

**Bazy Danych
i Zarządzanie Informacją**

Projekt zaliczeniowy

Uniwersytet Marii Curie-Skłodowskiej
w Lublinie

Aleksander Modzelewski
Kierunek Informatyka
Rok 2019/20 Gr. Lab. 6

2020

Spis treści

1. Opis przedsiębiorstwa ModAl
2. Uzasadnienie stworzenia oprogramowania na miarę
3. Przypadki użycia oprogramowania
4. Stan zastany
5. Wybór silnika relacyjnej bazy danych
6. Zalety
7. Wady
8. Diagram relacji
9. Analiza przypadków zmiany postaci normalnej
10. Spis przechowywanych danych
11. Skrypt generujący schemat bazy danych
12. Skrypt insertujący dane

1. Opis przedsiębiorstwa ModAl

Przedsiębiorstwo ModAl zajmuje się wynajmem aut osobowych dla klientów indywidualnych i przedsiębiorstw. Zwiększenie skali działania wymaga utworzenia oprogramowania wspomagającego operacyjne działanie firmy.

Zespół opiekunów klienta zajmuje się bieżącą obsługą oraz dbaniem o stan pojazdów. Oprogramowanie powinno rejestrować przeglądy techniczne oraz naprawy. Po wypożyczeniu auta specjaliści ds. obsługi klienta wystawiają fakturę.

Właściciel firmy zdecydował utworzenie oprogramowania „na miarę”, a więc pisanego na zamówienie.

2. Uzasadnienie stworzenia oprogramowania na miarę

Oprogramowanie tego typu pozwoli na idealne dopasowanie oprogramowania do potrzeb pracowników firmy ModAl. Wykonawca będzie także dalej utrzymywał projekt, który będzie mógł dostawać nowe funkcjonalności wraz ze wzrostem przedsiębiorstwa.

3. Przypadki użycia oprogramowania

Kilka typowych przypadków użycia

- Logowanie do systemu przy pomocy loginu i hasła
- Wylistowanie wypożyczonych pojazdów: marka i model pojazdu, imię i nazwisko osoby wypożyczającej, data wypożyczenia, kontakt do Klienta
- Rejestracja przeglądu technicznego pojazdu: data przeglądu, koszt przeglądu netto
- Przyjęcie wypożyczonego pojazdu: wypełnienie daty i godziny zwrotu, naniesienie uwag z oględzin auta
- 4. Wystawienie faktury VAT: wypełnienie danych klienta, w tym danych osobowych, kontaktowych, adresowych, ewentualnej nazwy firmy i numeru NIP, wyliczenie kosztu wynajmu, wystawienie faktury, wypełniając koszt w linii faktury na bazie danych wynajęcia pojazdu dla Klienta

5. Stan zastany

Firma do tej pory posługuje się dokumentacją papierową. Faktury wystawia przy pomocy prostego programu do wystawiania faktur VAT bez możliwości automatycznego importu danych z wypożyczenia auta.

6. Wybór silnika relacyjnej bazy danych

Wykonawca zdecydował się na darmową relacyjną bazę firmy Oracle o nazwie Oracle 11xe. Wybór renomowanego dostawcy gwarantuje długi cykl życia silnika RDBMS. Z kolei istniejące rozwiązania płatne umożliwią przejście na bardziej zaawansowane w momencie rozwoju przedsiębiorstwa ModAl a tym samym wzrostu wymagań niefunkcjonalnych i funkcjonalnych.

7. Zalety wybranego RDBMS

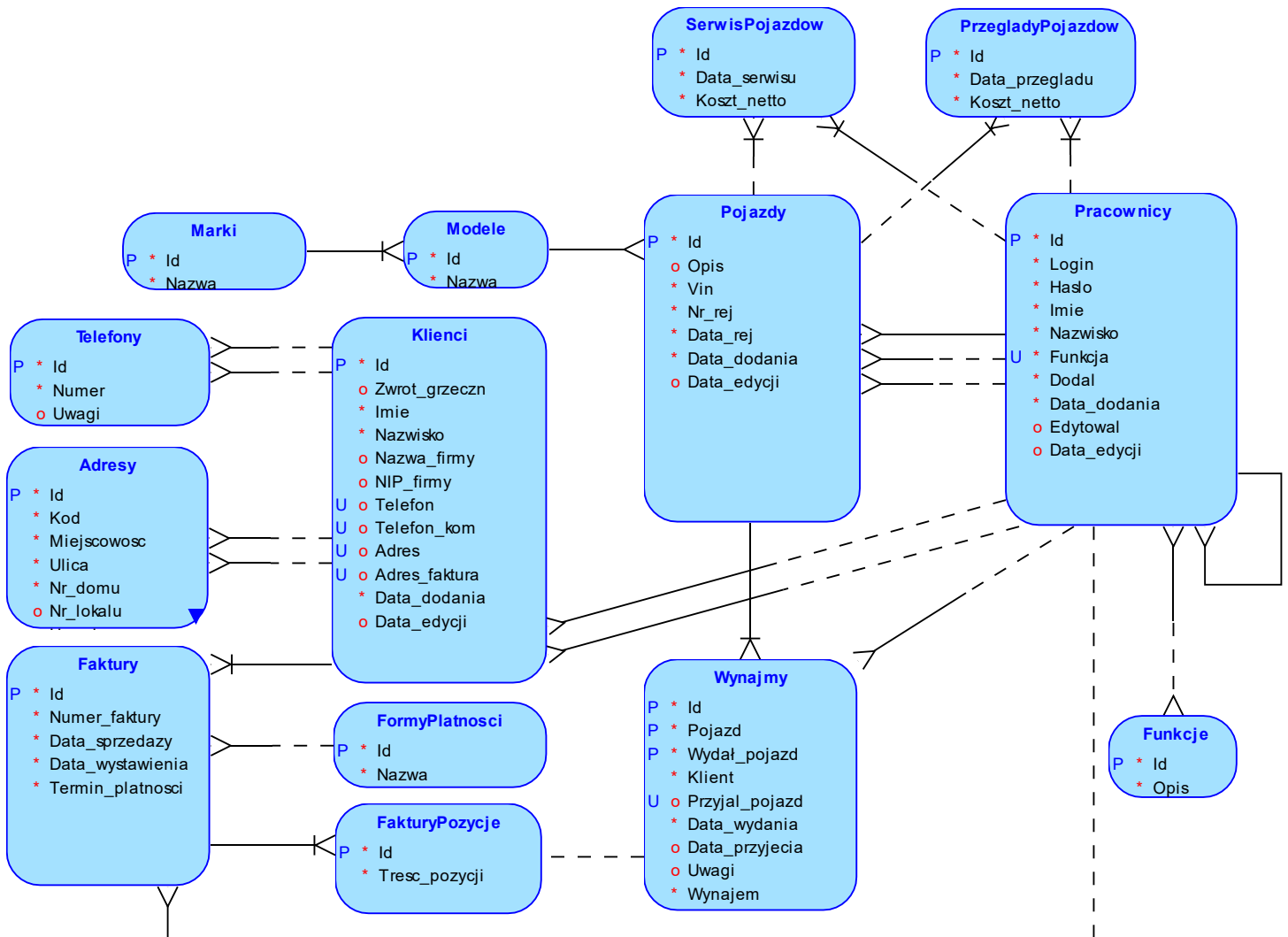
- Renomowany dostawca
- Dostępna pełna oferta dotycząca przetwarzania oprogramowania, włącznie z hurtownią danych cz Business Intelligence

8. Wady

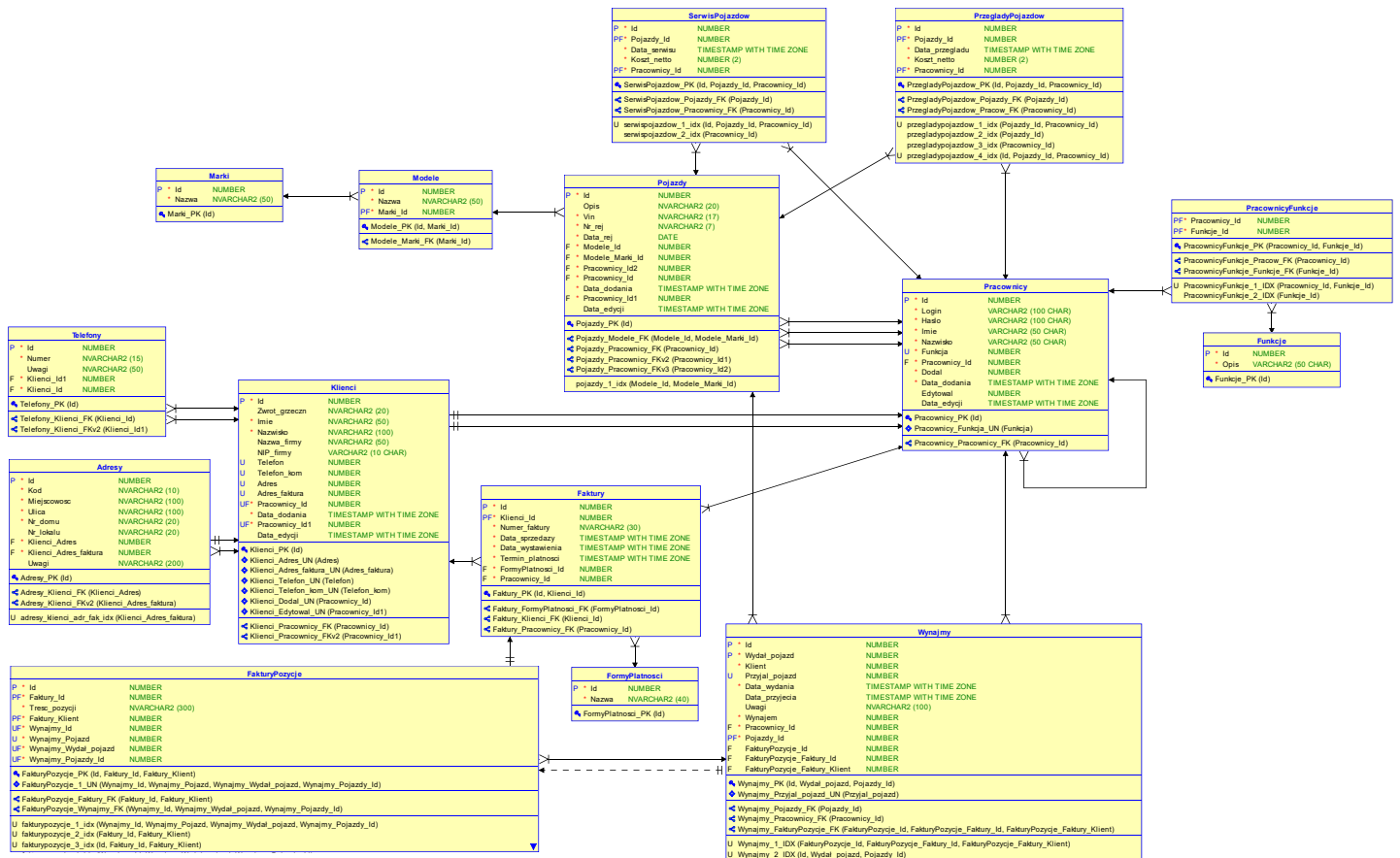
- Kosztowne wersje komercyjne
- Duże wymagania odnośnie sprzętu serwerowego oraz systemu operacyjnego serwera

9. Diagram relacji

- Model logiczny



Model relacyjny



10. Analiza przypadków zmiany postaci normalnej

Podjęto kompromis projektowy odnośnie wyposażenia pojazdów. Na ten moment, tabela przechowująca model pojazdu obejmuje także wyposażenie. Na potrzeby rozróżnienia dodano pole Opis.

11. Spis przechowywanych danych

12. Skrypt generujący schemat bazy danych

-- Generated by Oracle SQL Developer Data Modeler 19.4.0.350.1424

-- at: 2020-06-28 22:55:32 CEST

-- site: Oracle Database 11g

-- type: Oracle Database 11g

CREATE TABLE adresy (

```

    id          NUMBER NOT NULL,
    kod         NVARCHAR2(10) NOT NULL,
    miejscowosc NVARCHAR2(100) NOT NULL,
    ulica       NVARCHAR2(100) NOT NULL,
    nr_domu     NVARCHAR2(20) NOT NULL,
    nr_lokalu   NVARCHAR2(20),
    klienci_adres NUMBER NOT NULL,
    klienci_adres_faktura NUMBER NOT NULL,
    uwagi      NVARCHAR2(200)

```

);

```
CREATE UNIQUE INDEX adresy_klienci_adr_fak_idx ON
  adresy (
    klienci_adres_faktura
  ASC );
```

```
CREATE UNIQUE INDEX adresy_id_idx ON
  adresy (
    id
  ASC );
```

```
CREATE INDEX adresy_klienci_adres_idx ON
  adresy (
    klienci_adres
  ASC );
```

```
ALTER TABLE adresy ADD CONSTRAINT adresy_pk PRIMARY KEY ( id );
```

```
CREATE TABLE faktury (
  id          NUMBER NOT NULL,
  klienci_id  NUMBER NOT NULL,
  numer_faktury NVARCHAR2(30) NOT NULL,
  data_sprzedazy  TIMESTAMP WITH TIME ZONE NOT NULL,
  data_wystawienia  TIMESTAMP WITH TIME ZONE NOT NULL,
  termin_platnosci  TIMESTAMP WITH TIME ZONE NOT NULL,
  formyplatnosci_id NUMBER NOT NULL,
  pracownicy_id    NUMBER NOT NULL
);
```

```
CREATE UNIQUE INDEX faktury_id_klienci_id_idx ON
  faktury (
    id
  ASC,
    klienci_id
  ASC );
```

```
CREATE INDEX faktury_formyplatnosci_id_idx ON
  faktury (
    formyplatnosci_id
  ASC );
```

```
CREATE INDEX faktury_klienci_id_idx ON
  faktury (
    klienci_id
  ASC );
```

```
CREATE INDEX faktury_pracownicy_id_idx ON
  faktury (
    pracownicy_id
  ASC );
```

```
ALTER TABLE faktury ADD CONSTRAINT faktury_pk PRIMARY KEY ( id,
```

klienci_id);

```
CREATE TABLE fakturypozycje (  
  id          NUMBER NOT NULL,  
  faktury_id   NUMBER NOT NULL,  
  tresc_pozycji NVARCHAR2(300) NOT NULL,  
  faktury_klient NUMBER NOT NULL,  
  wynajmy_id   NUMBER NOT NULL,  
  wynajmy_pojazd NUMBER NOT NULL,  
  wynajmy_wydał_pojazd NUMBER NOT NULL,  
  wynajmy_pojazdy_id NUMBER NOT NULL  
);
```

```
CREATE UNIQUE INDEX fakturypozycje_1_idx ON  
  fakturypozycje (  
    wynajmy_id  
  ASC,  
    wynajmy_pojazd  
  ASC,  
    wynajmy_wydał_pojazd  
  ASC,  
    wynajmy_pojazdy_id  
  ASC );
```

```
CREATE UNIQUE INDEX fakturypozycje_2_idx ON  
  fakturypozycje (  
    faktury_id  
  ASC,  
    faktury_klient  
  ASC );
```

```
CREATE UNIQUE INDEX fakturypozycje_3_idx ON  
  fakturypozycje (  
    id  
  ASC,  
    faktury_id  
  ASC,  
    faktury_klient  
  ASC );
```

```
CREATE INDEX fakturypozycje_4_idx ON  
  fakturypozycje (  
    wynajmy_id  
  ASC,  
    wynajmy_wydał_pojazd  
  ASC,  
    wynajmy_pojazdy_id  
  ASC );
```

```
ALTER TABLE fakturypozycje  
  ADD CONSTRAINT fakturypozycje_pk PRIMARY KEY ( id,  
                                                  faktury_id,  
                                                  faktury_klient );
```

```
ALTER TABLE fakturypozycje
  ADD CONSTRAINT fakturypozycje_1_un UNIQUE ( wynajmy_id,
                                             wynajmy_pojazd,
                                             wynajmy_wydał_pojazd,
                                             wynajmy_pojazdy_id );
```

```
CREATE TABLE formyplatnosci (
  id   NUMBER NOT NULL,
  nazwa NVARCHAR2(40) NOT NULL
);
```

```
CREATE UNIQUE INDEX formyplatnosci_id_idx ON
  formyplatnosci (
    id
  ASC );
```

```
ALTER TABLE formyplatnosci ADD CONSTRAINT formyplatnosci_pk PRIMARY KEY ( id );
```

```
CREATE TABLE funkcje (
  id   NUMBER NOT NULL,
  opis VARCHAR2(50 CHAR) NOT NULL
);
```

```
CREATE UNIQUE INDEX funkcje_id_idx ON
  funkcje (
    id
  ASC );
```

```
ALTER TABLE funkcje ADD CONSTRAINT funkcje_pk PRIMARY KEY ( id );
```

```
CREATE TABLE klienci (
  id          NUMBER NOT NULL,
  zwrot_grzeczyn NVARCHAR2(20),
  imie        NVARCHAR2(50) NOT NULL,
  nazwisko    NVARCHAR2(100) NOT NULL,
  nazwa_firmy NVARCHAR2(50),
  nip_firmy   VARCHAR2(10 CHAR),
  telefon     NUMBER,
  telefon_kom NUMBER,
  adres       NUMBER,
  adres_faktura NUMBER,
  pracownicy_id NUMBER NOT NULL,
  data_dodania  TIMESTAMP WITH TIME ZONE NOT NULL,
  pracownicy_id1 NUMBER NOT NULL,
  data_edycji   TIMESTAMP WITH TIME ZONE
);
```

```
CREATE UNIQUE INDEX klienci_id_idx ON
  klienci (
    id
  ASC );
```



```
CREATE UNIQUE INDEX klienci_adres_idx ON
  klienci (
    adres
  ASC );
```

```
CREATE UNIQUE INDEX klienci_adres_faktura_idx ON
  klienci (
    adres_faktura
  ASC );
```

```
CREATE UNIQUE INDEX klienci_telefon_idx ON
  klienci (
    telefon
  ASC );
```

```
CREATE UNIQUE INDEX klienci_telefon_kom_idx ON
  klienci (
    telefon_kom
  ASC );
```

```
CREATE UNIQUE INDEX klienci_pracownicy_id_idx ON
  klienci (
    pracownicy_id
  ASC );
```

```
CREATE UNIQUE INDEX klienci_pracownicy_id1_idx ON
  klienci (
    pracownicy_id1
  ASC );
```

```
ALTER TABLE klienci ADD CONSTRAINT klienci_pk PRIMARY KEY ( id );
```

```
ALTER TABLE klienci ADD CONSTRAINT klienci_adres_faktura_un UNIQUE ( adres_faktura );
```

```
ALTER TABLE klienci ADD CONSTRAINT klienci_adres_un UNIQUE ( adres );
```

```
ALTER TABLE klienci ADD CONSTRAINT klienci_dodal_un UNIQUE ( pracownicy_id );
```

```
ALTER TABLE klienci ADD CONSTRAINT klienci_edytowal_un UNIQUE ( pracownicy_id1 );
```

```
ALTER TABLE klienci ADD CONSTRAINT klienci_telefon_kom_un UNIQUE ( telefon_kom );
```

```
ALTER TABLE klienci ADD CONSTRAINT klienci_telefon_un UNIQUE ( telefon );
```

```
CREATE TABLE marki (
  id   NUMBER NOT NULL,
  nazwa NVARCHAR2(50) NOT NULL
);
```

```
CREATE UNIQUE INDEX marki_id_idx ON
  marki (
    id
  ASC );
```

```
ALTER TABLE marki ADD CONSTRAINT marki_pk PRIMARY KEY ( id );
```

```
CREATE TABLE modele (  
    id      NUMBER NOT NULL,  
    nazwa   NVARCHAR2(50) NOT NULL,  
    marki_id NUMBER NOT NULL  
);
```

```
CREATE UNIQUE INDEX modele_id_marki_id_idx ON  
    modele (  
        id  
    ASC,  
        marki_id  
    ASC );
```

```
CREATE INDEX modele_marki_id_idx ON  
    modele (  
        marki_id  
    ASC );
```

```
ALTER TABLE modele ADD CONSTRAINT modele_pk PRIMARY KEY ( id,  
                                                            marki_id );
```

```
CREATE TABLE pojazdy (  
    id          NUMBER NOT NULL,  
    opis        NVARCHAR2(20),  
    vin         NVARCHAR2(17) NOT NULL,  
    nr_rej      NVARCHAR2(7) NOT NULL,  
    data_rej    DATE NOT NULL,  
    modele_id   NUMBER NOT NULL,  
    modele_marki_id NUMBER NOT NULL,  
    pracownicy_id2 NUMBER NOT NULL,  
    pracownicy_id NUMBER NOT NULL,  
    data_dodania  TIMESTAMP WITH TIME ZONE NOT NULL,  
    pracownicy_id1 NUMBER NOT NULL,  
    data_edycji  TIMESTAMP WITH TIME ZONE  
);
```

```
CREATE INDEX pojazdy_1_idx ON  
    pojazdy (  
        modele_id  
    ASC,  
        modele_marki_id  
    ASC );
```

```
CREATE UNIQUE INDEX pojazdy_id_idx ON  
    pojazdy (  
        id  
    ASC );
```

```
CREATE INDEX pojazdy_pracownicy_id_idx ON  
    pojazdy (  
        pracownicy_id  
    ASC );
```

```
        pracownicy_id
    ASC );

CREATE INDEX pojazdy_pracownicy_id1_idx ON
    pojazdy (
        pracownicy_id1
    ASC );

CREATE INDEX pojazdy_pracownicy_id2_idx ON
    pojazdy (
        pracownicy_id2
    ASC );

ALTER TABLE pojazdy ADD CONSTRAINT pojazdy_pk PRIMARY KEY ( id );

CREATE TABLE pracownicy (
    id          NUMBER NOT NULL,
    login       VARCHAR2(100 CHAR) NOT NULL,
    haslo       VARCHAR2(100 CHAR) NOT NULL,
    imie        VARCHAR2(50 CHAR) NOT NULL,
    nazwisko    VARCHAR2(50 CHAR) NOT NULL,
    funkcja     NUMBER NOT NULL,
    przełożony  NUMBER NOT NULL,
    pracownicy_id NUMBER NOT NULL,
    dodal       NUMBER NOT NULL,
    data_dodania TIMESTAMP WITH TIME ZONE NOT NULL,
    edytował    NUMBER,
    data_edycji  TIMESTAMP WITH TIME ZONE
);

CREATE UNIQUE INDEX pracownicy_id_idx ON
    pracownicy (
        id
    ASC );

CREATE INDEX pracownicy_pracownicy_id_idx ON
    pracownicy (
        pracownicy_id
    ASC );

CREATE INDEX pracownicy_przełożony_idx ON
    pracownicy (
        przełożony
    ASC );

ALTER TABLE pracownicy ADD CONSTRAINT pracownicy_pk PRIMARY KEY ( id );

CREATE TABLE pracownicyfunkcje (
    pracownicy_id NUMBER NOT NULL,
    funkcje_id    NUMBER NOT NULL
);

CREATE UNIQUE INDEX pracownicyfunkcje_1_idx ON
```

```
    pracownicyfunkcje (
        pracownicy_id
    ASC,
        funkcje_id
    ASC );

CREATE INDEX pracownicyfunkcje_2_idx ON
    pracownicyfunkcje (
        funkcje_id
    ASC );

ALTER TABLE pracownicyfunkcje ADD CONSTRAINT pracownicyfunkcje_pk PRIMARY KEY (
    pracownicy_id,
                                                funkcje_id );

CREATE TABLE przegladypojazdow (
    id          NUMBER NOT NULL,
    pojazdy_id  NUMBER NOT NULL,
    data_przegladu  TIMESTAMP WITH TIME ZONE NOT NULL,
    koszt_netto  NUMBER(2) NOT NULL,
    pracownicy_id  NUMBER NOT NULL
);

CREATE UNIQUE INDEX przegladypojazdow_1_idx ON
    przegladypojazdow (
        pojazdy_id
    ASC,
        pracownicy_id
    ASC );

CREATE INDEX przegladypojazdow_2_idx ON
    przegladypojazdow (
        pojazdy_id
    ASC );

CREATE INDEX przegladypojazdow_3_idx ON
    przegladypojazdow (
        pracownicy_id
    ASC );

CREATE UNIQUE INDEX przegladypojazdow_4_idx ON
    przegladypojazdow (
        id
    ASC,
        pojazdy_id
    ASC,
        pracownicy_id
    ASC );

ALTER TABLE przegladypojazdow
    ADD CONSTRAINT przegladypojazdow_pk PRIMARY KEY ( id,
                                                pojazdy_id,
                                                pracownicy_id );
```

```
CREATE TABLE serwispojazdow (  
    id          NUMBER NOT NULL,  
    pojazdy_id  NUMBER NOT NULL,  
    data_serwisu  TIMESTAMP WITH TIME ZONE NOT NULL,  
    koszt_netto  NUMBER(2) NOT NULL,  
    pracownicy_id NUMBER NOT NULL  
);
```

```
CREATE UNIQUE INDEX serwispojazdow_1_idx ON  
    serwispojazdow (  
        id  
    ASC,  
        pojazdy_id  
    ASC,  
        pracownicy_id  
    ASC );
```

```
CREATE INDEX serwispojazdow_2_idx ON  
    serwispojazdow (  
        pracownicy_id  
    ASC );
```

```
CREATE INDEX serwispojazdow_pojazdy_id_idx ON  
    serwispojazdow (  
        pojazdy_id  
    ASC );
```

```
ALTER TABLE serwispojazdow  
    ADD CONSTRAINT serwispojazdow_pk PRIMARY KEY ( id,  
                                                    pojazdy_id,  
                                                    pracownicy_id );
```

```
CREATE TABLE telefony (  
    id          NUMBER NOT NULL,  
    numer       NVARCHAR2(15) NOT NULL,  
    uwagi       NVARCHAR2(50),  
    klienci_id1 NUMBER NOT NULL,  
    klienci_id  NUMBER NOT NULL  
);
```

```
CREATE UNIQUE INDEX telefony_id_idx ON  
    telefony (  
        id  
    ASC );
```

```
CREATE INDEX telefony_klienci_id_idx ON  
    telefony (  
        klienci_id  
    ASC );
```

```
CREATE INDEX telefony_klienci_id1_idx ON  
    telefony (  
        klienci_id1  
    ASC );
```

```
    klienci_id1  
    ASC );
```

```
ALTER TABLE telefony ADD CONSTRAINT telefony_pk PRIMARY KEY ( id );
```

```
CREATE TABLE wynajmy (  
    id                NUMBER NOT NULL,  
    wydał_pojazd     NUMBER NOT NULL,  
    klient            NUMBER NOT NULL,  
    przyjął_pojazd    NUMBER,  
    data_wydania       TIMESTAMP WITH TIME ZONE NOT NULL,  
    data_przyjęcia     TIMESTAMP WITH TIME ZONE,  
    uwagi              NVARCHAR2(100),  
    wynajem            NUMBER NOT NULL,  
    pracownicy_id     NUMBER NOT NULL,  
    pojazdy_id         NUMBER NOT NULL,  
    fakturypozycje_id NUMBER,  
    fakturypozycje_faktury_id NUMBER,  
    fakturypozycje_faktury_klient NUMBER  
);
```

```
CREATE UNIQUE INDEX wynajmy_1_idx ON  
    wynajmy (  
        fakturypozycje_id  
    ASC,  
        fakturypozycje_faktury_id  
    ASC,  
        fakturypozycje_faktury_klient  
    ASC );
```

```
CREATE UNIQUE INDEX wynajmy_2_idx ON  
    wynajmy (  
        id  
    ASC,  
        wydał_pojazd  
    ASC,  
        pojazdy_id  
    ASC );
```

```
CREATE UNIQUE INDEX wynajmy_przyjął_pojazd_idx ON  
    wynajmy (  
        przyjął_pojazd  
    ASC );
```

```
CREATE INDEX wynajmy_pojazdy_id_idx ON  
    wynajmy (  
        pojazdy_id  
    ASC );
```

```
CREATE INDEX wynajmy_pracownicy_id_idx ON  
    wynajmy (  
        pracownicy_id  
    ASC );
```

```
ALTER TABLE wynajmy
  ADD CONSTRAINT wynajmy_pk PRIMARY KEY ( id,
                                     wydał_pojazd,
                                     pojazdy_id );

ALTER TABLE wynajmy ADD CONSTRAINT wynajmy_przyjał_pojazd_un UNIQUE ( przyjał_pojazd );

ALTER TABLE adresy
  ADD CONSTRAINT adresy_klienci_fk FOREIGN KEY ( klienci_adres )
    REFERENCES klienci ( adres );

ALTER TABLE adresy
  ADD CONSTRAINT adresy_klienci_fkv2 FOREIGN KEY ( klienci_adres_faktura )
    REFERENCES klienci ( adres_faktura );

ALTER TABLE faktury
  ADD CONSTRAINT faktury_formyplatnosci_fk FOREIGN KEY ( formyplatnosci_id )
    REFERENCES formyplatnosci ( id );

ALTER TABLE faktury
  ADD CONSTRAINT faktury_klienci_fk FOREIGN KEY ( klienci_id )
    REFERENCES klienci ( id );

ALTER TABLE faktury
  ADD CONSTRAINT faktury_pracownicy_fk FOREIGN KEY ( pracownicy_id )
    REFERENCES pracownicy ( id );

ALTER TABLE fakturypozycje
  ADD CONSTRAINT fakturypozycje_faktury_fk FOREIGN KEY ( faktury_id,
                                                         faktury_klient )
    REFERENCES faktury ( id,
                        klienci_id );

ALTER TABLE fakturypozycje
  ADD CONSTRAINT fakturypozycje_wynajmy_fk FOREIGN KEY ( wynajmy_id,
                                                         wynajmy_wydał_pojazd,
                                                         wynajmy_pojazdy_id )
    REFERENCES wynajmy ( id,
                        wydał_pojazd,
                        pojazdy_id );

ALTER TABLE klienci
  ADD CONSTRAINT klienci_pracownicy_fk FOREIGN KEY ( pracownicy_id )
    REFERENCES pracownicy ( id );

ALTER TABLE klienci
  ADD CONSTRAINT klienci_pracownicy_fkv2 FOREIGN KEY ( pracownicy_id1 )
    REFERENCES pracownicy ( id );

ALTER TABLE modele
  ADD CONSTRAINT modele_marki_fk FOREIGN KEY ( marki_id )
    REFERENCES marki ( id );
```

```
ALTER TABLE pojazdy
  ADD CONSTRAINT pojazdy_modele_fk FOREIGN KEY ( modele_id,
                                                modele_marki_id )
    REFERENCES modele ( id,
                        marki_id );

ALTER TABLE pojazdy
  ADD CONSTRAINT pojazdy_pracownicy_fk FOREIGN KEY ( pracownicy_id )
    REFERENCES pracownicy ( id );

ALTER TABLE pojazdy
  ADD CONSTRAINT pojazdy_pracownicy_fkv2 FOREIGN KEY ( pracownicy_id1 )
    REFERENCES pracownicy ( id );

ALTER TABLE pojazdy
  ADD CONSTRAINT pojazdy_pracownicy_fkv3 FOREIGN KEY ( pracownicy_id2 )
    REFERENCES pracownicy ( id );

ALTER TABLE pracownicy
  ADD CONSTRAINT pracownicy_pracow_fk FOREIGN KEY ( pracownicy_id )
    REFERENCES pracownicy ( id );

ALTER TABLE pracownicy
  ADD CONSTRAINT pracownicy_pracow_fk1 FOREIGN KEY ( przelezony )
    REFERENCES pracownicy ( id );

ALTER TABLE pracownicyfunkcje
  ADD CONSTRAINT pracownicyfunkcje_funkcje_fk FOREIGN KEY ( funkcje_id )
    REFERENCES funkcje ( id );

ALTER TABLE pracownicyfunkcje
  ADD CONSTRAINT pracownicyfunkcje_pracow_fk FOREIGN KEY ( pracownicy_id )
    REFERENCES pracownicy ( id );

ALTER TABLE przegladypojazdow
  ADD CONSTRAINT przegladypojazdow_pojazdy_fk FOREIGN KEY ( pojazdy_id )
    REFERENCES pojazdy ( id );

ALTER TABLE przegladypojazdow
  ADD CONSTRAINT przegladypojazdow_pracow_fk FOREIGN KEY ( pracownicy_id )
    REFERENCES pracownicy ( id );

ALTER TABLE serwispojazdow
  ADD CONSTRAINT serwispojazdow_pojazdy_fk FOREIGN KEY ( pojazdy_id )
    REFERENCES pojazdy ( id );

ALTER TABLE serwispojazdow
  ADD CONSTRAINT serwispojazdow_pracownicy_fk FOREIGN KEY ( pracownicy_id )
    REFERENCES pracownicy ( id );

ALTER TABLE telefony
  ADD CONSTRAINT telefony_klienci_fk FOREIGN KEY ( klienci_id )
```



```
REFERENCES klienci ( id );

ALTER TABLE telefony
ADD CONSTRAINT telefony_klienci_fk2 FOREIGN KEY ( klienci_id1 )
REFERENCES klienci ( id );

ALTER TABLE wynajmy
ADD CONSTRAINT wynajmy_fakturypozycje_fk FOREIGN KEY ( fakturypozycje_id,
                                                    fakturypozycje_faktury_id,
                                                    fakturypozycje_faktury_klient )
REFERENCES fakturypozycje ( id,
                             faktury_id,
                             faktury_klient );

ALTER TABLE wynajmy
ADD CONSTRAINT wynajmy_pojazdy_fk FOREIGN KEY ( pojazdy_id )
REFERENCES pojazdy ( id );

ALTER TABLE wynajmy
ADD CONSTRAINT wynajmy_pracownicy_fk FOREIGN KEY ( pracownicy_id )
REFERENCES pracownicy ( id );

CREATE SEQUENCE adresy_id_seq START WITH 1 NOCACHE ORDER;

CREATE OR REPLACE TRIGGER adresy_1 BEFORE
INSERT ON adresy
FOR EACH ROW
WHEN ( new.id IS NULL )
BEGIN
:new.id := adresy_id_seq.nextval;
END;
/

CREATE SEQUENCE faktury_id_seq START WITH 1 NOCACHE ORDER;

CREATE OR REPLACE TRIGGER faktury_1 BEFORE
INSERT ON faktury
FOR EACH ROW
WHEN ( new.id IS NULL )
BEGIN
:new.id := faktury_id_seq.nextval;
END;
/

CREATE SEQUENCE faktury_klienci_id_seq START WITH 1 NOCACHE ORDER;

CREATE OR REPLACE TRIGGER faktury_2 BEFORE
INSERT ON faktury
FOR EACH ROW
WHEN ( new.klienci_id IS NULL )
BEGIN
:new.klienci_id := faktury_klienci_id_seq.nextval;
END;
```

/

```
CREATE SEQUENCE faktury_formyplatnosci_id_seq START WITH 1 NOCACHE ORDER;
```

```
CREATE OR REPLACE TRIGGER faktury_3 BEFORE
  INSERT ON faktury
  FOR EACH ROW
  WHEN ( new.formyplatnosci_id IS NULL )
BEGIN
  :new.formyplatnosci_id := faktury_formyplatnosci_id_seq.nextval;
END;
/
```

```
CREATE SEQUENCE faktury_pracownicy_id_seq START WITH 1 NOCACHE ORDER;
```

```
CREATE OR REPLACE TRIGGER faktury_4 BEFORE
  INSERT ON faktury
  FOR EACH ROW
  WHEN ( new.pracownicy_id IS NULL )
BEGIN
  :new.pracownicy_id := faktury_pracownicy_id_seq.nextval;
END;
/
```

```
CREATE SEQUENCE fakturypozycje_id_seq START WITH 1 NOCACHE ORDER;
```

```
CREATE OR REPLACE TRIGGER fakturypozycje_1 BEFORE
  INSERT ON fakturypozycje
  FOR EACH ROW
  WHEN ( new.id IS NULL )
BEGIN
  :new.id := fakturypozycje_id_seq.nextval;
END;
/
```

```
CREATE SEQUENCE formyplatnosci_id_seq START WITH 1 NOCACHE ORDER;
```

```
CREATE OR REPLACE TRIGGER formyplatnosci_1 BEFORE
  INSERT ON formyplatnosci
  FOR EACH ROW
  WHEN ( new.id IS NULL )
BEGIN
  :new.id := formyplatnosci_id_seq.nextval;
END;
/
```

```
CREATE SEQUENCE funkcje_id_seq START WITH 1 NOCACHE ORDER;
```

```
CREATE OR REPLACE TRIGGER funkcje_1 BEFORE
  INSERT ON funkcje
  FOR EACH ROW
  WHEN ( new.id IS NULL )
BEGIN
```

```
    :new.id := funkcje_id_seq.nextval;  
END;  
/
```

```
CREATE SEQUENCE klienci_id_seq START WITH 1 NOCACHE ORDER;
```

```
CREATE OR REPLACE TRIGGER klienci_1 BEFORE  
  INSERT ON klienci  
  FOR EACH ROW  
  WHEN ( new.id IS NULL )  
BEGIN  
  :new.id := klienci_id_seq.nextval;  
END;  
/
```

```
CREATE SEQUENCE marki_id_seq START WITH 1 NOCACHE ORDER;
```

```
CREATE OR REPLACE TRIGGER marki_1 BEFORE  
  INSERT ON marki  
  FOR EACH ROW  
  WHEN ( new.id IS NULL )  
BEGIN  
  :new.id := marki_id_seq.nextval;  
END;  
/
```

```
CREATE SEQUENCE modele_id_seq START WITH 1 NOCACHE ORDER;
```

```
CREATE OR REPLACE TRIGGER modele_1 BEFORE  
  INSERT ON modele  
  FOR EACH ROW  
  WHEN ( new.id IS NULL )  
BEGIN  
  :new.id := modele_id_seq.nextval;  
END;  
/
```

```
CREATE SEQUENCE pojazdy_id_seq START WITH 1 NOCACHE ORDER;
```

```
CREATE OR REPLACE TRIGGER pojazdy_1 BEFORE  
  INSERT ON pojazdy  
  FOR EACH ROW  
  WHEN ( new.id IS NULL )  
BEGIN  
  :new.id := pojazdy_id_seq.nextval;  
END;  
/
```

```
CREATE SEQUENCE pracownicy_id_seq START WITH 1 NOCACHE ORDER;
```

```
CREATE OR REPLACE TRIGGER pracownicy_1 BEFORE  
  INSERT ON pracownicy  
  FOR EACH ROW
```

```
    WHEN ( new.id IS NULL )
BEGIN
    :new.id := pracownicy_id_seq.nextval;
END;
/

CREATE SEQUENCE przegladypojazdow_id_seq START WITH 1 NOCACHE ORDER;

CREATE OR REPLACE TRIGGER przegladypojazdow_ BEFORE
    INSERT ON przegladypojazdow
    FOR EACH ROW
    WHEN ( new.id IS NULL )
BEGIN
    :new.id := przegladypojazdow_id_seq.nextval;
END;
/

CREATE SEQUENCE przegladypojazdow_pojazdy_id START WITH 1 NOCACHE ORDER;

CREATE OR REPLACE TRIGGER przegladypojazdow_2 BEFORE
    INSERT ON przegladypojazdow
    FOR EACH ROW
    WHEN ( new.pojazdy_id IS NULL )
BEGIN
    :new.pojazdy_id := przegladypojazdow_pojazdy_id.nextval;
END;
/

CREATE SEQUENCE przegladypojazdow_pracownicy_i START WITH 1 NOCACHE ORDER;

CREATE OR REPLACE TRIGGER przegladypojazdow_3 BEFORE
    INSERT ON przegladypojazdow
    FOR EACH ROW
    WHEN ( new.pracownicy_id IS NULL )
BEGIN
    :new.pracownicy_id := przegladypojazdow_pracownicy_i.nextval;
END;
/

CREATE SEQUENCE serwispojazdow_id_seq START WITH 1 NOCACHE ORDER;

CREATE OR REPLACE TRIGGER serwispojazdow_1 BEFORE
    INSERT ON serwispojazdow
    FOR EACH ROW
    WHEN ( new.id IS NULL )
BEGIN
    :new.id := serwispojazdow_id_seq.nextval;
END;
/

CREATE SEQUENCE serwispojazdow_pojazdy_id_seq START WITH 1 NOCACHE ORDER;

CREATE OR REPLACE TRIGGER serwispojazdow_2 BEFORE
```

```
INSERT ON serwispojazdow
FOR EACH ROW
WHEN ( new.pojazdy_id IS NULL )
BEGIN
:new.pojazdy_id := serwispojazdow_pojazdy_id_seq.nextval;
END;
/

CREATE SEQUENCE serwispojazdow_pracownicy_id START WITH 1 NOCACHE ORDER;

CREATE OR REPLACE TRIGGER serwispojazdow_3 BEFORE
INSERT ON serwispojazdow
FOR EACH ROW
WHEN ( new.pracownicy_id IS NULL )
BEGIN
:new.pracownicy_id := serwispojazdow_pracownicy_id.nextval;
END;
/

CREATE SEQUENCE telefony_id_seq START WITH 1 NOCACHE ORDER;

CREATE OR REPLACE TRIGGER telefony_1 BEFORE
INSERT ON telefony
FOR EACH ROW
WHEN ( new.id IS NULL )
BEGIN
:new.id := telefony_id_seq.nextval;
END;
/

CREATE SEQUENCE wynajmy_id_seq START WITH 1 NOCACHE ORDER;

CREATE OR REPLACE TRIGGER wynajmy_1 BEFORE
INSERT ON wynajmy
FOR EACH ROW
WHEN ( new.id IS NULL )
BEGIN
:new.id := wynajmy_id_seq.nextval;
END;
/

-- Oracle SQL Developer Data Modeler Summary Report:
--
-- CREATE TABLE          15
-- CREATE INDEX           48
-- ALTER TABLE           50
-- CREATE VIEW             0
-- ALTER VIEW             0
-- CREATE PACKAGE          0
-- CREATE PACKAGE BODY     0
-- CREATE PROCEDURE        0
```

```
-- CREATE FUNCTION          0
-- CREATE TRIGGER           21
-- ALTER TRIGGER             0
-- CREATE COLLECTION TYPE    0
-- CREATE STRUCTURED TYPE    0
-- CREATE STRUCTURED TYPE BODY 0
-- CREATE CLUSTER            0
-- CREATE CONTEXT            0
-- CREATE DATABASE           0
-- CREATE DIMENSION          0
-- CREATE DIRECTORY          0
-- CREATE DISK GROUP         0
-- CREATE ROLE                0
-- CREATE ROLLBACK SEGMENT   0
-- CREATE SEQUENCE           21
-- CREATE MATERIALIZED VIEW   0
-- CREATE MATERIALIZED VIEW LOG 0
-- CREATE SYNONYM            0
-- CREATE TABLESPACE        0
-- CREATE USER               0
--
-- DROP TABLESPACE          0
-- DROP DATABASE             0
--
-- REDACTION POLICY          0
--
-- ORDS DROP SCHEMA          0
-- ORDS ENABLE SCHEMA        0
-- ORDS ENABLE OBJECT        0
--
-- ERRORS                    0
-- WARNINGS                  0
```

13. Skrypt insertujący dane

```
insert into marki values (marki_id_seq.nextval, 'Alfa Romeo');
insert into marki values (marki_id_seq.nextval, 'Audi');
insert into marki values (marki_id_seq.nextval, 'BMW');
insert into marki values (marki_id_seq.nextval, 'Fiat');
insert into marki values (marki_id_seq.nextval, 'Ford');
insert into marki values (marki_id_seq.nextval, 'Honda');
insert into marki values (marki_id_seq.nextval, 'Hyundai');
insert into marki values (marki_id_seq.nextval, 'Mazda');
insert into marki values (marki_id_seq.nextval, 'Mercedes');
insert into marki values (marki_id_seq.nextval, 'Opel');
insert into marki values (marki_id_seq.nextval, 'Peugeot');
insert into marki values (marki_id_seq.nextval, 'Renault');
insert into marki values (marki_id_seq.nextval, 'Seat');
insert into marki values (marki_id_seq.nextval, 'Volkswagen');
```

```
insert into modele values (modele_id_seq.nextval, 'Giulia', 1);
insert into modele values (modele_id_seq.nextval, 'Giulietta', 1);
```

```
insert into modele values (modele_id_seq.nextval, 'Quattro', 2);
insert into modele values (modele_id_seq.nextval, 'M5', 3);
insert into modele values (modele_id_seq.nextval, 'Tipo', 4);
insert into modele values (modele_id_seq.nextval, 'Umo', 4);
insert into modele values (modele_id_seq.nextval, 'Punto', 4);
insert into modele values (modele_id_seq.nextval, 'Ka', 5);
insert into modele values (modele_id_seq.nextval, 'Civic', 6);
insert into modele values (modele_id_seq.nextval, 'i20', 7);
insert into modele values (modele_id_seq.nextval, 'i30', 7);
insert into modele values (modele_id_seq.nextval, '6', 8);
insert into modele values (modele_id_seq.nextval, 'C200', 9);
insert into modele values (modele_id_seq.nextval, 'Astra', 10);
insert into modele values (modele_id_seq.nextval, 'Signum', 10);
insert into modele values (modele_id_seq.nextval, '206', 11);
insert into modele values (modele_id_seq.nextval, 'Clio', 12);
insert into modele values (modele_id_seq.nextval, 'Leon', 13);
insert into modele values (modele_id_seq.nextval, 'Polo', 14);
```

```
insert into funkcje values (funkcje_id_seq.nextval, 'Waściciel');
insert into funkcje values (funkcje_id_seq.nextval, 'Dyrektor operacyjny');
insert into funkcje values (funkcje_id_seq.nextval, 'Kierownik BOK');
insert into funkcje values (funkcje_id_seq.nextval, 'Specjalista ds. klientów');
```

```
insert into pracownicy values (pracownicy_id_seq.nextval, 'jan.kowalski', '123', 'Jan', 'Kowalski', 1, 1,
1, 1, '01/06/17 12:27:31,660000000 EUROPE/BELGRADE', null, null);
insert into pracownicy values (pracownicy_id_seq.nextval, 'marek.nowak', '123', 'Marek', 'Nowak', 2,
1, 2, 1, '02/06/17 09:42:11,660000000 EUROPE/BELGRADE', null, null);
insert into pracownicy values (pracownicy_id_seq.nextval, 'dariusz.stanek', '123', 'Dariusz', 'Stanek',
3, 2, 3, 2, '01/06/17 16:20:31,660000000 EUROPE/BELGRADE', null, null);
insert into pracownicy values (pracownicy_id_seq.nextval, 'jacek.dodalski', '123', 'Jacek', 'Dodalski', 4,
3, 4, 3, '03/06/17 12:47:41,660000000 EUROPE/BELGRADE', null, null);
insert into pracownicy values (pracownicy_id_seq.nextval, 'robert.ujemski', '123', 'Robert', 'Ujemski',
4, 3, 5, 3, '04/06/17 12:11:51,660000000 EUROPE/BELGRADE', null, null);
insert into pracownicy values (pracownicy_id_seq.nextval, 'alex.mnozynski', '123', 'Alex', 'Mnożyński',
4, 3, 6, 3, '05/06/17 10:24:11,660000000 EUROPE/BELGRADE', null, null);
insert into pracownicy values (pracownicy_id_seq.nextval, 'zofia.dzielinska', '123', 'Zofia', 'Dzielińska',
4, 3, 7, 3, '06/06/17 11:20:31,660000000 EUROPE/BELGRADE', null, null);
insert into pracownicy values (pracownicy_id_seq.nextval, 'natalia.potega', '123', 'Natalia', 'Potęga',
4, 3, 8, 3, '06/06/17 09:17:31,660000000 EUROPE/BELGRADE', null, null);
```

```
insert into pracownicyfunkcje values (1, 1);
insert into pracownicyfunkcje values (2, 2);
insert into pracownicyfunkcje values (3, 3);
insert into pracownicyfunkcje values (4, 4);
insert into pracownicyfunkcje values (5, 4);
insert into pracownicyfunkcje values (6, 4);
insert into pracownicyfunkcje values (7, 4);
insert into pracownicyfunkcje values (8, 4);
```

```
insert into klienci values (klienci_id_seq.nextval, '', 'Alojzy', 'Grzelak', '', '', null, null, null, null, 1,
'06/06/17 09:17:31,660000000 EUROPE/BELGRADE', null, null);
insert into klienci values (klienci_id_seq.nextval, '', 'Adam', 'Sodowy', '', '', null, null, null, null, 1,
'07/06/17 19:25:31,660000000 EUROPE/BELGRADE', null, null);
```

```
insert into klienci values (klienci_id_seq.nextval, '', 'Mariusz', 'Koń', '', '', null, null, null, null, 1, '07/06/17 11:12:11,660000000 EUROPE/BELGRADE', null, null);
```

```
insert into pojazdy values (pojazdy_id_seq.nextval, 'Bialy', '12345678912345678', 'LU1234S', '06/07/18', 1, 1, 1, 1, '02/06/20 11:12:11,660000000 EUROPE/BELGRADE', 1, null);
```

```
insert into pojazdy values (pojazdy_id_seq.nextval, 'Bialy', '12345678912345678', 'LU1234S', '06/07/18', 3, 2, 1, 1, '02/06/20 11:12:11,660000000 EUROPE/BELGRADE', 2, null);
```

```
insert into pojazdy values (pojazdy_id_seq.nextval, 'Bialy', '12345678912345678', 'LU1234S', '06/07/18', 7, 4, 1, 1, '02/06/20 11:12:11,660000000 EUROPE/BELGRADE', 2, null);
```

14. Zapytania

- Projekcje

```
select id, nazwa from modele;  
select id, nazwa from marki;  
select id, imie, nazwisko, login from pracownicy;  
select pracownicy_id, funkcje_id from pracownicyfunkcje;  
select id, opis from pojazdy;  
select id, nr_rej, vin from pojazdy;
```

- Selekcje

```
select id, nazwa from modele where id between 1 and 5;  
select id, nazwa from modele where nazwa like 'G%';  
select id, imie, nazwisko, login from pracownicy where data_edycji is not null;  
select pracownicy_id, funkcje_id from pracownicyfunkcje where funkcje_id = 1;
```

- Zapytania łączące 3 tabele

```
select p.nr_rej, p.vin, mo.nazwa, ma.nazwa from pojazdy p  
join modele mo on mo.id = p.modele_id  
join marki ma on ma.id = mo.marki_id;
```

```
select p.id, mo.nazwa, ma.nazwa, p.data_rej  
from pojazdy p  
join modele mo on mo.id = p.modele_id  
join marki ma on ma.id = mo.marki_id  
where p.data_rej > '01/01/10';
```