



WSEI

5 STYCZNIA 2020

WYPOŻYCZALNIA SAMOCHODOWA

PROJEKT ZALICZENIOWY Z PRZEDMIOTU „WPROWADZENIE DO BAZ
DANYCH”

BARTŁOMIEJ ŁĄKA

WYŻSZA SZKOŁA EKONOMII I INFORMATYKI W KRAKOWIE
LAB 7/1/IEN – NR ID. 12356

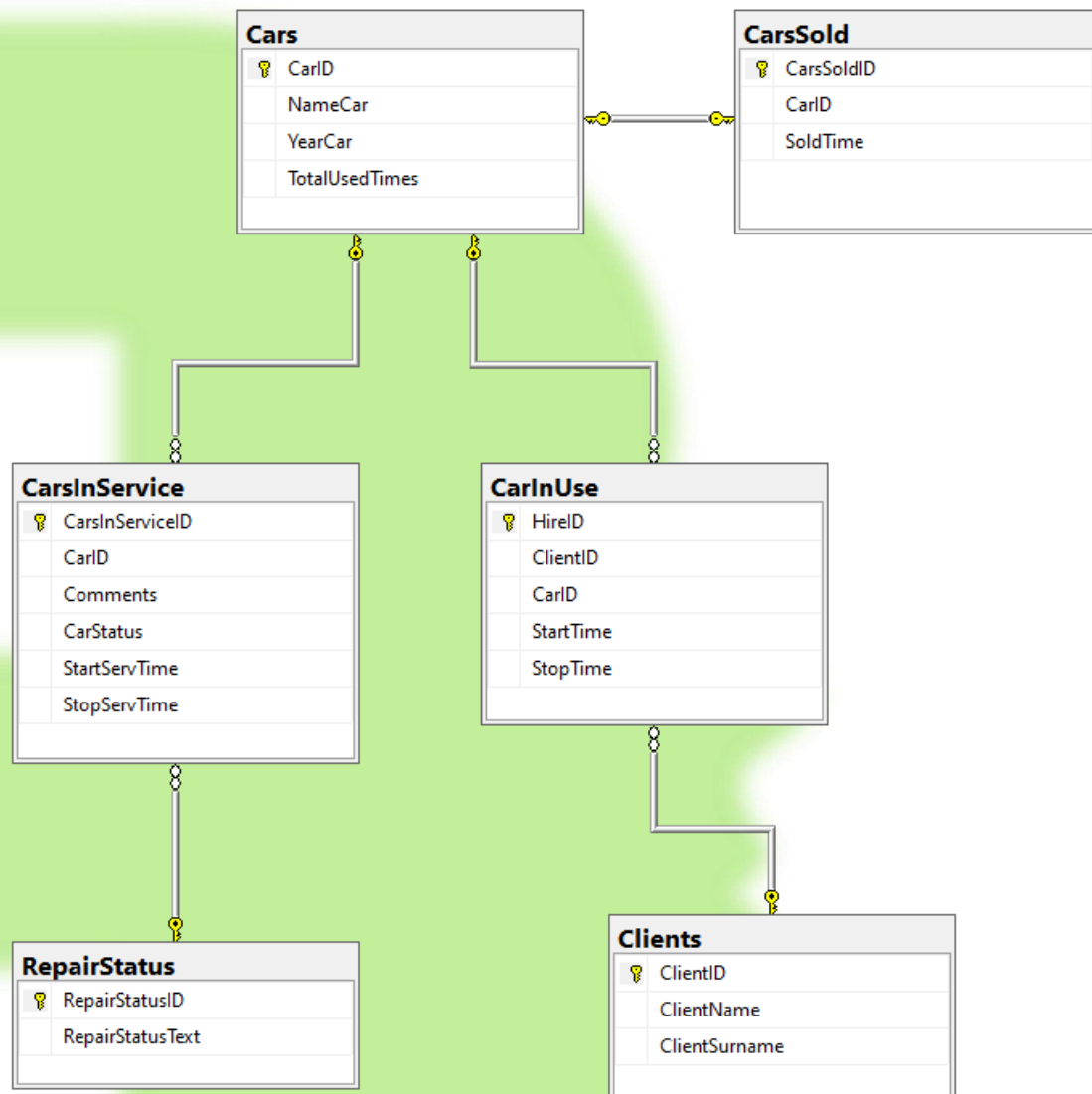
Opis Problemu

Firma „QuickCar” zajmuje się wypożyczaniem samochodów dla klientów indywidualnych na terenie Polski. Każdy klient może wypożyczyć jeden samochód do swoich potrzeb na określony czas. Samochód, który jest już wypożyczony, nie może zostać użyty w tym samym czasie ponownie. Dodatkowo też, niektóre pojazdy są serwisowane ze względu na uszkodzenia lub sprawdzenia, przez co są wyłączone z użytku. Firma zawarła współpracę z trzema serwisami samochodowymi. Każdy samochód może zostać wypożyczony najwyżej 50 razy – po tym czasie zostaje sprzedany na rynku samochodów używanych. W firmie „QuickCar” czas odgrywa kluczową rolę, dlatego też każdy samochód musi być dokładnie opisany kiedy nadaje się do użytku, a kiedy nie.

Opis Bazy Danych

Do rozwiązania tego problemu, użyłem sześciu tabel – tabela „Cars” jest tabelą bazową, na której opierają się pozostałe tabele. Zawiera dane o wszystkich samochodach, które były częścią firmy. W tabeli „Cars” możemy znaleźć podstawowe informacje na temat pojazdów – nazwę pojazdu, rok produkcji oraz liczbę wypożyczeń danego samochodu. Tabela „CarsSold” służy do katalogowania informacji o pojazdach wystawionych na sprzedaż/sprzedanych. „SoldTime” jak nazwa wskazuje mówi nam o czasie w którym dany samochód został sprzedany. „CarsInService” mówi nam o samochodach które aktualnie nie są zdane do wypożyczenia ze względu na stan techniczny. W tabeli zawierają się informacje o czasie wykonaniu naprawy („StartServTime” oraz „StopServTime”), statusie pojazdu (powiązana z tabelą tekstową „RepairStatus”) oraz „Comments” – czyli komentarz tekstowy aby dodać dane „od siebie” – na przykład informacja o tym że w samochodzie wymieniane są opony. „CarInUse” działa na podobnej zasadzie jak „CarsInService”, z tym, że samochody w tej tabeli są aktualnie w użyciu przez klienta (tabela „Clients”). Klienci są spisywani tylko i wyłącznie z imienia i nazwiska, oraz mają swój własny numer ID, dzięki czemu nie ma możliwości pomylenia dwóch klientów.

Schemat Relacji Tabel



Kod Źródłowy SQL

```
CREATE DATABASE QuickCar
```

```
GO
```

```
USE QuickCar
```

```
GO
```

```
CREATE TABLE Cars(
```

```
CarID int IDENTITY(1,1) PRIMARY KEY Not Null,
```

```
NameCar nchar(50) Not Null,
```

```
YearCar int DEFAULT YEAR(GetDate()) Not Null CHECK(YearCar  
BETWEEN YEAR(GetDate())-10 AND YEAR(GetDate())),
```

```
TotalUsedTimes tinyint CHECK(TotalUsedTimes BETWEEN 0 AND 50)  
DEFAULT 0
```

```
)
```

```
CREATE TABLE Clients(
```

```
ClientID int IDENTITY(1,1) PRIMARY KEY Not Null,
```

```
ClientName nchar(30) Not Null,
```

```
ClientSurname nchar(30) Not Null,
```

```
)
```

```
CREATE TABLE CarInUse(
```

```
HireID int IDENTITY(1,1) PRIMARY KEY Not Null,  
ClientID int Not Null,  
CarID int Not Null,  
StartTime datetime DEFAULT GetDate() CHECK(StartTime>=  
GetDate()),  
StopTime datetime CHECK (StopTime>= GetDate()+1) Not Null,  
CONSTRAINT FK_Client_CarInUse FOREIGN KEY (ClientID)  
REFERENCES Clients (ClientID) ON DELETE NO ACTION,  
CONSTRAINT FK_Car_CarInUse FOREIGN KEY (CarID) REFERENCES  
Cars (CarID) ON DELETE NO ACTION  
)
```

```
CREATE TABLE RepairStatus(  
RepairStatusID int IDENTITY(1,1) PRIMARY KEY,  
RepairStatusText nchar(50) UNIQUE Not Null,  
)
```

```
CREATE TABLE CarsInService(  
CarsInServiceID int IDENTITY(1,1) PRIMARY KEY Not Null,  
CarID int Not Null,  
Comments nchar(250),  
CarStatus int Not Null,  
StartServTime datetime DEFAULT GetDate() CHECK(StartServTime>=  
GetDate()),  
StopServTime datetime CHECK(StopServTime>= GetDate()+1),
```

```
CONSTRAINT FK_CarInService_RepairStatus FOREIGN KEY (CarStatus)
REFERENCES RepairStatus (RepairStatusID) ON DELETE NO ACTION,
CONSTRAINT FK_Car_CarInService FOREIGN KEY (CarID) REFERENCES
Cars (CarID) ON DELETE NO ACTION
)
```

```
CREATE TABLE CarsSold(
CarsSoldID int IDENTITY(1,1) PRIMARY KEY Not Null,
CarID int UNIQUE Not Null,
SoldTime datetime DEFAULT GetDate()
CONSTRAINT FK_Cars_CarsSold FOREIGN KEY (CarID) REFERENCES
Cars (CarID) ON DELETE NO ACTION
)
```

```
INSERT into Cars (NameCar,YearCar) values ('Pontiac GTO','2019'),
('Ford Fiesta','2017'), ('Porsche GT3 RS','2018'), ('Opel Corsa F',
'2019'), ('Ford Mustang GT', '2016')
```

```
INSERT into Clients (ClientName, ClientSurname) values ('David',
'Beckham'), ('John', 'Cena'), ('Tom', 'Hanks')
```

```
INSERT into RepairStatus (RepairStatusText) values ('InQueue'),
('Repairing'), ('Repaired')
```

```
INSERT into CarsSold (CarID) values ('1'), ('4')
```

```
INSERT into CarsInService (CarID, Comments, CarStatus,  
StopServTime) values ('2', 'Changing Tyres to Winter.', '1',  
GetDate()+2), ('5', 'Replacing the front bumper.', '2',  
GetDate()+1.5672)
```

```
INSERT into CarInUse (ClientID, CarID, StopTime) values ('1', '3',  
GetDate()+3)
```

```
GO
```


Spis treści

Opis Problemu	1
Opis Bazy Danych	2
Schemat Relacji Tabel	3
Kod Źródłowy SQL	4
Spis treści	8