

Raport z ćwiczeń dotyczących Oracle PL/SQL

Krzysztof Hardek

2. Wypełnianie tabeli przykładowymi danymi danymi

2.1. 10 Osób

```
INSERT INTO osoby (imie, nazwisko, pesel, kontakt)
VALUES('Adam', 'Kowalski', '87654321', 'tel: 6623');
```

```
INSERT INTO osoby (imie, nazwisko, pesel, kontakt)
VALUES('Jan', 'Nowak', '12345678', 'tel: 2312, dzwonić po 18.00');
```

```
insert into osoby (imie, nazwisko, pesel, kontakt)
values ('Karol', 'Wojtyla', '56908317', 'tel: 3865');
```

```
insert into osoby (imie, nazwisko, pesel, kontakt)
values ('Adam', 'Mickiewicz', '34213431', 'tel: 9163');
```

```
insert into osoby (imie, nazwisko, pesel, kontakt)
values ('Fryderyk', 'Chopin', '42622586', 'tel: 4799');
```

```
insert into osoby (imie, nazwisko, pesel, kontakt)
values ('Jan', 'Matejko', '51816061', '8676');
```

```
insert into osoby (imie, nazwisko, pesel, kontakt)
values ('Maria', 'Sklodowska', '67646380', 'tel: 3473');
```

```
insert into osoby (imie, nazwisko, pesel, kontakt)
values ('Mikolaj', 'Kopernik', '27940067', 'tel: 4312');
```

```
insert into osoby (imie, nazwisko, pesel, kontakt)
values ('Mariusz', 'Pudzianowski', '48834964', 'tel: 4732');
```

```
insert into osoby (imie, nazwisko, pesel, kontakt)
values ('Dawid', 'Sliwinski', '52488086', 'tel: 5128');
```

2.2. 5 wycieczek

```
INSERT INTO wycieczki (nazwa, kraj, data, opis, liczba_miejsc)
VALUES ('Wycieczka do Paryża','Francja',TO_DATE('2016-01-01'),'Ciekawa
wycieczka ...',3);
```

```
INSERT INTO wycieczki (nazwa, kraj, data, opis, liczba_miejsc)
VALUES ('Piękny Kraków','Polska',TO_DATE('2017-02-03'),'Najciekawa wycieczka
... ',2);
```

```
INSERT INTO wycieczki (nazwa, kraj, data, opis, liczba_miejsc)
VALUES ('Słoneczny Wrocław','Polska',TO_DATE('2025-07-18'),'Super Zabawa',3);
```

```
INSERT INTO wycieczki (nazwa, kraj, data, opis, liczba_miejsc)
VALUES ('Ciekawy Berlin','Niemcy',TO_DATE('2023-05-06'),'Zabytki Historyczne',1);
```

```
INSERT INTO wycieczki (nazwa, kraj, data, opis, liczba_miejsc)
VALUES ('Wieliczka','Polska',TO_DATE('2017-03-03'),'Zadziwiająca kopalnia ...',2);
```

2.3. 11 rezerwacji

```
INSERT INTO rezerwacje(id_wycieczki, id_osoby, status)
VALUES (21,1,'N');
```

```
INSERT INTO rezerwacje(id_wycieczki, id_osoby, status)
VALUES (22,2,'P');
```

```
INSERT INTO rezerwacje(id_wycieczki, id_osoby, status)
VALUES (21,41,'P');
```

```
INSERT INTO rezerwacje(id_wycieczki, id_osoby, status)
VALUES (21,42,'Z');
```

```
INSERT INTO rezerwacje(id_wycieczki, id_osoby, status)
VALUES (22,43,'A');
```

```
INSERT INTO rezerwacje(id_wycieczki, id_osoby, status)
VALUES (23,44,'N');
```

```
INSERT INTO rezerwacje(id_wycieczki, id_osoby, status)
```

```
VALUES (23,45,'P');
```

```
INSERT INTO rezerwacje(id_wycieczki, id_osoby, status)
VALUES (41,46,'A');
```

```
INSERT INTO rezerwacje(id_wycieczki, id_osoby, status)
VALUES (41,47,'Z');
```

```
INSERT INTO rezerwacje(id_wycieczki, id_osoby, status)
VALUES (41,48,'A');
```

```
INSERT INTO rezerwacje(id_wycieczki, id_osoby, status)
VALUES (42,48,'A');
```

Wynik wypełnienia danymi

	ID_OSOBY	IMIE	NAZWISKO	PESEL	KONTAKT
1	1	Adam	Kowalski	87654321	tel: 6623
2	2	Jan	Nowak	12345678	tel: 2312, dzwonić po 18.00
3	41	Karol	Wojtyla	56908317	tel: 3865
4	42	Adam	Mickiewicz	34213431	tel: 9163
5	43	Fryderyk	Chopin	42622586	tel: 4799
6	44	Jan	Matejko	51816061	8676
7	45	Maria	Skłodowska	67646380	tel: 3473
8	46	Mikołaj	Kopernik	27940067	tel: 4312
9	47	Mariusz	Pudziański	48834964	tel: 4732
10	48	Dawid	Sliński	52488086	tel: 5128

	NR_REZERWACJI	ID_WYCIĘZKI	ID_OSOBY	STATUS
1	21	21	1	N
2	22	22	2	P
3	41	21	41	P
4	42	21	42	Z
5	44	22	43	A
6	45	23	44	N
7	46	23	45	P
8	47	41	46	A
9	48	41	47	Z
10	49	41	48	A
11	50	42	48	Z

	ID_WYCIECZKI	NAZWA	KRAJ	DATA	OPIS	LICZBA_MIEJSC	LICZBA_WOLNYCH_MIEJSC
1	21	Wycieczka do Paryża	Francja	2016-01-01 00:00:00	Ciekawa wycieczka ...	50	<null>
2	22	Piękny Kraków	Polska	2017-02-03 00:00:00	Najciekawa wycieczka ...	2	<null>
3	23	Wieliczka	Polska	2017-03-03 00:00:00	Zadziwiająca kopalnia ...	2	<null>
4	41	Słoneczny Wrocław	Polska	2025-07-18 00:00:00	Super Zabawa	3	<null>
5	42	Ciekawy Berlin	Niemcy	2023-05-06 00:00:00	Zabytki Historyczne	1	<null>

3. Tworzenie widoków

Funkcje pomocnicze użyte w widokach

3.1. Funkcja obliczająca liczbę zajętych miejsc

```
create or replace function LiczbaZajetychMiejsc (id in number) return number
is
    zajeteMiejsc number;
begin
    select count(*)
        into zajeteMiejsc
    from REZERWACJE
    where STATUS not like 'A' and id_wycieczki = id;
    return zajeteMiejsc;
end;
```

"LICZBAZAJETYCHMIEJSC(21)"	
1	3

3.2. Funkcja obliczająca liczbę wolnych miejsc

```
create or replace function LiczbaWolnychMiejsc (id in number) return number
is
    limitMiejsc number;
```

```

begin
  select liczba_miejsc
  into limitMiejsc
  from WYCIECZKI
  where id_wycieczki = id;
  return limitMiejsc - LiczbaZajetychMiejsc(id);
end;

```

	"LICZBAWOLNYCHMIEJSC(21)"	
1		47

Widoki

3.3. RzerwacjeWszystkie(kraj,data, nazwa_wycieczki, imie,nazwisko,status_rezerwacji)

```

create view RezerwacjeWszystkie
as
select
  w.KRAJ,
  w.DATA,
  w.NAZWA,
  o.IMIE,
  o.NAZWISKO,
  r.STATUS
from WYCIECZKI w
join REZERWACJE r
on w.ID_WYCIECZKI = r.ID_WYCIECZKI
join OSOBY o on r.ID_OSOBY = o.ID_OSOBY;

```

	KRAJ	DATA	NAZWA	IMIE	NAZWISKO	STATUS
1	Francja	2016-01-01 00:00:00	Wycieczka do Paryza	Adam	Kowalski	N
2	Polska	2017-02-03 00:00:00	Piękny Kraków	Jan	Nowak	P
3	Francja	2016-01-01 00:00:00	Wycieczka do Paryza	Karol	Wojtyła	P
4	Francja	2016-01-01 00:00:00	Wycieczka do Paryza	Adam	Mickiewicz	Z
5	Polska	2017-02-03 00:00:00	Piękny Kraków	Fryderyk	Chopin	A
6	Polska	2017-03-03 00:00:00	Wieliczka	Jan	Matejko	N
7	Polska	2017-03-03 00:00:00	Wieliczka	Maria	Skłodowska	P
8	Polska	2025-07-18 00:00:00	Słoneczny Wrocław	Mikołaj	Kopernik	A
9	Polska	2025-07-18 00:00:00	Słoneczny Wrocław	Mariusz	Pudziański	Z
10	Polska	2025-07-18 00:00:00	Słoneczny Wrocław	Dawid	Słowiński	A
11	Niemcy	2023-05-06 00:00:00	Ciekawy Berlin	Dawid	Słowiński	Z

3.4. RezerwacjePotwierdzone (kraj,data, nazwa_wycieczki, imie, nazwisko,status_rezerwacji)

```
create view RezerwacjePotwierdzone
as
select
    KRAJ,
    DATA,
    NAZWA,
    IMIE,
    NAZWISKO,
    STATUS
from RezerwacjeWszystkie
where STATUS like 'P' or STATUS like 'Z';
```

	KRAJ	DATA	NAZWA	IMIE	NAZWISKO	STATUS
1	Polska	2017-02-03 00:00:00	Piękny Kraków	Jan	Nowak	P
2	Francja	2016-01-01 00:00:00	Wycieczka do Paryża	Karol	Wojtyła	P
3	Francja	2016-01-01 00:00:00	Wycieczka do Paryża	Adam	Mickiewicz	Z
4	Polska	2017-03-03 00:00:00	Wieliczka	Maria	Skłodowska	P
5	Polska	2025-07-18 00:00:00	Słoneczny Wrocław	Mariusz	Pudziański	Z
6	Niemcy	2023-05-06 00:00:00	Ciekawy Berlin	Dawid	Słowiński	Z

3.5. RezerwacjeWPrzyszlosci (kraj,data, nazwa_wycieczki, imie, nazwisko,status_rezerwacji)

```
create view RezerwacjeWPrzyszlosci
as
select
    KRAJ,
    DATA,
    NAZWA,
    IMIE,
    NAZWISKO,
    STATUS
from RezerwacjeWszystkie
where DATA > SYSDATE;
```

	KRAJ	DATA	NAZWA	IMIE	NAZWISKO	STATUS
1	Polska	2025-07-18 00:00:00	Słoneczny Wrocław	Mikołaj	Kopernik	A
2	Polska	2025-07-18 00:00:00	Słoneczny Wrocław	Mariusz	Pudziański	Z
3	Polska	2025-07-18 00:00:00	Słoneczny Wrocław	Dawid	Słowiński	A
4	Niemcy	2023-05-06 00:00:00	Ciekawy Berlin	Dawid	Słowiński	Z

3.6. WycieczkiMiejsca(kraj,data, nazwa_wycieczki,liczba_miejsc, liczba_wolnych_miejsc)

```
create view WycieczkiMiejsca
as
select unique
    kraj,
    data,
    nazwa,
    liczba_miejsc,
    LiczbaWolnychMiejsc(id_wycieczki) as Liczba_Wolnych_Miejsc
from WYCIECZKI
```

	KRAJ	DATA	NAZWA	LICZBA_MIEJSC	LICZBA_WOLNYCH_MIEJSC
1	Niemcy	2023-05-06 00:00:00	Ciekawy Berlin	1	0
2	Polska	2017-02-03 00:00:00	Piękny Kraków	2	1
3	Polska	2017-03-03 00:00:00	Wieliczka	2	0
4	Polska	2025-07-18 00:00:00	Śloneczny Wrocław	3	2
5	Francja	2016-01-01 00:00:00	Wycieczka do Paryża	50	47

3.7. WycieczkiDostepne(kraj,data, nazwa_wycieczki,liczba_miejsc, liczba_wolnych_miejsc)

```
create view WycieczkiDostepne
as
select
    kraj,
    data,
    nazwa,
    liczba_miejsc,
    Liczba_Wolnych_Miejsc
from WycieczkiMiejsca
where Liczba_Wolnych_Miejsc > 0 and data > SYSDATE
```

	KRAJ	DATA	NAZWA	LICZBA_MIEJSC	LICZBA_WOLNYCH_MIEJSC
1	Polska	2025-07-18 00:00:00	Śloneczny Wrocław	3	2

4. Tworzenie procedur/funkcji pobierających dane

Typy pomocnicze użyte w funkcjach

4.1. Rezerwacje

```
create or replace type uczestnicyWycieczki_rekord as object (
    kraj VARCHAR2(50),
    data date,
    nazwa varchar2(100),
    imie varchar2(50),
    nazwisko varchar2(50),
    status char(1)
);
```

```
create or replace type uczestnicyWycieczki_tabela as table of
uczestnicyWycieczki_rekord;
```

4.2. Wycieczki

```
create or replace type wycieczki_rekord as object (
    nazwa varchar2(100),
    kraj varchar2(50),
    data date,
    liczba_miejsc number,
    opis varchar2(200)
);
```

```
create or replace type wycieczki_tabela as table of wycieczki_rekord;
```

Funkcje

4.3. UczestnicyWycieczki (id_wycieczki), funkcja ma zwracać podobny zestaw danych jak widok RezerwacjeWszystkie

```
create or replace function UczestnicyWycieczki (id IN number) return
uczestnicyWycieczki_tabela
is
    uczestnicy uczestnicyWycieczki_tabela;
    czyWycieczkaIstnieje number; -- 0 - fałsz 1 - prawda
begin
    select
        count(*)
    into czyWycieczkaIstnieje
    from WYCIECZKI
    where ID_WYCIECZKI = id;
    if czyWycieczkaIstnieje = 0 then
        raise_application_error(-20000, 'Wycieczka nie istnieje');
```



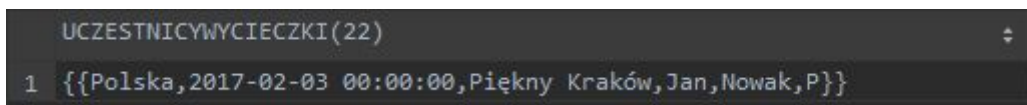
```

end if;

select uczestnicyWycieczki_rekord(w.kraj, w.data, w.nazwa, o.imie,
o.NAZWISKO, r.STATUS)
  bulk collect
into uczestnicy
from REZERWACJE r
join WYCIECZKI w on r.ID_WYCIECZKI = w.ID_WYCIECZKI
join OSOBY o on r.ID_OSOBY = o.ID_OSOBY
where STATUS not like 'A' and r.ID_WYCIECZKI = id;

return uczestnicy;
end;

```



UCZESTNICYWYCIECZKI(22)

1	{{Polska,2017-02-03 00:00:00,Piękny Kraków,Jan,Nowak,P}}
---	--

4.4. RezerwacjeOsoby(id_osoby), funkcja ma zwracać podobny zestaw danych jak widok wycieczki_osoby

```

create or replace function RezerwacjeOsoby(id number) return
uczestnicyWycieczki_tabela
is
  rezerwacjeOsoby uczestnicyWycieczki_tabela;
  czyOsobaIstnieje number; -- 1 - prawda 0 - fałsz
begin
  select
    count(*)
  into czyOsobaIstnieje
  from OSOBY
  where ID_OSOBY = id;

  if czyOsobaIstnieje = 0 then
    raise_application_error(-20001, 'Osoba nie istnieje');
  end if;

  select uczestnicyWycieczki_rekord(w.kraj, w.data, w.nazwa, o.imie, o.nazwisko,
r.status)
    bulk collect
  into rezerwacjeOsoby
  from rezerwacje r

```

```

join osoby o on r.ID_OSOBY = o.ID_OSOBY
join WYCIECZKI w on r.ID_WYCIECZKI = w.ID_WYCIECZKI
where r.ID_OSOBY = id;

return rezerwacjeOsoby;
end;

```

```

REZERWACJEOSOBY(47)
1  {{Polska,2025-07-18 00:00:00,Sloneczny Wrocław,Mariusz,Pudzianowski,Z}}

```

4.5. DostepneWycieczki(kraj, data_od, data_do)

```

create or replace function DostepneWycieczki(kraj_w varchar2, data_od date,
data_do date) return wycieczki_tabela
is
    dostepneWycieczki wycieczki_tabela;
begin
    if data_od > data_do then
        raise_application_error(-20002, 'data_od jest pozniej niz data_do ');
    end if;

    select wycieczki_rekord(nazwa, kraj, data, LICZBA_MIEJSC, opis)
        bulk collect
    into dostepneWycieczki
    from WYCIECZKI
    where data >= data_od and data <= data_do and
LICZBAWOLNYCHMIEJSC(ID_WYCIECZKI) > 0 and KRAJ like kraj_w;

    return dostepneWycieczki;
end;

```

```

DOSTEPNEWYCIECZKI('POLSKA','1999-01-01','2018-01-01')
1  {{Piękny Kraków,Polska,2017-02-03 00:00:00,2,Najciekawa wycieczka ...}}

```

5. Tworzenie procedur modyfikujących dane

5.1. DodajRezerwacje(id_wycieczki, id_osoby), procedura powinna kontrolować czy wycieczka jeszcze się nie odbyła, i czy sa wolne miejsca

```

create or replace procedure DodajRezerwacje(id_w in number, id_o in number)
as
    data_wycieczki date;
    wolne_miejsca number;
begin
    select Data
    into data_wycieczki
    from wycieczki
    where id_wycieczki = id_w;

    if data_wycieczki < sysdate then
        raise_application_error(-20003, 'Wycieczka juz sie odbyla');
    end if;

    select LICZBAWOLNYCHMIEJSC(id_w)
    into wolne_miejsca
    from dual;

    if wolne_miejsca <= 0 then
        raise_application_error(-20004, 'Brak wolnych miejsc');
    end if;

    insert into rezerwacje(ID_WYCIECZKI, ID_OSOBY, STATUS)
    values (id_w, id_o, 'N');
end;

```

The screenshot shows a SQL IDE with a dark background. On the left, a line editor displays the following code:

```

24
25 begin
26     DODAJREZERWACJE(21, 48);
27 end;
28

```

A green box highlights the procedure call on line 26. Below the editor, a red error message box is visible with the following text:

```

[72000][20003] ORA-20003: Wycieczka juz sie odbyla
ORA-06512: przy "BD_HARDEK.DODAJREZERWACJE", linia 13
ORA-06512: przy linia 2
Position: 0

```

```
24
25  begin
26      DODAJREZERWACJE(42, 48);
27  end;
28
```

[72000][20004] ORA-20004: Brak wolnych miejsc
ORA-06512: przy "BD_HARDEK.DODAJREZERWACJE", linia 21
ORA-06512: przy linia 2
Position: 0

5.2. ZmienStatusRezerwacji(nr_rezerwacji, status), procedura powinna kontrolować czy możliwa jest zmiana statusu, np. zmiana statusu już anulowanej wycieczki (przywrócenie do stanu aktywnego nie zawsze jest możliwa – może już nie być miejsc)

```
create or replace procedure ZmienStatusRezerwacji(nr_r in number, status_r char)
as
    wolne_miejsca number;
    id_w number;
    stary_status char(1);
begin
    select ID_WYCIEZKI, NR_REZERWACJI
    into id_w, stary_status
    from REZERWACJE
    where NR_REZERWACJI = nr_r;

    select LICZBAWOLNYCHMIEJSC(id_w)
    into wolne_miejsca
    from dual;

    if wolne_miejsca <= 0 and status_r not like 'A' and stary_status like 'A' then
        raise_application_error(-20004, 'Brak wolnych miejsc');
    end if;

    update REZERWACJE
    set STATUS = status_r
    where NR_REZERWACJI = nr_r;
```

```
end;
```

5.3. ZmienLiczbeMiejsc(id_wycieczki, liczba_miejsc), nie wszystkie zmiany liczby miejsc są dozwolone, nie można zmniejszyć liczby miejsc na wartość poniżej liczby zarezerwowanych miejsc

```
create or replace procedure ZmienLiczbeMiejsc(id_w in number, nowe_miejsca
number)
as
    zajete_miejsca number;
begin
    select LICZBAZAJETYCHMIEJSC(id_w)
    into zajete_miejsca
    from dual;

    if zajete_miejsca > nowe_miejsca then
        raise_application_error(-20005, 'Liczba zarezerwowanych miejsc jest większa
niż nowy limit miejsc');
    end if;

    update WYCIECZKI
    set LICZBA_MIEJSC = nowe_miejsca
    where ID_WYCIECZKI = id_w;
end;
```

6. Dodawanie tabeli dziennikującej zmiany statusu rezerwacji oraz modyfikacja procedur

6.1. Tworzenie nowej tabeli

```
CREATE TABLE Rezerwacje_Log
(
    id int generated always as identity not null,
    id_rezerwacji int,
    data date,
    status char(1),
    CONSTRAINT Rezerwacje_Log_PK PRIMARY KEY
(
```

```

        id
    )
ENABLE
);

ALTER TABLE Rezerwacje_Log
ADD CONSTRAINT Rezerwacje_log_FK1 FOREIGN KEY
(
    id_rezerwacji
)
REFERENCES Rezerwacje
(
    NR_REZERWACJI
)
enable;

```

```

alter table Rezerwacje_Log
Add constraint Rezerwacje_Log_CHK1 check
(
    status in ('N', 'P', 'Z', 'A')
)
enable;

```

Modyfikacja procedur

6.2. ZmienStatusRezerwacji

```

create or replace procedure ZmienStatusRezerwacji(nr_r in number, status_r char)
as
    wolne_miejsca number;
    id_w number;
    stary_status rezerwacje.status%type;
begin
    select ID_WYCIEZKI, NR_REZERWACJI
    into id_w, stary_status
    from REZERWACJE
    where NR_REZERWACJI = nr_r;

    select LICZBAWOLNYCHMIEJSC(id_w)
    into wolne_miejsca
    from dual;

```

```

if wolne_miejsca <= 0 and status_r not like 'A' and stary_status like 'A' then
    raise_application_error(-20004, 'Brak wolnych miejsc');
end if;

update REZERWACJE
set STATUS = status_r
where NR_REZERWACJI = nr_r;

insert into REZERWACJE_LOG(ID_REZERWACJI, data, STATUS)
values (nr_r, current_date, stary_status);
end;

```

6.3. DodajRezerwacje

```

create or replace procedure DodajRezerwacje(id_w in number, id_o in number)
as
    data_wycieczki date;
    wolne_miejsca number;
    id_r number;
begin
    select Data
    into data_wycieczki
    from wycieczki
    where id_wycieczki = id_w;

    if data_wycieczki < sysdate then
        raise_application_error(-20003, 'Wycieczka juz sie odbyla');
    end if;

    select LICZBAWOLNYCHMIEJSC(id_w)
    into wolne_miejsca
    from dual;

    if wolne_miejsca <= 0 then
        raise_application_error(-20004, 'Brak wolnych miejsc');
    end if;

    insert into rezerwacje(ID_WYCIECZKI, ID_OSOBY, STATUS)
    values (id_w, id_o, 'N')
    returning NR_REZERWACJI into id_r;

    insert into Rezerwacje_Log(id_rezerwacji, data, status)

```

```

values (id_r, current_date, 'N');
end;

```

7. Zmiana struktury bazy danych

7.1. Dodanie tabeli

```

alter table WYCIECZKI
add liczba_wolnych_miejsc number;

```

7.2. Tworzenie procedury uaktualniającej liczbę wolnych miejsc

```

create or replace procedure przelicz(id_w in number)
as
    nowa_liczba_miejsc number;
begin
    select LICZBAWOLNYCHMIEJSC(id_w)
    into nowa_liczba_miejsc
    from dual;

    update WYCIECZKI
    set WYCIECZKI.liczba_wolnych_miejsc = nowa_liczba_miejsc
    where ID_WYCIECZKI = id_w;
end;

```

ID_WYCIECZKI	NAZWA	KRAJ	DATA	OPIS	LICZBA_MIEJSC	LICZBA_WOLNYCH_MIEJSC
21	Wycieczka do Paryża	Francja	2016-01-01 00:00:00	Ciekawa wycieczka ...	50	47
22	Piękny Kraków	Polska	2017-02-03 00:00:00	Najciekawa wycieczka ...	2	1
23	Wieliczka	Polska	2017-03-03 00:00:00	Zadziwiająca kopalnia ...	2	0
41	Słoneczny Wrocław	Polska	2025-07-18 00:00:00	Super Zabawa	3	2
42	Ciekawy Berlin	Niemcy	2023-05-06 00:00:00	Zabytki Historyczne	1	0

Nowe widoki

7.3. WycieczkiDostepne2

```

create view WycieczkiDostepne2
as
select
    kraj,
    data,

```



```
    nazwa,  
    liczba_miejsc,  
    LICZBA_WOLNYCH_MIEJSC  
from WYCIECZKI  
where Liczba_Wolnych_Miejsc > 0 and data > SYSDATE;
```

7.4. WycieczkiMiejsca2

```
create view WycieczkiMiejsca2  
as  
select unique  
    kraj,  
    data,  
    nazwa,  
    liczba_miejsc,  
    LICZBA_WOLNYCH_MIEJSC  
from WYCIECZKI;
```

Nowe procedury i funkcje

7.5. DodajRezerwacje2

```
create or replace procedure DodajRezerwacje2(id_w in number, id_o in number)  
as  
    data_wycieczki date;  
    wolne_miejsca number;  
    id_r number;  
begin  
    select Data  
    into data_wycieczki  
    from wycieczki  
    where id_wycieczki = id_w;  
  
    if data_wycieczki < sysdate then  
        raise_application_error(-20003, 'Wycieczka juz sie odbyla');  
    end if;  
  
    select LICZBA_WOLNYCH_MIEJSC  
    into wolne_miejsca  
    from WYCIECZKI  
    where ID_WYCIECZKI = id_w;
```

```

if wolne_miejsca <= 0 then
    raise_application_error(-20004, 'Brak wolnych miejsc');
end if;

insert into rezerwacje(ID_WYCIECZKI, ID_OSOBY, STATUS)
values (id_w, id_o, 'N')
returning NR_REZERWACJI into id_r;

insert into Rezerwacje_Log(id_rezerwacji, data, status)
values (id_r, current_date, 'N');

update WYCIECZKI
set WYCIECZKI.liczba_wolnych_miejsc = WYCIECZKI.liczba_wolnych_miejsc - 1
where id_w = ID_WYCIECZKI;
end;

```

7.6. ZmienLiczbeMiejsc2

```

create or replace procedure ZmienLiczbeMiejsc2(id_w in number, nowe_miejsca
number)
as
    zajete_miejsca number;
begin
    select LICZBA_MIEJSC - liczba_wolnych_miejsc
    into zajete_miejsca
    from WYCIECZKI;

    if zajete_miejsca > nowe_miejsca then
        raise_application_error(-20005, 'Liczba zarezerwowanych miejsc jest większa
niż nowy limit miejsc');
    end if;

    update WYCIECZKI
    set LICZBA_MIEJSC = nowe_miejsca,
    WYCIECZKI.liczba_wolnych_miejsc = nowe_miejsca - zajete_miejsca
    where ID_WYCIECZKI = id_w;
end;

```

7.7. ZmienStatusRezerwacji2

```

create or replace procedure ZmienStatusRezerwacji2(nr_r in number, status_r char)
as
    wolne_miejsca number;
    id_w number;
    stary_status rezerwacje.status%type;
begin
    select ID_WYCIECZKI, NR_REZERWACJI
    into id_w, stary_status
    from REZERWACJE
    where NR_REZERWACJI = nr_r;

    select LICZBA_WOLNYCH_MIEJSC
    into wolne_miejsca
    from WYCIECZKI
    where ID_WYCIECZKI = id_w;

    if wolne_miejsca <= 0 and status_r not like 'A' and stary_status like 'A' then
        raise_application_error(-20004, 'Brak wolnych miejsc');
    end if;

    update REZERWACJE
    set STATUS = status_r
    where NR_REZERWACJI = nr_r;

    insert into REZERWACJE_LOG(ID_REZERWACJI, data, STATUS)
    values (nr_r, current_date, stary_status);

    if status_r not like 'A' and stary_status like 'A' then
        update WYCIECZKI
        set WYCIECZKI.liczba_wolnych_miejsc = WYCIECZKI.liczba_wolnych_miejsc
- 1;
    end if;

    if status_r like 'A' and stary_status not like 'A' then
        update WYCIECZKI
        set WYCIECZKI.liczba_wolnych_miejsc = WYCIECZKI.liczba_wolnych_miejsc
+ 1;
    end if;
end;

```

7.8. DostępneWycieczki2

```
create or replace function DostępneWycieczki2(kraj_w varchar2, data_od date,
data_do date) return wycieczki_tabela
is
    dostępneWycieczki wycieczki_tabela;
begin
    if data_od > data_do then
        raise_application_error(-20002, 'data_od jest później niż data_do ');
    end if;

    select wycieczki_rekord(nazwa, kraj, data, LICZBA_MIEJSC, opis)
        bulk collect
    into dostępneWycieczki
    from WYCIECZKI
    where data >= data_od and data <= data_do and LICZBA_WOLNYCH_MIEJSC
> 0 and KRAJ like kraj_w;

    return dostępneWycieczki;
end;
```

8. Zmiana strategii zapisywania do dziennika rezerwacji. Realizacja przy pomocy triggerów

8.1. DodanieRezerwacjiTrigger

```
create or replace trigger DodanieRezerwacjiTrigger
After insert
on rezerwacje
for each row
begin
    insert into REZERWACJE_LOG (ID_REZERWACJI, DATA,
REZERWACJE_LOG.STATUS)
    values (:new.NR_REZERWACJI, current_date, :new.STATUS);

    if :new.STATUS not like 'A' then
        update WYCIECZKI
        set WYCIECZKI.liczba_wolnych_miejsc =
WYCIECZKI.liczba_wolnych_miejsc - 1
        where wycieczki.ID_WYCIECZKI = :new.ID_WYCIECZKI;
    end if;
```

```
end;
```

8.2. ZmianaStatusuTrigger

```
create or replace trigger ZmianaStatusuTrigger
after update on REZERWACJE
for each row
begin
    insert into REZERWACJE_LOG (ID_REZERWACJI, DATA,
REZERWACJE_LOG.STATUS)
    values (:new.NR_REZERWACJI, current_date, :new.STATUS);

    if :new.STATUS not like 'A' and :old.STATUS like 'A' then
        update WYCIECZKI
        set WYCIECZKI.liczba_wolnych_miejsc =
WYCIECZKI.liczba_wolnych_miejsc - 1
        where ID_WYCIECZKI = :new.ID_WYCIECZKI;
    end if;

    if :new.STATUS like 'A' and :old.STATUS not like 'A' then
        update WYCIECZKI
        set WYCIECZKI.liczba_wolnych_miejsc =
WYCIECZKI.liczba_wolnych_miejsc + 1
        where ID_WYCIECZKI = :new.ID_WYCIECZKI;
    end if;

end;
```

8.3. BlokadaUsuwaniaRezerwacjiTrigger

```
create or replace trigger BlokadaUsuwaniaRezerwacjiTrigger
before delete on REZERWACJE
for each row
begin
    raise_application_error(-20006, 'Nie mozna usuwac rezerwacji');
end;
```

```
[72000][20006] ORA-20006: Nie mozna usuwac rezerwacji
ORA-06512: przy "BD_HARDEK.BLOKADA_USUWANIA_REZERWACJI_TRIGGER", linia 2
ORA-04088: błąd w trakcie wykonywania wyzwalacza 'BD_HARDEK.BLOKADA_USUWANIA_REZERWACJI_TRIGGER'
Position: 12
```

8.4. ZmianaLiczbyMiejscTrigger

```
create or replace trigger ZmianaLiczbyMiejscTrigger
after update on WYCIECZKI
for each row
begin
    update WYCIECZKI
    set WYCIECZKI.liczba_wolnych_miejsc = WYCIECZKI.liczba_wolnych_miejsc +
(:new.liczba_miejsc - :old.liczba_miejsc)
    where ID_WYCIECZKI = :new.id_wycieczki;
end;
```

Nowe procedury

8.5. ZmienLiczbaMiejsc3

```
create or replace procedure ZmienLiczbaMiejsc3(id_w in number, nowe_miejsca
number)
as
    zajete_miejsca number;
begin
    select LICZBA_MIEJSC - liczba_wolnych_miejsc
    into zajete_miejsca
    from WYCIECZKI;

    if zajete_miejsca > nowe_miejsca then
        raise_application_error(-20005, 'Liczba zarezerwowanych miejsc jest większa
niż nowy limit miejsc');
    end if;

    update WYCIECZKI
    set LICZBA_MIEJSC = nowe_miejsca
    where ID_WYCIECZKI = id_w;
end;
```

8.6. ZmienStatusRezerwacji3

```
create or replace procedure ZmienStatusRezerwacji3(nr_r in number, status_r char)
as
    wolne_miejsca number;
    id_w number;
```

```

    stary_status rezerwacje.status%type;
begin
    select ID_WYCIECZKI, NR_REZERWACJI
    into id_w, stary_status
    from REZERWACJE
    where NR_REZERWACJI = nr_r;

    select LICZBA_WOLNYCH_MIEJSC
    into wolne_miejsca
    from WYCIECZKI
    where id_w = ID_WYCIECZKI;

    if wolne_miejsca <= 0 and status_r not like 'A' and stary_status like 'A' then
        raise_application_error(-20004, 'Brak wolnych miejsc');
    end if;

    update REZERWACJE
    set STATUS = status_r
    where NR_REZERWACJI = nr_r;
end;

```

8.7. DodajRezerwacje3

```

create or replace procedure DodajRezerwacje3(id_w in number, id_o in number)
as
    data_wycieczki date;
    wolne_miejsca number;
begin
    select Data
    into data_wycieczki
    from wycieczki
    where id_wycieczki = id_w;

    if data_wycieczki < sysdate then
        raise_application_error(-20003, 'Wycieczka juz sie odbyla');
    end if;

    select LICZBA_WOLNYCH_MIEJSC
    into wolne_miejsca

```

```
from WYCIECZKI
where ID_WYCIECZKI = id_w;

if wolne_miejsca <= 0 then
    raise_application_error(-20004, 'Brak wolnych miejsc');
end if;

insert into rezerwacje(ID_WYCIECZKI, ID_OSOBY, STATUS)
values (id_w, id_o, 'N');
end;
```