

## **1. Dokumentacja projektu:**

**Imię i nazwisko autora:** *Bartłomiej Florek*

**Numer albumu:** *125115*

**Temat projektu:** *wypożyczalnia samochodów*

**Nazwa przedmiotu:** *Programowanie obiektowe*

**Grupa laboratoryjna:** *lab1*

**Data oddania projektu:** 22.01.2024r.

Program „wypożyczalnia samochodów” udostępnia możliwość przeglądania, dodawania, edytowania oraz usuwania samochodów do wypożyczalni. Użytkownik może wydrukować raport w postaci pliku .xlsx, w którym będą się znajdowały wszystkie samochody z bazy danych.

## *Spis treści*

*a) Dokumentacja projektu:*

*b) Wprowadzenie:*

*c) Instalacja:*

*d) Interfejs użytkownika i funkcjonalność:*

*e) Bezpieczeństwo:*

*f) Dokumentacja zawartych klas i metod:*

*g) Przydatne informacje:*

## 2. Wprowadzenie:

1. Celem tej aplikacji jest umożliwienie przeglądania samochodów dostępnych do wypożyczenia. Aplikacja posiada łatwy w obsłudze, intuicyjny interfejs zawierający 5 przycisków z odpowiednimi funkcjami. Wcielamy się w rolę administratora tej aplikacji, który może dodawać nowe samochody, usuwać je, edytować oraz drukować zestawienie aby przekazać je w formie papierowej dla klienta na życzenie.
2. Wymagania sprzętowe:
  - minimum 4GB pamięci RAM
  - minimum 10GB przestrzeni dyskowej
  - dostęp do internetu
  - IntelliJ IDEA Community Edition 2022.3.2
3. Diagram ERD:

cars samochod	
ID	int(11)
MARKA	varchar(50)
MODEL	text
NADWOZIE	varchar(25)
SILNIK	float
KM	int(4)
ROCZNIK	int(4)
LICZNIK	int(11)
REJESTRACJA	text
CZY_WYPOZYCZONY	tinyint(1)

ID – klucz główny autoinkrementowany

MARKA(varchar) – ciąg liter odpowiadający realnej marce samochodów

MODEL(text) – ciąg znaków, który opisuje model samochodu, typ text ze względu na to, że istnieją samochody np. z liczbami w nazwie: rs6

NADWOZIE(varchar) – ciąg liter odpowiadający nadwoziu samochodu

SILNIK(float) – silnik może być wartością zmiennoprzecinkową np. 1.8

KM(int) – kilometry mechaniczne podaje się w liczbach całkowitych

ROCZNIK(int) – rocznik produkcji samochodu jest liczbą całkowitą

LICZNIK(int) – liczniki nie posiadają połówki kilometrów tylko całkowite liczby

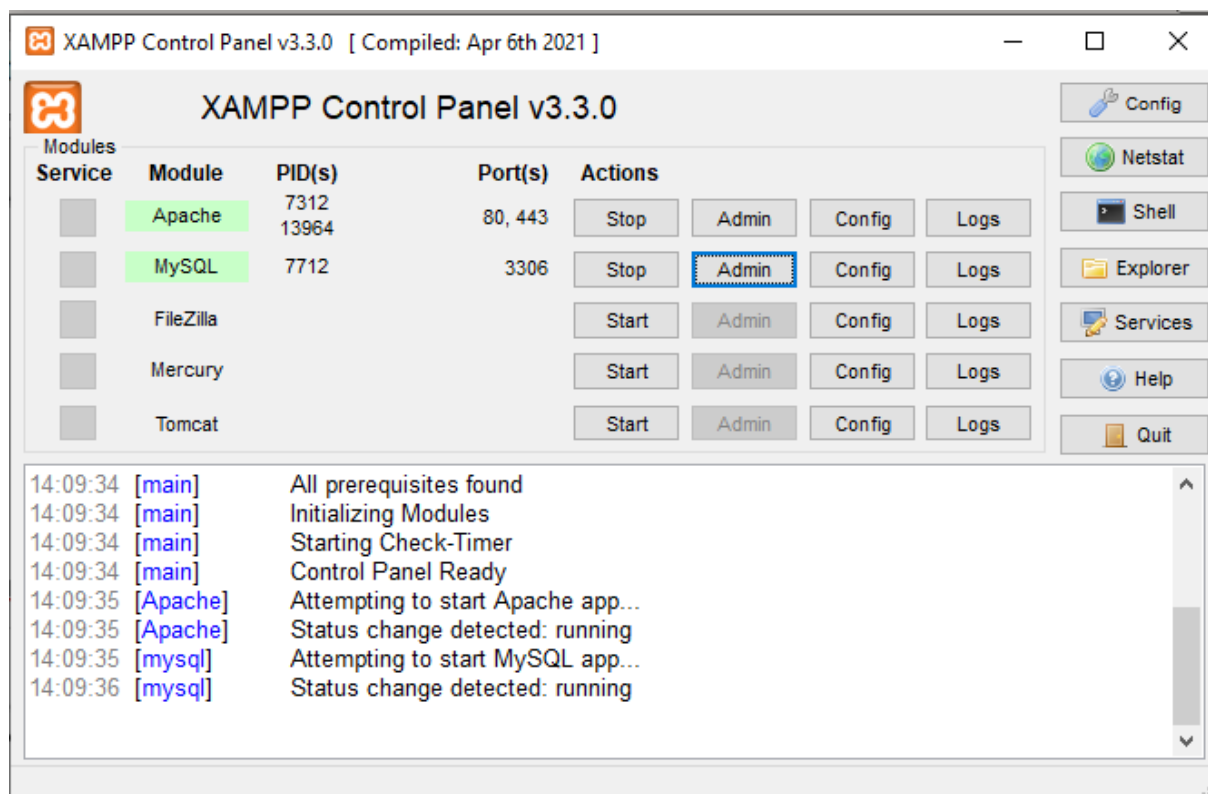
REJESTRACJA(text) – rejestracja może zawierać liczby i litery

CZY\_WYPOZYCZONY(tinyint) – wartość 0 albo 1 w zależności

### 3. Instalacja:

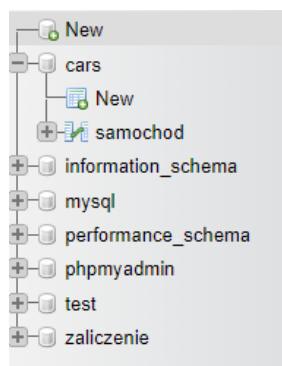
Pobieramy i instalujemy XAMPP-a z linku:  
<https://www.apachefriends.org/pl/index.html>

Następnie uruchamiamy XAMPP Control panel i klikamy w start przy Apache i MySQL:



Kiedy Apache i MySQL zaświecą się na zielono klikamy w przycisk Admin, uruchomi się wtedy okno przeglądarki z phpmyadmin

Klikamy w New:



Wpisujemy nazwę „cars” i klikamy Create:

Create database

cars utf8mb4\_general\_ci Create

Po utworzeniu, w górnym menu szukamy opcji Import

Server: 127.0.0.1 Database: xd

Structure SQL Szukaj Zapytanie Export Import

⚠ Nie znaleziono żadnych tabel w bazie danych.

Create new table

Nazwa tabeli Liczba kolumn

Create

Importujemy

plik

cars.sql

File to import:

File may be compressed (gzip, bzip2) or uncompressed.  
A compressed file's name must end in `.[format].[compression]`. Example: `.sql.zip`

Browse your computer: (Maksymalny rozmiar: 40MB)

Wybierz plik Nie wybrano pliku

Możesz także Nie wybrano pliku cić plik na dowolnej stronie.

Character set of the file:

Jeśli wszystko poszło zgodnie z planem powinniśmy uzyskać taki oto wynik:

✓ Pokazano wiersze 0 - 5 (6 total, Wykonanie zapytania trwało 0,0002 sekund(y).)

SELECT \* FROM `samochod`

☐ Profilowanie [Edytuj w linii] [Edit] [Explain SQL] [Create PHP code] [Refresh]

☐ Show all | Liczba wierszy: 25 | Filter rows: Przeszukaj tę tabelę | Sortuj wg klucza: Zaden

Extra options

	ID	MARKA	MODEL	NADWOZIE	SILNIK	KM	ROCZNIK	LICZNIK	REJESTRACJA	CZY_WYPOZYCZONY
<input type="checkbox"/> Edit Copy Delete	39	AUDI	A4	KOMBI	2	144	2008	156000	RSR12312	0
<input type="checkbox"/> Edit Copy Delete	40	LYKAN	HYPERSPORT	HYPERCAR	6.2	670	2020	5000	RZ123DS	0
<input type="checkbox"/> Edit Copy Delete	41	FERRARI	LAFERRARI	HYPERCAR	6.6	500	2019	23000	XD32EAQ	1
<input type="checkbox"/> Edit Copy Delete	42	FORD	RAPTOR	PICKUP	5.7	680	2015	56002	RZE1X1X1	0
<input type="checkbox"/> Edit Copy Delete	43	DODGE	CHARGER	SEDAN	3.2	380	2018	32123	RKRXDQWE	1
<input type="checkbox"/> Edit Copy Delete	44	DODGE	CHALLENGER	COUPE	3.6	380	2015	123450	RSR132OI	1

#### 4. Interfejs użytkownika i funkcjonalność:

# Wypożyczalnia samochodów

DODAJ

DRUKUJ

ODSWIEŻ

USUN

EDYTUJ

Wyszukaj samochód:

ID	MARKA	MODEL	NADWOZIE	SILNIK	KM	ROCZNIK	LICZNIK(w k...	REJESTRACJA	CZY_WYPOZ
39	AUDI	A4	KOMBI	2.0	144	2008	156000	RSR12312	false
40	LYKAN	HYPERSPORT	HYPERCAR	6.2	670	2020	5000	RZ123DS	false
41	FERRARI	LAFERRARI	HYPERCAR	6.6	500	2019	23000	XD32EAQ	true
42	FORD	RAPTOR	PICKUP	5.7	680	2015	56002	RZE1X1X1	false
43	DODGE	CHARGER	SEDAN	3.2	380	2018	32123	RKRXDQWE	true
44	DODGE	CHALLENGER	COUPE	3.6	380	2015	123450	RSR132OI	true

Jak na załączonym obrazku widać, interfejs jest bardzo prosty w obsłudze, posiada on główną tabelę, która pobiera wartości z bazy danych, przyciski funkcyjne oraz wyszukiwarkę.

Przycisk „dodaj”:

## Dodaj samochód

MARKA

MODEL

NADWOZIE

SILNIK(z .)

KM

ROCZNIK

LICZNIK(w km)

REJESTRACJA

WYPOZYCZONY (t/n)

Zapisz

Wyczyść

Po kliknięciu w ten przycisk uruchamia nam się menu z możliwością dodania nowego samochodu. Po wpisaniu wszystkich pól możemy zatwierdzić dane klikając w przycisk zapisz, który doda nasz samochód do bazy albo wyczyść, który usunie wszystkie wartości z pól.

Przycisk „odśwież” nie robi nic innego jak odświeża tabelę.

## Przycisk „usuń”

Po kliknięciu uruchamia się okno z możliwością podania odpowiedniego ID samochodu, którego chcemy usunąć. Przyciski usuń i wyczyść działają tak samo jak w przypadku przycisku „dodaj”

Przycisk „drukuj” tworzy plik w excelu oraz uzupełnia go o dane z tabeli:



Autozapis

Info\_o\_samochodach...

Bartłomiej Florek

Plik

Narzędzia główne

Wstawianie

Układ strony

Formuły

Dane

Recenzja

Widok

Automatyzacja

Wklej

Czcionka

Wyrównanie

Liczba

Formatowanie warunkowe

Formatuj jako tabelę

Style komórki

Komórki

Edytowanie

Schowek

Style

A1

ID

	A	B	C	D	E	F	G	H	I	J	K
1	ID	MARKA	MODEL	NADWOZIE	SILNIK	KM	ROCZNIK	LICZNIK	REJESTRACJA	WYPOZYCZONY	
2	39	AUDI	A4	KOMBI	2	144	2008	156000	RSR12312	0	
3	40	LYKAN	HYPERS	HYPERCAR	6,2	670	2020	5000	RZ123DS	0	
4	41	FERRARI	LAFERR	HYPERCAR	6,6	500	2019	23000	XD32EAQ	1	
5	42	LYKAN	HYPERS	HYPERCAR	6,2	670	2020	5000	RJSXD123	0	
6	43	DODGE	CHARG	SEDAN	3,2	380	2018	32123	RKRXDQWE	1	
7	44	DODGE	CHALL	COUPE	3,6	380	2015	123450	RSR132OI	1	
8											

Przycisk „edytuj” pozwala edytować wartości w tabeli:



# Edytuj samochody

EDYTUJ

X

ID

Marka

Model

Nadwozie

Silnik

KM

Rocznik

Licznik

Rejestracja

Wypozycz

ID	MARKA	MODEL	NADWOZIE	SILNIK	KM	ROCZNIK	LICZNIK(w k...	REJESTRACJA	CZY_WYPOZ
39	AUDI	A4	KOMBI	2.0	144	2008	156000	RSR12312	false
40	LYKAN	HYPERSPORT	HYPERCAR	6.2	670	2020	5000	RZ123DS	false
41	FERRARI	LAFERRARI	HYPERCAR	6.6	500	2019	23000	XD32EAQ	true
42	FORD	RAPTOR	PICKUP	5.7	680	2015	56002	RZE1X1X1	false
43	DODGE	CHARGER	SEDAN	3.2	380	2018	32123	RKRXDQWE	true
44	DODGE	CHALLENGER	COUPE	3.6	380	2015	123450	RSR132OI	true

Uruchamia się nowe okno, w którym mamy możliwość wybrania samochodu i edytowania go. Wybrany samochód pojawia się w polach ustawionych wyżej i tam dokonuje się edycji. Po wprowadzonych zmianach klikamy przycisk „**edytuj**” który zapisze zmiany.

42

LYKAN

HYPERSPOF

HYPERCAR

6.2

670

2020

5000

RZ123DS

false

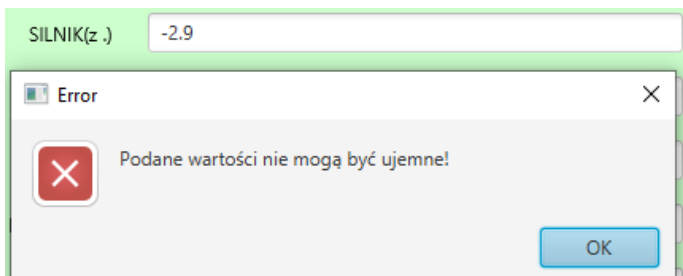
ID	MARKA	MODEL	NADWOZIE	SILNIK	KM	ROCZNIK	LICZNIK(w k...	REJESTRACJA	CZY_WYPOZ
39	AUDI	A4	KOMBI	2.0	144	2008	156000	RSR12312	false
40	LYKAN	HYPERSPORT	HYPERCAR	6.2	670	2020	5000	RZ123DS	false
41	FERRARI	LAFERRARI	HYPERCAR	6.6	500	2019	23000	XD32EAQ	true
42	LYKAN	HYPERSPORT	HYPERCAR	6.2	670	2020	5000	RZ123DS	false

Diagram UML:

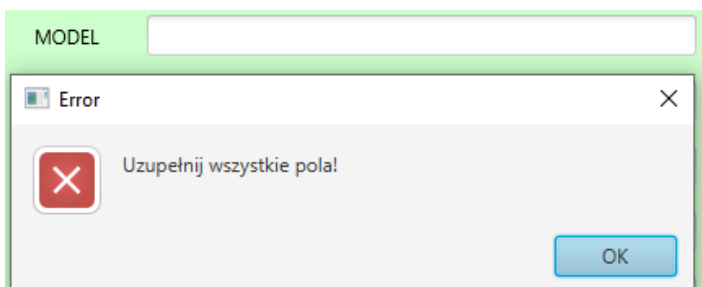


## 5. Bezpieczeństwo:

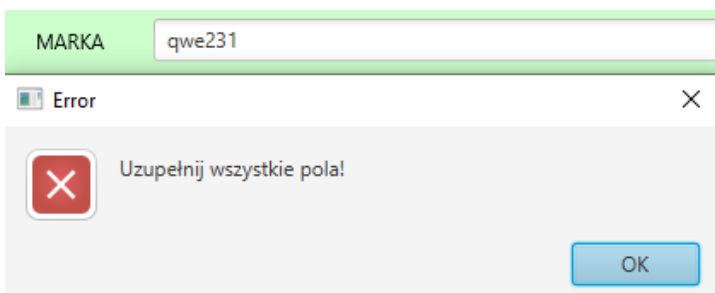
Aplikacja zawiera szereg zabezpieczeń, które uniemożliwiają dodanie do bazy danych samochodów o źle wprowadzonych danych. Przykładowo nie możemy dodać samochodu o ujemnych wartościach w polach liczbowych np. silnik -2.0, wtedy aplikacja wyrzuci nam błąd:



Tak samo mamy w przypadku podania pustych pól




Albo jeśli chcemy wprowadzić markę samochodu albo nadwozie które zawiera w sobie liczby, a jak dobrze wiemy nie może tam się znajdować liczba.



Takie same zabezpieczenia znajdują się w edycji samochodu

LYKAN2	HYPERSPOF	HYPERCAR	6.2	670	2020
--------	-----------	----------	-----	-----	------


Warning

 uzupełnij pola

OK

SPOF	HYPERCAR	-6.2	670	2020	5000
------	----------	------	-----	------	------


Warning

 podano błędne wartości

OK

41	Marka	LAFERRARI	HYPERCAR	6.6	500
----	-------	-----------	----------	-----	-----

Warning

 uzupełnij pola

OK

## 6. Dokumentacja zawartych klas i metod:

Klasa Cars.java to konstruktor, zawierający wszystkie pola tabeli, potrzebny do dodawania nowych samochodów jak i uzupełniania tabeli.

Klasa Connect.java odpowiada za połączenie się z baza danych, jeśli nie uda się to wyrzuci odpowiedni wyjątek.

Klasa Main.java, to główny plik odpowiedzialny za włączenie aplikacji. Posiada szereg metod takich jak: **start()**, która uruchamia widok tabeli, **ADD()** uruchamia widok dodawania nowego samochodu, **CLOSE()** zamyka aplikację, **EDTI()** uruchamia okno edycji samochodu, **PRINT()** tworzy nowy skoroszyt w excelu o podanych nazwach a następnie uzupełnia go danymi z tabeli, **REFRESH()** odświeża widok głównej tabeli, **DELETE()** uruchamia okno usuwania samochodu, **loadDate()** ładuje dane do tabeli która znajduje się w głównym oknie aplikacji, **getCars()** metoda wykorzystywana do wyszukiwarki, **search\_car()** metoda obsługując wyszukiwanie danych w tabeli.

Klasa Addcar.java, **SAVE()** pobiera wartości podane przez użytkownika z pól, sprawdza czy są poprawne, jeśli tak to dodaje nowy samochód do bazy, w przeciwnym wypadku wyrzuca błąd,

**isLetter(String input)**, sprawdza czy w danym ciągu znajduje się inny znak niż litera, **CLEAN()** usuwa wszystkie wartości, podane przez użytkownika

Klasa Deletcar.java, **DELETE()** usuwa samochód o podanym przez użytkownika ID, jeśli podany się nie zgadza aplikacja wyrzuci odpowiedni błąd, **CLEAN()** usuwa wartości podane przez użytkownika

Klasa Editcar.java, **start()** pokazuje aplikację do edycji danych w tabeli, **loadData()** ładuje do tabeli dane zawarte z bazy, **REFRESH()** odświeża dane w tabeli, **selectTextFields()** pozwala na zaznaczenie z tabeli rekordu i wypisanie jego wartości do pól tekstowych, **CLOSE()** zamyka aplikację, **carEdit()** pobiera nowe wartości z pól tekstowych, sprawdza je czy są poprawne i jeśli się zgadza to aktualizuje rekord a w innym przypadku zwraca odpowiedni błąd, **isLetter(String input)**, sprawdza czy w danym ciągu znajduje się inny znak niż litera

## **7. Przydatne informacje:**

Aplikacja jest prosta w użyciu i bardzo intuicyjna, lecz nie ma możliwości przypisania samochodu do osoby, która go wypożyczyła. Należałoby wtedy wykonać drugą tabelę z informacją o osobach, które są zarejestrowane w danej wypożyczalni i jaki samochód wypożyczyli.