Laboratorium 1 : Oracle PL/SQL

Bartłomiej Kozera

# Tabele

## Tabela person

Obraz zawierający tekst

Opis wygenerowany automatycznie

## Tabela country

Obraz zawierający tekst

Opis wygenerowany automatycznie

## Tabela trip wraz z kluczem obcym country

Obraz zawierający tekst

Opis wygenerowany automatycznie

## Tabela reservation wraz z kluczami obcymi i sprawdzaniem poprawności danych

Obraz zawierający tekst

Opis wygenerowany automatycznie

# Dane poszczególnych tabel

Wszystkie dane dodawane były do tabel za pomocą poleceni insert into [nazwa tabeli] (param\_1, param\_ 2, …, param\_ n) values (wartość\_1, wartość\_2, …, wartość\_n).

## Dane tabeli person

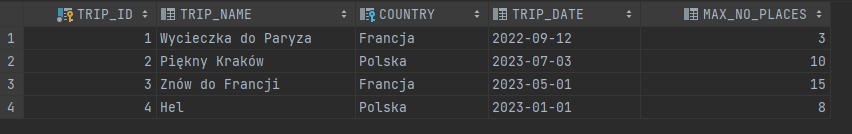
Obraz zawierający tekst, sprzęt elektroniczny, czarny, ekran

Opis wygenerowany automatycznie

## Dane tabeli country



## Dane tabeli trip



## Dane tabeli reservation

Obraz zawierający tekst, wewnątrz, sprzęt elektroniczny, komputer

Opis wygenerowany automatycznie

# Widoki

## Widok Reservations

Obraz zawierający tekst

Opis wygenerowany automatycznie

Obraz zawierający tekst, monitor, sprzęt elektroniczny, czarny

Opis wygenerowany automatycznie

## Widok Trips

Obraz zawierający tekst

Opis wygenerowany automatycznie

Obraz zawierający tekst

Opis wygenerowany automatycznie

## Widok AvailableTrips

Obraz zawierający tekst

Opis wygenerowany automatycznie

Obraz zawierający tekst

Opis wygenerowany automatycznie

## Wnioski

Tworzenie widoku przypomina jego konstrukcję w MsSQL, jedyną sporą różnicą konstrukcyjną która zauważyłem jest sformułowanie ‘create or replace’ które pozwala nam na stworzenie lub jeśli taki widok istnieje, uaktualnienie go. Widoki z podpunktu 3.2. oraz 3.3. nie różnią się wyjściem ze względu na dane, nie ma wycieczki na której nie było by miejsc.

# Procedury

## Obiekty

Aby w PL/SQL stworzyć procedurę która zwraca nam tabelę z odpowiadającymi nam danymi, należy stworzyć 2 typy pól. Pierwszym z nich jest obiekt, który będzie zawierał odpowiednie pola, wraz z poprawnymi typami tych pól oraz tabele tychże obiektów. Kolejna różnica między tym językiem a MsSQL, gdzie były 2 typy procedur, jedne zwracały wartość, drugie tabele, tutaj procedury tylko zwracają wartość, stąd potrzeba tworzenie obiektów oraz ich tabel.

Obraz zawierający tekst

Opis wygenerowany automatycznie

## Funkcja TripParticipants

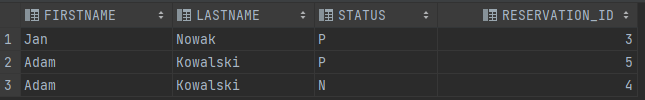
Obraz zawierający tekst

Opis wygenerowany automatycznie

Po wykonaniu tego polecenia:



Wynikami są:



## Funkcja PersonReservation

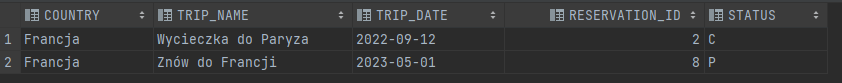
Obraz zawierający tekst

Opis wygenerowany automatycznie

Po wykonaniu tego polecenia:



Wynikami są:



## Funkcja FAvailableTrips

Obraz zawierający tekst

Opis wygenerowany automatycznie

Po wykonaniu tego polecenia:



Wynikami są:



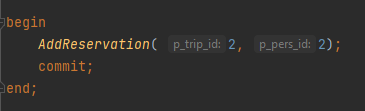
# Procedury modyfikujące

## Procedura AddReservation

Obraz zawierający tekst

Opis wygenerowany automatycznie

Po wykonania poniższego polecenia



Wyniki prawidłowo dodają się do tabeli z rezerwacjami

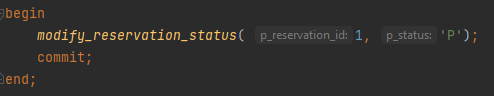


## Procedure modify\_reservation\_status

Obraz zawierający tekst

Opis wygenerowany automatycznie







Jak widać procedura poprawnie modyfikuje status zamówienia

## Procedura modify\_no\_places

Obraz zawierający tekst

Opis wygenerowany automatycznie



Po wykonaniu poniższego fragmentu kodu.

Obraz zawierający tekst

Opis wygenerowany automatycznie

Wartość pola max\_no\_places zmienia się.



## Wnioski

Procedury praktycznie nie różniące się od tych w ms SQL, Możemy zauważyć ciekawy sposób na wywoływanie błędów jako brak wartości dla zadeklarowanych przez nas argumentów pomocniczych. Nie da się stworzyć argumentów w sekcji begin … end.

# Tabela log

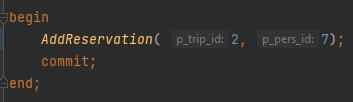
Kod do stworzenia tej tabeli był podany w raz z materiałami do zajęć, nie będę go tutaj umieszczał.

## Modyfikacja procedury AddReservation

Obraz zawierający tekst

Opis wygenerowany automatycznie

Do powyższej procedury dodano dodawanie danych do tabeli log.







Po wykonanie AddReservation, dane poprawnie dodają się do naszych tabel.

## Modyfikacja procedury modify\_reservation\_status

## Obraz zawierający tekst Opis wygenerowany automatycznie

W powyższej procedurze jedyną zmianą do 5.2 jest dodanie nowego rekordu do log.

Obraz zawierający tekst

Opis wygenerowany automatycznie



## Wnioski

Tabela log jest ciekawym dodatkiem monitorującym działania w naszej bazie, jednak modyfikacja jej w procedurach może być uciążliwa. Lepszym pomysłem jest wydziedziczenie tego do triggera działającego przy zmianie danych w tabeli Reservations.

# Zmiana podejścia do modyfikacji

## Triggery

Obraz zawierający tekst

Opis wygenerowany automatycznie

tr\_update\_log dodaje rekord do tabeli log, za każdym razem kiedy dodajemy rekord do tabeli, lub zmieniamy jakieś dane.

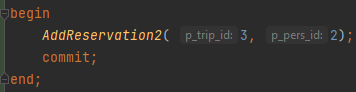
tr\_dell\_log odpowiada za uniemożliwienie usunięcia rekordu z tabeli reservations.

## Procedura AddReservation2

Obraz zawierający tekst

Opis wygenerowany automatycznie

Po wykonaniu polecenia



Otrzymujemy w tabeli Reservations



Oraz w tabeli log



## Procedura Modify\_reservation\_status

Obraz zawierający tekst

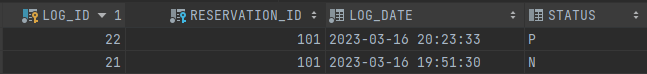
Opis wygenerowany automatycznie

Obraz zawierający tekst

Opis wygenerowany automatycznie

Po wywołaniu powyższej funkcji, modyfikowana jest wartość w tabeli reservations, oraz dodawany jest rekord w tabeli log.





## Wnioski

Takie rozwiązanie odrobinkę zmniejsza nam ilość kodu w procedurach oraz pozwala na utrzymanie bardziej czytelnej struktury kodu. Triggery same w sobie nie różnią się znacząco z tymi z MS Sql.

# Kontrola dostępności miejsc w trigerach

## Trigger tr\_controll\_places\_add

Obraz zawierający tekst

Opis wygenerowany automatycznie

Po wprowadzeniu tego triggera, procedura dodająca dane będzie wyglądała następująco:

Obraz zawierający tekst

Opis wygenerowany automatycznie

Przy próbie dodania rezerwacji do niedostępnej już wycieczki pojawia nam się oczekiwana odmowa

Obraz zawierający tekst

Opis wygenerowany automatycznie

Natomiast dodanie danych do istniejącej tabelii powiedzie się:

Obraz zawierający tekst

Opis wygenerowany automatycznie



## Tr\_controll\_places\_update

Obraz zawierający tekst

Opis wygenerowany automatycznie

Po dodaniu tego triggera procedura modyfikująca rezerwacje może wyglądać następująco:

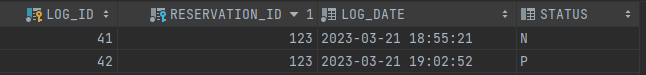
Obraz zawierający tekst

Opis wygenerowany automatycznie

Zmiana statusu wycieczki, której status możemy zmieniać jest dale j bez problemowy, natomiast zmiana statusu na wycieczkę bez wolnych miejsc jest niemożliwy.

Obraz zawierający tekst

Opis wygenerowany automatycznie



## Wnioski

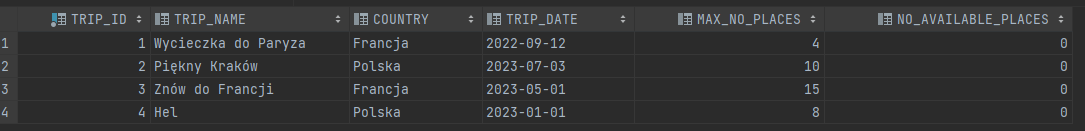
Dodanie tych triggerów sprawiło że kod procedur stał się czytelniejszy, oraz uzyskaliśmy jaśniejszy podział kodu na część modyfikującą i część sprawdzającą.

# Dodanie pola no\_available\_places

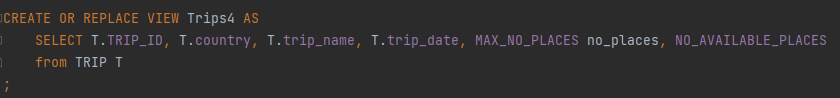
Kolumnę dodaje za pomocą polecenia



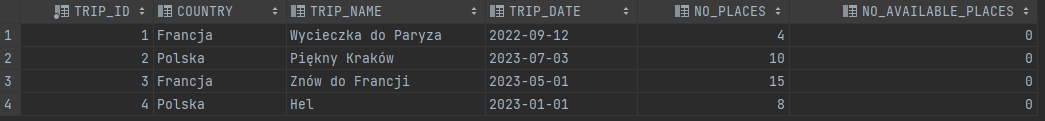
Ustawiając wartość domyślną na 0.



Po dodaniu tego pola można zmodyfikować widoki Trips oraz AvailableTrips







Obraz zawierający tekst

Opis wygenerowany automatycznie





Po stworzeniu procedury liczącej miejsca dostępne na danej wycieczce

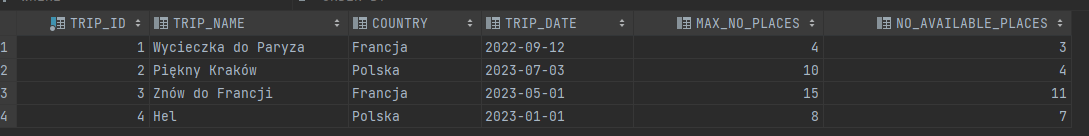
Obraz zawierający tekst

Opis wygenerowany automatycznie

I wykonaniu jej na wszystkich elementach, np.

Obraz zawierający tekst

Opis wygenerowany automatycznie



## 9.1 Modyfikacja funkcji

Obraz zawierający tekst

Opis wygenerowany automatycznie





## 9.2. Modyfikacja Procedur modyfikujących dane

Obraz zawierający tekst

Opis wygenerowany automatycznie

Po wykonaniu tego kodu, dodana została nowa wartość w tabeli Reservations oraz została na nowo obliczona wartość dodanego pola w tabeli Trip

Obraz zawierający tekst, monitor, komputer, elektronika

Opis wygenerowany automatycznie

Obraz zawierający tekst, elektronika, zrzut ekranu

Opis wygenerowany automatycznie

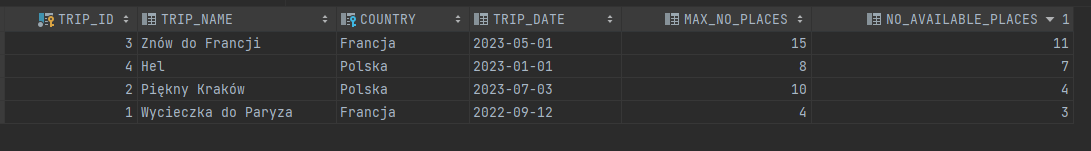
Po wykonaniu poniższego kawałka kodu, zmienia się zarówno status rezerwacji, jak i ilość miejsc wolnych na daną wycieczkę.

Obraz zawierający tekst

Opis wygenerowany automatycznie

Obraz zawierający tekst, w pomieszczeniu, elektronika, komputer

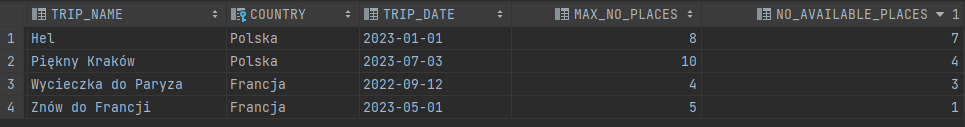
Opis wygenerowany automatycznie



Po lekkiej korekcie, można też zmienić ilość miejsc na dana wycieczkę

Obraz zawierający tekst

Opis wygenerowany automatycznie



# Zmiana podejścia do pola no\_available\_places

## Dodanie triggera na dodanie rezerwacji

Obraz zawierający tekst, zrzut ekranu, ekran

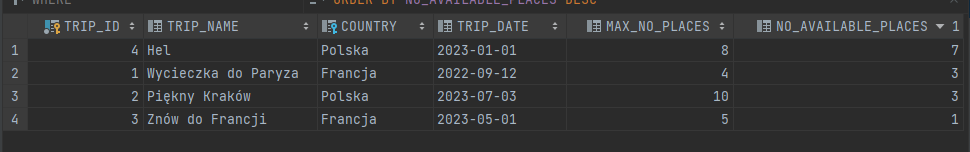
Opis wygenerowany automatycznie

Modyfikuję procedurę AddReservation

Obraz zawierający tekst

Opis wygenerowany automatycznie

Po jej wykonaniu otrzymuje w tabeli Trip



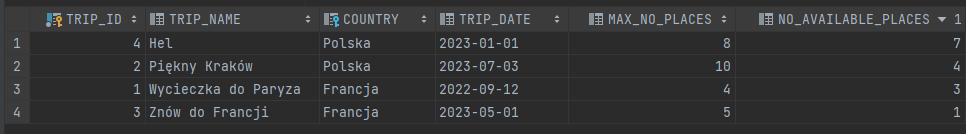
## Trigger na modyfikację danych

Obraz zawierający tekst

Opis wygenerowany automatycznie

Obraz zawierający tekst

Opis wygenerowany automatycznie



## Trigger na zmianę liczb miejsc w wycieczce

Obraz zawierający tekst

Opis wygenerowany automatycznie

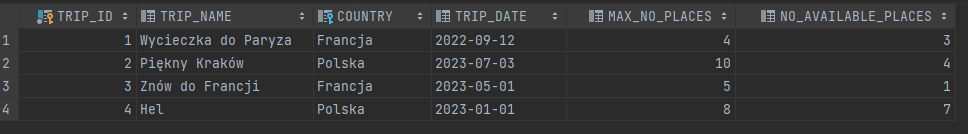
Obraz zawierający tekst

Opis wygenerowany automatycznie

Obraz zawierający tekst

Opis wygenerowany automatycznie

Uzyskano oczekiwany błąd.



## Wnioski

Dodanie powyższych triggerów dodaje przejrzystości kodu, oraz zmniejsza objętość procedur.