|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Graphical user interface, text  Description automatically generated with medium confidence | **Politechnika Bydgoska**  Wydział Telekomunikacji,  Informatyki i Elektrotechniki | | Logo, company name  Description automatically generated | |
| **Przedmiot** | Podstawy Baz Danych | | | |
| **Prowadzący** | mgr inż. Mateusz Szczepański | | | |
| **Temat** | Wprowadzenie do programu MySQL Workbench i składni MySQL | | | |
| **Imię i nazwisko** | Jędrzej Sikorski | | | |
| **Nr albumu** | 118854 | **Grupa** | | 5 |
| **Nr lab.** | 2 | **Data wykonania** | | 03.09.2022 |

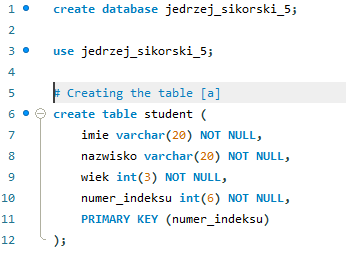
1. **Cel i zakres ćwiczenia**

Sprawdzenie poziomu wiedzy z podstawowych komend języka MySQL i przeprowadzania operacji na bazach danych.

1. **Realizacja zadań do samodzielnego wykonania**

**A)**

*Utwórz tabelę „student” z kolumnami „imie”, „nazwisko”, „wiek” oraz „numer\_indeksu”. Dobierz typ danych dla każdej kolumny i uzasadnij swój wybór. Zablokuj możliwość wprowadzania wartości NULL dla każdej kolumny. Ustaw kolumnę „numer\_indeksu” jako klucz główny.*

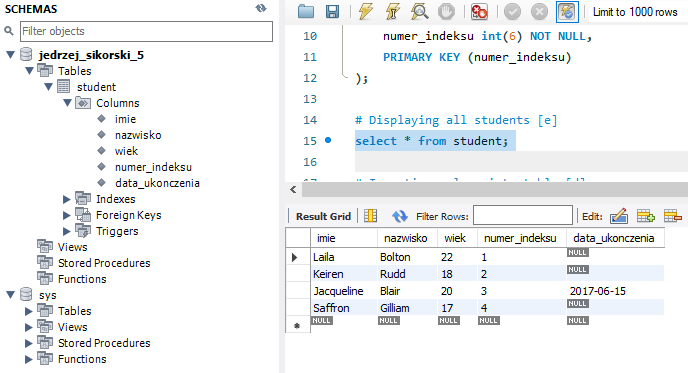


Tworzenie tabeli za pomocą komendy CREATE TABLE *name.* Typ danychkolumn imie i nazwisko przechowuje ciągi znaków. Typ danych użyty w kolumnach wiek i numer\_indeksu, przechowuje liczby. NOT NULL sprawia, że wiersz nie może być pusty, a PRIMARY KEY ustawia podane pole na klucz główny.

**B & C)**

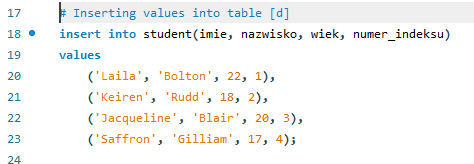
*Wyświetl tabele w bazie danych. &*

*Wyświetl informacje o stworzonej tabeli.*



**D)**

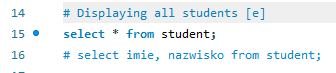
*Wymyśl 4 różnych studentów i dodaj ich do tabeli. Niech mają różny wiek. Czy jesteś w stanie to zrobić w jednym zapytaniu?*



Poleceniem insert into *name* dodamy do tabeli wiersze i wypełnimy je wartościami podanymi w nawiasach.

**E)**

*Wypisz wszystkich studentów z tabeli.*



Pierwsze polecenie zwróci całą tabelę, drugie tylko pola imie i nazwisko z tabeli student.

**F)**

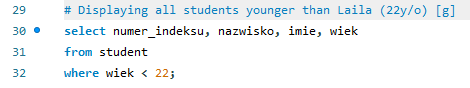
*Wypisz imiona wszystkich studentów z tabeli posortowane alfabetycznie.*



Polecenie zwraca imiona z tabeli student i sortuje je alfabetycznie względem pola imie.

**G)**

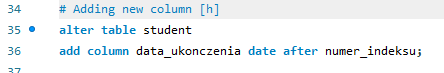
*Wypisz wszystkich studentów młodszych od najstarszego dodanego przez siebie studenta. Wprowadź wartość bezpośrednio.*



Zawarcie WHERE w zapytaniu pozwala dokładniej określić szukane wartości. W tym przypadku zapytanie szuka w tabeli student pól wiek z wartością mniejszą od 22.

**H)**

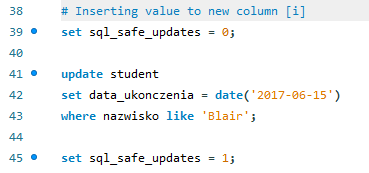
*Dodaj to istniejącej tabeli „student” kolumnę „data\_ukonczenia”. Potwierdź rezultat.*

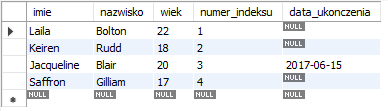


Alter table pozwala na dodanie do tabeli kolejnej kolumny. Za pomocą polecenia dodano kol. data\_ukonczenia o typie danych date. Kolumnę dodano za kolumną numer\_indeksu.

**I)**

*Zaktualizuj wartość kolumny „data\_ukonczenia” dla wybranego przez siebie studenta. Potwierdź rezultat.*

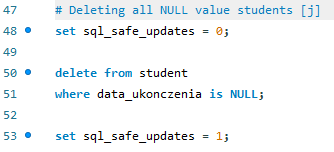


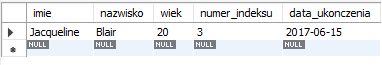


Update pozwala zaktualizować dane w tabeli. Do wykonania polecenia potrzebne było wyłączenie bezpiecznego trybu w tabeli.

**J)**

*Usuń wszystkich studentów z wartością NULL w kolumnie „data\_ukonczenia” jednym zapytaniem. W tym celu konieczne będzie wyłączenie trybu bezpiecznego. Aktywuj go ponownie po usunięciu studentów. Potwierdź rezultat.*





Wyczyszczono wszystkie wiersze, w których kolumna data\_ukonczenia przybierała wartość NULL.

1. **Podsumowanie**

Zadanie pozwoliło mi na powtórzenie i przypomnienie sobie podstawowej składni zapytań MySQL, a także pozwoliło mi zapoznać się z narzędziem MySQL Workbench.

Podczas wykonywania zadań stworzyłem nową bazę danych, dodałem do niej wymagane kolumny i wypełniłem wiersze danymi studentów. Następnie uaktualniłem i zmieniłem odpowiednie pola w bazie danych, a następnie usunąłem z niej kilku z dodanych studentów.