

Wstęp do baz danych

Projekt: Serwis Samochodowy

1. Zakres i cel projektu (opis założeń funkcjonalnych projektowanej bazy danych)

2. Definicja systemu

2.1 Perspektywy użytkowników

3. Model konceptualny

3.1 Definicja zbiorów encji określonych w projekcie (decyzje projektowe)

3.2 Ustalenie związków i ich typów między encjami

3.3 Klucze kandydujące i główne (decyzje projektowe)

3.4 Schemat ER na poziomie konceptualnym

3.5 Problem pułapek szczelinowych i wachlarzowych - analiza i przykłady

4. Model logiczny

4.1 Charakterystyka modelu relacyjnego

4.2 Usunięcie właściwości niekompatybilnych z modelem relacyjnym - przykłady

4.3 Proces normalizacji - analiza i przykłady

4.4 Schemat ER na poziomie modelu logicznego

4.5 Więzy integralności

4.6 Proces denormalizacji - analiza i przykłady

5. Faza fizyczna

5.1 Projekt transakcji i weryfikacja ich wykonalności

5.2 Strojanie bazy danych - dobór indeksów

5.3 Skrypt SQL zakładający bazę danych

5.4 Przykłady zapytań i poleceń SQL odnoszących się do bazy danych

1. Zakres i cel projektu

Celem projektu jest zaprojektowanie relacyjnej bazy danych na poziomie koncepcyjnym i logicznym oraz jej fizyczna implementacja. Baza danych obsługuje serwis samochodowy i będzie oparta na rozwiązaniach firmy Oracle, z wykorzystaniem języka SQL

Oprogramowanie użyte w realizacji projektu:

- TOAD Data Modeler
- SQL Developer
- Database Oracle 11g Release 2

Założenia funkcjonalne:

Projekt dotyczy sieci serwisów samochodowych zarządzanych, przez główny Serwis. Serwis może posiadać jednego lub kilku właścicieli oraz zatrudniać pracowników, którzy muszą być przydzieleni do konkretnej placówki. Każdy pracownik otrzymuje miesięczne wynagrodzenie adekwatne do obejmowanego stanowiska. Serwis przyjmuje samochody na wizytę i wykonuje określone naprawy w ramach wizyty. Wizytę klient może zarezerwować oraz zobaczyć status swojej naprawy.

2. Definicja systemu

Funkcjonalność systemu zakłada:

1. Dodawanie, usuwanie, modyfikacja Serwisu
2. Podgląd danych Serwisu
3. Dodawanie, usuwanie, modyfikacja właścicieli
4. Podgląd danych właścicieli
5. Dodawanie, usuwanie, modyfikacja placówek
6. Podgląd danych placówki
7. Dodawanie, usuwanie, modyfikacja pracowników
8. Podgląd danych pracownika
9. Dodawanie, usuwanie, modyfikacja wynagrodzeń pracowników
10. Podgląd wynagrodzeń pracownika
11. Dodawanie, usuwanie, modyfikacja stanowisk w placówce
12. Podgląd stanowiska
13. Podgląd pracowników na danym stanowisku
14. Dodawanie, usuwanie, przyjętych obecnie samochodów
15. Podgląd danych przyjętych samochodów
16. Dodawanie, usuwanie, modyfikacja danych klientów
17. Podgląd danych klienta
18. Dodawanie, usuwanie, modyfikacja historii przyjęć
19. Podgląd danych dotyczących napraw
20. Dodawanie, usuwanie, modyfikacja napraw

21. Podgląd danych dotyczących trwających napraw
22. Dodawanie, usuwanie, modyfikacja rezerwacji
23. Podgląd rezerwacji
24. Nadawanie uprawnień do modyfikacji bazy danych

2.1 Perspektywy użytkowników

- Administrator

Posiada dostęp do wszystkich funkcjonalności systemu i modyfikacji bazy danych. Ma uprawnienia administratora bazy danych Oracle.

- Właściciel Serwisu

Ma dostęp do wszystkich danych dotyczących serwisu oraz placówek pod niego podlegających. Ma możliwość podglądu informacji związanych z pracownikami, wynagrodzeniami, stanowiskami, klientami i naprawionymi samochodami.

- Kierownik placówki

Ma możliwość podglądu danych związanych z własną Warsztat i jej klientami.

Ma dostęp do danych pracowników, stanowisk, wynagrodzeń, naprawianych samochodów, napraw i samochodów zastępczych w obrębie zarządzanej placówki.

- Księgowy

Ma możliwość zarządzania finansami placówki i sprawami związanymi z pracownikami. Dostęp do wszystkich danych pracowników oraz informacji o transakcjach dokonanych przez klientów.

-Pracownik

Pomoc i obsługa klienta - odpowiedzialny za przyjęcie samochodu do naprawy,, wydanie samochodu zastępczego, rezerwacje terminu naprawy oraz informacji o aktualnym dostępności serwisu.

- Klient

Ma wgląd we własne dane, ma możliwość modyfikowania niektórych z nich(nr kontaktowy). Ma podgląd dokonanej rezerwacji i zaawansowania naprawy.

Tabela przedstawia funkcje, które są dostępne dla każdej z powyższych perspektyw:

FUNKCJA	Administrator	Właściciel Serwisu	Kierownik Warsztatu	Księgowy	Pracownik	Klient
1	+	-	-	-	-	-
2	+	+	+	+	+	+
3	+	-	-	-	-	-
4	+	+	+	+	-	-
5	+	-	-	-	-	-
6	+	+	+	+	+	+
7	+	-	-	+	-	-
8	+	+	+	+	-	-
9	+	-	-	+	-	-
10	+	+	+	+	-	-
11	+	-	-	-	-	-
12	+	+	+	+	-	-
13	+	+	+	+	-	-
14	+	+	+	-	-	+
15	+	+	+	+	-	+
16	+	+	+	-	-	+
17	+	+	+	-	+	+
18	+	+	+	-	-	-
20	+	-	+	-	-	-
21	+	+	+	-	+	+
22	+	+	+	-	+	+
23	+	+	+	-	+	+
24	+	-	-	-	-	-

3. Model konceptualny

3.1 Definicja zbiorów encji określonych w projekcie

Serwis - istnieje tylko jeden obiekt tej encji, a wszystkie atrybuty przedstawiono poniżej.

Nazwa atrybutu	Typ, dziedzina	Obowiązkowy	Unikatowy	Opis
Serwis_ID	Integer	TAK	TAK	Klucz główny, numer identyfikujący serwis
Nazwa	VarChar(30)	TAK	TAK	Nazwa serwisu
Adres	VarChar(50)	TAK	NIE	Adres Serwisu, pole segmentowe
Właściciel	VarChar(100)	TAK	NIE	Imię, nazwisko, pole wielowartościowe
Data_zalozenia	Date	TAK	NIE	Data założenia serwisu
NIP	Char(10)	TAK	TAK	Numer Identyfikacji Podatkowej
Nr_telefonu	VarChar(12)	TAK	TAK	Numer telefonu do serwisu

Warsztat - istnieje wiele obiektów tej encji. Atrybuty w tabelce opisują poszczególne cechy warsztatu.

Nazwa atrybutu	Typ, dziedzina	Obowiązkowy	Unikatowy	Opis
Warsztat_ID	Integer	TAK	TAK	Klucz główny, numer identyfikujący warsztat

Nazwa	VarChar(30)	TAK	TAK	Nazwa warsztatu
Adres	VarChar(50)	TAK	NIE	Adres placówki, pole segmentowe
Kierownik	VarChar(50)	TAK	Nie	Imię nazwisko kierownika, pole segmentowe
Godz_otwarcia	Time	NIE	Nie	Godzina otwarcia warsztatu
Godz_zamknięcia	Time	NIE	Nie	Godzina zamknięcia Warsztatu
Data_otwarcia	Date	TAK	NIE	data otwarcia serwisu

Pracownik - istnieje wiele obiektów tej encji. Atrybuty w tabelce opisują poszczególne cechy pracownika.

Nazwa atrybutu	Typ, dziedzina	Obowiązkowy	Unikatowy	Opis
Pracownik_ID	Integer	TAK	TAK	Unikalny identyfikator pracownika
Imię	VarChar(20)	TAK	NIE	Imię Pracownika
Nazwisko	VarChar(50)	Tak	NIE	Nazwisko pracownika
Pesel	VarChar(11)	TAK	TAK	Pesel Pracownika
Data_urodzenia	Date	TAK	NIE	Data urodzenia pracownika
Stanowisko	VarChar(30)	TAK	NIE	Stanowisko pracownika

Data_Zatrudnienia	Date	TAK	NIE	Data zatrudnienia pracownika
Wynagrodzenie	Money	TAK	NIE	Wynagrodzenie pracownika
Adres	VarChar(50)	TAK	NIE	Adres zamieszkania pracownika
Email	VarChar(50)	NIE	TAK	Email pracownika
Nr_telefonu	VarChar(12)	NIE	NIE	Numer telefonu pracownika

Klient- istnieje wiele obiektów tej encji. Atrybuty w tabelce opisują poszczególne cechy klienta.

Nazwa atrybutu	Typ, dziedzina	Obowiązkowy	Unikatowy	Opis
Klient_ID	INTEGER	TAK	TAK	ID klienta
Imię	VarChar(20)	TAK	NIE	Imię klienta
Nazwisko	VarChar(50)	TAK	NIE	Nazwisko Klienta
Nr_telefonu	VarChar(12)	TAK	NIE	Numer kontaktowy Telefonu klienta
Email	VarChar(50)	TAK	TAK	Email klienta

Samochód - istnieje wiele obiektów tej encji. Atrybuty w tabelce opisują poszczególne cechy samochodu.

Nazwa atrybutu	Typ, dziedzina	Obowiązkowy	Unikatowy	Opis
Samochod_ID	INTEGER	TAK	TAK	ID samochodu
Marka	VarChar(15)	TAK	NIE	Marka samochodu
Model	VarChar(15)	TAK	NIE	Model samochodu
Nr_rejestracyjny	VarChar(10)	TAK	TAK	Numer tablicy rejestracyjnej samochodu
Vin	VarChar(18)	TAK	TAK	Numer vin karoserii
Data_przeglądu	Date	TAK	NIE	Data ważności ostatniego przeglądu

Część - istnieje wiele obiektów tej encji. Atrybuty w tabelce opisują poszczególne cechy części.

Nazwa atrybutu	Typ, dziedzina	Obowiązkowy	Unikatowy	Opis
czesci_ID	INTEGER	TAK	TAK	ID danej części
Nazwa	VarChar(20)	TAK	NIE	Nazwa części
Cena	Money	NIE	NIE	Cena danej części

Naprawa - istnieje wiele obiektów tej encji. Atrybuty w tabelce opisują poszczególne cechy naprawy.

Nazwa atrybutu	Typ, dziedzina	Obowiązkowy	Unikatowy	Opis
Naprawa_ID	INTEGER	TAK	TAK	ID naprawy

Rodzaj_naprawy	VarChar(20)	TAK	TAK	Nazwa naprawy
Koszta	Money	TAK	NIE	Koszt za daną naprawę

Wizyta - istnieje wiele obiektów tej encji. Atrybuty w tabelce opisują poszczególne cechy wizyty.

Nazwa atrybutu	Typ, dziedzina	Obowiązkowy	Unikatowy	Opis
Wizyta_ID	INTEGER	TAK	TAK	ID wizyty
Data	Date	TAK	NIE	Data wizyty
Czy zrealizowana	Boolean	TAK	NIE	Status realizacji wizyty
Opis	VarChar(500)	NIE	NIE	Opis przebiegu wizyty
Koszt	Money	TAK	NIE	Koszt całkowity wizyty

Rezerwacja - istnieje wiele obiektów tej encji. Atrybuty w tabelce opisują poszczególne cechy rezerwacji.

Nazwa atrybutu	Typ, dziedzina	Obowiązkowy	Unikatowy	Opis
Rezerwacja_ID	INTEGER	TAK	TAK	ID rezerwacji
Data	Date	TAK	NIE	Data rezerwacji

3.2 Ustalenie związków i ich typów między encjami

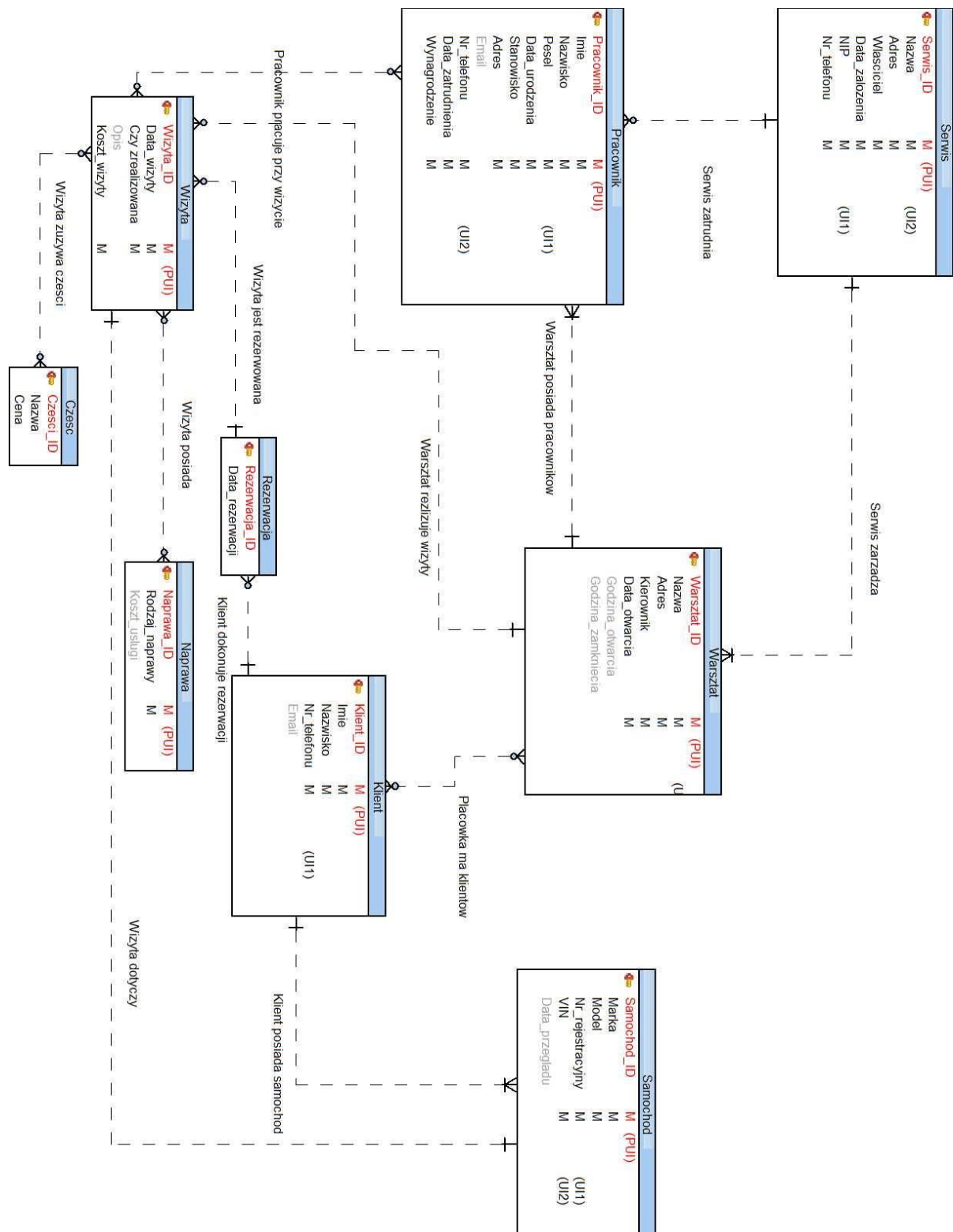
Encja 1	Encja 2	Typ związku	Typ uczestnictwa	Stopień związku	Nazwa
serwis	warsztat	1:n	obowiązkowy-obowiązkowy	binarny	Serwis ma placówki
serwis	pracownik	1:n	opcjonalny-opcjonalny	binarny	Serwis ma pracowników
warsztat	klient	m:n	opcjonalny-opcjonalny	binarny	Warsztat ma klientów
klient	samochód	1:n	obowiązkowy-obowiązkowy	binarny	Klient posiada samochód
warsztat	wizyta	1:n	obowiązkowy-obowiązkowy	binarny	Warsztat przyjmuje na wizyty
wizyta	naprawy	m:n	opcjonalny-opcjonalny	binarny	Wizyta zawiera naprawy
rezerwacja	wizyta	1:n	obowiązkowy-opcjonalny	binarny	Wizyta jest rezerwowana
warsztat	pracownik	1:n	obowiązkowy-obowiązkowy	binarny	Warsztat ma pracowników
klient	rezerwacja	1:n	obowiązkowy-opcjonalny	binarny	Klient dokonuje rezerwacji
wizyta	części	1:n	opcjonalny-opcjonalny	binarny	Wizyta zużywa części
pracownik	wizyta	m:n	opcjonalny-opcjonalny	binarny	Pracownik pracuje przy wizycie

3.3 Klucze kandydujące i główne (decyzje projektowe)

Każdy klucz główny został stworzony sztucznie - nazwa_encji_ID, aby uniknąć sytuacji, gdzie klucz kandydujący okaże się niejednoznacznym kluczem.

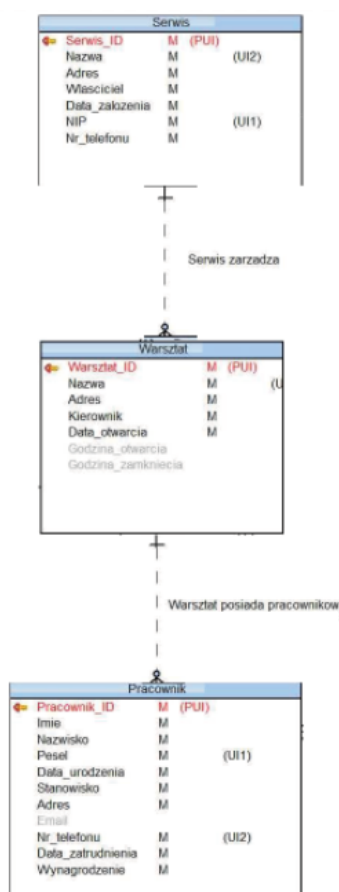
Nazwa encji	Potencjalny klucz kandydujący
Serwis	Nazwa, NIP
Warsztat	nazwa
Pracownik	PESEL,numer telefonu
Klient	nr_telefonu, email
samochód	VIN, nr_rejestracyjny
części	-
naprawa	rodzaj naprawy
Wizyta	-
rezerwacja	-

3.4 Schemat ER na poziomie konceptualnym

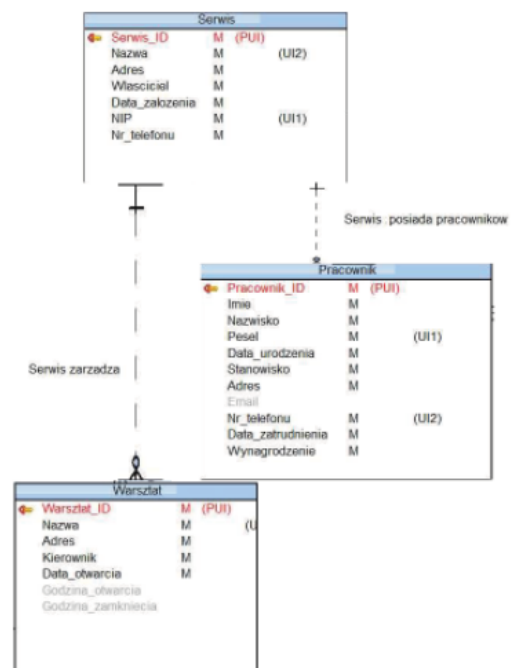


3.5 Problem pułapek szczelinowych i wachlarzowych - analiza i przykłady

Problem pułapki szczelinowej oraz wachlarzowej mogliśmy zauważyć w przypadku encji Serwis, Warsztat i Pracownik. Model z rys. a) pokazuje problem pułapki szczelinowej- model ten zakłada, że każdy Pracownik musi być zatrudniony przez Warsztat, nie uwzględnia więc sytuacji, w której Serwis zatrudnia swoich pracowników. Model z rys. b) pokazuje problem pułapki wachlarzowej- w takim przypadku nie jesteśmy w stanie określić, który pracownik należy do konkretnego Warsztatu. Problem pułapek został rozwiązany poprzez dodanie związku 1:n między Serwisem a Pracownikiem.



a)



b)

4. Model logiczny

4.1 Charakterystyka modelu relacyjnego

Model logiczny został wstępnie stworzony przez konwersję modelu konceptualnego w programie Toad Data Modeler 6.3. W celu uzyskania kompletnego modelu logicznego należało wyeliminować niekompatybilności z modelem relacyjnym.

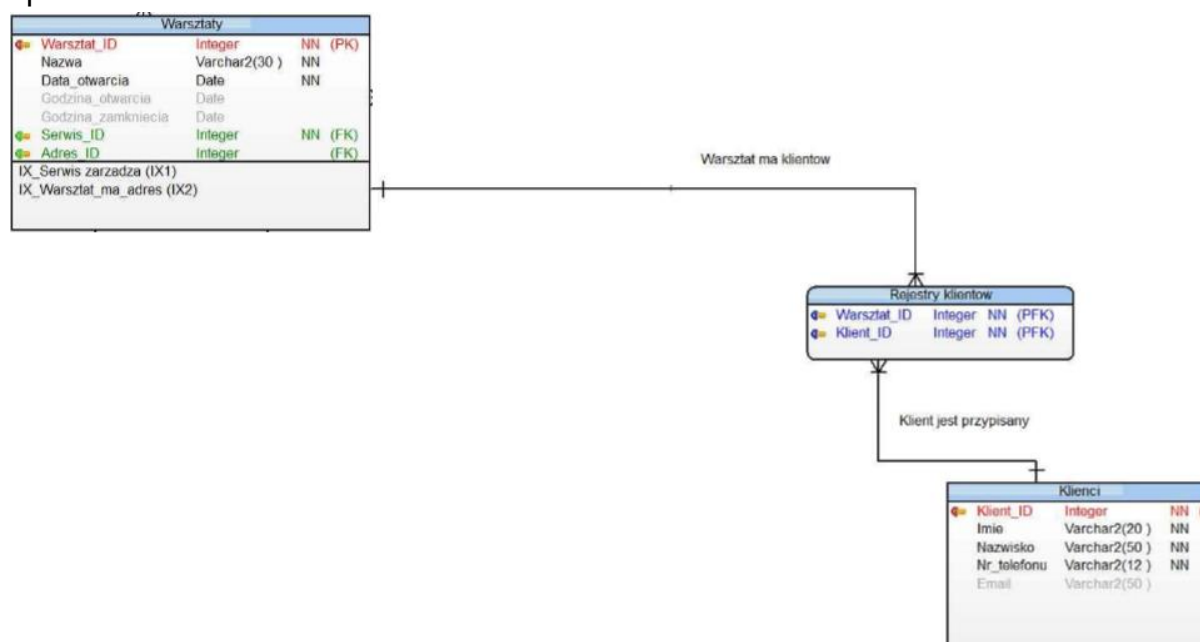
Przed wszystkim wszystkie związki “wielu do wielu” zostały zastąpione przez dwa związki “jeden do wielu” i tzw. tablicę bridge’ującą pomiędzy nimi, zawierającą klucze główne relacji wchodzących w skład związku. Wszystkie nazwy relacji otrzymały nazwy w liczbie mnogiej, gdyż tabele reprezentują na tym etapie zbiór encji. Każda z encji zawierała sztucznie dodany atrybut w postaci identyfikatora. W wyniku przekształcenia modelu stał się on kluczem głównym każdej z relacji. Wszystkie pozostałe atrybuty encji stają się w naturalny sposób nie głównymi atrybutami relacji. Konwersja modelu spowodowała wstawienie do odpowiednich tabel kluczy obcych encji związanych w stosunku “jeden do wielu”. W następnej kolejności dokonano procesów normalizacji i denormalizacji w celu uzyskania trzeciej postaci normalnej i spełnienia pozostałych wymagań.

4.2 Usunięcie właściwości niekompatybilnych z modelem relacyjnym - przykłady

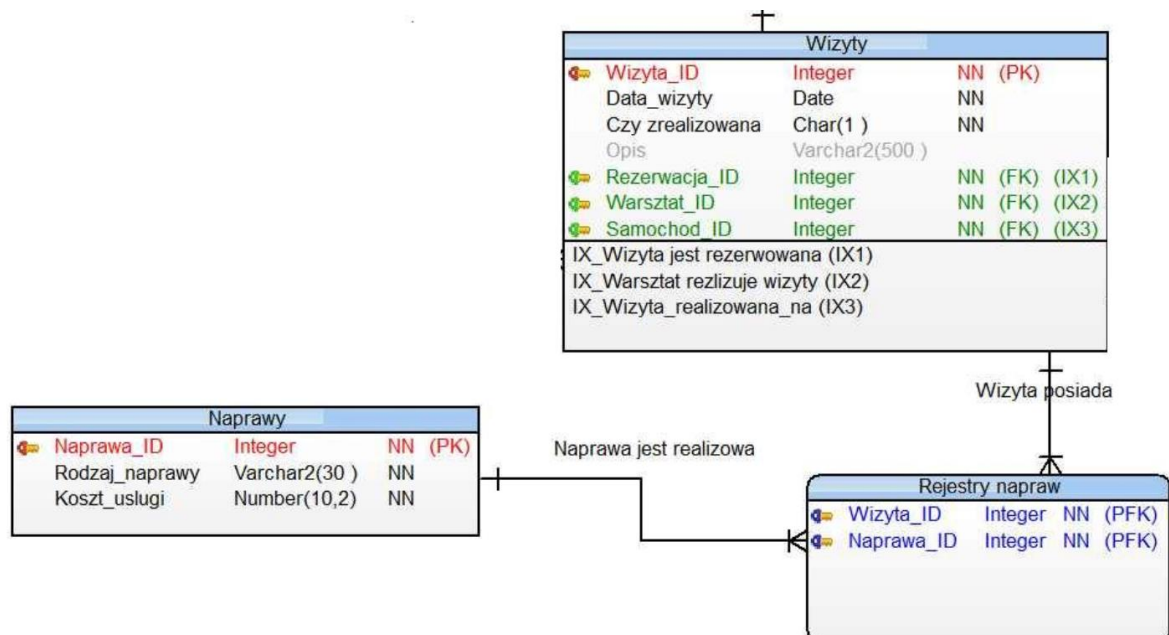
4.2.1 Usunięcie związków “wiele do wielu”

W modelu konceptualnym występowały dwa związki “wiele do wielu”. Były to związki pomiędzy encjami Pracownik - Klient, Wizyta - Naprawa oraz Pracownik - Wizyta. W miejsce tych związków pojawiły się trzy związki “jeden do wielu” oraz nowa relacja (bridge’ująca), zawierająca klucze główne encji wchodzących w skład takiego związku.

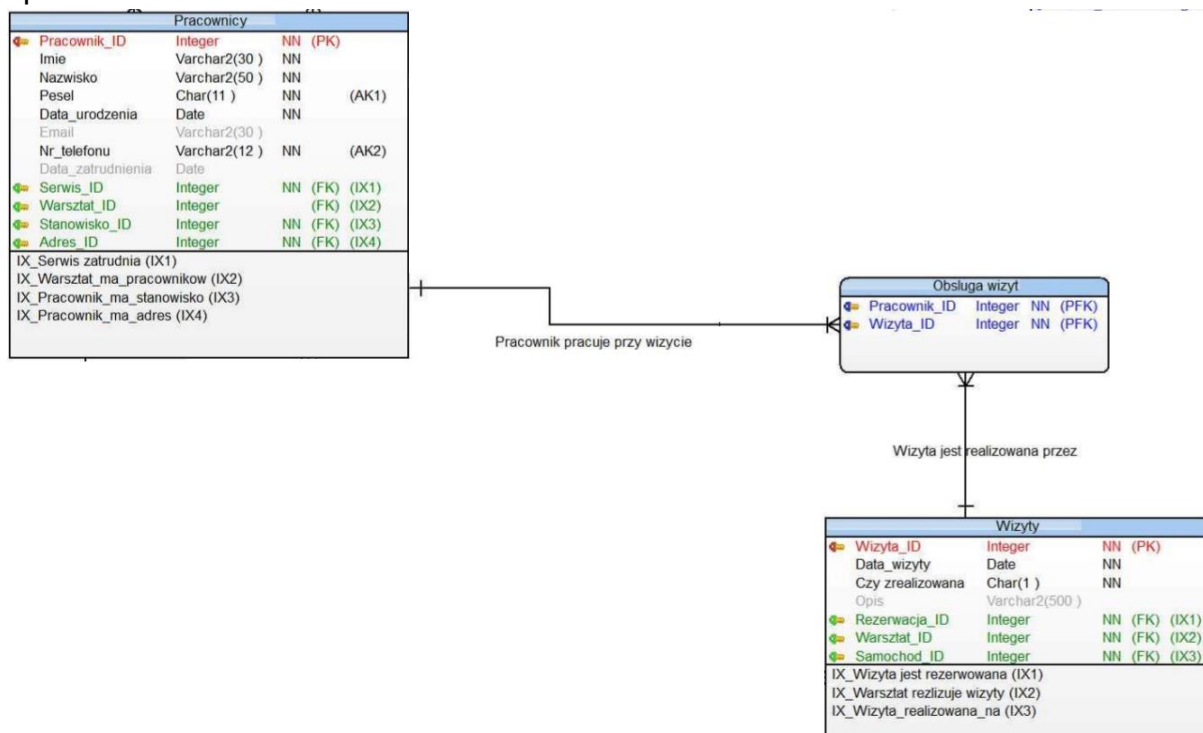
Związek pomiędzy encjami Warsztat - Klient został przekształcony w następujący sposób:



Związek pomiędzy encjami Wizyta - Naprawa został przekształcony w następujący sposób:



Związek pomiędzy encjami Pracownik - Wizyta został przekształcony w następujący sposób:



4.3 Proces normalizacji - analiza i przykłady

Projekt bazy danych zakładał, że ostatecznie będzie ona w co najmniej 3 postaci normalnej. W tym celu każdą relację występującą w zaprojektowanej bazie przekształcono do kolejnych postaci normalnych.

4.3.1 Przekształcenie relacji do 1 postaci normalnej

Relacja znajduje się w pierwszej postaci normalnej, jeśli każdy atrybut tej relacji jest atomowy tzn. nierozkładalny i nie ma powtarzających się grup. W celu przekształcenia relacji do 1 postaci normalnej należy wszystkie atrybuty wielowartościowe i segmentowe zastąpić nową relacją lub zestawem atrybutów, w przypadku gdy ich liczba jest stała i nieduża.

Przykładem może być tutaj atrybut Właściciel występujący w relacji Serwis. Pole to jest wielowartościowe i segmentowe, gdyż biurem może zarządzać wielu właścicieli, a każdy z nich ma imię, nazwisko i wiele innych atrybutów. Zostało więc ono usunięte z relacji a w jego miejsce powstała nowa relacja Właściciele.

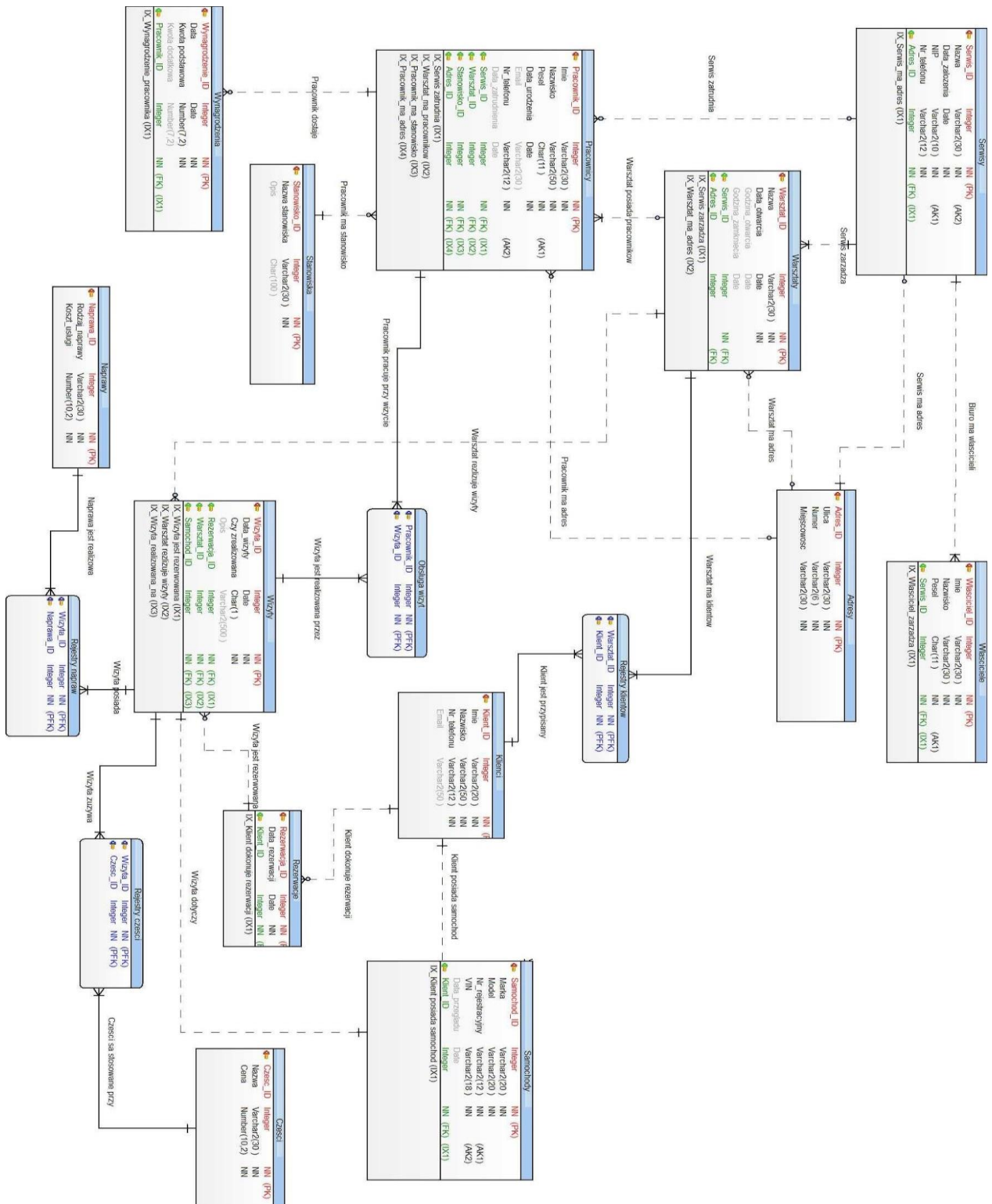
W podobny sposób dokonano następujących zmian:

- Atrybut segmentowy Adres występujący w relacjach Serwisu, Warsztatu oraz atrybut segmentowy Adres zamieszkania z relacji Pracownicy został zastąpiony nową relacją Adresy.
- Atrybut Stanowisko z relacji Pracownik został zastąpiony nową relacją słownikową Stanowiska.
- Atrybut wielowartościowy Wynagrodzenie z relacji Pracownik został zastąpiony nową relacją Wynagrodzenia.
- Atrybut segmentowy Kierownik z relacji Warsztat został usunięty, gdyż uznano że posiadając relacje Pracownicy i Stanowiska oraz związek pomiędzy relacjami Warsztat i Pracownicy, możliwe jest zidentyfikowanie kierownika danego warsztatu.

4.3.1 Przekształcenie relacji do 2 i 3 postaci normalnej

Relacja znajduje się w drugiej postaci normalnej, jeśli jest w pierwszej postaci normalnej oraz każdy atrybut niewchodzący w skład żadnego klucza potencjalnego, jest w pełni funkcyjnie zależny od wszystkich kluczy potencjalnych tej relacji oraz wszystkie klucze potencjalne są kluczami prostymi. Relacja jest w trzeciej postaci normalnej, jeśli znajduje się w drugiej postaci normalnej oraz każdy atrybut tej relacji, nie wchodzący w skład klucza potencjalnego nie jest przechodnio funkcyjnie zależny od żadnego klucza potencjalnego tej relacji. Po przekształceniu wszystkich relacji do 1 postaci normalnej okazało się, że każda z nich znajduje się zarówno w 2 jak i w 3 postaci normalnej, gdyż wszystkie klucze były kluczami prostymi już na etapie koncepcyjnym oraz każdy atrybut niebędący częścią klucza potencjalnego mógł zostać jednoznacznie określony poprzez każdy taki klucz. Nie występowały również zależności przechodnie między kluczami potencjalnymi, a atrybutami niewchodzącymi w skład kluczy.

4.4 Schemat ER na poziomie modelu logicznego



4.5 Więzy integralności

W poprzednich etapach zadbano, by wszystkie atrybuty były atomowe tzn. nierozkładalne. Wszystkie klucze główne oznaczoną są jako UNIQUE. Na wszystkie atrybuty, których brak miałby istotny wpływ na błędne działanie bazy danych, nałożono ograniczenie NOT NULL.

4.6 Proces denormalizacji - analiza i przykłady

Dokonując analizy struktury danych po procesie normalizacji stwierdzono, że nie zachodzi potrzeba wprowadzenie denormalizacji w celu przyspieszenia działania systemu. Uznano, że wzrost wydajności systemu jaki można by w tym procesie otrzymać, byłby nieadekwatny do utraconej elastyczności systemu oraz zmniejszonej wydajności aktualizacji danych.

5. Faza fizyczna

5.1 Projekt transakcji i weryfikacja ich wykonalności

- **Perspektywa Klienta**

- Podgląd całkowitego kosztu swojej wizyty (zapytanie umieszczone w podpunkcie 5.4).
- Aktualizacja swoich danych.
- Dokonanie rezerwacji na wizytę.
- Uzyskanie informacji o danej wizycie
- Uzyskanie informacji o Warsztacie.

- **Perspektywa właściciela Serwisu**

- Pogląd danych Serwisu.
- Podgląd danych pracowników (zapytanie umieszczone w podpunkcie 5.4).

- **Perspektywa kierownika Warsztatu**

- Dodawanie/usuwanie Wizyt.
- Uzyskanie informacji o liczbie pracowników na danym stanowisku (zapytanie umieszczone w podpunkcie 5.4).
- Uzyskanie informacji o liczbie rezerwacji na dany dzień.
- Uzyskanie informacji o tym, na którą godzinę dokonywano największej liczby rezerwacji.
- Uzyskanie informacji o najbardziej obleganych terminach/godzinach wizyt

- **Perspektywa księgowego**

- Uzyskanie informacji o łącznej sumie wynagrodzeń w danym okresie.
- Uzyskanie informacji o łącznej sumie wynagrodzeń danego pracownika w danym okresie.

- Wprowadzanie miesięcznych wynagrodzeń danego pracownika.
- Aktualizacja danych pracownika.
- Uzyskiwanie informacji o przychodzie ze sprzedaży części w danym okresie (zapytanie umieszczone w podpunkcie 5.4).

- **Perspektywa pracownika**

- Przyjęcie na wizytę
- Podgląd wolnych miejsc w warsztacie.

5.2 Strojenie bazy danych - dobór indeksów

Utworzone indeksy są najczęściej wybieranymi danymi w relacji według określonego atrybutu:

- podgląd Wizyty po dacie CREATE INDEX "IX_Wizyta_odbywajaca_sie" ON "Wizyty" ("Data")
- podgląd klienta po nazwisku CREATE INDEX "IX_Klient_o_nazwisku" ON "Klienci" ("Nazwisko")

5.3 Skrypt SQL tworzący bazę danych

Skrypt w załączniku nr 1.

5.4 Przykłady zapytań i poleceń SQL odnoszących się do bazy danych

Skrypt uzupełniający bazę danymi w załączniku nr 2.

- **Podgląd całkowitego kosztu wizyty**

```
SELECT SUM(kosztwizyty) Koszt_Wizyty
FROM
(
    SELECT SUM( n . "Koszt_uslugi") kosztwizyty
    FROM "Naprawy" n JOIN "Rejestry napraw" w ON n."Naprawa_ID" = w."Naprawa_ID"
    JOIN "Wizyty" v ON w."Wizyta_ID"=v."Wizyta_ID"
    WHERE v."Wizyta_ID" = 1
    UNION ALL
    SELECT SUM( c . "Cena") kosztwizyty
    FROM "Czesci" c
    JOIN "Rejestry czesci" d ON c."Czesc_ID" = d."Czesc_ID"
    JOIN "Wizyty" v ON d."Wizyta_ID"=v."Wizyta_ID"
    WHERE v."Wizyta_ID" = 1
)
```

The screenshot shows a SQL query result window with two tabs: 'Script Output' and 'Query Result'. The 'Query Result' tab is active, displaying the results of the SQL query. The window title is 'Query Result x'. Below the title bar, there are icons for a folder, a document, a magnifying glass, and a SQL icon. The text 'All Rows Fetched: 1 in 0,001 seconds' is displayed. The table has one column named 'KOSZT_WIZYT' and one row with the value '87'.

KOSZT_WIZYT
87

- Uzyskanie o ilości pracowników na danym stanowisku

```
SELECT COUNT(p."Pracownik_ID") AS "Mechanicy"
FROM "Pracownicy" p
WHERE "Stanowisko ID" = 4;
```

	Mechanicy
1	4

- Podgląd danych pracowników

```
SELECT
p."Imie",
p."Nazwisko",
s."Nazwa stanowiska",
p."Data_zatrudnienia",
p."Pesel",
p."Data_urodzenia",
p."Nr_telefonu",
a."Ulica",
a."Numer",
a."Miejscowosc"
FROM "Pracownicy" p
JOIN "Stanowiska" s ON p."Stanowisko ID"=s."Stanowisko_ID"
JOIN "Adresy" a ON p."Adres ID"=a."Adres_ID"
WHERE p."Serwis_ID" = 1;
```

	Imie	Nazwisko	Nazwa stanowiska	Data_zatrudnienia	Pesel	Data_urodzenia	Nr_telefonu	Ulica	Numer	Miejscowosc
1	Jan	Taki	Administrator	11/10/12	45505004919	85/11/04	198741254	Warynskiego	12	Warszawa
2	Sebastian	Nowak	Kierownik	04/11/12	45505500919	56/08/12	298741254	Piekna	60A	Wroclaw
3	Janusz	Golec	Mechanik	08/11/04	4550506919	93/02/01	398741254	Marszałkowska	78B	Warszawa
4	Tadeusz	Bolec	Mechanik	09/02/03	45505880919	94/10/03	498741254	Warynskiego	12	Warszawa
5	Szymon	Psikuta	Kierownik	10/09/08	45505080919	95/11/12	598741254	Obozowa	81	Warszawa
6	John	Smith	Mechanik	06/06/07	45505007919	92/05/09	698741254	Wiejska	14	Wroclaw
7	Krystyna	Czubowna	Administrator	10/05/09	45205000919	89/10/12	798741254	Sosnowa	03	Wroclaw
8	Anna	Wanna	Ksiegowy	16/10/11	45505000599	87/08/05	898741254	Koscielna	07	Wroclaw
9	Marian	Dziendziel	Pomocnik	13/10/09	45504000919	84/05/03	998741254	Koscielna	10	Wroclaw
10	Witold	Grosz	Mechanik	17/10/12	45505000511	87/10/02	108744254	Koscielna	15	Wroclaw

- Uzyskanie informacji o przychodzie ze sprzedaży części w danym okresie

```
SELECT SUM( c . "Cena") AS "Suma_czesci_za_rok_2018"
FROM "Czesci" c
JOIN "Rejestry_czesci" d ON c."Czesc_ID" = d."Czesc_ID"
JOIN "Wizyty" v ON d."Wizyta_ID"=v."Wizyta_ID"
WHERE v."Data_wizyty"
BETWEEN '2018-01-01' AND '2018-12-31'
```

	Suma_czesci_za_rok_2018
1	5

ZAŁĄCZNIK NR.1

Skrypt tworzący bazę danych.

```
1.  /*
2.  Created: 05.05.2018
3.  Modified: 07.05.2018
4.  Model: Serwis
5.  Database: Oracle 11g Release 2
6.  */
7.
8.
9.  -- Create tables section -----
10.
11. -- Table Serwis
12.
13. CREATE TABLE "Serwis"(
14.   "Serwis_ID" INTEGER NOT NULL,
15.   "Nazwa" Varchar2(30 ) NOT NULL,
16.   "Data_zalozenia" DATE NOT NULL,
17.   "NIP" Varchar2(10 ) NOT NULL,
18.   "Nr_telefonu" Varchar2(12 ) NOT NULL,
19.   "Adres_ID" INTEGER NOT NULL
20. )
21. /
22.
23. -- Create indexes for table Serwis
24.
25. CREATE INDEX "IX_Serwis_ma_adres" ON "Serwis" ("Adres_ID")
26. /
27.
28. -- Add keys for table Serwis
29.
30. ALTER TABLE "Serwis" ADD CONSTRAINT "Serwis_PK" PRIMARY KEY ("Serwis_ID")
31. /
32.
33. ALTER TABLE "Serwis" ADD CONSTRAINT "NIP_AK" UNIQUE ("NIP")
34. /
35.
36. ALTER TABLE "Serwis" ADD CONSTRAINT "Nazwa_AK" UNIQUE ("Nazwa")
37. /
38.
39. -- Table Warsztaty
40.
41. CREATE TABLE "Warsztaty"(
42.   "Warsztat_ID" INTEGER NOT NULL,
43.   "Nazwa" Varchar2(30 ) NOT NULL,
44.   "Data_otwarcia" DATE NOT NULL,
45.   "Godzina_otwarcia" DATE,
46.   "Godzina_zamknienia" DATE,
47.   "Serwis_ID" INTEGER NOT NULL,
48.   "Adres_ID" INTEGER
49. )
50. /
51.
52. -- Create indexes for table Warsztaty
53.
54. CREATE INDEX "IX_Serwis zaradza" ON "Warsztaty" ("Serwis_ID")
55. /
56.
57. CREATE INDEX "IX_Warsztat_ma_adres" ON "Warsztaty" ("Adres_ID")
58. /
59.
60. -- Add keys for table Warsztaty
61.
```

```

62. ALTER TABLE "Warsztaty" ADD CONSTRAINT "Warsztat_PK" PRIMARY KEY ("Warsztat_ID")
63. /
64.
65. ALTER TABLE "Warsztaty" ADD CONSTRAINT "Nazwaa" UNIQUE ("Nazwa")
66. /
67.
68. -- Table Pracownicy
69.
70. CREATE TABLE "Pracownicy"(
71.     "Pracownik_ID" INTEGER NOT NULL,
72.     "Imie" Varchar2(30 ) NOT NULL,
73.     "Nazwisko" Varchar2(50 ) NOT NULL,
74.     "Pesel" CHAR(11 ) NOT NULL,
75.     "Data_urodzenia" DATE NOT NULL,
76.     "Email" Varchar2(30 ),
77.     "Nr_telefonu" Varchar2(12 ) NOT NULL,
78.     "Data_zatrudnienia" DATE,
79.     "Serwis_ID" INTEGER NOT NULL,
80.     "Warsztat_ID" INTEGER,
81.     "Stanowisko ID" INTEGER NOT NULL,
82.     "Adres ID" INTEGER NOT NULL
83. )
84. /
85.
86. -- Create indexes for table Pracownicy
87.
88. CREATE INDEX "IX_Serwis zatrudnia" ON "Pracownicy" ("Serwis_ID")
89. /
90.
91. CREATE INDEX "IX_Warsztat_ma_pracownikow" ON "Pracownicy" ("Warsztat_ID")
92. /
93.
94. CREATE INDEX "IX_Pracownik_ma_stanowisko" ON "Pracownicy" ("Stanowisko ID")
95. /
96.
97. CREATE INDEX "IX_Pracownik_ma_adres" ON "Pracownicy" ("Adres ID")
98. /
99.
100.     -- Add keys for table Pracownicy
101.
102.     ALTER TABLE "Pracownicy" ADD CONSTRAINT "Pracownik_PK" PRIMARY KEY ("Pracownik_ID")
103.     /
104.
105.     ALTER TABLE "Pracownicy" ADD CONSTRAINT "PESEL" UNIQUE ("Pesel")
106.     /
107.
108.     ALTER TABLE "Pracownicy" ADD CONSTRAINT "Nr_AK" UNIQUE ("Nr_telefonu")
109.     /
110.
111. -- Table Klienci
112.
113. CREATE TABLE "Klienci"(
114.     "Klient_ID" INTEGER NOT NULL,
115.     "Imie" Varchar2(20 ) NOT NULL,
116.     "Nazwisko" Varchar2(50 ) NOT NULL,
117.     "Nr_telefonu" Varchar2(12 ) NOT NULL,
118.     "Email" Varchar2(50 )
119. )
120. /
121.
122. -- Add keys for table Klienci
123.
124. ALTER TABLE "Klienci" ADD CONSTRAINT "Unique_Identifier4" PRIMARY KEY ("Klient_ID")
125. /
126.

```

```

127. ALTER TABLE "Klienci" ADD CONSTRAINT "nr_telefonu" UNIQUE ("Nr_telefonu")
128. /
129.
130. -- Table Samochody
131.
132. CREATE TABLE "Samochody"(
133.     "Samochod_ID" INTEGER NOT NULL,
134.     "Marka" Varchar2(20 ) NOT NULL,
135.     "Model" Varchar2(20 ) NOT NULL,
136.     "Nr_rejestracyjny" Varchar2(12 ) NOT NULL,
137.     "VIN" Varchar2(18 ) NOT NULL,
138.     "Data_przeglądu" DATE,
139.     "Klient_ID" INTEGER NOT NULL
140. )
141. /
142.
143. -- Create indexes for table Samochody
144.
145. CREATE INDEX "IX_Klient posiada samochod" ON "Samochody" ("Klient_ID")
146. /
147.
148. -- Add keys for table Samochody
149.
150. ALTER TABLE "Samochody" ADD CONSTRAINT "Samochod_PK" PRIMARY KEY ("Samochod_ID")
151. /
152.
153. ALTER TABLE "Samochody" ADD CONSTRAINT "Nr_rejestracyjny" UNIQUE ("Nr_rejestracyjny
154. ")
155. /
156. ALTER TABLE "Samochody" ADD CONSTRAINT "VIN_AK" UNIQUE ("VIN")
157. /
158.
159. -- Table Rezerwacje
160.
161. CREATE TABLE "Rezerwacje"(
162.     "Rezerwacja_ID" INTEGER NOT NULL,
163.     "Data_rezerwacji" DATE NOT NULL,
164.     "Klient_ID" INTEGER NOT NULL
165. )
166. /
167.
168. -- Create indexes for table Rezerwacje
169.
170. CREATE INDEX "IX_Klient dokonuje rezerwacji" ON "Rezerwacje" ("Klient_ID")
171. /
172.
173. -- Add keys for table Rezerwacje
174.
175. ALTER TABLE "Rezerwacje" ADD CONSTRAINT "Rezerwacja_PK" PRIMARY KEY ("Rezerwacja_ID
176. ")
177. /
178. -- Table Wizyty
179.
180. CREATE TABLE "Wizyty"(
181.     "Wizyta_ID" INTEGER NOT NULL,
182.     "Data_wizyty" DATE NOT NULL,
183.     "Czy_zrealizowana" CHAR(1 ) NOT NULL,
184.     "Opis" Varchar2(500 ),
185.     "Rezerwacja_ID" INTEGER NOT NULL,
186.     "Warsztat_ID" INTEGER NOT NULL,
187.     "Samochod_ID" INTEGER NOT NULL
188. )
189. /

```

```

190.
191.      -- Create indexes for table Wizyty
192.
193.      CREATE INDEX "IX_Wizyta jest rezerwowana" ON "Wizyty" ("Rezerwacja_ID")
194.      /
195.
196.      CREATE INDEX "IX_Warsztat rezlizuje wizyty" ON "Wizyty" ("Warsztat_ID")
197.      /
198.
199.      CREATE INDEX "IX_Wizyta_realizowana_na" ON "Wizyty" ("Samochod_ID")
200.      /
201.
202.      -- Add keys for table Wizyty
203.
204.      ALTER TABLE "Wizyty" ADD CONSTRAINT "Wizyta_PK" PRIMARY KEY ("Wizyta_ID")
205.      /
206.
207.      -- Table Czesci
208.
209.      CREATE TABLE "Czesci"(
210.          "Czesc_ID" INTEGER NOT NULL,
211.          "Nazwa" Varchar2(30 ) NOT NULL,
212.          "Cena" NUMBER(10,2) NOT NULL
213.      )
214.      /
215.
216.      -- Add keys for table Czesci
217.
218.      ALTER TABLE "Czesci" ADD CONSTRAINT "Czesc_PK" PRIMARY KEY ("Czesc_ID")
219.      /
220.
221.      -- Table Naprawy
222.
223.      CREATE TABLE "Naprawy"(
224.          "Naprawa_ID" INTEGER NOT NULL,
225.          "Rodzaj_naprawy" Varchar2(30 ) NOT NULL,
226.          "Koszt_uslugi" NUMBER(10,2) NOT NULL
227.      )
228.      /
229.
230.      -- Add keys for table Naprawy
231.
232.      ALTER TABLE "Naprawy" ADD CONSTRAINT "Naprawa_PK" PRIMARY KEY ("Naprawa_ID")
233.      /
234.
235.      -- Table Rejestry klientow
236.
237.      CREATE TABLE "Rejestry klientow"(
238.          "Warsztat_ID" INTEGER NOT NULL,
239.          "Klient_ID" INTEGER NOT NULL
240.      )
241.      /
242.
243.      -- Table Rejestry napraw
244.
245.      CREATE TABLE "Rejestry napraw"(
246.          "Wizyta_ID" INTEGER NOT NULL,
247.          "Naprawa_ID" INTEGER NOT NULL
248.      )
249.      /
250.
251.
252.      -- Table Obsluga wizyt
253.
254.      CREATE TABLE "Obsluga wizyt"(

```



```

255.         "Pracownik_ID" INTEGER NOT NULL,
256.         "Wizyta_ID" INTEGER NOT NULL
257.     )
258. /
259.
260. -- Table Stanowiska
261.
262. CREATE TABLE "Stanowiska"(
263.     "Stanowisko_ID" INTEGER NOT NULL,
264.     "Nazwa stanowiska" Varchar2(30 ) NOT NULL,
265.     "Opis" CHAR(100 )
266. )
267. /
268.
269. -- Add keys for table Stanowiska
270.
271. ALTER TABLE "Stanowiska" ADD CONSTRAINT "Stanowisko_PK" PRIMARY KEY ("Stanowisko_ID
272. ")
273. /
274.
275. -- Table Wynagrodzenia
276.
277. CREATE TABLE "Wynagrodzenia"(
278.     "Wynagrodzenie_ID" INTEGER NOT NULL,
279.     "Data" DATE NOT NULL,
280.     "Kwota podstawowa" NUMBER(7,2) NOT NULL,
281.     "Kwota dodatkowa" NUMBER(7,2),
282.     "Pracownik_ID" INTEGER NOT NULL
283. )
284. /
285.
286. -- Create indexes for table Wynagrodzenia
287.
288. CREATE INDEX "IX_Wynagrodzenie_pracownika" ON "Wynagrodzenia" ("Pracownik_ID")
289. /
290.
291. -- Add keys for table Wynagrodzenia
292.
293. ALTER TABLE "Wynagrodzenia" ADD CONSTRAINT "Wynagrodzenie_PK" PRIMARY KEY ("Wynagro
294. dzenie_ID")
295. /
296.
297. -- Table Adresy
298.
299. CREATE TABLE "Adresy"(
300.     "Adres_ID" INTEGER NOT NULL,
301.     "Ulica" Varchar2(30 ) NOT NULL,
302.     "Numer" Varchar2(6 ) NOT NULL,
303.     "Miejscowosc" Varchar2(30 ) NOT NULL
304. )
305. /
306.
307. -- Add keys for table Adresy
308.
309. ALTER TABLE "Adresy" ADD CONSTRAINT "Adres_PK" PRIMARY KEY ("Adres_ID")
310. /
311.
312. -- Table Wlasciciele
313.
314. CREATE TABLE "Wlasciciele"(
315.     "Wlasciciel_ID" INTEGER NOT NULL,
316.     "Imie" Varchar2(30 ) NOT NULL,
317.     "Nazwisko" Varchar2(30 ) NOT NULL,
318.     "Pesel" CHAR(11 ) NOT NULL,
319.     "Serwis_ID" INTEGER NOT NULL

```

```

318.      )
319.      /
320.
321.      -- Create indexes for table Wlasciciele
322.
323.      CREATE INDEX "IX_Wlasciciel_zaradza" ON "Wlasciciele" ("Serwis_ID")
324.      /
325.
326.      -- Add keys for table Wlasciciele
327.
328.      ALTER TABLE "Wlasciciele" ADD CONSTRAINT "Wlasciciel_PK" PRIMARY KEY ("Wlasciciel_I
329.      D")
330.      /
331.      ALTER TABLE "Wlasciciele" ADD CONSTRAINT "Pesel_AK" UNIQUE ("Pesel")
332.      /
333.
334.      -- Table Rejestry czesci
335.
336.      CREATE TABLE "Rejestry czesci"(
337.          "Wizyta_ID" INTEGER NOT NULL,
338.          "Czesc_ID" INTEGER NOT NULL
339.      )
340.      /
341.
342.      CREATE INDEX "IX_Wizyta_odbywajaca_sie" ON "Wizyty" ("Data")
343.      /
344.
345.      CREATE INDEX "IX_Klient_o_nazwisku" ON "Klienci" ("Nazwisko")
346.      /
347.      -- Create foreign keys (relationships) section -----
348.      -----
349.      ALTER TABLE "Pracownicy" ADD CONSTRAINT "Serwis
350.      zatrudnia" FOREIGN KEY ("Serwis_ID") REFERENCES "Serwisy" ("Serwis_ID")
351.      /
352.
353.      ALTER TABLE "Warsztaty" ADD CONSTRAINT "Serwis
354.      zaradza" FOREIGN KEY ("Serwis_ID") REFERENCES "Serwisy" ("Serwis_ID")
355.      /
356.
357.      ALTER TABLE "Pracownicy" ADD CONSTRAINT "Warsztat posiada
358.      pracownikow" FOREIGN KEY ("Warsztat_ID") REFERENCES "Warsztaty" ("Warsztat_ID")
359.      /
360.
361.      ALTER TABLE "Samochody" ADD CONSTRAINT "Klient posiada
362.      samochod" FOREIGN KEY ("Klient_ID") REFERENCES "Klienci" ("Klient_ID")
363.      /
364.
365.      ALTER TABLE "Rezerwacje" ADD CONSTRAINT "Klient dokonuje
366.      rezerwacji" FOREIGN KEY ("Klient_ID") REFERENCES "Klienci" ("Klient_ID")
367.      /
368.
369.      ALTER TABLE "Wizyty" ADD CONSTRAINT "Wizyta jest
370.      rezerwowana" FOREIGN KEY ("Rezerwacja_ID") REFERENCES "Rezerwacje" ("Rezerwacja_ID")
371.      /
372.
373.      ALTER TABLE "Wizyty" ADD CONSTRAINT "Warsztat rezlizuje
374.      wizyty" FOREIGN KEY ("Warsztat_ID") REFERENCES "Warsztaty" ("Warsztat_ID")

```

```

374.      /
375.
376.
377.      ALTER TABLE "Pracownicy" ADD CONSTRAINT "Pracownik ma
stanowisko" FOREIGN KEY ("Stanowisko_ID") REFERENCES "Stanowiska" ("Stanowisko_ID")
378.      /
379.
380.
381.      ALTER TABLE "Wynagrodzenia" ADD CONSTRAINT "Pracownik
dostaje" FOREIGN KEY ("Pracownik_ID") REFERENCES "Pracownicy" ("Pracownik_ID")
382.      /
383.
384.
385.      ALTER TABLE "Serwisy" ADD CONSTRAINT "Serwis ma
adres" FOREIGN KEY ("Adres_ID") REFERENCES "Adresy" ("Adres_ID")
386.      /
387.
388.
389.      ALTER TABLE "Warsztaty" ADD CONSTRAINT "Warsztat ma adres" FOREIGN KEY ("Adres
ID") REFERENCES "Adresy" ("Adres_ID")
390.      /
391.
392.
393.      ALTER TABLE "Pracownicy" ADD CONSTRAINT "Pracownik ma adres" FOREIGN KEY ("Adres
ID") REFERENCES "Adresy" ("Adres_ID")
394.      /
395.
396.
397.      ALTER TABLE "Wlasciciele" ADD CONSTRAINT "Biuro ma
wlascicieli" FOREIGN KEY ("Serwis_ID") REFERENCES "Serwisy" ("Serwis_ID")
398.      /
399.
400.
401.      ALTER TABLE "Wizyty" ADD CONSTRAINT "Wizyta
dotyczy" FOREIGN KEY ("Samochod_ID") REFERENCES "Samochody" ("Samochod_ID")
402.      /

```

ZAŁĄCZNIK NR.2

Skrypt wypełniający bazę danych.

```
1. INSERT INTO "Adresy" VALUES (1, 'Plocka', '15', 'Warszawa');
2. INSERT INTO "Adresy" VALUES (2, 'Polna', '7', 'Warszawa');
3. INSERT INTO "Adresy" VALUES (3, 'Wolska', '5', 'Wroclaw');
4. INSERT INTO "Adresy" VALUES (4, 'Warynskiego', '12', 'Warszawa');
5. INSERT INTO "Adresy" VALUES (5, 'Piekna', '60A', 'Wroclaw');
6. INSERT INTO "Adresy" VALUES (6, 'Marszałkowska', '78B', 'Warszawa');
7. INSERT INTO "Adresy" VALUES (7, 'Warynskiego', '12', 'Warszawa');
8. INSERT INTO "Adresy" VALUES (8, 'Obozowa', '81', 'Warszawa');
9. INSERT INTO "Adresy" VALUES (9, 'Wiejska', '14', 'Wroclaw');
10. INSERT INTO "Adresy" VALUES (10, 'Sosnowa', '03', 'Wroclaw');
11. INSERT INTO "Adresy" VALUES (11, 'Koscielna', '07', 'Wroclaw');
12. INSERT INTO "Adresy" VALUES (12, 'Koscielna', '10', 'Wroclaw');
13. INSERT INTO "Adresy" VALUES (13, 'Koscielna', '15', 'Wroclaw');
14.
15. INSERT INTO "Serwisy" VALUES (1, 'AUTO-SERWIS', '2001-09-28', 1234567890, 705689652, 1);
16.
17. INSERT INTO "Warsztaty" VALUES (1, 'AUTOVIWA', '2002-10-12', NULL, NULL, 1, 2);
18. INSERT INTO "Warsztaty" VALUES (2, 'MOTOPARK', '2008-05-15', NULL, NULL, 1, 3);
19.
20. INSERT INTO "Wlasciciele" VALUES (1, 'Janusz', 'Korwin', 25083502137, 1);
21.
22. INSERT INTO "Klienci" VALUES (1, 'Krystian', 'Boczek', 691987205, NULL);
23. INSERT INTO "Klienci" VALUES (2, 'Wojciech', 'Cejrowski', 691957205, NULL);
24. INSERT INTO "Klienci" VALUES (3, 'Alicja', 'Policja', 691956205, NULL);
25. INSERT INTO "Klienci" VALUES (4, 'Brajan', 'Czerep', 691987555, NULL);
26. INSERT INTO "Klienci" VALUES (5, 'Jessica', 'Abba', 692987205, NULL);
27. INSERT INTO "Klienci" VALUES (6, 'Grażyna', 'Smuteczek', 291987205, NULL);
28. INSERT INTO "Klienci" VALUES (7, 'Tadeusz', 'Wajcha', 391987205, NULL);
29. INSERT INTO "Klienci" VALUES (8, 'Zygfryd', 'Jaki', 591987205, NULL);
30.
31. INSERT INTO "Stanowiska" VALUES(1, 'Administrator', 'Administrator bazy danych');
32. INSERT INTO "Stanowiska" VALUES(2, 'Ksiegowy', 'Ksiegowy');
33. INSERT INTO "Stanowiska" VALUES(3, 'Kierownik', 'Kierownik Warsztatu');
34. INSERT INTO "Stanowiska" VALUES(4, 'Mechanik', 'Naprawa aut');
35. INSERT INTO "Stanowiska" VALUES(5, 'Pomocnik', 'Pomocnik przy naprawach');
36.
37. INSERT INTO "Pracownicy" VALUES (1, 'Jan', 'Taki', 45505004919, '1985-11-04', 'admin@krol.pl', 198741254, '2011-10-12', 1, NULL, 1, 4);
38. INSERT INTO "Pracownicy" VALUES (2, 'Sebastian', 'Nowak', 45505500919, '1956-08-12', 'kiro@o2.pl', 298741254, '2004-11-12', 1, 1, 3, 5);
39. INSERT INTO "Pracownicy" VALUES (3, 'Janusz', 'Golec', 4550506919, '1993-02-01', 'kasztan@zloty.pl', 398741254, '2008-11-04', 1, 1, 4, 6);
40. INSERT INTO "Pracownicy" VALUES (4, 'Tadeusz', 'Bolec', 45505880919, '1994-10-03', 'kot@pies.pl', 498741254, '2009-02-03', 1, 1, 4, 7);
41. INSERT INTO "Pracownicy" VALUES (5, 'Szymon', 'Psikuta', 45505080919, '1995-11-12', 'kielbasa@did.pl', 598741254, '2010-09-08', 1, 2, 3, 8);
42. INSERT INTO "Pracownicy" VALUES (6, 'John', 'Smith', 45505007919, '1992-05-09', 'kaszalot@gmail', 698741254, '2006-06-07', 1, 2, 4, 9);
43. INSERT INTO "Pracownicy" VALUES (7, 'Krystyna', 'Czubuwna', 45205000919, '1989-10-12', 'kalajak@ruby.pl', 798741254, '2010-05-09', 1, 2, 1, 10);
44. INSERT INTO "Pracownicy" VALUES (8, 'Anna', 'Wanna', 45505000599, '1987-08-05', 'lojon@yahoo.pl', 898741254, '2016-10-11', 1, NULL, 2, 11);
45. INSERT INTO "Pracownicy" VALUES (9, 'Marian', 'Dziendziel', 45504000919, '1984-05-03', 'jroger@moore.pl', 998741254, '2013-10-09', 1, 1, 5, 12);
46. INSERT INTO "Pracownicy" VALUES (10, 'Witold', 'Orosz', 45505000511, '1987-10-02', 'zabrac@go.pl', 108744254, '2017-10-12', 1, 2, 4, 13);
47.
48.
49.
```

```

50. INSERT INTO "Samochody" VALUES (1,'Opel','Corsa', 'WX4543', '33FG2R123423534626', NULL, 1);
51. INSERT INTO "Samochody" VALUES (2,'BMW','X5M', 'WG6554', 'R487RY483RH43R43R', NULL, 2);
52. INSERT INTO "Samochody" VALUES (3,'LOTUS','EXIGE', 'WI25689', 'SDAKJFNA48404F4F4J', NULL, 3);
53. INSERT INTO "Samochody" VALUES (4,'Fiat','Panda', 'WGC6554', '1832DHNI3DN2UDN3E3', NULL, 4);
54. INSERT INTO "Samochody" VALUES (5,'Opel','Astra', 'WPN2137', '2832DHNI3DN2UDN3E3', NULL, 5);
55. INSERT INTO "Samochody" VALUES (6,'BMW','E36', 'CNG6789', '3832DHNI3DN32UN3E3', NULL, 5);
56. INSERT INTO "Samochody" VALUES (7,'Ford','Mustang', 'PJ45WG6', '4832DHNI3D32UDN3E3', NULL, 6)
;
57. INSERT INTO "Samochody" VALUES (8,'Lancia','Delta', 'WSIHW34', '5832DHNI3D32UDN3E3', NULL, 7)
;
58. INSERT INTO "Samochody" VALUES (9,'Audi','80', 'W0LUXUS', '6832DHNI3DN32UN3E3', NULL, 8);
59. INSERT INTO "Samochody" VALUES (10,'ARRINERA','HUSSARYA', 'P0SSIJ', '7832DHNI3DN32UDN3E3', NULL, 8);
60.
61. INSERT INTO "Czesci" VALUES (1,'kolo_pasowe','34,0');
62. INSERT INTO "Czesci" VALUES (2,'alternator','5');
63. INSERT INTO "Czesci" VALUES (3,'rozsusznik','8');
64. INSERT INTO "Czesci" VALUES (4,'aparat_zaplonowy','27');
65.
66. INSERT INTO "Naprawy" VALUES (1,'wymiana kola','34,0');
67. INSERT INTO "Naprawy" VALUES (2,'geometria_kol','5');
68. INSERT INTO "Naprawy" VALUES (3,'przegląd','6');
69. INSERT INTO "Naprawy" VALUES (4,'wypalanie_dp','7');
70.
71.
72.
73. INSERT INTO "Rezerwacje" VALUES (1,'2017-11-01',1);
74. INSERT INTO "Rezerwacje" VALUES (2,'2017-12-01',1);
75. INSERT INTO "Rezerwacje" VALUES (3,'2017-3-21',3);
76. INSERT INTO "Rezerwacje" VALUES (4,'2018-2-01',4);
77. INSERT INTO "Rezerwacje" VALUES (5,'2017-1-01',5);
78. INSERT INTO "Rezerwacje" VALUES (6,'2018-7-01',6);
79. INSERT INTO "Rezerwacje" VALUES (7,'2017-9-01',7);
80.
81. INSERT INTO "Wizyty" VALUES (1,'2017-12-01', 0, NULL, 1, 1, 1);
82. INSERT INTO "Wizyty" VALUES (2,'2017-2-01', 0, NULL, 2, 2, 2);
83. INSERT INTO "Wizyty" VALUES (3,'2017-5-01', 0, NULL, 3, 1, 3);
84. INSERT INTO "Wizyty" VALUES (4,'2018-7-01', 1, NULL, 4, 2, 4);
85. INSERT INTO "Wizyty" VALUES (5,'2018-9-01', 0, NULL, 5, 1, 5);
86.
87.
88.
89.
90.
91.
92. INSERT INTO "Rejestry klientow" VALUES (1,1);
93. INSERT INTO "Rejestry klientow" VALUES (1,2);
94. INSERT INTO "Rejestry klientow" VALUES (2,1);
95. INSERT INTO "Rejestry klientow" VALUES (3,2);
96. INSERT INTO "Rejestry klientow" VALUES (4,2);
97. INSERT INTO "Rejestry klientow" VALUES (6,1);
98. INSERT INTO "Rejestry klientow" VALUES (7,1);
99.
100. INSERT INTO "Rejestry napraw" VALUES (1,1);
101. INSERT INTO "Rejestry napraw" VALUES (1,3);
102. INSERT INTO "Rejestry napraw" VALUES (2,1);
103. INSERT INTO "Rejestry napraw" VALUES (3,4);
104. INSERT INTO "Rejestry napraw" VALUES (4,2);
105.
106.
107. INSERT INTO "Rejestry czesci" VALUES (1,1);
108. INSERT INTO "Rejestry czesci" VALUES (1,2);
109. INSERT INTO "Rejestry czesci" VALUES (1,3);
110. INSERT INTO "Rejestry czesci" VALUES (3,1);
111. INSERT INTO "Rejestry czesci" VALUES (4,2);

```

```

112.
113.      INSERT INTO "Obsluga wizyt" VALUES (3,1);
114.      INSERT INTO "Obsluga wizyt" VALUES (10,1);
115.      INSERT INTO "Obsluga wizyt" VALUES (8,5);
116.      INSERT INTO "Obsluga wizyt" VALUES (6,4);
117.      INSERT INTO "Obsluga wizyt" VALUES (5,3);
118.
119.
120.      INSERT INTO "Wynagrodzenia" VALUES (1, '2017-11-01', '10000,0', NULL, 1);
121.      INSERT INTO "Wynagrodzenia" VALUES (2, '2017-11-01', '10000,0', NULL, 1);
122.      INSERT INTO "Wynagrodzenia" VALUES (3, '2017-12-01', '10000,0', NULL, 1);
123.      INSERT INTO "Wynagrodzenia" VALUES (4, '2017-01-01', '10000,0', NULL, 1);
124.      INSERT INTO "Wynagrodzenia" VALUES (5, '2016-11-01', '7000,0', NULL, 2);
125.      INSERT INTO "Wynagrodzenia" VALUES (6, '2016-12-01', '7000,0', NULL, 2);
126.      INSERT INTO "Wynagrodzenia" VALUES (7, '2017-01-01', '7000,0', '400,0', 2);
127.      INSERT INTO "Wynagrodzenia" VALUES (8, '2017-10-01', '12000,0', NULL, 3);
128.      INSERT INTO "Wynagrodzenia" VALUES (9, '2017-11-01', '12000,0', NULL, 3);
129.      INSERT INTO "Wynagrodzenia" VALUES (10, '2017-12-01', '12000,0', NULL, 3);
130.      INSERT INTO "Wynagrodzenia" VALUES (11, '2017-11-01', '2000,0', NULL, 4);
131.      INSERT INTO "Wynagrodzenia" VALUES (12, '2017-12-01', '2000,0', NULL, 4);
132.      INSERT INTO "Wynagrodzenia" VALUES (13, '2018-01-01', '2000,0', '100,0', 4);
133.      INSERT INTO "Wynagrodzenia" VALUES (14, '2017-11-01', '6000,0', NULL, 5);
134.      INSERT INTO "Wynagrodzenia" VALUES (15, '2017-12-01', '6000,0', NULL, 5);
135.      INSERT INTO "Wynagrodzenia" VALUES (16, '2018-01-01', '6000,0', NULL, 5);
136.      INSERT INTO "Wynagrodzenia" VALUES (17, '2018-07-01', '11000,0', NULL, 10);
137.      INSERT INTO "Wynagrodzenia" VALUES (18, '2018-08-01', '11000,0', NULL, 10);
138.      INSERT INTO "Wynagrodzenia" VALUES (19, '2018-07-01', '6000,0', NULL, 7);
139.      INSERT INTO "Wynagrodzenia" VALUES (20, '2018-08-01', '6000,0', '200,0', 7);
140.      INSERT INTO "Wynagrodzenia" VALUES (21, '2018-07-01', '1800,0', NULL, 8);
141.      INSERT INTO "Wynagrodzenia" VALUES (22, '2018-08-01', '1800,0', NULL, 8);

```