

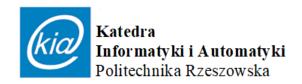


Baza danych Prezentacja SQL

Temat: "Baza danych do obsługi wypożyczalni samochodów"

Kamil Madej 161876 3EF-ZI





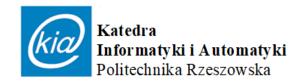
1.Tworzenie przykładowych tabel

1.1 Tabela wypożyczenia

```
CREATE TABLE `wypożyczenia` (
 'id' int NOT NULL AUTO_INCREMENT,
 `klient_id` int NOT NULL,
  `samochód_nr_rejestracyjny` varchar(10) NOT NULL,
  `pracownik_id` int NOT NULL,
  `data_wypożyczenia` date NOT NULL,
  'data_zwrotu' date DEFAULT NULL,
  `status_wypożyczenia_id` int NOT NULL,
  PRIMARY KEY ('id'),
 UNIQUE KEY 'id_UNIQUE' ('id'),
 KEY `fk_wypożyczenie_samochód1_idx` (`samochód_nr_rejestracyjny`),
 KEY `fk_wypożyczenie_status_wypożyczenia1_idx` (`status_wypożyczenia_id`),
 KEY `fk wypożyczenia klient idx` (`klient id`),
 KEY `fk_wypożyczenia_pracownik_idx` (`pracownik_id`),
 CONSTRAINT `fk_wypożyczenia_klient` FOREIGN KEY (`klient_id`) REFERENCES `klient` (`id`),
 CONSTRAINT `fk_wypożyczenia_pracownik` FOREIGN KEY (`pracownik_id`) REFERENCES `pracownik` ('id`),
 CONSTRAINT `fk_wypożyczenie_samochód1` FOREIGN KEY (`samochód_nr_rejestracyjny`), REFERENCES `samochód1` (`nr_rejestracyjny`),
 CONSTRAINT `fk_wypożyczenie_status_wypożyczenia1` FOREIGN KEY (`status_wypożyczenia_id`) REFERENCES `status_wypożyczenia` (`id`)
```

1.2 Tabela pracownik





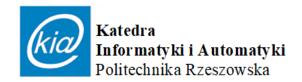
1.3 Tabela klient

```
CREATE TABLE `klient` (
    `id` int NOT NULL AUTO_INCREMENT,
    `prawo_jazdy` varchar(15) NOT NULL,
    `adres_id` int NOT NULL,
    `osoba_id` int NOT NULL,
    PRIMARY KEY (`id`),
    UNIQUE KEY `id_UNIQUE` (`id`),
    KEY `fk_klient_osoba_idx` (`id`,`osoba_id`),
    KEY `fk_klient_osoba_idx1` (`osoba_id`),
    KEY `fk_klient_adres_idx` (`adres_id`),
    CONSTRAINT `fk_klient_adres` FOREIGN KEY (`adres_id`) REFERENCES `adres` (`id`) ON DELETE CASCADE,
    CONSTRAINT `fk_klient_osoba` FOREIGN KEY (`osoba_id`) REFERENCES `osoba` (`id`) ON DELETE CASCADE
```

1.4 Tabela samochód

```
CREATE TABLE `samochód` (
  `nr rejestracyjny` varchar(10) NOT NULL,
  `przebieg` varchar(15) NOT NULL,
  `pojemność silnika` varchar(6) NOT NULL,
 `moc_silnika` varchar(4) NOT NULL,
  `rok_produkcji` varchar(4) NOT NULL,
 'typ paliwa' varchar(10) NOT NULL,
 `kolor` varchar(20) NOT NULL,
 `model id` int NOT NULL,
 `segment_auta_id` int NOT NULL,
  `dostępność` int NOT NULL,
  PRIMARY KEY (`nr_rejestracyjny`),
  UNIQUE KEY `id_UNIQUE` (`nr_rejestracyjny`),
  KEY `fk_dostępność_idx` (`dostępność`),
  KEY `fk_Samochód_Model1_idx` (`model_id`),
  KEY `fk_samochód_segment_auta_idx` (`segment_auta_id`),
  CONSTRAINT `fk_dostepność` FOREIGN KEY (`dostepność`) REFERENCES `dostepność` ('id'),
  CONSTRAINT `fk_samochód_model` FOREIGN KEY (`model_id`) REFERENCES `model` (`id`),
  CONSTRAINT `fk_samochód_segment_auta` FOREIGN KEY (`segment_auta_id`) REFERENCES `segment_auta` (`id`)
```





2. Wstawianie danych do tabel

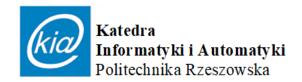
2.1 Tabela wypożyczenia

```
INSERT INTO `wypożyczenia` VALUES (1,10,'RNI 12HJ',2,'2020-01-01',2);
INSERT INTO `wypożyczenia` VALUES (2,20,'MFG DVD1',2,'2020-04-10',1);
INSERT INTO `wypożyczenia` VALUES (3,23,'RLE LCD4',2,'2019-12-19',1);
INSERT INTO `wypożyczenia` VALUES (4,24,'RNI 2137',2,'2020-01-09',2);
INSERT INTO `wypożyczenia` VALUES (5,25,'RNI 3H67',2,'2020-05-22',1);
INSERT INTO `wypożyczenia` VALUES (6,26,'RNI 456H',2,'2020-07-07',1);
INSERT INTO `wypożyczenia` VALUES (7,15,'GHD 9JK5',3,'2020-04-15',2);
```

2.2 Tabela samochód

```
INSERT INTO `samochód` VALUES ('ASDA RTG4','200000','40000','420','2017','Benzyna','Czarny',44,3,1);
INSERT INTO `samochód` VALUES ('FFF 5555','80000','2000','150','2010','Benzyna','Czarny',41,7,1);
INSERT INTO `samochód` VALUES ('GHD 9JK5','78000','1200','50','2013','Benzyna','Czarny',48,6,1);
INSERT INTO `samochód` VALUES ('KGU 0896', '543535', '3200', '310', '2010', 'Diesel', 'Czerwony', 5, 1, 1);
INSERT INTO `samochód` VALUES ('MFG DVD1','250000','1800','133','2000','Benzyna','Granatowy',9,10,2);
INSERT INTO `samochód` VALUES ('RIO SD55', '54660', '1550', '120', '2029', 'Diesel', 'Srebrny', 42,7,1);
INSERT INTO `samochód` VALUES ('RKL 9KGF','147200','2100','143','2006','Benzyna','Zielony',45,5,1);
INSERT INTO `samochód` VALUES ('RLE 10j7','125000','5000','4000','2012','Benzyna','Biały',46,5,1);
INSERT INTO `samochód` VALUES ('RLE LCD4','100000','1900','110','2012','Benzyna','Czarny',43,10,2);
INSERT INTO `samochód` VALUES ('RLU 4ULO', '20000', '1300', '260', '2017', 'Benzyna', 'Czarny', 47, 12, 1);
INSERT INTO `samochód` VALUES ('RNI 12HJ','14300','2500','180','2019','Benzyna','brqzowy',1,1,1);
INSERT INTO `samochód` VALUES ('RNI 2137','89000','2500','1570','2017','Benzyna','czerwony',17,2,1);
INSERT INTO `samochód` VALUES ('RNI 220P','47000','2800','193','2015','Benzyna','żółty',13,3,1);
INSERT INTO `samochód` VALUES ('RNI 22TM','123000','1400','75','2015','Benzyna','czarny',5,1,1);
INSERT INTO `samochód` VALUES ('RNI 3H67','90000','1600','90','2016','Benzyna','srebrny',21,6,2);
INSERT INTO `samochód` VALUES ('RNI 41TL','102000','1600','105','2008','Diesel','grantaowy',12,7,1);
INSERT INTO `samochód` VALUES ('RNI 44TL','99000','1800','125','2008','benzyna','czarny',6,8,1);
INSERT INTO `samochód` VALUES ('RNI 456H', '95600', '1900', '123', '2013', 'Benzyna', 'zielony', 17,5,2);
INSERT INTO `samochód` VALUES ('RNI 5HJ8','30000','3000','210','2019','Benzyna','czarny',7,6,1);
INSERT INTO `samochód` VALUES ('RNI 5J7F','2100000','1100','63','2010','Benzyna','czarny',4,3,1);
INSERT INTO `samochód` VALUES ('RNI 66LI','22000','1600','105','2017','Diesel','zielony',19,1,1);
INSERT INTO `samochód` VALUES ('RNI 66QR','180000','1800','125','2009','Benzyna','srebrny',22,2,1);
INSERT INTO `samochód` VALUES ('RNI 75TM','50000','2000','150','2015','Benzyna','czerwony',30,6,1);
INSERT INTO `samochód` VALUES ('RNI 9999','100000','1400','75','2005','Diesel','tososiowy',40,3,1);
INSERT INTO `samochód` VALUES ('RNI 99TL','73000','2600','210','2004','Benzyna','czarny',5,5,1);
INSERT INTO `samochód` VALUES ('RNI ISTF','173000','900','40','2009','Benzyna','beżowy',1,2,1);
INSERT INTO `samochód` VALUES ('RNI KK55','50000','3000','220','2009','Benzyna','Fioletowy',25,11,1);
INSERT INTO `samochód` VALUES ('UFO TR86', '80000', '1400', '86', '2017', 'Diesel', 'Srebrny', 40,6,1);
```





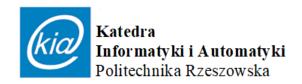
2.3 Tabela klient

```
INSERT INTO `klient` VALUES (10, 'Posiada',99,19);
INSERT INTO `klient` VALUES (11, 'Posiada', 101, 21);
INSERT INTO `klient` VALUES (12, 'Posiada', 102, 22);
INSERT INTO `klient` VALUES (13, 'Posiada', 103, 23);
INSERT INTO `klient` VALUES (14, 'Posiada', 104, 24);
INSERT INTO `klient` VALUES (15, 'Posiada', 105, 25);
INSERT INTO `klient` VALUES (16, 'Posiada', 106, 26);
INSERT INTO `klient` VALUES (17, 'Posiada', 107, 27);
INSERT INTO `klient` VALUES (18, 'Posiada', 108, 28);
INSERT INTO `klient` VALUES (19, 'Posiada', 109, 29);
INSERT INTO `klient` VALUES (20, 'Posiada',110,30);
INSERT INTO `klient` VALUES (21, 'Posiada',111,31);
INSERT INTO `klient` VALUES (22, 'Posiada', 112, 32);
INSERT INTO `klient` VALUES (23, 'Posiada',113,33);
INSERT INTO `klient` VALUES (24, 'Posiada',114,34);
INSERT INTO `klient` VALUES (25, 'Posiada',115,35);
INSERT INTO `klient` VALUES (26, 'Posiada',116,36);
INSERT INTO `klient` VALUES (27, 'Posiada', 117, 37);
```

2.4 Tabela pracownik

```
INSERT INTO `pracownik` VALUES (2,'Sprzedawca',5000,100,20);
INSERT INTO `pracownik` VALUES (3,'Sprzedawca',4000,118,38);
INSERT INTO `pracownik` VALUES (4,'Sprzedawca',2900,120,40);
```





3. Procedury

3.11Procedura "dodaj_klienta"

```
CREATE DEFINER* root @ localhost PROCEDURE 'dodaj_klienta' (
inic varchar(20), nazwisko varchar(29), data_urodzenia DATE, pesel varchar(11), nr_telefonu varchar(10), email varchar(45),
miejscowość varchar(25), kod_pocztowy varchar(18), ulica varchar(25),
prawo_jazdy varchar(15)
)

BEGIN

DECLARE idAdres INT DEFAULT 0;

DECLARE idOsoba INT DEFAULT 0;

IT NOT EXISTS (SELECT pesel,nr_telefonu,email from osoba NHERE osoba.pesel = pesel OR osoba.nr_telefonu = nr_telefonu OR osoba.email = email) THEN

INSERT INTO osoba(imie, nazwisko, data_urodzenia, pesel, nr_telefonu, email) VALUES(imie, nazwisko, data_urodzenia, pesel, nr_telefonu, email);
INSERT INTO adres(miejscowość, kod_pocztowy, ulica) VALUES(miejscowość, kod_pocztowy, ulica);

set idAdres = (select projekt_wypożyczalnia_samochodów.zwróc_adresID(miejscowość, kod_pocztowy, ulica) LIMIT 1);

set idOsoba = (select projekt_wypożyczalnia_samochodów.zwróc_id_osoba(imie, nazwisko, data_urodzenia, pesel, nr_telefonu, email) LIMIT 1);

INSERT INTO klient(prawo_jazdy, adres_id, osoba_id) VALUES(prawo_jazdy,idAdres, idOsoba);

ELSE

SELECT 'Osoba o podanych danych widnieje w bazie';

END IF;

END
```

3.12Procedura "dodaj_pracownika"

```
CREATE DEFINER= root @ localhost PROCEDURE 'dodaj_pracownika'(
imie varchar(20), nazwisko varchar(20), data_urodzenia DATE, pesel varchar(11), nn_telefonu varchar(10), email varchar(45),
miejscowość varchar(25), kod_pocttowy varchar(10), ulica varchar(25),
stanowisko varchar(25), pensja INT
)

BEGIN

DECLARE idAdres INT DEFAULT 0;

DECLARE idAdres INT DEFAULT 0;

IF NOT EXISTS (SELECT pesel,nn_telefonu,email from osoba WHERE osoba.pesel = pesel OR osoba.nn_telefonu = nn_telefonu OR osoba.email = email) THEN

INSERT INTO osoba(imie, nazwisko, data_urodzenia, pesel, nn_telefonu, email) VALUES(imie, nazwisko, data_urodzenia, pesel, nn_telefonu, email);
INSERT INTO adres(miejscowość, kod_pocztowy, ulica) VALUES(miejscowość, kod_pocztowy, ulica);

set idAdres = (select projekt_wypożyczalnia_samochodów.zwróc_adresID(miejscowość, kod_pocztowy, ulica) LIMIT 1);
set idOsoba = (select projekt_wypożyczalnia_samochodów.zwróc_id_osoba(imie, nazwisko, data_urodzenia, pesel, nn_telefonu, email) LIMIT 1);

INSERT INTO pracownik(stanowisko, pensja, adres, osoba) VALUES(stanowisko, pensja,idAdres, idOsoba);

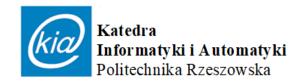
ELSE

SELECT 'Osoba o podanych danych widnieje w bazie';

END IF;

END
```





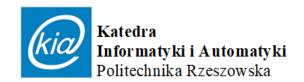
3.13 Procedura "dodaj_samochód"

```
nr_rejestracyjny varchar(10),przebieg VARCHAR(15), pojemność_silnika varchar(6), moc_silnika varchar(4), rok_produkcji varchar(4),
typ_paliwa varchar(10), kolor varchar(20),
marka nazwa varchar(20).
  odel_nazwa varchar(20),
segment_auta_nazwa varchar(20)
DECLARE idModel INT DEFAULT 0;
     INSERT INTO segment_auta(nazwa) values (segment_auta_nazwa);
     set idSegmentAuta = (select projekt_wypożyczalnia_samochodów.zwróć_id_segment_auta(segment_auta_nazwa));
IF NOT EXISTS (SELECT nazwa from marka WHERE marka, nazwa = marka nazwa ) THEN
     INSERT INTO marka(nazwa) values(marka_nazwa);
set idMarka = (select projekt_wypożyczalnia_samochodów.zwróć_id_marka(marka_nazwa));
IF NOT EXISTS (SELECT nazwa from model WHERE model.nazwa = model_nazwa ) THEN
     INSERT INTO model(nazwa, marka_id) values(model_nazwa, idMarka);
     set idModel = (select projekt_wypożyczalnia_samochodów.zwróć_id_model(model_nazwa));
IF NOT EXISTS (SELECT nr_rejestracyjny FROM samochód WHERE samochód.nr_rejestracyjny = nr_rejestracyjny) TH
   INSERT INTO samochod(nr_rejestracyjny, przebieg, pojemność_silnika, moc_silnika, rok_produkcji, typ_paliwa, kolor, model_id, segment_auta_id, dostępność)
VALUES(nr_rejestracyjny, przebieg, pojemność_silnika, moc_silnika, rok_produkcji, typ_paliwa, kolor, idModel, idSegmentAuta,i);
```

3.14 Procedura "dodaj_płatność"

```
root'@'localhost' PROCEDURE 'dodaj_płatność'(
wypożyczenie_id INT
    DECLARE idKlient INT DEFAULT 0;
    DECLARE idNrRejestracyjny TEXT DEFAULT '';
    DECLARE idSegmentAuta TEXT DEFAULT '';
    DECLARE idCennik INT DEFAULT 0;
    DECLARE różnicaDni INT DEFAULT 0;
    DECLARE cena INT DEFAULT 0;
    \textbf{set} \  \  \textbf{idKlient} = (\textbf{select} \  \, \textbf{projekt\_wypożyczalnia\_samochodów.zwr} \\ \textit{6-id\_klienta\_z\_wypożyczenia-(wypożyczenie\_id)} \  \, \textbf{LIMIT} \  \, \textbf{1});
    set idNrRejestracyjny = (select projekt_wypożyczalnia_samochodów.zwróć_nr_rejestracyjny_z_wypożyczenia(wypożyczenie_id));
    set idSegmentAuta = (select projekt_wypożyczalnia_samochodów.zwróć_segment_auta_z_nr_rejestracyjnego(idNrRejestracyjny));
    set idCennik = (SELECT id FROM cennik WHERE cennik.segment_auta_id = idSegmentAuta LIMIT 1);
    set różnicaDni = datediff((select data_zwrotu from wypożyczenia where wypożyczenia.id = wypożyczenie_id),
(select data_wypożyczenia from wypożyczenia where wypożyczenia.id = wypożyczenie_id);
    set cena = rożnicaDni * (select cena_za_dobe from cennik where cennik.id = idCennik);
    INSERT INTO płatność(wypożyczenia, klient, cennik,cena) VALUES(wypożyczenie_id, idKlient,idCennik, cena);
    call projekt_wypożyczalnia_samochodów.zmień_dostępność_samochodu(idNrRejestracyjny,1);
    UPDATE wypożyczenia
    SET status_wypożyczenia_id = 2
    WHERE wypożyczenia.id = wypożyczenie_id;
END
```





3.15Procedura "dodaj_wypożyczenie"

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `dodaj_wypożyczenie`(
klient_id INT, samochód_nr_rejestracyjny VARCHAR(10), pracownik_id INT, data_wypożyczenia DATE, data_zwrotu DATE
)

BEGIN

INSERT INTO wypożyczenia(klient_id, samochód_nr_rejestracyjny, pracownik_id, data_wypożyczenia, data_zwrotu, status_wypożyczenia_id)

VALUES (klient_id, samochód_nr_rejestracyjny, pracownik_id, data_wypożyczenia, data_zwrotu, 1);

call projekt_wypożyczalnia_samochodów.zmień_dostępność_samochodu(idNrRejestracyjny,2);
END
```

3.16Procedura "usuń_adres"

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `usuń_adres`(id INT)

BEGIN

delete from adres

where adres.id = id;

END
```

3.17Procedura "usuń_klienta"

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `usuń_klienta`(id INT)

BEGIN

DELETE adres, osoba, klient

FROM klient

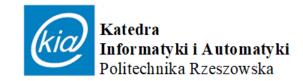
INNER JOIN adres on klient.adres_id = adres.id

INNER JOIN osoba on klient.osoba_id = osoba.id

WHERE klient.id = id;

END
```





3.18 Procedura "usuń_pracownika"

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `usuń_pracownika`(id INT)

BEGIN

DELETE adres, osoba, pracownik

FROM pracownik

INNER JOIN adres on pracownik.adres = adres.id

INNER JOIN osoba on pracownik.osoba = osoba.id

WHERE pracownik.id = id;

END
```

3.19 Procedura "usuń_samochód"

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `usuń_samochód`(nr_rejestracyjny varchar(10))

BEGIN

DELETE SAMOCHÓD

FROM samochód

WHERE (samochód.nr_rejestracyjny = nr_rejestracyjny);

END
```

3.20 Procedura "wyświetl_ilość_auta_wszystkich_marek"

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `wyświetl_ilość_auta_wszystkich_marek`()

BEGIN

SELECT COUNT(marka_id) as Ilość, marka.nazwa as 'Marka'

FROM model

INNER JOIN marka

ON model.marka_id = marka.id

group by marka.nazwa

ORDER BY Ilość DESC;
```





3.21 Procedura "wyświetl_płatności"

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `wyświetl_płatności`()
BEGIN
    select płatność.id, wypożyczenia as "NR wypożyczenia", osoba.imie as 'Imię klienta', osoba.nazwisko as 'Nazwisko klienta',
    samochód.nr_rejestracyjny as 'NR rejestracyjnya smochodu', cennik.cena_za_dobe as 'Cena za dobe',płatność.cena as 'Cena'
   from płatność
   TNNER JOTN klient
    on płatność.klient = klient.id
   INNER JOIN osoba
   on klient.osoba_id = osoba.id
   INNER JOIN cennik
   on płatność.cennik = cennik.id
   INNER JOIN wypożyczenia
   on płatność.wypożyczenia = wypożyczenia.id
    INNER JOIN samochód
    on wypożyczenia.samochód_nr_rejestracyjny = samochód.nr_rejestracyjny;
END
```

3.22Procedura "wyświetl_samochód_na_podstawie_rejestracji"

```
CREATE DEFINER='root'@'localhost' PROCEDURE 'wyświetl_samochód_na_podstawie_rejestracji'(nr_rejestracyjny varchar(20))

BEGIN

SELECT marka.nazwa as "Marka", model.nazwa as 'Model', rok_produkcji as 'Rok produkcji', przebieg as 'Przebieg', moc_silnika as 'Moc',
pojemność_silnika as 'Pojemność', nr_rejestracyjny as 'Nr rejestracyjny', typ_paliwa as 'Typ paliwa', kolor as 'Kolor', segment_auta.nazwa as 'Segment',
dostępność.Opis
   FROM samochód

INNER JOIN model

ON samochód.model_id = model.id

INNER JOIN marka
on model.marka_id = marka.id

INNER JOIN dostępność
ON samochód.dostępność = dostępność.id

INNER JOIN segment_auta
ON samochód.segment_auta

ON samochód.segment_auta_id = segment_auta.id

WHERE samochód.nr_rejestracyjny = nr_rejestracyjny;
END
```





3.23 Procedura "wyświetl_wszystkich_klientów"

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `wyświetl_wszystkich_klientów`(
)

BEGIN

SELECT klient.id, prawo_jazdy, osoba.imie, osoba.nazwisko, osoba.data_urodzenia, osoba.pesel,

osoba.nr_telefonu, osoba.email, adres.miejscowość, adres.kod_pocztowy, adres.ulica

FROM klient

INNER JOIN osoba

ON klient.osoba_id = osoba.id

INNER JOIN adres

ON klient.adres_id = adres.id;

END
```

3.24 Procedura "wyświetl_wszystkich_pracowników"

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `wyświetl_wszystkich_pracowników`()

BEGIN

SELECT pracownik.id,stanowisko, pensja, osoba.imie, osoba.nazwisko, osoba.data_urodzenia, osoba.pesel,

osoba.nr_telefonu, osoba.email, adres.miejscowość, adres.kod_pocztowy, adres.ulica

FROM pracownik

INNER JOIN osoba

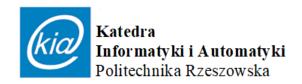
ON pracownik.osoba = osoba.id

INNER JOIN adres

ON pracownik.adres = adres.id;

END
```





3.27 Procedura "wyświetl_wszystkie_samochodyi"

```
CREATE DEFINER='root'@'localhost' PROCEDURE 'wyświetl_wszystkie_samochody'()

BEGIN

SELECT marka.nazwa as "Marka", model.nazwa as 'Model', rok_produkcji as 'Rok produkcji',przebieg as 'Przebieg',
moc_silnika as 'Moc',pojemność_silnika as 'Pojemność', nr_rejestracyjny as 'Nr rejestracyjny',typ_paliwa as 'Typ paliwa', kolor as 'Kolor',
segment_auta.nazwa as 'Segment', dostępność.Opis

FROM samochód

INNER JOIN model

ON samochód.model_id = model.id

INNER JOIN marka
on model.marka_id = marka.id

INNER JOIN dostępność
ON samochód.dostępność = dostępność.id

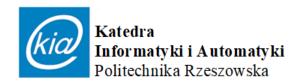
INNER JOIN segment_auta
ON samochód.segment_auta
ON samochód.segment_auta_id = segment_auta.id

ORDER BY rok_produkcji DESC;
```

3.28 Procedura "wyświetl_wszystkie_wypożyczenia"

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `wyświetl_wszystkie_wypożyczenia`()
BEGIN
SELECT wypożyczenia.id, osoba.imie, osoba. nazwisko , marka.nazwa as marka, model.nazwa as model,
samochód.nr_rejestracyjny,data_wypożyczenia,data_zwrotu, status_wypożyczenia.opis as status_wypożyczenia
   FROM wypożyczenia
   INNER JOIN samochód
   ON wypożyczenia.samochód_nr_rejestracyjny = samochód.nr_rejestracyjny
   INNER JOIN model
   ON samochód.model id = model.id
   INNER JOIN marka
   ON model.marka_id = marka.id
   INNER JOIN klient
   on wypożyczenia.klient_id = klient.id
   INNER JOIN osoba
   ON klient.osoba_id = osoba.id
   INNER JOIN status_wypożyczenia
   ON wypożyczenia.status_wypożyczenia_id = status_wypożyczenia.id
   ORDER BY data_wypożyczenia;
```





3.29 Procedura "wyświetl_wypożyczenia_zrealizowane"

```
CREATE DEFINER='root'@'localhost' PROCEDURE 'wyświetl_wypożyczenia_zrealizowane'()
BEGIN

SELECT wypożyczenia.id, osoba.imie, osoba. nazwisko , marka.nazwa as marka,
model.nazwa as model, samochód.nr_rejestracyjny,data_wypożyczenia,data_zwrotu

FROM wypożyczenia

INNER JOIN samochód
ON wypożyczenia.samochód_nr_rejestracyjny = samochód.nr_rejestracyjny

INNER JOIN model
ON samochód.model_id = model.id

INNER JOIN marka
ON model.marka_id = marka.id

INNER JOIN klient
on wypożyczenia.klient_id = klient.id

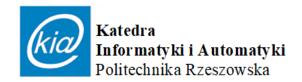
INNER JOIN osoba
ON klient.osoba_id = osoba.id

WHERE wypożyczenia.status_wypożyczenia_id = 2;
END
```

3.30 Procedura "wyświetl_wypożyczenia_w_trakcie_realizacji"

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `wyświetl_wypożyczenia_w_trakcie_realizacji`()
SELECT wypożyczenia.id, osoba.imie, osoba. nazwisko , marka.nazwa as marka, model.nazwa as model,
samochód.nr_rejestracyjny,data_wypożyczenia,data_zwrotu
   FROM wypożyczenia
   INNER JOIN samochód
   ON wypożyczenia.samochód_nr_rejestracyjny = samochód.nr_rejestracyjny
   INNER JOIN model
   ON samochód.model id = model.id
   INNER JOIN marka
   ON model.marka_id = marka.id
   INNER JOIN klient
   on wypożyczenia.klient_id = klient.id
   INNER JOIN osoba
   ON klient.osoba_id = osoba.id
   WHERE wypożyczenia.status_wypożyczenia_id = 1;
END
```





3.31 Procedura "wyświetl_wypożyczenie_na_podstawie_parametru"

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `wyświetl_wypożyczenie_na_podstawie_parametru` (Wpisz_zrealizowane_lub_niezrealizowane varchar(20))

BEGIN

IF(Wpisz_zrealizowane_lub_niezrealizowane = 'zrealizowane') THEN

call projekt_wypożyczalnia_samochodów.wyświetl_wypożyczenia_zrealizowane();

ELSEIF (Wpisz_zrealizowane_lub_niezrealizowane = 'niezrealizowane') THEN

call projekt_wypożyczalnia_samochodów.wyświetl_wypożyczenia_w_trakcie_realizacji();

ELSE

SELECT 'Wpisano złe dane';

END IF;
```

3.32 Procedura "zmień_dostępność_samochodu"

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `zmień_dostępność_samochodu`(nr_rejestracyjny varchar(10), parametr INT)

BEGIN

IF(parametr = 1)THEN

UPDATE samochód

SET dostępność = 1

WHERE samochód.nr_rejestracyjny = nr_rejestracyjny;

ELSEIF(parametr = 2) THEN

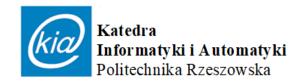
UPDATE samochód

SET dostępność = 2

WHERE samochód.nr_rejestracyjny = nr_rejestracyjny;

END IF;
```





4.Funkcje

4.1Funkcja "zwróć_adresID"

```
CREATE DEFINER='root'@'localhost' FUNCTION 'zwróc_adresID'(
miejscowość varchar(25), kod_pocztowy varchar(10), ulica varchar(25)
) RETURNS int
BEGIN

DECLARE idAdres INT DEFAULT 0;

set idAdres = (select id from adres where adres.miejscowość = miejscowość AND adres.kod_pocztowy = kod_pocztowy AND adres.ulica = ulica LIMIT 1);

RETURN idAdres;
END
```

4.2 Funkcja "zwróć_id_klienta_z_wypożyczenia"

```
CREATE DEFINER=`root`@`localhost` FUNCTION `zwróć_id_klienta_z_wypożyczenia`(id_wypożyczenia INT) RETURNS int

BEGIN

DECLARE idKlient INT DEFAULT 0;

set idKlient = (select klient_id from wypożyczenia where wypożyczenia.id = id_wypożyczenia LIMIT 1);

RETURN idKlient;

END
```

4.3 Funkcja "zwróć_id_marka"

```
CREATE DEFINER=`root`@`localhost` FUNCTION `zwróć_id_marka`(nazwa varchar(20)) RETURNS int

BEGIN

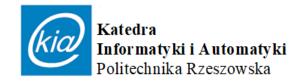
DECLARE idMarka INT DEFAULT 0;

set idMarka = (select id from marka where marka.nazwa = nazwa LIMIT 1);

RETURN idMarka;

END
```





4.4 Funkcja "zwróć_id_model"

```
CREATE DEFINER=`root`@`localhost` FUNCTION `zwróć_id_model`(nazwa varchar(20)) RETURNS int

BEGIN

DECLARE idModel INT DEFAULT 0;

set idModel = (select id from model where model.nazwa = nazwa LIMIT 1);

RETURN idModel;
END
```

4.5 Funkcja "zwróć_id_osoba"

```
CREATE DEFINER='root'@'localhost' FUNCTION 'zwróc_id_osoba'(
imie varchar(20), nazwisko varchar(20), data_urodzenia DATE, pesel varchar(11),nr_telefonu varchar(10), email varchar(45)

) RETURNS INT

BEGIN

DECLARE idOsoba INT DEFAULT 0;

set idOsoba = (select id from osoba where osoba.imie = imie AND osoba.nazwisko = nazwisko AND osoba.data_urodzenia = data_urodzenia AND osoba.pesel = pesel

AND |
DSoba.nr_telefonu = nr_telefonu AND osoba.email = email LIMIT 1);

RETURN idOsoba;

END
```

4.6 Funkcja "zwróć_id_segment_auta"

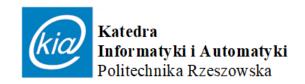
```
CREATE DEFINER=`root`@`localhost` FUNCTION `zwróć_id_segment_auta`(nazwa varchar(20)) RETURNS int
BEGIN

DECLARE idSegment INT DEFAULT 0;

set idSegment = (select id from segment_auta where segment_auta.nazwa = nazwa LIMIT 1);

RETURN idSegment;
END
```





4.7 Funkcja "zwróć_nr_rejestracyjny_z_wypożyczenia"

```
CREATE DEFINER=`root`@`localhost` FUNCTION `zwróć_nr_rejestracyjny_z_wypożyczenia`(id_wypożyczenia INT) RETURNS text CHARSET utf8mb4
BEGIN

DECLARE nr_rejestracyjny TEXT DEFAULT '';

set nr_rejestracyjny = (select samochód_nr_rejestracyjny from wypożyczenia where wypożyczenia.id = id_wypożyczenia LIMIT 1);

RETURN nr_rejestracyjny;
END
```

4.8 Funkcja "zwróć_segment_auta_z_nr_rejestracyjnego"

```
CREATE DEFINER='root'@'localhost' FUNCTION 'zwróć_segment_auta_z_nr_rejestracyjnego'(nr_rejestracyjny varchar(10)) RETURNS text CHARSET utf8mb4

BEGIN

DECLARE segment_auta TEXT DEFAULT '';

set segment_auta = (SELECT segment_auta_id FROM samochód WHERE samochód.nr_rejestracyjny = nr_rejestracyjny);

RETURN segment_auta;

END
```