Temat: System zarządzania komisem samochodowym

Autorzy: Szymon Budziosz, Piotr Knapik

Projekt bazy danych dla komisu samochodowego Bazy danych Prowadzący: dr inż. Robert Marcjan Autorzy: Szymon Budziosz, Piotr Knapik Grupa 1K333/SI Informatyka (Niestacjonarne) Rok III.

**1.Opis systemu**

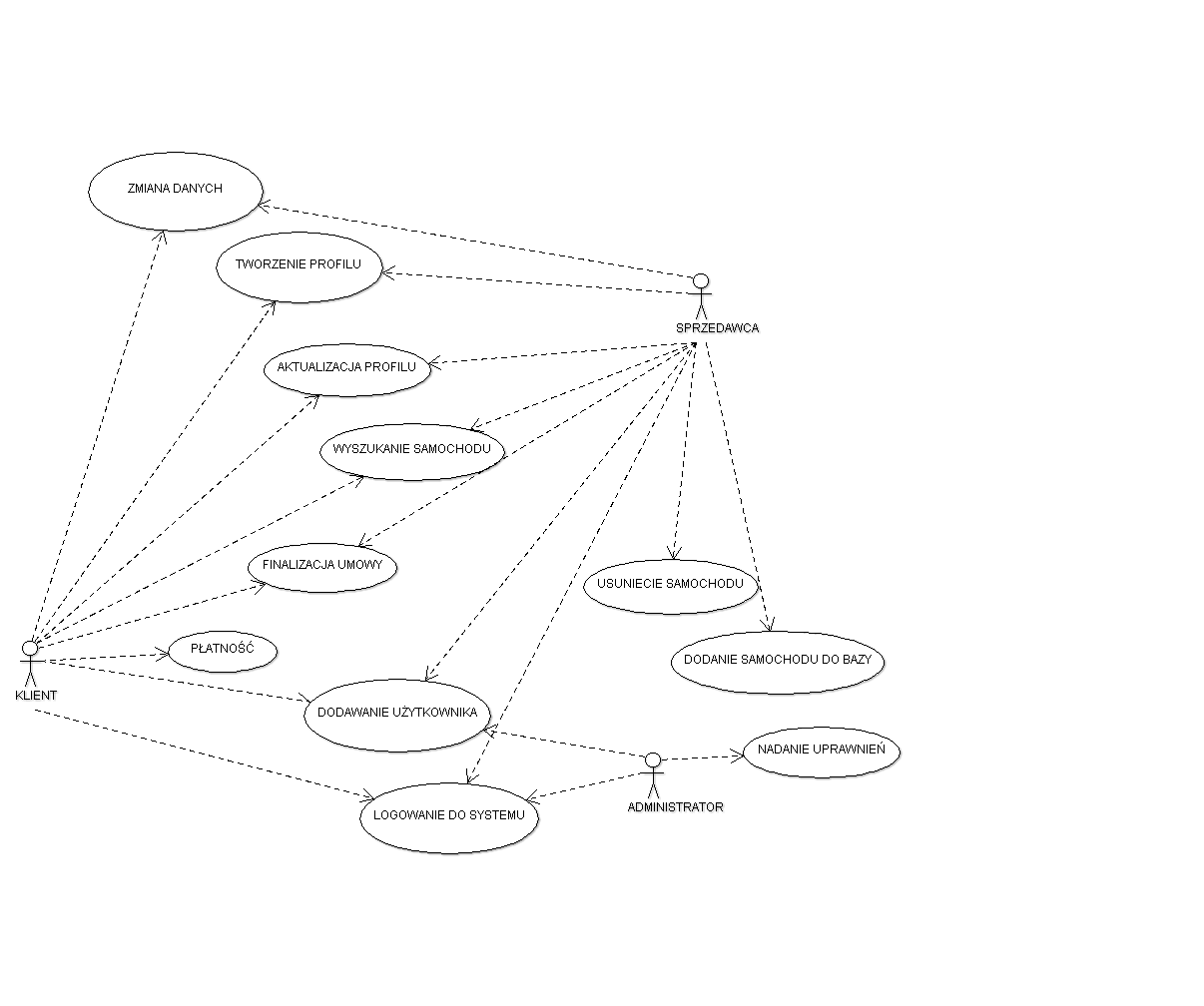
1. Aplikacja jest przeznaczona do ewidencjonowania pojazdów, klientów oraz rejestrowania transakcji zakupu i sprzedaży pojazdów. Umożliwia prowadzenie kartoteki klientów komisu, a także ewidencjonuje pojazdy, które przeszły przez komis. Dostarcza danych do wystawienia faktury kupna/sprzedaży, a także umożliwia określenie, które pojazdy są na sprzedaż. Pozwala w łatwy sposób znaleźć oferty odpowiadające oczekiwaniom klienta poprzez zastosowane filtry. Klient z łatwością może je porównać między sobą. Kontakt pomiędzy dealerem, a kupującym dzięki aplikacji będzie szybszy i bardziej rzeczowy . Klient dowie się o możliwości kredytowania zakupu samochodu. Pozna możliwości ubezpieczenia (dane ubezpieczyciela, koszt, zakres ubezpieczenia).

2. Przepływ danych Klient Zamówienie/ oferta sprzedaży Przyjęcie zgłoszenia kupna/sprzedaży Dane o zgłoszeniu Weryfikacja zgłoszenia: Dobór odpowiadającego klientowi auta (przedstawienie oferty sprzedaży)/ Sprawdzenie stanu technicznego i wycena w celu zakupu Przegląd dostępnych aut Finalizowanie transakcji Kartoteka klientów Sprzedaż / zakup auta Kartoteka pojazdów Dane o kliencie Dane o pojeździe Dane o transakcji Kartoteka transakcji kupna/sprzedaży.

Definicja bytów z określeniem ich cech. Klient (nazwisko, imię, pesel, , kod pocztowy, miasto, ulica, nr domu, nr lokalu, adres e-mail, numer telefonu). Pojazd (marka, model, rok produkcji, nr Vin , przebieg, pojemność, rodzaj paliwa, kolor, nadwozie, wyposażenie). Transakcja (rodzaj transakcji, data transakcji, kwota brutto, , id faktury, sposób zapłaty, czy zapłacono).

1. **Przypadki użycia**

1. Diagram przypadków użycia



2. Opis przypadków użycia

Dla każdego przypadku użycia opis w formie tabelki

|  |
| --- |
| Nazwa: dodanie użytkownika klient  Aktorzy: klient ,sprzedawca |
| Scenariusz:  1.Aktor wpisuje na stronie rejestracji swoje imię , nazwisko , adres email i tel.  2.Klika na przycisk ‘rejestracja’  3.System sprawdza unikalność danych  4.System generuje dla użytkownika hasło oraz login i wyświetla w oknie przeglądarki |

|  |
| --- |
| Nazwa: usuwanie użytkownika z systemu  Aktorzy: klient ,sprzedawca |
| Scenariusz:  1.Aktor loguje się do systemu  2.Aktor klika na zakładkę dane użytkownika i przycisk usuń dane z systemu |

|  |
| --- |
| Nazwa: dodanie użytkownika sprzedawca  Aktorzy: administrator |
| Scenariusz:  1.Aktor loguje się do systemu  2.Aktor wyszukuje użytkownika  3.Aktor nadaje uprawnienia z zakresu – administrator ,sprzedawca |

|  |
| --- |
| Nazwa: utworzenie profilu użytkownika  Aktorzy: klient ,sprzedawca |
| Scenariusz:  1.Aktor loguje się do systemu  2.W zakładce ustawienia użytkownika Aktor określa swoje ogólne preferencje co do samochodu takie jak -model,rok\_produkcji,przebieg\_do,pojemnosc\_od,pojemnosc\_do,paliwo,nadwozie  oraz maksymalnie 10 dodatków szczegółowych:  ( '2 drzwi'),('3 drzwi'),('4/5 drzwi'),('2 miejsca'),('5 miejsc'),('7 miejsc'),('na koła przednie'),('na koła tylne'),('4x4'),('manualna'),('automatyczna'),('klimatyzacja manualna'),('brak klimatyzacji'),('klimatyzacja automatyczna'),( 'oświetlenie do jazdy dziennej'),('światła led'),('światła przeciwmgłowe'),('światła ksenonowe'),('ABS'),('ESP'),('kontrola trakcji'),('asystent parkowania'),('czujniki parkowania'),('czujnik deszczu'),('immobilizer'),('poduszka pasażera'),('poduszka boczna'),(‘kamera cofania'),('elektryczne szyby przednie'),('elektryczne szyby tylne'),('elektryczne lusterka'),('podgrzewane lusterka'),('kierownica wielofunkcyjna'),('podgrzewane siedzenia'),('wspomaganie kierownicy'),('przyciemniane szyby'),('radio fabryczne'),('radio niefabryczne'),('mp3'),('zmieniarka CD'),('nawigacja GPS'),('bluetooth'),('alufelgi'),('centralny zamek'),('panoramiczny dach'),( 'hak'),('komputer pokładowy'),('tempomat'),('metalik'),('akryl'),('matowy'),('perłowy'),('krajowe'),('importowany'),('bezwypadkowy'),('pierwszy właściciel'),('serwisowany w aso'),('zarejestrowany w polsce'),('czarny'),('czerwony'),('zielony'),('biały'),('srebrny'),('niebieski')  3.Klika przycisk ‘Zapisz’, system wysyła dane do bazy danych |
|  |

|  |
| --- |
| Nazwa: modyfikacja profilu użytkownika  Aktorzy: klient |
| Scenariusz:  1.Aktor loguje się do systemu  2.W zakładce ustawienia użytkownika Aktor zmienia swoje preferencje co do samochodu  3.Zapisuje, system wysyła dane do bazy danych |

|  |
| --- |
| Nazwa: dodanie samochodu do bazy  Aktorzy: sprzedawca |
| Scenariusz:  1.W zakładce DODAJ Aktor wpisuje dane dotyczące samochodu który ma znaleźć się w komisie: ([producent],[model],[nadwozia],[cena],[przebieg],[data\_produkcji],[kraj\_pochodzenia],[ksiazka\_serwisowa],[pojemnosc],[paliwo])  3.Zapisuje, system wysyła dane do bazy danych |

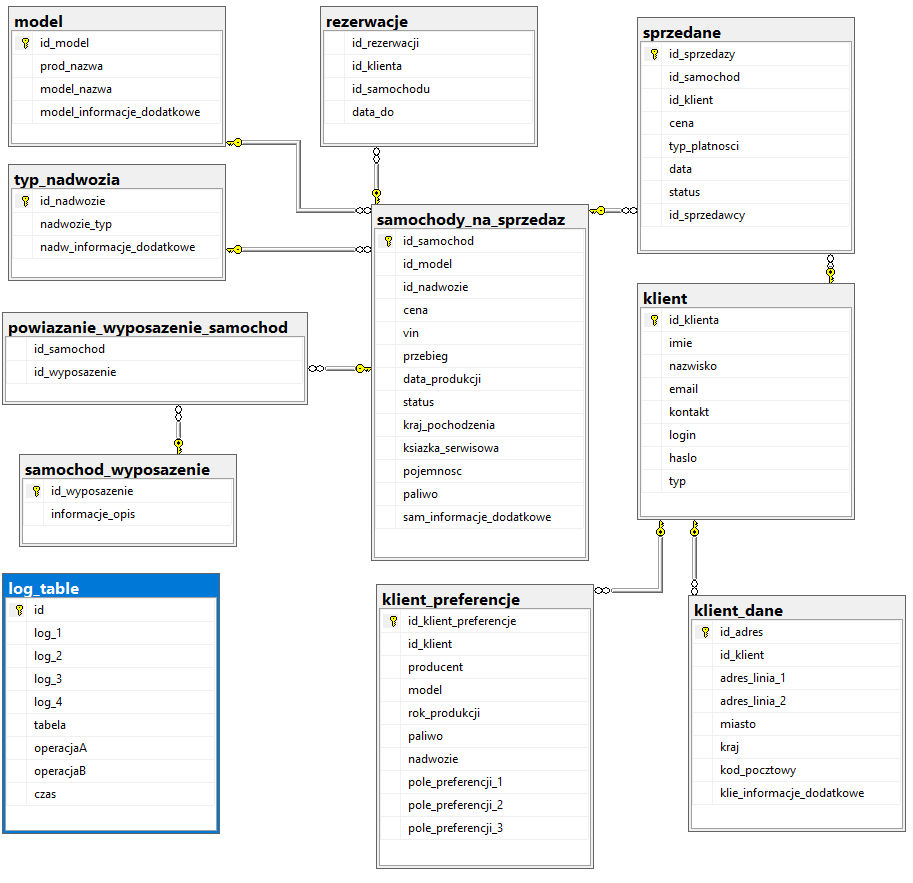
|  |
| --- |
| Nazwa: usunięcie samochodu z bazy  Aktorzy: sprzedawca |
| Scenariusz:  1.Aktory wyszukuje samochod po nr. Id lub vin  2.Zapisuje, system wysyła dane do bazy danych |

|  |
| --- |
| Nazwa: sprzedaż samochodu  Aktorzy: sprzedawca |
| Scenariusz:  2.W zakładce SPRZEDAŻ Aktor wpisuje dane do wyszukania samochodu (nr.rejestracyjny ,właściciel ,nr.umowy) oraz dane kupującego (nazwisko,pesel)  3.Wygenerowanie umowy na podstawie wyszukanych danych  4.Zapisuje, system wysyła dane do bazy danych |

|  |
| --- |
| Nazwa: Logowanie  Aktorzy: klient ,sprzedawca ,administrator |
| Scenariusz:  1. Aktor wpisuje login i hasło w formularzu logowania  2. Aktor klika przycisk ‘login’  3. Aktor otrzymuje dostęp do systemu wg. Uprawnień przypisanych do jego loginu |

|  |
| --- |
| Nazwa: płatność  Aktorzy: sprzedawca |
| 1.Aktor w zakładce ‘płatności’ wybiera formę płatności (gotówka , kredyt)  2.Aktor wpisuje kwotę należności i klika przycisk akceptuj |

SCHEMAT BAZY DANYCH



Tabele

[dbo].[klient]

|  |  |  |
| --- | --- | --- |
| KOLUMNA | TYP | OPIS |
| [id\_klienta] | [int] | KLUCZ GŁÓWNY |
| [imie] | [varchar](30) |  |
| [nazwisko] | [varchar](50) |  |
| [email] | [varchar](50) |  |
| [kontakt] | [varchar](20) |  |
| [login] | [varchar](15) | GENEROWANE PRZEZ PROCEDURĘ |
| [haslo] | [varchar](15) | GENEROWANE PRZEZ PROCEDURĘ |
| [typ] | [char](1) | D-DEALER S-KLIENT A- ADMIN |

[dbo].[klient\_dane]

|  |  |  |
| --- | --- | --- |
| KOLUMNA | TYP | OPIS |
| [id\_adres] | [int] | KLUCZ GŁÓWNY |
| [id\_klient] | [int] | KLUCZ OBCY |
| [adres\_linia\_1] | [varchar](50) |  |
| [adres\_linia\_2] | [varchar](50) |  |
| [miasto] | [varchar](40) |  |
| [kraj] | [varchar](25) |  |
| [kod\_pocztowy] | [varchar](15) |  |
| [klie\_informacje\_dodatkowe] | [varchar](50) |  |

[dbo].[klient\_preferencje]

|  |  |  |
| --- | --- | --- |
| KOLUMNA | TYP | OPIS |
| [id\_klient\_preferencje] | [int] | KLUCZ GŁÓWNY |
| [id\_klient] | [int] | KLUCZ OBCY |
| [producent] | [varchar](50) | W POLA KLIENT MOŻE WPISAĆ ATRYBUTY WEDŁUG KTÓRYCH MOŻLIWE JEST WYCZUKANIE SAMOCHODU  POLA PREFERENCJI PRZYJMUJĄ WARTOŚC LICZBOWĄ ODPPOWIADAJACĄ TABELI [samochod\_wyposazenie] |
| [model] | [varchar](50) |
| [rok\_produkcji] | [int] |
| [paliwo] | [char](1) |
| [nadwozie] | [int] |
| [pole\_preferencji\_1] | [int] |
| [pole\_preferencji\_2] | [int] |
| [pole\_preferencji\_3] | [int] |

[dbo].[samochody\_na\_sprzedaz]

|  |  |  |
| --- | --- | --- |
| KOLUMNA | TYP | OPIS |
| [id\_samochod] | [int] | KLUCZ GŁÓWNY |
| [id\_model] | [int] | KLUCZ OBCY |
| [id\_nadwozie] | [int] | KLUCZ OBCY |
| [cena] | [int] | CENA ŻĄDANA |
| [vin] | [varchar](30) |  |
| [przebieg] | [int] |  |
| [data\_produkcji] | [int] | ROK PRODUKCJI |
| [status] | [char](1) | S-SPRZEDANY,R-REZERWACJA,W-WOLNY |
| [kraj\_pochodzenia] | [char](3) |  |
| [ksiazka\_serwisowa] | [char](1) | T-TAK,N-NIE |
| [pojemnosc] | [int] |  |
| [paliwo] | [char](1) | B,G,O |
| [sam\_informacje\_dodatkowe] | [varchar](50) |  |

[dbo].[rezerwacje]

|  |  |  |
| --- | --- | --- |
| KOLUMNA | TYP | OPIS |
| [id\_rezerwacji] | [int] | KLUCZ GŁÓWNY |
| [id\_klienta] | [int] |  |
| [id\_samochodu] | [int] |  |
| [data\_do] | [smalldatetime] | DATA KOŃCA REZERWACJI |

[dbo].[sprzedane]

|  |  |  |
| --- | --- | --- |
| KOLUMNA | TYP | OPIS |
| [id\_sprzedazy] | [int] | KLUCZ GŁÓWNY |
| [id\_samochod] | [int] | KLUCZ OBCY |
| [id\_klient] | [int] | KLUCZ OBCY |
| [cena] | [int] | KWOTA OTRZYMANA |
| [typ\_platnosci] | [char](1) | L-LEASING,G-GOTÓWKA,K-KREDYT |
| [data] | [date] | DATA SPRZEDAŻY |
| [status] | [char](1) | T,N |
| [id\_sprzedawcy] | [int] | ID DEALERA DOKONUJACEGO SPRZEDAZY |

[dbo].[log\_table]

|  |  |  |
| --- | --- | --- |
| KOLUMNA | TYP | OPIS |
| [id] | [int] | KLUCZ GŁÓWNY |
| [log\_1] | [varchar](50) | POLA NA INFORMACJE LOGOWANE |
| [log\_2] | [varchar](50) |
| [log\_3] | [varchar](50) |
| [log\_4] | [varchar](50) |
| [tabela] | [varchar](50) | NAZWA TABELI NA KTÓREJ WYKONANO OPERACJE |
| [operacjaA] | [varchar](20) | TYP OPERACJI:INSERT/DELETE  /UPDATE LUB NAZWA UŻYTKOWNIKA |
| [operacjaB] | [varchar](20) |
| [czas] | [smalldatetime] | DATA OPERACJI |

[dbo].[samochod\_wyposazenie]

|  |  |  |
| --- | --- | --- |
| KOLUMNA | TYP | OPIS |
| [id\_wyposazenie] | [int] | KLUCZ GŁÓWNY |
| [informacje\_opis] | [varchar](50) | NAZWA WYPOSAŻENIA |

[dbo].[typ\_nadwozia]

|  |  |  |
| --- | --- | --- |
| KOLUMNA | TYP | OPIS |
| [id\_nadwozie] | [int] | KLUCZ GŁÓWNY |
| [nadwozie\_typ] | [char](20) | NAZWA NADWOZIA |
| [nadw\_informacje\_dodatkowe] | [varchar](50) |  |

[dbo].[model]

|  |  |  |
| --- | --- | --- |
| KOLUMNA | TYP | OPIS |
| [id\_model] | [int] | KLUCZ GŁÓWNY |
| [prod\_nazwa] | [varchar](50) | NAZWA PRODUCENTA |
| [model\_nazwa] | [char](50) | NAZWA MODELU |
| [model\_informacje\_dodatkowe] | [varchar](200) |  |

[dbo].[powiazanie\_wyposazenie\_samochod]

|  |  |  |
| --- | --- | --- |
| KOLUMNA | TYP | OPIS |
| [id\_samochod] | [int] | KLUCZ OBCY |
| [id\_wyposazenie] | [int] | KLUCZ OBCY |

WIDOKI

1. Widok przedstawia podstawowe dane klientów (S)

CREATE VIEW [dbo].[v\_klient\_dane]

AS

SELECT dbo.klient.imie,

dbo.klient.nazwisko,

dbo.klient.email,

dbo.klient.kontakt,

dbo.klient\_dane.adres\_linia\_1,

dbo.klient\_dane.adres\_linia\_2,

dbo.klient\_dane.miasto,

dbo.klient\_dane.kod\_pocztowy,

dbo.klient.typ,

dbo.klient.id\_klienta

FROM dbo.klient

LEFT OUTER JOIN dbo.klient\_dane ON

dbo.klient.id\_klienta = dbo.klient\_dane.id\_klient

WHERE(dbo.klient.typ = 'S')



1. Widok preferencji klientów

CREATE VIEW dbo.v\_klienci\_preferencje

AS

SELECT dbo.klient.imie,

dbo.klient.nazwisko,

dbo.klient\_preferencje.producent,

dbo.klient\_preferencje.model,

dbo.klient\_preferencje.rok\_produkcji,

dbo.klient\_preferencje.paliwo,

(

SELECT nadwozie\_typ

FROM dbo.typ\_nadwozia

WHERE dbo.klient\_preferencje.nadwozie = id\_nadwozie

) AS Expr1,

(

SELECT informacje\_opis

FROM dbo.samochod\_wyposazenie

WHERE dbo.klient\_preferencje.pole\_preferencji\_1 = id\_wyposazenie

) AS Pref\_1,

(

SELECT informacje\_opis

FROM dbo.samochod\_wyposazenie AS samochod\_wyposazenie\_2

WHERE dbo.klient\_preferencje.pole\_preferencji\_2 = id\_wyposazenie

) AS Pref\_2,

(

SELECT informacje\_opis

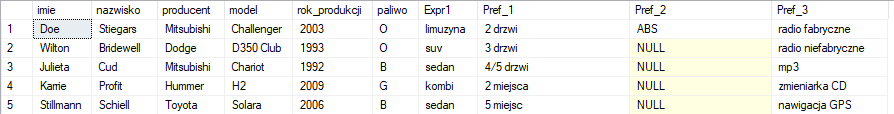
FROM dbo.samochod\_wyposazenie AS samochod\_wyposazenie\_1

WHERE dbo.klient\_preferencje.pole\_preferencji\_3 = id\_wyposazenie

) AS Pref\_3

FROM dbo.klient

INNER JOIN dbo.klient\_preferencje ON dbo.klient.id\_klienta = dbo.klient\_preferencje.id\_klient



1. Widok najpopularniejszych marek wybranych przez klientów w swoich preferencjach

CREATE VIEW [dbo].[v\_preferencje\_popularni\_producenci]

AS

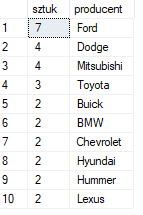
SELECT TOP 10 COUNT(producent) AS sztuk,

producent

FROM v\_klienci\_preferencje

GROUP BY producent

ORDER BY sztuk DESC



1. Widok pokazuje ilość samochodów poszczególnych marek na stanie komisu

CREATE VIEW [dbo].[v\_marki\_na\_stanie]

AS

SELECT prod\_nazwa AS marka,

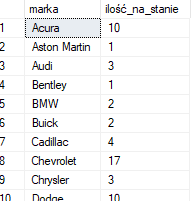
COUNT(\*) AS ilość\_na\_stanie

FROM dbo.v\_samochody\_do\_sprzedania

GROUP BY prod\_nazwa,

status;

GO



1. Widok samochodów do sprzedania ( bez rezerwacji )

CREATE VIEW [dbo].[v\_samochody\_do\_sprzedania]

AS

SELECT dbo.model.prod\_nazwa,

dbo.model.model\_nazwa,

dbo.samochody\_na\_sprzedaz.cena,

dbo.samochody\_na\_sprzedaz.przebieg,

dbo.samochody\_na\_sprzedaz.data\_produkcji,

dbo.samochody\_na\_sprzedaz.ksiazka\_serwisowa,

dbo.samochody\_na\_sprzedaz.pojemnosc,

dbo.samochody\_na\_sprzedaz.paliwo,

dbo.typ\_nadwozia.nadwozie\_typ,

dbo.samochody\_na\_sprzedaz.status,

dbo.samochody\_na\_sprzedaz.id\_samochod

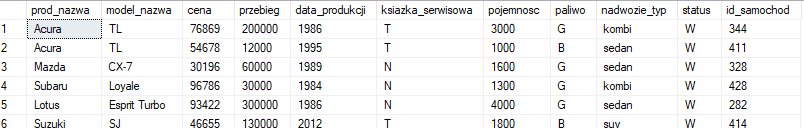
FROM dbo.model

LEFT OUTER JOIN dbo.samochody\_na\_sprzedaz ON dbo.model.id\_model = dbo.samochody\_na\_sprzedaz.id\_model

LEFT OUTER JOIN dbo.typ\_nadwozia ON dbo.samochody\_na\_sprzedaz.id\_nadwozie = dbo.typ\_nadwozia.id\_nadwozie

WHERE(dbo.samochody\_na\_sprzedaz.status = 'W');

GO



1. Widok przedstawia wszystkie sprzedane samochody

CREATE VIEW [dbo].[v\_samochody\_sprzedane]

AS

SELECT dbo.model.prod\_nazwa,

dbo.model.model\_nazwa,

dbo.samochody\_na\_sprzedaz.id\_model,

dbo.samochody\_na\_sprzedaz.cena,

dbo.samochody\_na\_sprzedaz.przebieg,

dbo.samochody\_na\_sprzedaz.data\_produkcji,

dbo.samochody\_na\_sprzedaz.ksiazka\_serwisowa,

dbo.samochody\_na\_sprzedaz.pojemnosc,

dbo.samochody\_na\_sprzedaz.paliwo,

dbo.typ\_nadwozia.nadwozie\_typ,

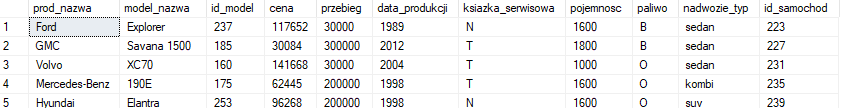
dbo.samochody\_na\_sprzedaz.id\_samochod

FROM dbo.model

INNER JOIN dbo.samochody\_na\_sprzedaz ON dbo.model.id\_model = dbo.samochody\_na\_sprzedaz.id\_model

INNER JOIN dbo.typ\_nadwozia ON dbo.samochody\_na\_sprzedaz.id\_nadwozie = dbo.typ\_nadwozia.id\_nadwozie

WHERE(dbo.samochody\_na\_sprzedaz.status = 'S')



1. Widok przedstawia raport pogrupowany po typach płatności (kredyt,leasing,gotówka)

CREATE VIEW [dbo].[v\_raport\_typu\_platnosci]

AS

SELECT SUM(cena) AS Suma,

typ\_platnosci,

COUNT(prod\_nazwa) AS ilość\_sprzedanych\_samochodów

FROM dbo.v\_raport\_sprzedaz

GROUP BY typ\_platnosci



1. Raport sprzedaży szczegółowy

CREATE VIEW [dbo].[v\_raport\_sprzedaz]

AS

SELECT dbo.model.prod\_nazwa,

dbo.model.model\_nazwa,

dbo.samochody\_na\_sprzedaz.vin,

dbo.sprzedane.cena,

dbo.sprzedane.data,

dbo.sprzedane.status,

dbo.sprzedane.id\_sprzedawcy,

dbo.sprzedane.typ\_platnosci,

dbo.klient.imie,

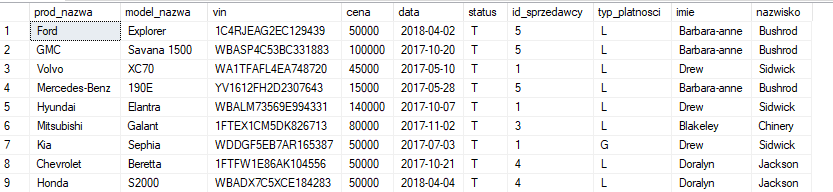
dbo.klient.nazwisko

FROM dbo.samochody\_na\_sprzedaz

INNER JOIN dbo.sprzedane ON dbo.samochody\_na\_sprzedaz.id\_samochod = dbo.sprzedane.id\_samochod

INNER JOIN dbo.model ON dbo.samochody\_na\_sprzedaz.id\_model = dbo.model.id\_model

INNER JOIN dbo.klient ON dbo.sprzedane.id\_sprzedawcy = dbo.klient.id\_klienta



1. Raport sprzedaży ogólny

CREATE VIEW [dbo].[wyniki\_sprzedawcy]

AS

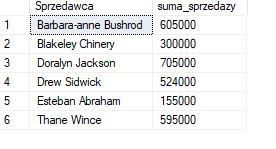
SELECT imie+' '+nazwisko AS Sprzedawca,

SUM(cena) AS suma\_sprzedazy

FROM dbo.v\_raport\_sprzedaz

GROUP BY nazwisko,

imie



1. Raport rezerwacji

CREATE VIEW [dbo].[v\_rezerwacje\_klient\_samochod]

AS

SELECT dbo.klient.imie,

dbo.klient.nazwisko,

dbo.klient.kontakt,

dbo.rezerwacje.id\_samochodu,

dbo.rezerwacje.id\_klienta,

dbo.rezerwacje.data\_do,

dbo.samochody\_na\_sprzedaz.vin,

dbo.samochody\_na\_sprzedaz.status,

dbo.model.prod\_nazwa,

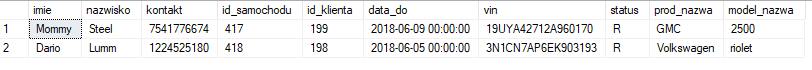
dbo.model.model\_nazwa

FROM dbo.klient

INNER JOIN dbo.rezerwacje ON dbo.klient.id\_klienta = dbo.rezerwacje.id\_klienta

INNER JOIN dbo.samochody\_na\_sprzedaz ON dbo.rezerwacje.id\_samochodu = dbo.samochody\_na\_sprzedaz.id\_samochod

INNER JOIN dbo.model ON dbo.samochody\_na\_sprzedaz.id\_model = dbo.model.id\_model



**PROCEDURY**

a.procedura dodania klienta , procedura zwraca haslo i login dla kienta

CREATE PROC dbo.dodaj\_klienta @imie VARCHAR(15),

@nazw VARCHAR(20),

@email VARCHAR(20),

@kontakt VARCHAR(20),

@typ CHAR(1)

AS

BEGIN

DECLARE @output VARCHAR(10), @login VARCHAR(10), @haslo VARCHAR(10);

EXEC gen\_haslo\_login

@output OUTPUT;

SET @login = @output;

EXEC gen\_haslo\_login

@output OUTPUT;

IF @imie IS NULL OR @nazw IS NULL OR @kontakt IS NULL

BEGIN

PRINT 'BRAK ARGUMENTU';

RETURN;

END;

IF EXISTS

(

SELECT \* FROM klient

WHERE imie = @imie AND nazwisko = @nazw

)

BEGIN

PRINT 'uzytkownik'+@imie+@nazw+'jest już w systemie';

RETURN;

END;

IF EXISTS

(

SELECT \* FROM klient

WHERE email = @email

)

BEGIN

PRINT 'adres mejlowy '+@email+'jest już w systemie';

RETURN;

END;

INSERT INTO klient (imie, nazwisko, email, kontakt, login, haslo, typ)

VALUES (@imie, @nazw, @email, @kontakt, @login, @output, @typ);

BEGIN

PRINT ' twój login '+@login+' i hasło '+@output;

END;

END;

GO

b. procedura usuniecia klienta

CREATE PROCEDURE [dbo].[p\_usun\_klienta] @nazwisko VARCHAR(50)

AS

BEGIN

DECLARE @id\_k INT;

IF @nazwisko IS NULL

BEGIN

PRINT 'BRAK ARGUMENTU';

RETURN;

END;

SET @id\_k =

(

SELECT id\_klienta

FROM klient

WHERE nazwisko = @nazwisko

);

IF @id\_k IS NULL

BEGIN

PRINT 'nie ma takiego klienta';

END;

DELETE FROM klient

WHERE nazwisko = @nazwisko;

c. procedura wyszukania klienta po nazwisku

CREATE PROCEDURE [dbo].[p\_wyszukiwanie\_klienta] @klient VARCHAR(50)

AS

BEGIN

IF @klient IS NULL

BEGIN

PRINT 'BRAK ARGUMENTU';

RETURN;

END;

IF NOT EXISTS

(

SELECT \*

FROM v\_klient\_dane

WHERE nazwisko = @klient

)

BEGIN

PRINT 'klient ---'+STR(@klient)+'--- NIE ISTNIEJE';

RETURN;

END;

SELECT \*

FROM v\_klient\_dane

WHERE nazwisko = @klient;

END

d. procedura zwraca ciag znaków do procedury dodaj klienta

CREATE PROC [dbo].[gen\_haslo\_login] @output VARCHAR(10) OUTPUT

AS

DECLARE @zakres INT= 74;

DECLARE @min INT= 48;

DECLARE @char CHAR;

DECLARE @dlug INT= 7;

SET @output = '';

WHILE @dlug > 0

BEGIN

SELECT @char = CHAR(ROUND(RAND() \* @zakres + @min, 0));

SET @output+=@char;

SET @dlug = @dlug - 1;

END;

RETURN

e. procedura dodawania samochodu do bazy

CREATE PROC [dbo].[dodaj\_samochod] @id\_model INT,

@id\_nadwozie INT,

@cena INT,

@vin VARCHAR(30),

@przebieg INT,

@data\_prod INT,

@kraj\_p CHAR(3),

@ks CHAR(1),

@pojemnosc INT,

@paliwo CHAR(1),

@info VARCHAR(50)

AS

BEGIN

IF @id\_model IS NULL

OR @id\_nadwozie IS NULL

OR @cena IS NULL

OR @vin IS NULL

BEGIN

PRINT 'BRAK ARGUMENTU';

RETURN;

END;

IF EXISTS

(

SELECT id\_samochod

FROM samochody\_na\_sprzedaz

WHERE vin = @vin

)

BEGIN

PRINT 'samochod'+@vin+'jest już w systemie';

RETURN;

END;

INSERT INTO samochody\_na\_sprzedaz (id\_model, id\_nadwozie, cena, vin, przebieg, data\_produkcji, status, kraj\_pochodzenia, ksiazka\_serwisowa, pojemnosc, paliwo, sam\_informacje\_dodatkowe)

VALUES(@id\_model, @id\_nadwozie, @cena, @vin, @przebieg, @data\_prod, 'W', @kraj\_p, @ks, @pojemnosc, @paliwo, @info);

BEGIN

SET @id\_model =

(

SELECT id\_samochod

FROM samochody\_na\_sprzedaz

WHERE vin = @vin

);

PRINT 'dodano nowy samochod do bazy - nr. '+@id\_model;

END;

END;

f. usuwanie samochodu z bazy

CREATE PROCEDURE [dbo].[p\_usun\_samochod] @id INT,

@vin VARCHAR(30)

AS

BEGIN

IF @id IS NULL AND @vin IS NULL

BEGIN

PRINT 'BRAK ARGUMENTU';

RETURN;

END;

IF NOT EXISTS

(

SELECT id\_samochod

FROM samochody\_na\_sprzedaz

WHERE vin = @vin

OR id\_samochod = @id

)

BEGIN

PRINT 'brak samochodu w bazie';

RETURN;

END;

DELETE FROM samochody\_na\_sprzedaz

WHERE vin = @vin

OR id\_samochod = @id;

END

g. procedura wyszukiwania samochodów po marce

CREATE PROCEDURE [dbo].[p\_szukaj\_marki] @producent\_samochod VARCHAR(50)

AS

BEGIN

IF @producent\_samochod IS NULL

BEGIN

PRINT 'BRAK ARGUMENTU';

RETURN;

END;

IF NOT EXISTS

(

SELECT \*

FROM v\_samochody\_do\_sprzedania

WHERE prod\_nazwa = @producent\_samochod

)

BEGIN

PRINT 'BRAK TEGO MODELU '+@producent\_samochod;

RETURN;

END;

SELECT \*

FROM v\_samochody\_do\_sprzedania

WHERE prod\_nazwa = @producent\_samochod;

END

h. procedura sprzedaży

CREATE PROCEDURE dbo.p\_sprzedaz

@id\_s INT ,

@id\_k INT ,

@cena INT ,

@typ CHAR(1) ,

@data DATE ,

@status CHAR(1) ,

@id\_sp INT

AS

BEGIN

IF EXISTS ( SELECT \*

FROM samochody\_na\_sprzedaz

WHERE id\_samochod = @id\_s

)

BEGIN

IF EXISTS ( SELECT \*

FROM klient

WHERE id\_klienta = @id\_k

)

BEGIN

IF ( SELECT status

FROM samochody\_na\_sprzedaz

WHERE id\_samochod = @id\_s

) = 'W'

BEGIN

INSERT INTO sprzedane ( id\_samochod , id\_klient , cena , typ\_platnosci , data , status , id\_sprzedawcy

)

VALUES ( @id\_s , @id\_k , @cena , @typ , @data , @status , @id\_sp

);

UPDATE samochody\_na\_sprzedaz

SET status = 'S'

WHERE id\_samochod = @id\_s;

END;

END;

END;

END;

1. Usuwanie / dodanie rezerwacji

CREATE PROCEDURE [dbo].[p\_zmiena\_rezerwacji] @id\_samochod INT,

@status CHAR(1),

@id\_klient INT,

@data DATE

AS

BEGIN

DECLARE @stary\_status CHAR(1);

IF @id\_klient IS NULL

OR @status IS NULL

BEGIN

PRINT 'BRAK ARGUMENTU';

RETURN;

END;

IF @status = 'R'

BEGIN

IF @data IS NULL

OR @id\_samochod IS NULL

OR @data <= GETDATE()

BEGIN

PRINT 'BRAK ARGUMENTU';

RETURN;

END;

IF NOT EXISTS

(

SELECT \*

FROM v\_samochody\_do\_sprzedania

WHERE id\_samochod = @id\_samochod

)

BEGIN

PRINT 'BRAK samochodu '+STR(@id\_samochod);

RETURN;

END;

IF NOT EXISTS

(

SELECT \*

FROM [dbo].[v\_klient\_dane]

WHERE id\_klienta = @id\_klient

)

BEGIN

PRINT 'BRAK klienta o numerze '+STR(@id\_samochod);

RETURN;

END;

INSERT INTO rezerwacje

(id\_klienta,

id\_samochodu,

data\_do

)

VALUES

(@id\_klient,

@id\_samochod,

@data

);

UPDATE samochody\_na\_sprzedaz

SET

status = @status

WHERE id\_samochod = @id\_samochod;

END;

ELSE

IF @status = 'W'

BEGIN

SET @id\_samochod =

(

SELECT id\_samochodu

FROM rezerwacje

WHERE id\_klienta = @id\_klient

);

UPDATE samochody\_na\_sprzedaz

SET

status = @status

WHERE id\_samochod = @id\_samochod;

DELETE FROM rezerwacje

WHERE id\_klienta = @id\_klient;

END;

ELSE

BEGIN

PRINT 'W - usuwanie rezerwacji / R - dodanie rezerwacji ';

END;

END