхідності можна додавати будь-які службові атрибути та методи для будь-яких класів. Усе рішення має міститися в окремій папці з назвою cv\_builder.

class Skill:

category:

technologies:

name,

experience,

level:

beginner,

junior,

middle,

senior,

expert.

methodologies:

name,

experience,

level:

beginner,

junior,

middle,

senior,

expert.

languages:

name,

experience,

level:

beginner,

junior,

middle,

senior,

expert.

class Contact:

contact\_type:

'phone', 'email'

value

class JobExperience:

start\_date,

end\_date,

company,

position

class Person:

first\_name,

last\_name,

birth\_date,

list of Contact,

list of Skill,

list of JobExperience

id

def add:

def update:

def delete:

def dump\_to\_json:

def load\_from\_json:

def sort\_skils:

return { technologies:

name,

experience,

level:

beginner,

junior,

middle,

senior,

expert.

methodologies:

name,

experience,

level:

beginner,

junior,

middle,

senior,

expert.

languages:

name,

experience,

level:

beginner,

junior,

middle,

senior,

expert.

}

def sort\_ JobExperience:

(20.11.2021)

Написать способ чтобы создавался экземпляр класса Person из залитого файла json. То есть метод будет получать словарь со всеми данными для класса Person и записывать все в этот экземпляр. Нужно что бы метод получал словарь, а еще что бы все экземпляры других классов прописанные в этом классе тоже прописывались и добавлялись в список.