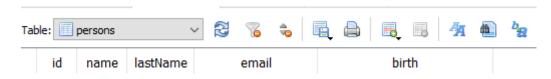
Dokumentacja projektu – baza danych osób, studentów, nauczyczieli, klas Autor – Maksym Buniak Data ostatniej aktualizacji projektu – 01.01.2022 Data ostatniej aktualizacji dokumentacji – 01.01.2022 Informacja ogólna: Baza danych osób, studentów, nauczyczieli, klas. Baza powinna być zapisywana do pliku (txt, CSV, JSON). Uwaga w zipie już jest baza, która zawiera niektóre dane!(lepsza opcja dla wizualizacji działania applikacji) Jeśli w folderze z applikacją nie będzie app.db, to wtedy będzie stworzona nowa pusta baza danych!!! (Po stworzeniu bazy danych)app.db można otwórzyć za pomocą programu -> DB Browser(SQLite) Operacje na bazie: # wstawianie # usuwanie # wyszukiwanie

# listowanie

## Relacje w tabelach:

Osoba w tabeli 'persons' – podstawowa jednostka



Student w tabli 'students' – osoba może być studentem

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

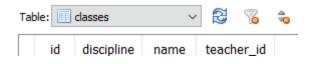


Nauczyciel w tabeli 'teachers' – osoba może być również nauczycielem

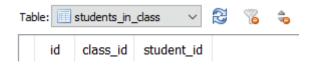
# 'person\_id' w tabeli "teachers" to Foreign Key(klucz obcy) do 'id' tabeli "persons".



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



Student w klasie – jednostka w tabeli 'students\_in\_class'



Jakie biblioteki są konieczne żeby uruchomić program(+ wersje bibliotek, które mam zainstalowane): - SQLAlchemy (my version = 1.4.27); \$ pip install SQLAlchemy - prettytable (my version = 2.4.0); \$ python -m pip install -U prettytable - Flask-SQLAlchemy (my version = 2.5.1)((gdy os jest inny niż Windows)) : \$ pip install Flask-SQLAlchemy Jak uruchomić applikację: 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) wyswietlanie (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 applikacji:

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