

Samochody

Wygenerowano przez Doxygen 1.8.17

1 Indeks klas	1
1 Indeks klas	1
1.1 Lista klas	1
2 Indeks plików	1
2.1 Lista plików	1
3 Dokumentacja klas	2
3.1 Dokumentacja struktury numer_rejestracyjny	2
3.1.1 Opis szczegółowy	2
3.2 Dokumentacja struktury samochod	2
3.2.1 Opis szczegółowy	3
3.3 Dokumentacja struktury wlasciciel	3
3.3.1 Opis szczegółowy	3
3.4 Dokumentacja struktury wypisany_wlasciciel	3
3.4.1 Opis szczegółowy	3
4 Dokumentacja plików	4
4.1 Dokumentacja pliku funkcje.h	4
4.1.1 Dokumentacja funkcji	4
4.2 Dokumentacja pliku struktury.h	11
Indeks	13

1 Indeks klas

1.1 Lista klas

Tutaj znajdują się klasy, struktury, unie i interfejsy wraz z ich krótkimi opisami:

numer_rejestracyjny	2
samochod	2
wlasciciel	3
wypisany_wlasciciel	3

2 Indeks plików

2.1 Lista plików

Tutaj znajduje się lista wszystkich udokumentowanych plików z ich krótkimi opisami:

funkcje.h	4
struktury.h	11

3 Dokumentacja klas

3.1 Dokumentacja struktury numer_rejestacyjny

```
#include <struktury.h>
```

Atrybuty publiczne

- `std::string data_rejestracji`
data w której numer rejestracyjny samochodu został zarejestrowany
- `std::string nr_rejestacyjny`
numer rejestracyjny
- `numer_rejestacyjny * pNext`
wskaźnik na następny numer rejestracyjny

3.1.1 Opis szczegółowy

Struktura reprezentująca numery rejestracyjne samochodu

Dokumentacja dla tej struktury została wygenerowana z pliku:

- [struktury.h](#)

3.2 Dokumentacja struktury samochod

```
#include <struktury.h>
```

Atrybuty publiczne

- `std::string marka`
marka samochodu
- `std::string model`
model samochodu
- `int rok_produkcji`
rok produkcji samochodu
- `int pojemnosc_silnika`
pojemność silnika samochodu
- `std::string nr_silnika`
numer silnika samochodu
- `std::string nr_nadwozia`
numer nadwozia samochodu
- `std::string data_pierwszej_rejestracji`
data pierwszej rejestracji samochodu
- `numer_rejestacyjny * rejestracje`
lista rejestracji samochodu
- `wlasciciel * wlasciciele`
lista właścicieli samochodu
- `samochod * pNext`
wskaźnik do następnego samochodu listy

3.2.1 Opis szczegółowy

Struktura reprezentująca samochód

Dokumentacja dla tej struktury została wygenerowana z pliku:

- [struktury.h](#)

3.3 Dokumentacja struktury wlasciciel

```
#include <struktury.h>
```

Atrybuty publiczne

- `std::string data_nabycia`
data nabycia samochodu przez właściciela
- `std::string nazwa`
nazwa właściciela
- `wlasciciel * pNext`
wskaźnik do następnego właściciela samochodu

3.3.1 Opis szczegółowy

Struktura reprezentująca właścicieli samochodu

Dokumentacja dla tej struktury została wygenerowana z pliku:

- [struktury.h](#)

3.4 Dokumentacja struktury wypisany_wlasciciel

```
#include <struktury.h>
```

Atrybuty publiczne

- `std::string nazwa`
nazwa wypisanego właściciela
- `wypisany_wlasciciel * pNext`
wskaźnik do następnego wypisanego właściciela

3.4.1 Opis szczegółowy

Struktura reprezentująca właścicieli dla których raport został już sporządzony

Dokumentacja dla tej struktury została wygenerowana z pliku:

- [struktury.h](#)

4 Dokumentacja plików

4.1 Dokumentacja pliku funkcje.h

```
#include <fstream>
#include <string>
#include "struktury.h"
```

Funkcje

- void [znajdzPrzelacznikInput](#) (int argc, char *argv[], ifstream &dane)
- void [znajdzPrzelacznikOutput](#) (int argc, char *argv[], ofstream &raport)
- void [nowySamochod](#) (samochod *pHead)
- [samochod](#) * [ostatniSamochod](#) ([samochod](#) *pHead)
- void [nowyNumerRejestracyjny](#) (numer_rejestracyjny *pHead, string data_rejestracji, string nr_rejestracyjny)
- [numer_rejestracyjny](#) * [ostatniNumerRejestracyjny](#) ([numer_rejestracyjny](#) *pHead)
- void [nowyWlasciciel](#) (wlasciciel *pHead, string data_nabycia, string nazwa)
- [wlasciciel](#) * [ostatniWlasciciel](#) ([wlasciciel](#) *pHead)
- void [nowyWypisanyWlasciciel](#) (wypisany_wlasciciel *pHead, string nazwa)
- void [stworzListeRejestracji](#) (samochod *samochody, ifstream &dane)
- void [stworzListeWlascieli](#) (samochod *samochody, ifstream &dane)
- void [stworzListeSamochodow](#) (samochod *&samochody, ifstream &dane)
- void [wypiszOkresPosiadania](#) (wlasciciel *wlasciele, ofstream &raport)
- void [wypiszSamochod](#) (ofstream &raport, [samochod](#) *samochody)
- [samochod](#) * [znajdzKolejnySamochodWlasciciela](#) ([samochod](#) *samochody, string nazwa_wlasciciela)
- [wlasciciel](#) * [znajdzWlascicielaSamochodu](#) ([samochod](#) *samochody, string nazwa_wlasciciela)
- void [wypiszSamochodyWlasciciela](#) (samochod *samochody, ofstream &raport, [wlasciciel](#) *wlasciele)
- bool [czyWlascicielZostalWypisany](#) (wypisany_wlasciciel *wypisani_wlasciele, string nazwa_wlasciciela)
- void [usunListeWypisanychWlascieli](#) (wypisany_wlasciciel *pHead)
- void [stworzRaport](#) (samochod *samochody, ofstream &raport)
- void [usunWlascieli](#) (wlasciciel *&wlasciele)
- void [usunRejestracje](#) (numer_rejestracyjny *&rejestracje)
- void [usunSamochody](#) (samochod *&samochody)

4.1.1 Dokumentacja funkcji

4.1.1.1 czyWlascicielZostalWypisany() bool czyWlascicielZostalWypisany (
 [wypisany_wlasciciel](#) * [wypisani_wlasciele](#),
 string [nazwa_wlasciciela](#))

Funkcja sprawdza czy raport danego właściciela został już wypisany żeby uniknąć podwójnego wypisania.

Parametry

wypisani_wlasciele	lista wypisanych właścicieli
nazwa_wlasciciela	szukana nazwa właściciela

Zwraca

true gdy powstał już raport dla tego właściciela
false gdy jeszcze nie ma raportu dla tego właściciela

4.1.1.2 nowyNumerRejestracyjny() void nowyNumerRejestracyjny (
 numer_rejestracyjny *& pHead,
 string data_rejestracji,
 string nr_rejestracyjny)

Funkcja tworzy nowy element struktury "numer_rejestracyjny" na końcu listy lub jako jej pierwszy element.

Parametry

<i>pHead</i>	przekazywany przez referencje wskaźnik na listę numerów rejestracyjnych
<i>data_rejestracji</i>	parametr przypisywany tworzonemu elementowi listy
<i>nr_rejestracyjny</i>	parametr przypisywany tworzonemu elementowi listy

4.1.1.3 nowySamochod() void nowySamochod (
 samochod *& pHead)

Funkcja tworzy nowy element struktury "samochod" na końcu listy lub jako jej pierwszy element.

Parametry

<i>pHead</i>	przekazywany przez referencje wskaźnik na listę samochodów
--------------	--

4.1.1.4 nowyWlasciciel() void nowyWlasciciel (
 wlasciciel *& pHead,
 string data_nabycia,
 string nazwa)

Funkcja tworzy nowy element struktury "wlasciciel" na końcu listy lub jako jej pierwszy element.

Parametry

<i>pHead</i>	przekazywany przez referencje wskaźnik na listę właścicieli
<i>data_nabycia</i>	parametr przypisywany tworzonemu elementowi listy
<i>nazwa</i>	parametr przypisywany tworzonemu elementowi listy

4.1.1.5 nowyWypisanyWlasciciel() `void nowyWypisanyWlasciciel (`
 `wypisany_wlasciciel *& pHead,`
 `string nazwa)`

tworzy nowy element struktury "wypisany_wlasciciel" na końcu listy lub jako jej pierwszy element.

Parametry

<i>pHead</i>	przekazywany przez referencje wskaźnik na listę wypisanych właścicieli
<i>nazwa</i>	parametr przypisywany tworzonemu elementowi listy

4.1.1.6 ostatniNumerRejestracyjny() `numer_rejestracyjny* ostatniNumerRejestracyjny (`
 `numer_rejestracyjny * pHead)`

Funkcja zwraca wskaźnik na ostatni element listy.

Parametry

<i>pHead</i>	przekazywany przez referencje wskaźnik na listę numerów rejestracyjnych
--------------	---

Zwraca

wskaźnik na ostatni element listy

4.1.1.7 ostatniSamochod() `samochod* ostatniSamochod (`
 `samochod * pHead)`

Funkcja zwraca wskaźnik na ostatni element listy.

Parametry

<i>pHead</i>	przekazywany przez referencje wskaźnik na listę samochodów
--------------	--

Zwraca

wskaźnik na ostatni element listy

4.1.1.8 ostatniWlasciciel() `wlasciciel* ostatniWlasciciel (`
 `wlasciciel * pHead)`

Funkcja zwraca wskaźnik na ostatni element listy.

Parametry

<i>pHead</i>	przekazywany przez referencje wskaźnik na listę właścicieli
--------------	---

Zwraca

wskaźnik na ostatni element listy

4.1.1.9 stworzListeRejestracji() `void stworzListeRejestracji (`
 `samochod * samochody,`
 `ifstream & dane)`

Funkcja odczytuje z pliku wejściowego dane o numerach rejestracyjnych samochodów i tworzy ich listę przypisaną do danego samochodu.

Parametry

<i>samochody</i>	lista wszystkich odczytanych do tej pory samochodów
<i>dane</i>	dane wejściowe

4.1.1.10 stworzListeSamochodow() `void stworzListeSamochodow (`
 `samochod *& samochody,`
 `ifstream & dane)`

Funkcja tworzy listę samochodów i przypisuje im dane z pliku wejściowego, następnie tworzy listę rejestracji i listę właścicieli na aktualnie stworzonym samochodzie.

Parametry

<i>samochody</i>	lista wszystkich odczytanych do tej pory samochodów
<i>dane</i>	dane wejściowe

4.1.1.11 stworzListeWlascicieli() `void stworzListeWlascicieli (`
 `samochod * samochody,`
 `ifstream & dane)`

Funkcja odczytuje z pliku wejściowego dane o właścicielach samochodów i tworzy ich listę przypisaną do danego samochodu. Funkcja znajduje przecinki i na tej podstawie określa ilość właścicieli w 1 linii.

Parametry

<i>samochody</i>	lista wszystkich odczytanych do tej pory samochodów
<i>dane</i>	dane wejściowe

4.1.1.12 stworzRaport() `void stworzRaport (`
 `samochod * samochody,`
 `ofstream & raport)`

Funkcja tworzy raport dla każdego właściciela o posiadanych przez niego samochodach.

Parametry

<i>samochody</i>	lista samochodów
<i>raport</i>	strumień wyjściowy

4.1.1.13 usunListeWypisanychWlascicieli() `void usunListeWypisanychWlascicieli (`
 `wypisany_wlasciciel *& pHead)`

Funkcja usuwa wszystkie elementy listy wypisanych właścicieli i zeruje ich wskaźniki.

Parametry

<i>pHead</i>	wskaźnik na listę wypisanych właścicieli
--------------	--

4.1.1.14 usunRejestracje() `void usunRejestracje (`
 `numer_rejestracyjny *& rejestracje)`

Funkcja usuwa wszystkie elementy listy numerów rejestracyjnych i zeruje ich wskaźniki.

Parametry

<i>rejestracje</i>	lista rejestracji
--------------------	-------------------

4.1.1.15 usunSamochody() `void usunSamochody (`
 `samochod *& samochody)`

Funkcja usuwa wszystkie elementy listy samochodów i usuwa listy właścicieli, i rejestracji przypisanych każdemu samochodowi. Zeruje wskaźniki wszystkich usuniętych elementów.

Parametry

<i>samochody</i>	lista samochodów
------------------	------------------

4.1.1.16 usunWlascicieli() `void usunWlascicieli (`
`wlaszciciel *& wlasciciele)`

Funkcja usuwa wszystkie elementy listy właścicieli i zeruje ich wskaźniki.

Parametry

<i>wlasciciele</i>	lista właścicieli
--------------------	-------------------

4.1.1.17 wypiszOkresPosiadania() `void wypiszOkresPosiadania (`
`wlaszciciel * wlasciciele,`
`ofstream & raport)`

Funkcja wyprowadza na strumień wyjściowy dane o okresie posiadania samochodu przez danego właściciela.

Parametry

<i>wlasciciele</i>	lista właścicieli
<i>raport</i>	strumień wyjściowy

4.1.1.18 wypiszSamochod() `void wypiszSamochod (`
`ofstream & raport,`
`samochod * samochody)`

Funkcja wypisuje dane o samochodzie do strumienia wyjściowego.

Parametry

<i>raport</i>	strumień wyjściowy
<i>samochody</i>	lista samochodów

4.1.1.19 wypiszSamochodyWlasciciela() `void wypiszSamochodyWlasciciela (`
`samochod * samochody,`
`ofstream & raport,`
`wlaszciciel * wlasciciele)`

Funkcja wypisuje wszystkie samochody posiadane przez danego właściciela.

Parametry

<i>samochody</i>	lista samochodów
<i>raport</i>	strumień wyjściowy
<i>wlasciele</i>	lista właścicieli

4.1.1.20 `znajdzKolejnySamochodWlasciciela()` `samochod*` `znajdzKolejnySamochodWlasciciela` (
 samochod * *samochody*,
 string *nazwa_wlasciciela*)

Funkcja przeszukuje liste samochodów i ich listy właścicieli w celu znalezienia odpowiadającej nazwy właściciela wprowadzonej w argumencie funkcji.

Parametry

<i>samochody</i>	lista przeszukiwanych samochodów
<i>nazwa_wlasciciela</i>	nazwa właściciela którego samochodu szukamy

Zwraca

wskaźnik pierwszego znalezionej samochodu właściciela

4.1.1.21 `znajdzPrzelacznikInput()` `void` `znajdzPrzelacznikInput` (
 int *argc*,
 char * *argv*[],
 ifstream & *dane*)

Funkcja znajduje przełącznik "-i", następnie otwiera plik o nazwie argumentu po przełączniku.

Parametry

<i>argc</i>	ilość argumentów
<i>argv</i>	wektor argumentów
<i>dane</i>	strumień wejściowy

4.1.1.22 `znajdzPrzelacznikOutput()` `void` `znajdzPrzelacznikOutput` (
 int *argc*,
 char * *argv*[],
 ofstream & *raport*)

Funkcja znajduje przełącznik "-o", następnie otwiera plik o nazwie argumentu po przełączniku.

Parametry

<i>argc</i>	ilość argumentów
<i>argv</i>	wektor argumentów
<i>raport</i>	strumień wyjściowy

4.1.1.23 `znajdzWlascicielaSamochodu()` `wlasciciel*` `znajdzWlascicielaSamochodu` (
 `samochod * samochody`,
 `string nazwa_wlasciciela`)

Funkcja zwraca wskaźnik do szukanego właściciela o podanej nazwie.

Parametry

<i>samochody</i>	lista samochodów
<i>nazwa_wlasciciela</i>	nazwa szukanego właściciela

Zwraca

wskaźnik szukanego właściciela

4.2 Dokumentacja pliku struktury.h

```
#include <string>
```

Komponenty

- struct `numer_rejestracyjny`
- struct `wlasciciel`
- struct `samochod`
- struct `wypisany_wlasciciel`

Indeks

czyWlascicielZostalWypisany
funkcje.h, 4

funkcje.h, 4
czyWlascicielZostalWypisany, 4
nowyNumerRejestracyjny, 5
nowySamochod, 5
nowyWlasciciel, 5
nowyWypisanyWlasciciel, 5
ostatniNumerRejestracyjny, 6
ostatniSamochod, 6
ostatniWlasciciel, 6
stworzListeRejestracji, 7
stworzListeSamochodow, 7
stworzListeWlascieli, 7
stworzRaport, 8
usunListeWypisanychWlascieli, 8
usunRejestracje, 8
usunSamochody, 8
usunWlascieli, 9
wypiszOkresPosiadania, 9
wypiszSamochod, 9
wypiszSamochodyWlasciciela, 9
znajdzKolejnySamochodWlasciciela, 10
znajdzPrzelacznikInput, 10
znajdzPrzelacznikOutput, 10
znajdzWlascicielaSamochodu, 11

nowyNumerRejestracyjny
funkcje.h, 5

nowySamochod
funkcje.h, 5

nowyWlasciciel
funkcje.h, 5

nowyWypisanyWlasciciel
funkcje.h, 5

numer_rejestracyjny, 2

ostatniNumerRejestracyjny
funkcje.h, 6

ostatniSamochod
funkcje.h, 6

ostatniWlasciciel
funkcje.h, 6

samochod, 2

struktury.h, 11

stworzListeRejestracji
funkcje.h, 7

stworzListeSamochodow
funkcje.h, 7

stworzListeWlascieli
funkcje.h, 7

stworzRaport
funkcje.h, 8

usunListeWypisanychWlascieli

funkcje.h, 8

usunRejestracje
funkcje.h, 8

usunSamochody
funkcje.h, 8

usunWlascieli
funkcje.h, 9

wlasciciel, 3

wypisany_wlasciciel, 3

wypiszOkresPosiadania
funkcje.h, 9

wypiszSamochod
funkcje.h, 9

wypiszSamochodyWlasciciela
funkcje.h, 9

znajdzKolejnySamochodWlasciciela
funkcje.h, 10

znajdzPrzelacznikInput
funkcje.h, 10

znajdzPrzelacznikOutput
funkcje.h, 10

znajdzWlascicielaSamochodu
funkcje.h, 11