

Baza danych dla wypożyczalni samochodów

Etap 2

Maciej Krasny 252863

Robert Walery 249000

April 2022

1 Sformułowanie wymagań dotyczących dostępu do bazy i jej zawartości

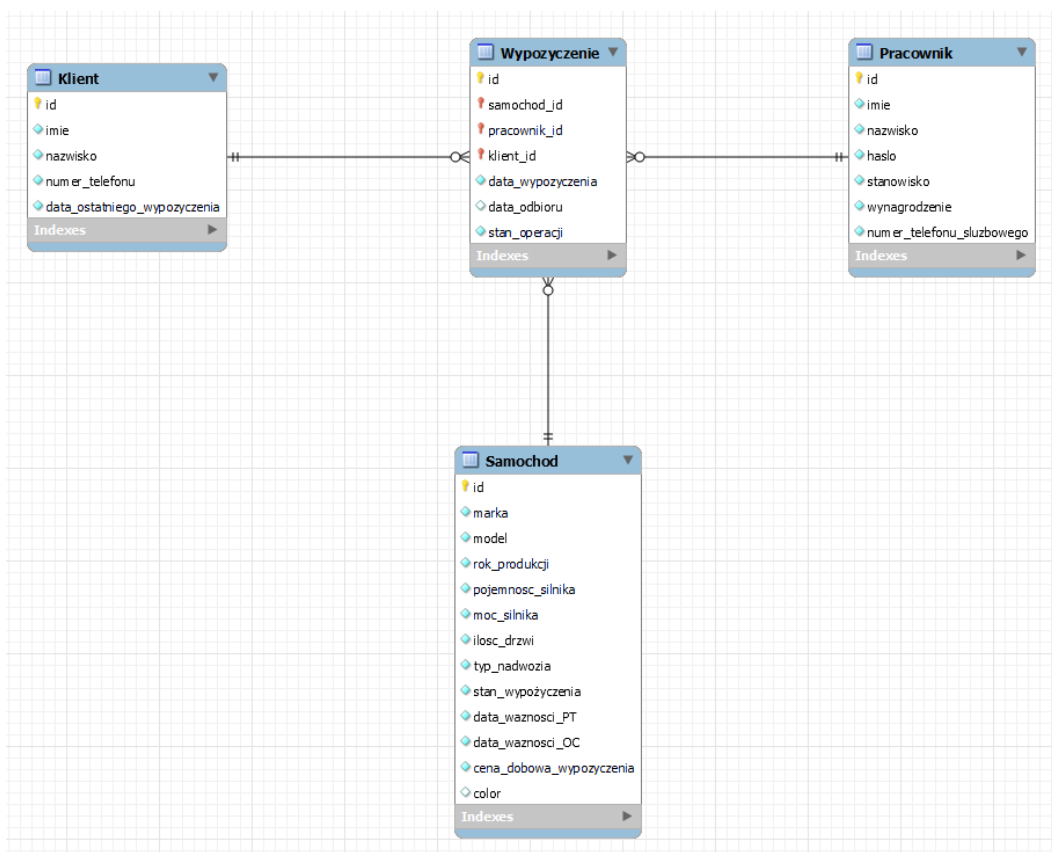
Wyróżniamy dwa rodzaje użytkowników bazy danych:

- Szef
- Pracownik niższego szczebla

Użytkownik "Pracownik niższego szczebla" ma dostęp do tabel Klient, Wypożyczenie oraz Samochód. W tabeli samochod nie może on dodawać, ani usuwać nowych rekordów (możliwe jest tylko modyfikowanie wartości kolumn "Stan wypożyczenia", "Data ważności OC" i "Data ważności PT"), w pozostałych dwóch tabelach (Klient i Wypożyczenia) może on dodawać, usuwać oraz modyfikować rekordy.

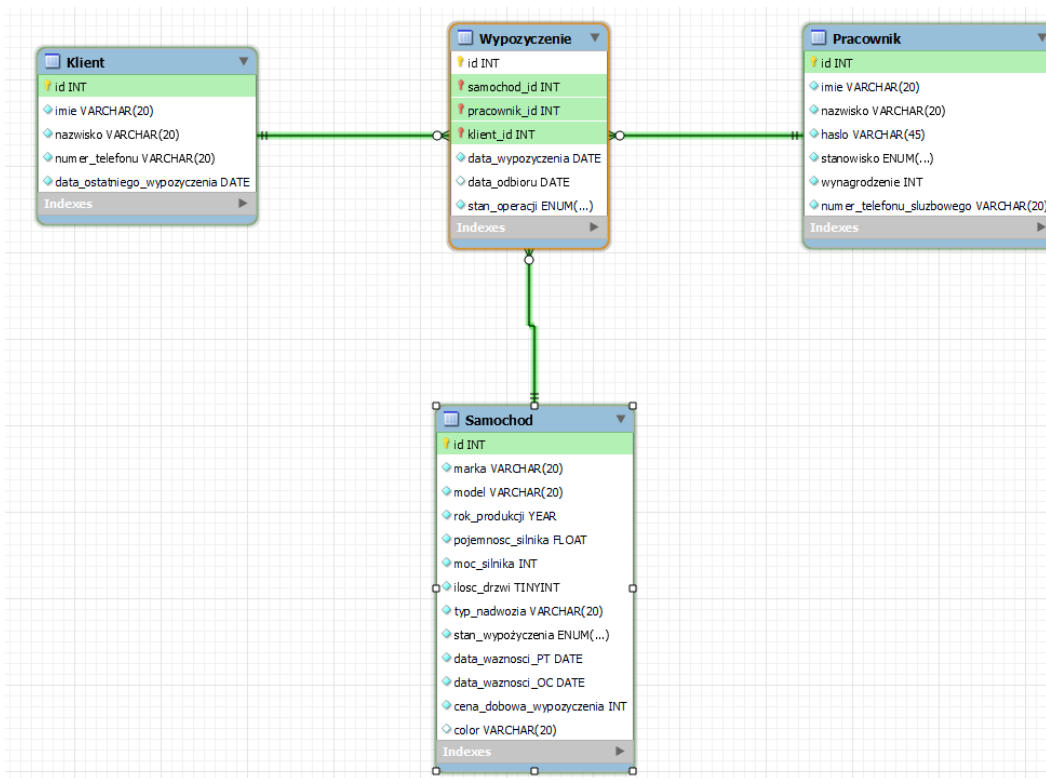
Użytkownik "Szef" posiada wszystkie uprawnienia dostępne dla użytkownika "Zwykły pracownik" i ponadto może on dodawać, usuwać lub modyfikować rekordy z tabel Samochód oraz Pracownik.

2 Diagram konceptualny modelu bazy danych na podstawie diagramu ERD w 3. postaci normalnej



Rysunek 1: Model konceptualny

3 Diagram fizycznego modelu bazy i utworzenie skryptu SQL definiującego bazę



Rysunek 2: Model fizyczny

W modelu fizycznym dla kolumn stan_operacji, stan_wypożyczenia i stanowisko wykorzystujemy typ zmiennych ENUM, który pozwala na przyjęcie wartości jedynie spośród wcześniej określonych.

W tabeli Samochód dla kolumn pojemnosc_silnika, moc_silnika, ilosc_drzwi zastosowano flagę UN, która gwarantuje, że wpisane tam dane będą dodatnie (zmiana typu danych np. z INT na unsigned INT).

Każdy klucz główny w danej tabeli jest unikalny za sprawą ustawienia flagi UQ.

Kolumna data_odbioru jest opcjonalna (może przyjmować wartość NULL), gdy wypożyczenie jest aktualne - dopóki klient nie zwróci samochodu do wypożyczalni.

```
1  — MySQL Workbench Forward Engineering
2
3  SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
4  SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;
5  SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,
6    NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION';
7
8  — Schema car_rental
9
10
11
12  — Schema car_rental
13
14  CREATE SCHEMA IF NOT EXISTS 'car_rental' DEFAULT CHARACTER SET utf8 ;
15  USE 'car_rental' ;
16
17
18  — Table 'car_rental'. 'Klient'
19
20  CREATE TABLE IF NOT EXISTS 'car_rental'. 'Klient' (
21    'id' INT NOT NULL AUTO_INCREMENT,
22    'imie' VARCHAR(20) NOT NULL,
23    'nazwisko' VARCHAR(20) NOT NULL,
24    'numer_telefonu' VARCHAR(20) NOT NULL,
25    'data_ostatniego_wypozyczenia' DATE NOT NULL,
26    PRIMARY KEY ('id'),
27    UNIQUE INDEX 'idKlient_UNIQUE' ('id' ASC) VISIBLE)
```

```

28 ENGINE = InnoDB
29 AUTO_INCREMENT = 5
30 DEFAULT CHARACTER SET = utf8mb3;
31
32
33
34 -- Table 'car_rental'. 'Pracownik'
35
36 CREATE TABLE IF NOT EXISTS 'car_rental'. 'Pracownik' (
37   'id' INT NOT NULL AUTO_INCREMENT,
38   'imie' VARCHAR(20) NOT NULL,
39   'nazwisko' VARCHAR(20) NOT NULL,
40   'haslo' VARCHAR(45) NOT NULL,
41   'stanowisko' ENUM('szef', 'zwykly_pracownik') NOT NULL DEFAULT 'zwykly_pracownik',
42   'wynagrodzenie' INT NOT NULL,
43   'numer_telefonu_sluzbowego' VARCHAR(20) NOT NULL,
44   PRIMARY KEY ('id'),
45   UNIQUE INDEX 'idPracownik_UNIQUE' ('id' ASC) VISIBLE,
46   UNIQUE INDEX 'Haslo_UNIQUE' ('haslo' ASC) VISIBLE)
47 ENGINE = InnoDB
48 AUTO_INCREMENT = 10
49 DEFAULT CHARACTER SET = utf8mb3;
50
51
52
53 -- Table 'car_rental'. 'Samochod'
54
55 CREATE TABLE IF NOT EXISTS 'car_rental'. 'Samochod' (
56   'id' INT NOT NULL AUTO_INCREMENT,
57   'marka' VARCHAR(20) NOT NULL,
58   'model' VARCHAR(20) NOT NULL,
59   'rok_produkcji' YEAR NOT NULL,
60   'pojemnosc_silnika' FLOAT UNSIGNED NOT NULL,
61   'moc_silnika' INT UNSIGNED NOT NULL,
62   'ilosc_drzwi' TINYINT UNSIGNED NOT NULL,
63   'typ_nadwozia' VARCHAR(20) NOT NULL,
64   'stan_wypo_yczenia' ENUM('wypożyczony', 'niewypożyczony') NOT NULL,
65   'data_waznosci_PT' DATE NOT NULL,
66   'data_waznosci_OC' DATE NOT NULL,
67   'cena_dobowa_wypozyczenia' INT UNSIGNED NOT NULL,
68   'color' VARCHAR(20) NULL DEFAULT NULL,
69   PRIMARY KEY ('id'),
70   UNIQUE INDEX 'idSamochod_UNIQUE' ('id' ASC) VISIBLE)
71 ENGINE = InnoDB
72 AUTO_INCREMENT = 2
73 DEFAULT CHARACTER SET = utf8mb3;
74
75
76
77 -- Table 'car_rental'. 'Wypozyczenie'
78
79 CREATE TABLE IF NOT EXISTS 'car_rental'. 'Wypozyczenie' (
80   'id' INT NOT NULL AUTO_INCREMENT,
81   'samochod_id' INT NOT NULL,
82   'pracownik_id' INT NOT NULL,
83   'klient_id' INT NOT NULL,
84   'data_wypozyczenia' DATE NOT NULL,
85   'data_odbioru' DATE NULL DEFAULT NULL,
86   'stan_operacji' ENUM('aktualne', 'archiwalne') NOT NULL,
87   PRIMARY KEY ('id', 'klient_id', 'pracownik_id', 'samochod_id'),
88   UNIQUE INDEX 'idWypo_yczenie_UNIQUE' ('id' ASC) VISIBLE,
89   INDEX 'fk_Wypo_yczenie_Samochod1_idx' ('samochod_id' ASC) VISIBLE,
90   INDEX 'fk_Wypo_yczenie_Pracownik1_idx' ('pracownik_id' ASC) VISIBLE,
91   INDEX 'fk_Wypo_yczenie_Klient1_idx' ('klient_id' ASC) VISIBLE,
92   CONSTRAINT 'fk_Wypo_yczenie_Klient1'
93     FOREIGN KEY ('klient_id')
94     REFERENCES 'car_rental'. 'Klient' ('id'),
95   CONSTRAINT 'fk_Wypo_yczenie_Pracownik1'
96     FOREIGN KEY ('pracownik_id')
97     REFERENCES 'car_rental'. 'Pracownik' ('id'),
98   CONSTRAINT 'fk_Wypo_yczenie_Samochod1'
99     FOREIGN KEY ('samochod_id')
100     REFERENCES 'car_rental'. 'Samochod' ('id'))
101 ENGINE = InnoDB
102 AUTO_INCREMENT = 3
103 DEFAULT CHARACTER SET = utf8mb3;
104
105

```

```

106 SET SQL_MODE=@OLD_SQL_MODE;
107 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
108 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;

```

Powyższy skrypt został wygenerowany przy pomocy programu MySQL Workbench. Polecenia zaczynające się od słowa kluczowego "SET" są związane z poprawną obsługą silnika bazy danych InnoDB. Są one używane w celu przyspieszenia importowania naszej bazy danych przez sewer MySQL.

W linie 14-tej tworzymy nowy schemat, czyli nową bazę danych (oczywiście, jeśli do tej pory ona nie istniała) o nazwie 'mydb' oraz ustawiamy kodowanie znaków na utf-8.

W linie 79 za pomocą polecenia 'create table' tworzymy nową tabelę w naszej bazie danych, jeśli jeszcze ona nie istnieje. Podajemy jej nazwę, a następnie wpisujemy wszystkie kolumny jakie mają być w niej umieszczone. Po nazwie kolumny ustawiamy typ przechowywanych w niej danych oraz flagi typu: not null, auto increment itp. Po stworzeniu kolumn podajemy, która kolumna ma być kluczem głównym, gwarantujemy jej unikalność, a następnie wpisujemy wszelkie informacje o kluczach obcych. Na końcu każdej tabeli ustawiamy silnik bazy danych - w naszym wypadku jest to InnoDB.

W ten sam sposób, analogicznie tworzymy pozostałe tabele.

4 Przetestowanie bazy danych

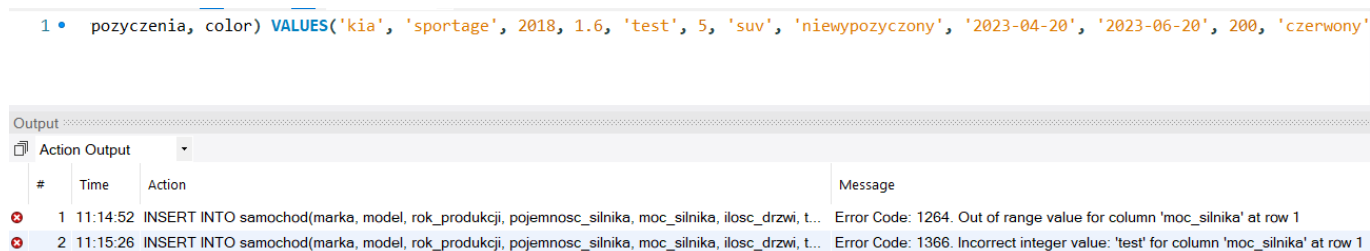
4.1 Integralność semantyczna

W celu sprawdzenia czy dane mają właściwy typ, rozmiar, format itp. wprowadzono do bazy danych przykładowe rekordy.

	id	imie	nazwisko	stanowisko	wynagrodzenie	numer_telefonu_sluzbowego
▶	1	Robert	Walery	zwykly_pracownik	3500	603855153
	2	Dawid	Walczak	zwykly_pracownik	3000	567234123
	3	Paweł	Skotnik	zwykly_pracownik	3000	667234123
*	NULL	NULL	NULL	NULL	NULL	NULL

Rysunek 3: Przykładowe dane w tabeli pracownik

Możliwość wprowadzenia danych świadczy o poprawnej integralności semantycznej. Napotkano na pewne problemy przy zapisie danych w tabeli Samochód, gdyż nie zaznaczono, że TINYINT powinien być wartości unsigned. Przez co przekraczano zakres tego typu. Ostatecznie udało się rozwiązać wszelkie problemy.
















Rysunek 4: Test poprawności typu danych

Przetestowano próbę dodania do tablicy Samochód rekordu z nieprawidłowymi typami danych. W pierwszej próbie moc_silnika, która jest typu unsigned INT miała wartość ujemną (Error nr 1) - dodanie rekordu nie powiodło się, otrzymano informację, że wartość znajduje się poza zakresem. W drugiej próbie do kolumny moc_silnika próbowano wpisać zmienną znakową 'test' - próba również zakończyła się niepowodzeniem, otrzymano błąd dotyczący niepoprawnej wartości zmiennej.

4.2 Integralność encji

Już na etapie wdrażania bazy danych zadbane, aby klucze główne każdej tabeli nie mogły być puste, były unikalne oraz samoinkrementujące się.

Table Name: Samochod											
Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression	
 id	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
 marka	VARCHAR(20)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
 model	VARCHAR(20)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
 rok_produkcji	YEAR(4)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
 pojemnosc_silnika	FLOAT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
 moc_silnika	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
 ilosc_drzwi	TINYINT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
 typ_nadwozia	VARCHAR(20)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
 stan_wypozyczenia	ENUM('wypozyc...	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
 data_waznosci_PT	DATE	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
 data_waznosci_OC	DATE	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
 cena_dobowa_wypozy...	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
 color	VARCHAR(20)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		

Rysunek 5: Właściwości kluczy głównych

```
1 INSERT INTO pracownik(id, imie, nazwisko, stanowisko, wynagrodzenie, numer_telefonu_sluzbowego) VALUES(2, 'Kamil', 'Bąk', 'z
```

Output			
Action Output			
#	Time	Action	Message
1	11:26:20	INSERT INTO pracownik(id, imie, nazwisko, stanowisko, wynagrodzenie, numer_telefonu_sluzbow...	Error Code: 1062. Duplicate entry '2' for key 'pracownik.PRIMARY'

Rysunek 6: Test unikalności klucza głównego

W powyższym teście sprawdzono, czy istnieje możliwość dodania nowego rekordu z kluczem głównym takim samym jak w istniejącym już w tablicy rekordzie - zgodnie z wcześniejszą konfiguracją dodanie takiego rekordu okazało się niemożliwe.

```
1 INSERT INTO pracownik(id, imie, nazwisko, stanowisko, wynagrodzenie, numer_telefonu_sluzbowego) VALUES(NULL,
```

Output			
Action Output			
#	Time	Action	Message
1	11:31:59	INSERT INTO pracownik(id, imie, nazwisko, stanowisko, wynagrodzenie, numer_telefonu_sluzbow...	1 row(s) affected

Rysunek 7: Test niepustości klucza głównego

Result Grid

Filter Rows:

Edit:

Export/Import:

Wrap Cell Content:

	id	imie	nazwisko	stanowisko	wynagrodzenie	numer_telefonu_sluzbowego
▶	1	Robert	Walery	zwykly_pracownik	3500	603855153
	2	Dawid	Walczak	zwykly_pracownik	3000	567234123
	3	Paweł	Skotnik	zwykly_pracownik	3000	667234123
	8	Kamil	Bąk	zwykly_pracownik	3000	567863456
	9	Kamil	Bąk	zwykly_pracownik	3000	567863456
*	NULL	NULL	NULL	NULL	NULL	NULL

pracownik 2 x

Output

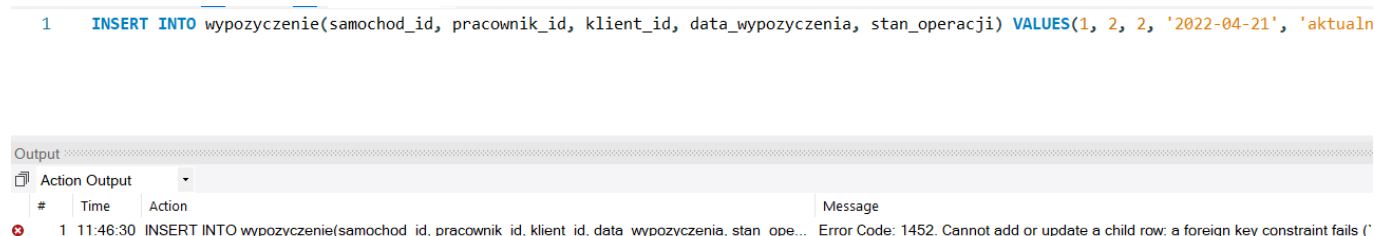
Action Output

#	Time	Action	Message
✓	1 11:34:26	INSERT INTO pracownik(id, imie, nazwisko, stanowisko, wynagrodzenie, numer_telefonu_sluzbow...	1 row(s) affected
✓	2 11:36:00	INSERT INTO pracownik(id, imie, nazwisko, stanowisko, wynagrodzenie, numer_telefonu_sluzbow...	1 row(s) affected

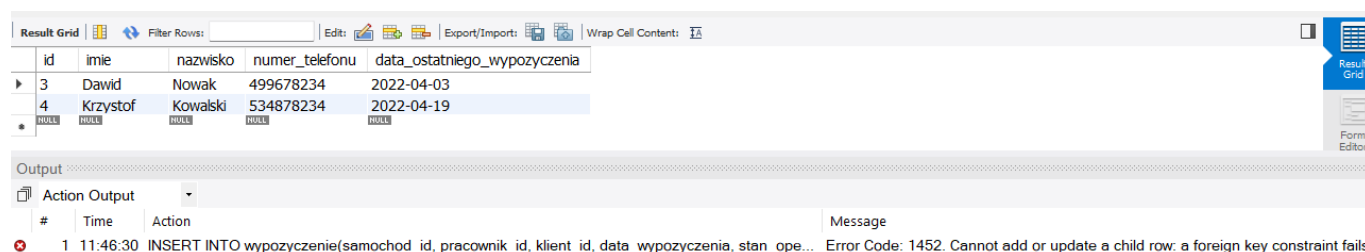
Rysunek 8: Test niepustości klucza głównego i działania jego samoinkrementacji

Na podstawie powyższych zrzutów ekranu widać, że tak skonfigurowana baza danych potrafi obsłużyć próbę dodania rekordu o pustym kluczu głównym. Dzięki zastosowaniu autoinkrementacji, mimo braku wartości klucza głównego w poleceniu INSERT INTO, wartości dodawane są do tabeli i w miejsce brakującego klucza wstawiany jest nowy unikalny klucz, w przypadku kilkukrotnego dodania wartości z kluczem ustawionym na NULL - wartość klucza rośnie (jeśli nie było pomiędzy nimi innych operacji to o 1), co potwierdza poprawne działanie autoinkrementacji.

4.3 Integralność referencji



Rysunek 9: Test integralności referencji



Rysunek 10: Zawartość tabeli Klient

Przetestowano próbę dodania rekordu do tabeli Wypożyczenia z id nieistniejącego w tabeli Klient klienta - operacja nie powiodła się, ze względu na brak referencji do obcego klucza (ponieważ taki nie istniał).