

Dokumentacja Projektu

Ksawery Józefowski

24 stycznia 2025

Spis treści

1	Spis użytych technologii	2
2	Lista plików i opis ich zawartości	2
3	Kolejność i sposób uruchamiania plików	2
4	Schemat projektu bazy danych	3
5	Lista zależności funkcyjnych z wyjaśnieniem	4
5.1	Pracownicy	4
5.2	Rodzaje Wycieczek	4
5.3	Kierunki Wycieczek	4
5.4	Wycieczki	4
5.5	Klienci	4
5.6	Udziały Wyjazdów	5
5.7	Zrealizowane Wyjazdy	5
5.8	Wyплаты Pracowników	5
6	Uzasadnienie, że baza danych jest w postaci <i>EKNF</i>	6
6.1	Analiza relacji	6
6.2	Wniosek	6
7	Najtrudniejsze elementy projektu	6

1 Spis użytych technologii

W projekcie wykorzystano następujące technologie:

- **Python** – do implementacji skryptów generujących dane i komunikacji z baza danych.
- **MySQL** – system zarządzania relacyjna baza danych.
- **mysql.connector** - biblioteka Pythona do integracji z baza danych.
- **Faker** – biblioteka Pythona do generowania danych testowych.
- **Unidecode** – biblioteka Pythona do obsługi znaków.
- **DBI** - pakiet pomagający połączyć R z systemami zarządzania bazami danych.
- **RMySQL** - pakiet do integracji R z MySQL.
- **knitr** - pakiet R do generowania Raportów.
- **RMarkdown** – do analizy danych i generowania raportów.
- **DrawSQL** - do wizualizacji schematu bazy danych.
- **Overleaf** - do tworzenia dokumentacji w LaTeX.

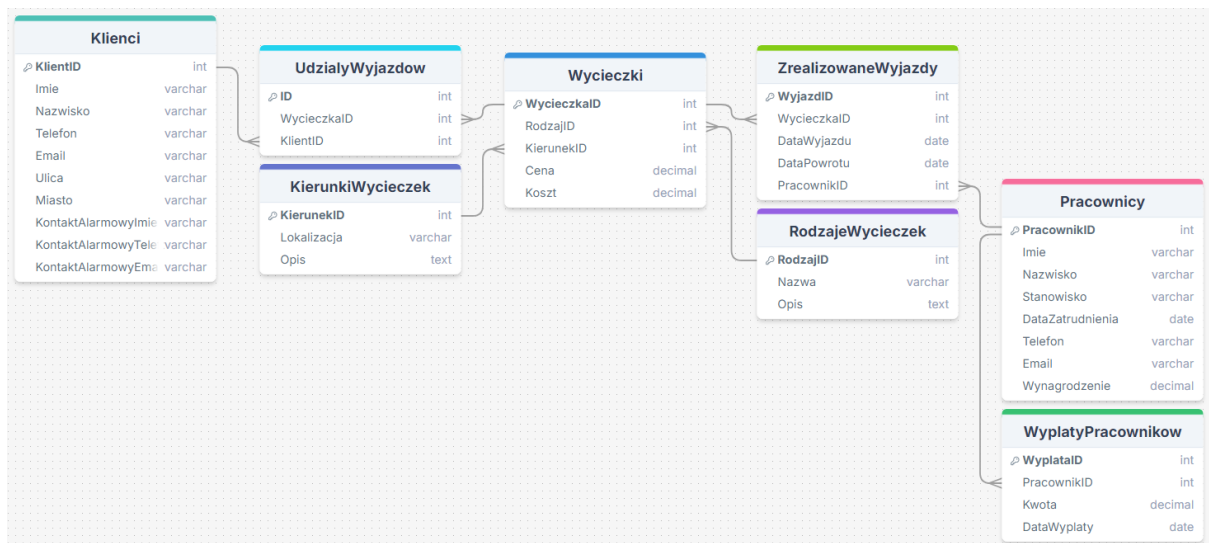
2 Lista plików i opis ich zawartości

- `main.py` – skrypt w języku Python, odpowiedzialny za generowanie danych testowych i wypełnianie bazy danych.
- `schemat.sql` – plik SQL definiujący strukturę bazy danych.
- `Raport_Projekt.rmd` – kod źródłowy raportu analitycznego w formacie RMarkdown.
- `Raport_Projekt.pdf` – wygenerowany raport w formacie PDF.
- `Dokumentacja.tex` – kod źródłowy dokumentacji w LaTeX.
- `Dokumentacja.pdf` – wygenerowana dokumentacja w formacie PDF.
- `schematbazy.png` - obraz wykorzystywany przez dokumentację.

3 Kolejność i sposób uruchamiania plików

1. Uruchomienie pliku `schemat.sql` w celu utworzenia struktury bazy danych.
2. Wykonanie skryptu `main.py`, który wypełnia bazę danych danymi testowymi.
3. Analiza danych za pomocą kodu w `Raport_Projekt.rmd` i wygenerowanie raportu PDF.

4 Schemat projektu bazy danych



5 Lista zależności funkcyjnych z wyjaśnieniem

5.1 Pracownicy

- Zależności funkcyjne:
 $\text{PracownikID} \rightarrow \{\text{Imie, Nazwisko, Stanowisko, DataZatrudnienia, Telefon, Email, Wynagrodzenie}\}$
- Wyjaśnienie:
PracownikID jest kluczem głównym i identyfikuje każdego pracownika.

5.2 Rodzaje Wycieczek

- Zależności funkcyjne:
 $\text{RodzajID} \rightarrow \{\text{Nazwa, Opis}\}$
- Wyjaśnienie:
RodzajID jest kluczem głównym tabeli i identyfikuje każdy rodzaj wycieczki.

5.3 Kierunki Wycieczek

- Zależności funkcyjne:
 $\text{KierunekID} \rightarrow \{\text{Lokalizacja, Opis}\}$
- Wyjaśnienie:
KierunekID jest kluczem głównym i identyfikuje lokalizacje i opis kierunku wycieczki.

5.4 Wycieczki

- Zależności funkcyjne:
 $\text{WycieczkaID} \rightarrow \{\text{RodzajID, KierunekID, Cena, Koszt}\}$
 $\text{RodzajID} \rightarrow \{\text{Nazwa, Opis}\}$ (*poprzez klucz obcy*)
 $\text{KierunekID} \rightarrow \{\text{Lokalizacja, Opis}\}$ (*poprzez klucz obcy*)
- Wyjaśnienie:
WycieczkaID jest kluczem głównym i identyfikuje atrybuty związane z danymi wycieczkami.

5.5 Klienci

- Zależności funkcyjne:
 $\text{KlientID} \rightarrow \{\text{Imie, Nazwisko, Telefon, Email, Ulica, Miasto, KontaktAlarmowyImie, KontaktAlarmowyTelefon, KontaktAlarmowyEmail}\}$
- Wyjaśnienie:
KlientID jest kluczem głównym i identyfikuje każdego klienta.

5.6 Udziały Wyjazdów

- Zależności funkcyjne:
ID \rightarrow {WycieczkaID, KlientID}
WycieczkaID \rightarrow {RodzajID, KierunekID, Cena, Koszt} (*poprzez klucz obcy*)
KlientID \rightarrow {Imie, Nazwisko, Telefon, Email, Ulica, Miasto, KontaktAlarmowy-Imie, KontaktAlarmowyTelefon, KontaktAlarmowyEmail} (*poprzez klucz obcy*)
- Wyjaśnienie:
ID jest kluczem głównym i identyfikuje identyfikatory wycieczki i klienta biorącego w niej udział.

5.7 Zrealizowane Wyjazdy

- Zależności funkcyjne:
WyjazdID \rightarrow {WycieczkaID, DataWyjazdu, DataPowrotu, PracownikID}
WycieczkaID \rightarrow {RodzajID, KierunekID, Cena, Koszt} (*poprzez klucz obcy*)
PracownikID \rightarrow {Imie, Nazwisko, Stanowisko, DataZatrudnienia, Telefon, Email, Wynagrodzenie} (*poprzez klucz obcy*)
- Wyjaśnienie:
WyjazdID jest kluczem głównym i identyfikuje szczegóły zrealizowanego wyjazdu.

5.8 Wyплаты Pracowników

- Zależności funkcyjne:
WyplataID \rightarrow {PracownikID, Kwota, DataWyplaty}
PracownikID \rightarrow {Imie, Nazwisko, Stanowisko, DataZatrudnienia, Telefon, Email, Wynagrodzenie} (*poprzez klucz obcy*)
- Wyjaśnienie:
WyplataID jest kluczem głównym i identyfikuje szczegóły wypłaty dla pracownika.

6 Uzasadnienie, że baza danych jest w postaci *EKNF*

6.1 Analiza relacji

- Wszystkie relacje są w *3NF*.
 - Wszystkie atrybuty są w pełni zależne od klucza głównego. Nie występują częściowe zależności funkcyjne, ponieważ każda relacja posiada jednoznaczny klucz główny, który determinuje wszystkie pozostałe atrybuty.
 - Nie występują tranzytywne zależności między atrybutami niekluczowymi. Jeśli istnieją powiązania między tabelami, są one reprezentowane przez klucze obce, które jednoznacznie wskazują odpowiednie dane w innych relacjach.
- W każdej relacji każda zależność funkcyjna $X \rightarrow Y$ jest spełniona przez nadklucz:
 - Zależności funkcyjne w każdej relacji odwołują się do kluczy głównych lub kluczy obcych, które jednoznacznie identyfikują wszystkie atrybuty.
 - Nie istnieją żadne atrybuty zależne od części klucza głównego.

6.2 Wniosek

Baza danych spełnia wymagania postaci normalnej (*EKNF*), ponieważ każda relacja:

- Nie zawiera częściowych zależności funkcyjnych.
- Każda zależność funkcyjna $X \rightarrow Y$ jest spełniona przez nadklucz.

7 Najtrudniejsze elementy projektu

- Zachowanie poprawności wszystkich wprowadzanych danych.
- Zapewnienie unikalności danych takich jak e-maile i numery telefonów.
- Utrzymanie spójności bazy danych przy złożonych zależnościach.