

Dokumentacja projektu – baza danych osób, studentów, nauczycieli, klas

Autor – Maksym Buniak

Data ostatniej aktualizacji projektu – 14.12.2021

Data ostatniej aktualizacji dokumentacji – 21.12.2021

Informacja ogólna :

Baza danych osób, studentów, nauczycieli, klas.

Baza powinna być zapisywana do pliku (txt, CSV, JSON).

Uwaga w zipie już jest baza, która zawiera niektóre dane!(lepsza opcja dla wizualizacji działania aplikacji)

Jeśli w folderze z aplikacją nie będzie app.db , to wtedy będzie stworzona nowa pusta baza danych!!!

(Po stworzeniu bazy danych)app.db można otworzyć za pomocą programu -> DB Browser(SQLite)

Operacje na bazie :

wstawianie

usuwanie

wyszukiwanie

listowanie


Relacje w tabelach :

Osoba w tabeli 'persons' – podstawowa jednostka

Table: <div>persons</div>									
id	name	lastName	email		birth				

Student w tabli 'students' – osoba może być studentem

'id' w tabeli "students" to Foreign Key(klucz obcy) do 'id' tabeli "persons".

Table: students																					
student_id	id																				
Filter	Filter																				

Nauczyciel w tabeli 'teachers' – osoba może być również nauczycielem

'person_id' w tabeli "teachers" to Foreign Key(klucz obcy) do 'id' tabeli "persons".

Table: teachers																			
teacher_id	person_id																		

Klasa ćwiczeniowa w tabeli 'classes' – jednostka w tabeli 'classes'

Table: <div>classes</div>					
id	discipline	name	teacher_id		

Student w klasie – jednostka w tabeli 'students_in_class'

Table: students_in_class			
id	class_id	student_id	

Jakie biblioteki są konieczne żeby uruchomić program :

- SQLAlchemy
- Prettytable
- Flask-SQLAlchemy (gdzie os jest inny niż Windows) :

```
$ pip install Flask-SQLAlchemy
```

Jak uruchomić aplikację :

Shell :

```
$folder_aplikacji/ python3 main.py
```

Funkcjonalność :

Główne f-e() :

1) wyświetlanie wszystkich tabel :

```
def show_tables() : # ...
```

2) wstawianie do bazy rekordu danych do pewnej tabeli :

```
def insert_to_table() : # ....
```

3) usuwanie rekordu danych z pewnej tabeli :

```
def delete_from_table() : # ...
```

4) wyświetlanie (sprawdzenie) rekordu danych w pewnej tabeli :

```
def retrieve_from_table() : # ...
```

5) eksportowanie pewnej tabeli do pliku (.txt,.csv lub .json) :

```
def export_table_to_file() : # ...
```

F-e() pomocnicze:

1) wyświetlanie głównego menu i listy opcji do wyboru :

```
def show_main_menu() : # ...
```

2) zwracanie tabeli w formie PrettyTable :

```
def ret_pretty_table(field_names, rows) : # ...
```

Struktura aplikacji :

- main.py (plik główny)
- app.db (gotowa baza danych, która zawiera niektóre dane dla lepszej wizualizacji i działania aplikacji)