

- Opis problemu i opis funkcjonalności udostępnianej przez API
 - Celem API jest umożliwienie obsługi struktury organizacyjnej firmy będącej strukturą hierarchiczną. Elementami w hierarchii są pracownicy wraz z ich danymi personalnymi.
- Opis typów danych oraz metod (funkcji) udostępnionych w ramach API
 - Typy tabeli przechowujące dane:
 - Employee
 - Level hierarchyid
 - Name nvarchar(30)
 - Position nvarchar(30)
 - Salary Int
 - Udostępnione funkcje:
 - AddEmployee
 - RemoveEmployee
 - RemoveAllEmployees
 - GetEmployee
 - GetEmployeeWithChildren
 - GetAllEmployees
 - GetMaxSalary
 - GetAverageSalary
 - GetSumSalary
- Opis implementacji
 - Api udostępnia procedury wykonujące operacje na rekordach tabeli pracowników (Employees) w bazie danych. Jest ono zaimplementowane w postaci klasy w języku C# wykonującej zapytania SQL do bazy danych uruchamiających procedury wykonujące żądane operacje. W procedurach wykonywane są instrukcje SQL korzystające z dobrodziejstw użytych typów danych jak np. hierarchyid
- Prezentacja przeprowadzonych testów jednostkowych
 - TestGetEmployees - testuje pobieranie wszystkich pracowników z bazy - sprawdza, czy ich liczba jest poprawna
 - TestGetEmployee - testuje pobieranie informacji o pracowniku na konkretnej pozycji - sprawdza, czy zostały pobrane poprawne dane
 - TestGetEmployeeWithChildren - testuje pobieranie pracownika wraz ze wszystkimi jego podwładnymi - sprawdza, czy liczba pobranych pracowników jest poprawna
 - TestAddEmployee - testuje dodawanie pracownika - sprawdzenie, czy liczba pracowników się zwiększyła, oraz pobranie danych i sprawdzenie ich poprawności
 - TestRemoveEmployee - testuje usuwanie pracownika - sprawdzenie, czy liczba pracowników się zmniejszyła
 - TestGetMaxSalary - testuje pobieranie informacji o maksymalnej wypłacie
 - TestGetAverageSalary - testuje pobieranie informacji o średniej wypłacie
 - TestGetSumSalary - testuje pobieranie informacji o sumie wypłat
- Podsumowanie, wnioski

- Typ hierarchiid udostępnia wiele funkcji, które są przydatne i normalnie przy rozwiązywaniu tego typu problemów musiałyby być dodatkowo zaimplementowane.
- Literatura:
 - https://www.plukasiewicz.net/Artykuly/ADO_NET
 - <https://docs.microsoft.com/pl-pl/aspnet/core/fundamentals/?view=aspnetcore-3.1&tabs=windows>
 - <https://softwarehut.com/blog/tech/hierarchyid-entity-framework>
 - <https://kariera.future-processing.pl/blog/hierarchy-in-the-entity-framework-6-wi-th-the-hierarchyid-type/>
 - <https://www.c-sharpcorner.com/article/a-basic-introduction-of-unit-test-for-beginners/>
- Kod źródłowy wszystkich skryptów T-SQL wykorzystanych w projekcie (np. tworzenie testowej bazy danych) oraz aplikacji w C#.ul style="list-style-type: none;">- W załączonym pliku .sql