

Twórcy:

Andrei Daniliuk, s24610

Artem Yumina, s25316

## Dokumentacja do Zadania: System zarządzania magazynem

### Co wykonaliśmy w danym projekcie:

#### Charakterystyka:

Aplikacja konsolowa.

#### Co udało się wykonać:

W danym projekcie zostały zaimplementowane funkcje wymagane w danym projekcie a mianowicie:

- Dodanie rekordu do bazy danych z ID i bez ID;
- Usunięcie rekordu z bazy danych;
- Aktualizacja danych rekordu z bazy danych: nazwę, opis, cenę oraz liczbę danego produktu;
- Wyszukanie rekordów w bazie danych;
- Utworzenie raportu;

Ostatnie dwie funkcje umożliwiają sortowanie rekordów dla wyprowadzenia na konsoli lub w raporcie.

#### Cele projektu:

Utworzenie prostej aplikacji dla użytkownika do pracy z bazą danych gdzie są przechowywane jakieś produkty. Każdy produkt na charakterystykę: id, nazwę, opis (opcjonalnie), cena, liczba produktów. Użytkownik ma możliwość korzystania z powyżej opisanych funkcji.

#### Wykorzystane metody:

W danym projekcie zostały wykorzystane takie biblioteki:

- sqlite3 – praca z bazą danych
- os – praca z plikami i folderami na urządzeniu
- datetime – praca z datą i czasem celem utworzenia raportu z unikalną nazwą
- tabulate – praca z danymi celem wyprowadzenia informacji w formie tabelarycznej

Utworzone dodatkowe klasy własnej implementacji:

- Pair – cel, przekazanie parę wartości boolean i opcjonalna, wykorzystywana celem zwrotu komunikatów błędnych w wypadku wartości False
- OrderBy – cel, klasa typu Enum, wykorzystywana celem klarowności kodu i sortowania danych w zapytaniach do bazy danych.

W danym projekcie staraliśmy się korzystać z dobrych praktyk programistycznych takich jak:

- Podział kodu na moduły;
- Tworzenie funkcje i klasy;

## Opis klas i funkcji:

Pliki zawierające informacje:

- DataBase.py -> zawiera skrypt współpracy z użytkownikiem
- DataBaseClass.py -> class DataBase, zawiera skrypt współpracy z bazą danych
- Order.py -> class OrderBy(Enum), wyżej opisana funkcja klasy
- PairClass.py -> class Pair, wyżej opisana funkcja klasy

Opis funkcji w plikach:

Plik: DataBase.py, - funkcje :

- choice() -> menu użytkownika
- add\_product\_with\_id () -> dodane rekordu z podaniem przez użytkownika ID
- add\_product\_without\_id () -> dodane rekordu bez podania przez użytkownika ID
- delete () -> usuń rekord
  - select\_all() -> wybierz wszystkie rekordy z bazy danych
  - select\_by\_id() -> wybierz rekordy z bazy danych za podanym ID
  - select\_by\_name() -> wybierz rekordy z bazy danych za podaną nazwą
- select() -> menu wyboru powyższych opcji zaczynających się z **select**
- report () -> generowanie raportu
  - select\_id\_for\_update() -> wyświetlenie listy rekordów z bazy danych i pobranie ID
  - update\_name() -> aktualizacja nazwy za podanym ID
  - update\_description() -> aktualizacja opisu za podanym ID
  - update\_price() -> aktualizacja ceny za podanym ID
  - update\_amount() -> aktualizacja liczby produktów za podanym ID
- update() -> menu wyboru powyższych opcji zaczynających się z **update**
- exit\_program() -> wyjście z programu

Plik: DataBaseClass.py, - funkcje :

- close\_connection -> zerwanie połączenia z bazą danych
- add\_product\_with\_id -> dodane rekordu z podaniem przez użytkownika ID
- add\_product\_without\_id -> dodane rekordu bez podania przez użytkownika ID
- delete -> usunięcie rekordu
- select\_all -> wybierz wszystkie rekordy z bazy danych
- select\_by\_id -> wybierz rekordy z bazy danych za podanym ID
- select\_by\_name -> wybierz rekordy z bazy danych za podaną nazwą
- make\_report -> generowanie raportu
- update\_name -> aktualizacja nazwy
- update\_description -> aktualizacja opisu
- update\_price -> aktualizacja ceny
- update\_amount -> aktualizacja liczby

- create\_base -> utworzenie nowej bazy danych
- is\_exist\_id -> weryfikacja czy za podanym ID istnieje rekord
- is\_correct\_amount -> weryfikacja czy podana wartość liczby jest poprawna
- is\_correct\_price -> weryfikacja czy podana wartość ceny jest poprawna
- is\_correct\_id -> weryfikacja czy podana wartość ID jest poprawna

powyższe funkcje w danym pliku zwracają komunikat (w formie class Pair) jeśli taki występuje

Dodatkowa funkcja:

- table\_to\_string -> pobiera tablicę i zwraca w formie tabelarycznej przyjaznej oku użytkownika