

# Primary School 'Python'

## Проект №3

### I. Условия на проекта

1. Създаване на конзолно приложение, което симулира сайт на училище.
  - 1.1 Могат да се регистрират ученици, учители и обикновени потребители.
  - 1.2. Влизането в сайта е посредством въвеждане на имейл адрес и парола.
2. Информация, която се пази след регистрацията
  - 2.1. Информация за учениците: име; фамилия; имейл адрес; парола; учебна група;
  - 2.2. Информация за учителите: име; фамилия; имейл адрес; парола; учебни групи, на които преподава; дата на раждане; дата на започване на работа като учител;
  - 2.3. Информация за потребителите: име; фамилия; имейл адрес; парола;
3. Групи
  - 3.1. Всяка учебна група има наименование и в нея се пази списък с учениците, които учат в нея, посредством съхранение във файл.
  - 3.2. Учебната група може да има само един преподавател.
  - 3.3. Един преподавател може да преподава на няколко учебни групи.
4. Информация, която се пази за училището
  - 4.1. Наименование, списък на учителите, списък на групите и списък на учениците
  - 4.2. Имейл адрес и пароли на администратора
5. Действия в системата
  - 5.1. Регистрацията чрез попълване на данни.
  - 5.2. Влизане в сайта чрез попълване на имейл адрес и парола.
  - 5.3. Визуализиране на всички групи, а при избор на група визуализиране на учениците в нея.
  - 5.4. Визуализиране на всички учители, а при избор на един учител само неговата информация.
  - 5.5. Опции за администратора относно промяна на информацията достъпна в училището.
  - 5.6. Запазване на данните във файлове (.csv).
  - 5.7. Опции за администратора за генериране на графики.
  - 5.8. След влизане в сайта, при излизането от всяка опция да се връща в предишното меню.

### II. Задачи

1. Да се реализира конзолно приложение съдържащо условията от точка 1.
2. Да се направи диаграма на класовете
3. Данните да се запазват във файлове, които могат да се редактират.
4. Да се направят графики за броя на студентите по групи, възрастта на учителите и стажа на учителите.
5. Да се напише описание на разработеното приложение, съдържащо обяснение на класовете, библиотеките и графиките.
6. Да се попълни информация поне за 20 записа.

### III. Описание на конзолното приложение

#### 1. Регистрации

- регистрация на ученици и потребители: въвеждане на име, фамилия, имейл адрес и парола.
- регистрация на учители: въвеждане на име, фамилия, имейл адрес, парола, дата на раждане и дата на започване на работа като учител.
- администраторът не се регистрира. Има предварително създадена регистрация - имейл адрес ([admin@pythonschool.com](mailto:admin@pythonschool.com)), парола (adminpassword) и специална парола (\*123\*).
- регистрации се съхраняват във файлове, разделени според вида на потребителя.
- администраторът може да одобри или отхвърли регистрация на учител или ученик, но не и на потребител. Ако регистрацията бъде отхвърлена, тя добива статус на обикновен потребител и се прехвърля във файла на потребителите.
- ако регистрацията на ученикът бъде одобрена, трябва да се добави учебна група. Регистрацията бива прехвърлена едновременно във файл с одобрените ученици и във файл с името на групата.
- ако регистрацията на учителя бъде одобрена, има опция да се добавят групи, на които преподава, но не е задължително. Регистрацията бива прехвърлена във файла с одобрените учители.

#### 2. Влизане в сайта

- всеки потребител, който си е направил регистрация може да влезе в системата след като въведе имейл адрес и парола.

#### 3. Опции в сайта

##### 3.1. Опции за потребителите и за учениците

- Опция за показване на списък с всички учители, съдържащ име, фамилия и имейл адрес. Може да бъде избран конкретен учител след въвеждането на неговия имейл адрес. Информацията, която се изписва за конкретния учител е име, фамилия, учебни групи, на които преподава и дата на започване на работа като учител.
- Опция за излизане от системата.

##### 3.2. Опции за учителите

- Опция за показване на списък с всички учители, съдържащ име, фамилия и имейл адрес. Може да бъде избран конкретен учител след въвеждането на неговия имейл адрес. Информацията, която се изписва за конкретния учител е име, фамилия, учебни групи, на които преподава и дата на започване на работа като учител.
- Опция за показване на учебните групи. Може да бъде избрана конкретна учебна група след въвеждане на името ѝ. Ако бъде избрана се визуализират имената, фамилиите и имейл адресите на всички ученици, които учат в нея.
- Опция за излизане от системата.

##### 3.3. Опции за администратора

###### А) Опция за управление на потребителите

- Опцията за управлението на потребителите съдържа в себе си следните вторични опции:

##### 1. Регистрации

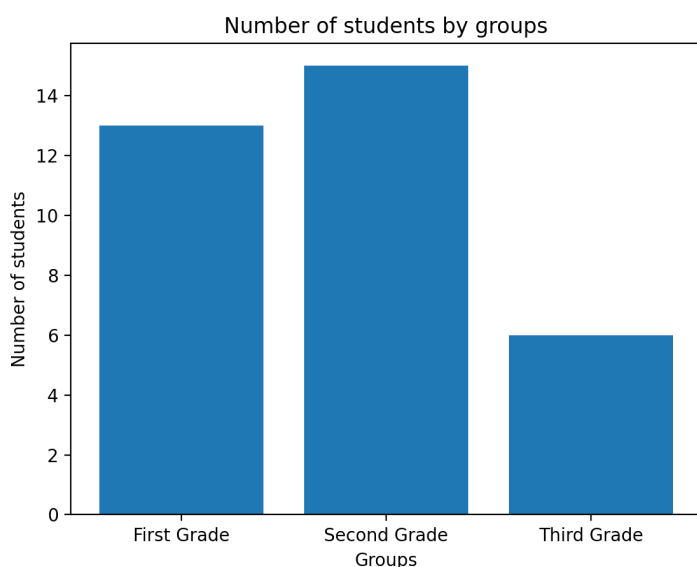
- Регистрации на ученици и отделно регистрации на учители. Могат да бъдат отхвърлени или одобрени.

##### 2. Потребителски профили

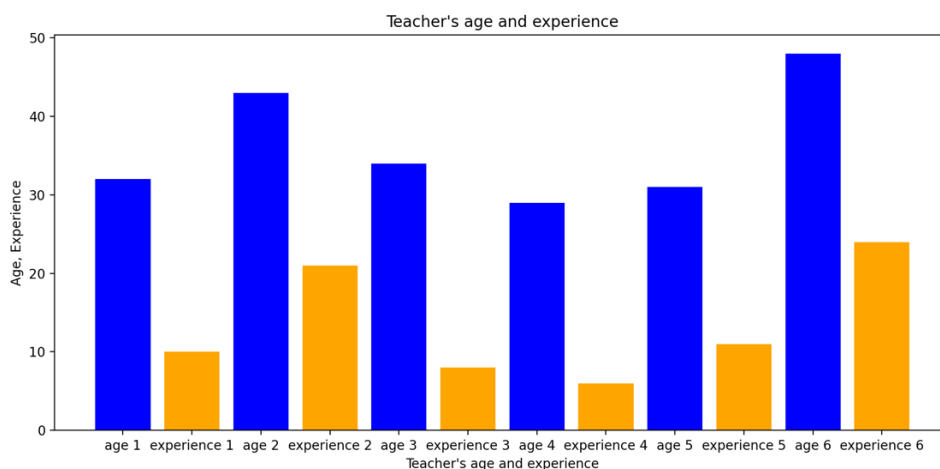
- Показване на потребители, редактиране или изтриване на потребителски профили.

###### Б) Опция за управление на учебните групи

- Опцията за управлението на групите съдържа в себе си следните вторични опции:
  1. Показване на всички учебни групи
  - Визуализиране на списък с всички учебни групи. Възможност за избор на определена учебна група и показване на учениците в нея.
  2. Създаване на нова учебна група
  - След като ѝ се даде име се създава файл, в който да се попълват учениците.
  3. Редактиране на учебна група
  - Редактиране на името на учебната група. Променя се и името на файла.
  4. Изтриване на учебна група
  - Едната възможност е да се изтрие файлът и цялата информация в него, другата възможност е да се изтрие файлът, но информацията да се прехвърли в друга група.
- В) Опция за показване на информацията за училището
- Опцията за показване на информацията за училището съдържа в себе си следните вторични опции:
  1. Показване на учениците по групи
  - Визуализиране на графика, която показва броя на учениците във всяка група.



2. Показване на информация за учителите
- Визуализиране на графика, която показва възрастта и годините опит на учителите.



- Г) Опция за излизане от системата.

#### IV. Класове

- Таблицы на класовете:



CLASS SCHOOL
os_path_exists_database(database)
show_students_database()
show_teachers_database()
delete_account(email, user_name)
find_account(database, email)
os_path_exists_registrations(database)
show_students_or_teachers_database_for_approval(database, user)
os_path_exists(email, database)
write_in_database(database, temp_list)
overwriting_the_database(database, user)
find_student(email)
find_email(database, email)
add_in_list(database)
find_teacher(email)
deny_user(email, user)
show_users_database()
edit_user()
edit_student_or_user(database, user_name, email)
edit_teacher(database, email)

## V. Използвани библиотеки

### 1. Csv (Comma Separated Values)

- Използван модул за запазване на информация във файлове с разширение .csv. Файловете могат да се редактират, презаписват и изтриват.

### 2. Os

- Използван модул за проверка дали даден път (файл) съществува чрез операционната система.

### 3. Copy

-Използван модул за дълбоко копиране, което конструира нов обект и след това рекурсивно вмъква копия в него на обектите, намерени в оригинала.

### 4. Datetime

- Използван модул предоставящ класове за манипулиране на дати и часове.

### 5. Matplotlib

-Използван модул с функцията .pyplot за генериране на графики.

6. Re

-Използван модул за операции с регулярни изрази.

КРАЙ!

БЛАГОДАРЯ ЗА ВНИМАНИЕТО!