|  |  |
| --- | --- |
| **http://www.po.opole.pl/media/logotypy/2014/logotyp-politechnika-opolska-02.jpg** | **Politechnika Opolska**  **Wydział Elektrotechniki, Automatyki i Informatyki**  **Katedra Informatyki** |
| **Rok akademicki** | 2022/2023 |
| **Przedmiot** | [Przedmiot wybieralny IX - Aplikacje bazodanowe](https://elearning.po.edu.pl/course/view.php?id=13408#section-2) |
| **Prowadzący zajęcia** | Dr inż. W. Pokuta |
| **Nr grupy** | - |

|  |
| --- |
| **Symulator aplikacji / Generator Danych** |

|  |  |
| --- | --- |
| **Nazwisko i imię** | **Nr indeksu** |
|
| **Daniel Morzejko** | **S64597** |

|  |
| --- |
| **Uwagi** |
|  |

# ****Treść zadania.****

Zadanie miało na celu wytworzenie aplikacji generującej dane do posiadanej już bazy danych.

Aplikacja miała się łączyć z bazą danych i w zależności od potrzeb pobierać z niej dane oraz dodawać nowe. Możliwość dodawania dużej ich ilości.

# Opis środowiska programistyczno testowego.

Zadanie wykonano w środowisku programistycznym **PyCharm**, wersja 2022.2.3 (Community Edition).

Język programowania to **Python**, a jego interpreter jest w wersji 3.11.

Baza danych jest lokalną bazą **Oracle** Database 21C Express Edition.

W celu wykonania testów oraz przeprowadzenia obserwacji działania aplikacji korzystano z **Oracle SQL Developer** w wersji 22.2.1.234

# Opis działania aplikacji.

W realizacji zadania podjęto się utworzenia aplikacji okienkowej(GUI) łączącej się z bazą na localhoście.

Wygląd okna programu:

Obraz zawierający tekst

Opis wygenerowany automatycznie

Rys.3.1. Wygląd okna aplikacji.

Okno pozwala na zaznaczenie jednej z trzech opcji wprowadzania danych.:

1. Dane do tabeli klient
2. Dane do tabel Obiekt oraz Przedmioty\_przeglądu
3. Dane do czterech tabel jednocześnie:

* Akumulator
* Centrala
* Typ
* Brama

Po wpisaniu w pole tekstowe liczby wpisów należy wcisnąć przycisk ‘Wypełnij bazę’ po którym wyświetla się okno informujące o tym, że wpisy zostały dodane:

Obraz zawierający tekst

Opis wygenerowany automatycznie

Rys.3.2. Wygląd okna powiadomień o pomyślnym wykonaniu operacji

Aplikacja ma zabezpieczenie przed wpisaniem wartości liczby całkowitej, jest to również wyskakujące okno (popup):

Obraz zawierający tekst

Opis wygenerowany automatycznie

Rys.3.3. Wygląd okna błędu.

Aplikacja wyświetla również informacje w konsoli PyCharm, po wciśnięciu przycisku ‘Wypełnij bazę’ aplikacja wyświetla napis „Uruchomiono generator: „ a następnie po wykonaniu pomyślnie informacji dopisuje w nowej linii: „Działa.” lub po błędzie analogicznie „Nie działa.”:

Obraz zawierający tekst

Opis wygenerowany automatycznie\

Rys.3.4. Wypisanie informacji w konsoli o działaniu aplikacji.

Aplikację można zamknąć przyciskiem Koniec, lub przyciskiem „x” eksploratora.

# Przykład działania aplikacji.

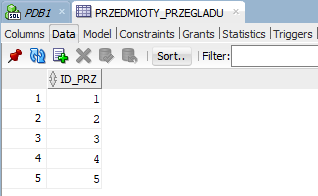
Przykładowe wykorzystanie aplikacji. Dla zobrazowania działania wykonano dwa scenariusze:

1. Dodanie po dziesięć insertów do pierwszych dwóch wyborów.

Widok bazy danych przed uruchomieniem aplikacji generującej dane:

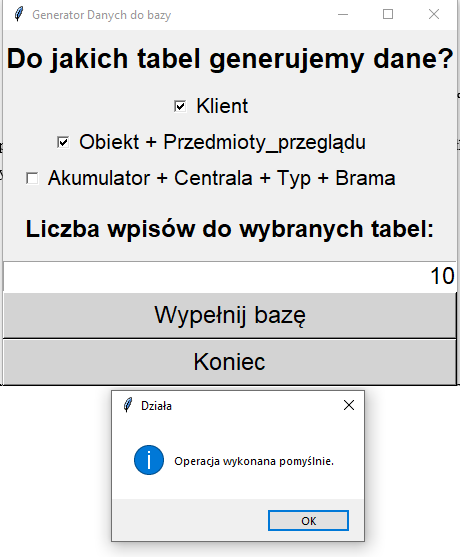
Obraz zawierający stół

Opis wygenerowany automatycznieObraz zawierający tekst

Opis wygenerowany automatycznie

Rys.4.1. Tabele przed uruchomieniem aplikacji testowej.

Uruchomienie aplikacji oraz zadanie dodania 10 insertów.

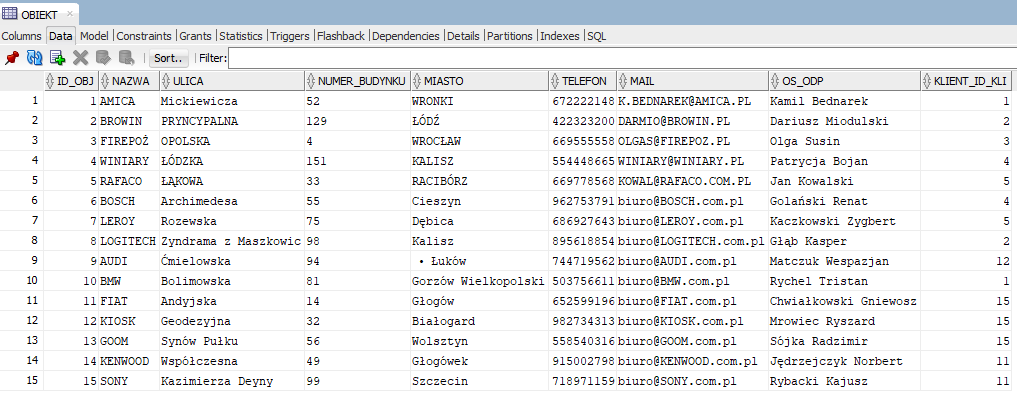


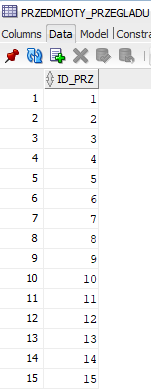
Rys.4.2. Okno aplikacji po zadziałaniu.

Dane zostały wygenerowane. Dowód:

Obraz zawierający stół

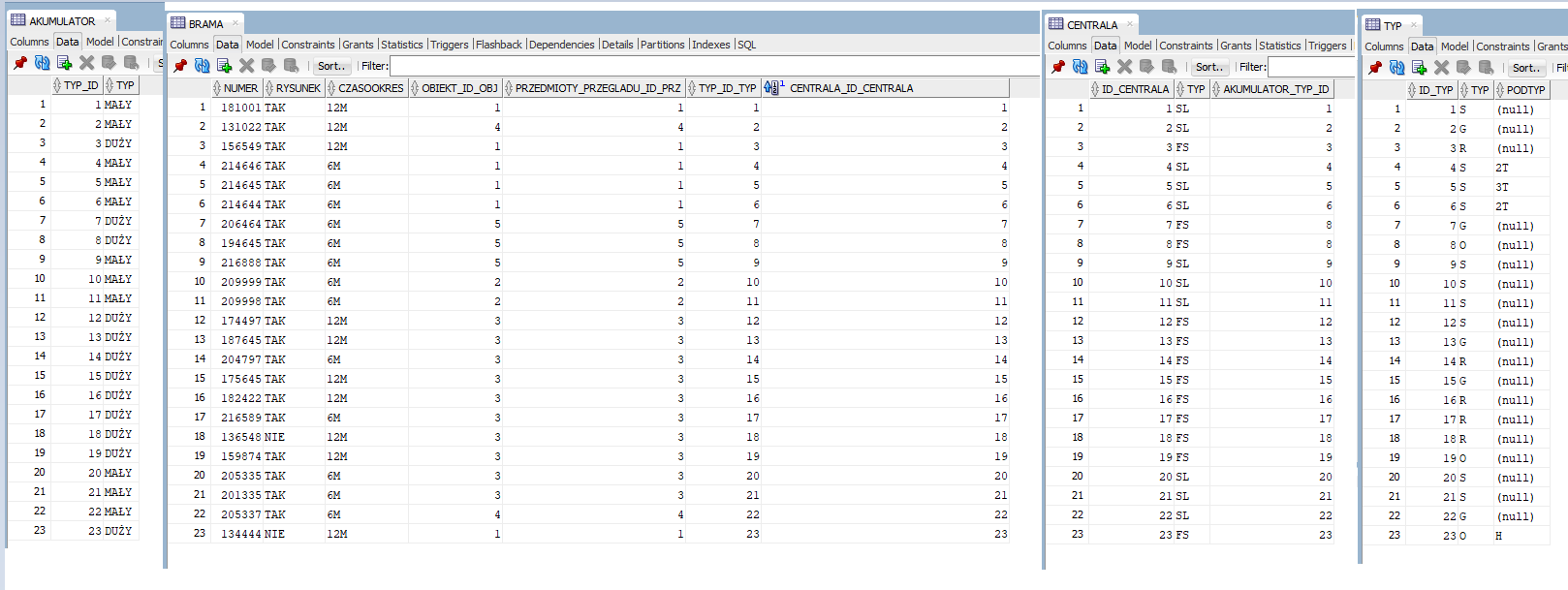
Opis wygenerowany automatycznie





Rys.4.3. Wygenerowane dane.

1. Dodanie po pięć insertów do pierwszych tabel Brama, Akumulator, Centrala oraz Typ.

Widok tabel przed operacją:

Rys.4.4. Dane przed modyfikacją.

Obraz zawierający tekst, zrzut ekranu, biały

Opis wygenerowany automatyczniePo zadziałaniu aplikacji do czterech wyżej podanych tabel zostały dodane wygenerowane dane:

Rys.4.5. Dane po modyfikacji.

# Opis działania aplikacji wraz z prezentacją kodu.

Aplikacja składa się z pięciu wykonywalnych plików .py, pliku z danymi logowania do bazy danych oraz folderu z danymi tekstowymi do generatora.

Obraz zawierający tekst

Opis wygenerowany automatycznie

Rys.5.1. Schemat plików.

# Połączenie z bazą danych.

Obraz zawierający tekst

Opis wygenerowany automatycznie

Rys.5.1.1. – Kod pliku umożliwiającego łączenie się z bazą danych

Kod pliku umożliwia kolejno: połączenie się z bazą danych danymi z pliku dbcon, tworzy kursor otwarcia bazy danych, zamknięcie połączenia poprzez usunięcie kursora, wykonanie operacji, zatwierdzenie operacji oraz na końcu pobranie danych.

# Generowanie danych.

Utworzenie klasy generator:

Obraz zawierający tekst

Opis wygenerowany automatycznie

Rys.5.2.1. Klasa Generator.

Dzięki tej klasie jesteśmy w stanie kolejno generować dane dla trzech opcji wyboru.

Jako przykład generowania danych niech posłuży nam tabela Obiekt oraz Przedmioty\_przeglądu:

Obraz zawierający tekst

Opis wygenerowany automatycznie

Rys5.2.2.