

Vehicle Base

Wygenerowano przez Doxygen 1.9.1

| | |
|---|----------|
| 1 Indeks hierarchiczny | 1 |
| 1.1 Hierarchia klas | 1 |
| 2 Indeks klas | 3 |
| 2.1 Lista klas | 3 |
| 3 Indeks plików | 5 |
| 3.1 Lista plików | 5 |
| 4 Dokumentacja klas | 7 |
| 4.1 Dokumentacja klasy Connections | 7 |
| 4.1.1 Opis szczegółowy | 8 |
| 4.1.2 Dokumentacja konstruktora i destruktora | 8 |
| 4.1.2.1 Connections() [1/2] | 8 |
| 4.1.2.2 Connections() [2/2] | 8 |
| 4.1.3 Dokumentacja funkcji składowych | 8 |
| 4.1.3.1 addConnection() | 8 |
| 4.1.3.2 addFromFile() | 9 |
| 4.1.3.3 addToConnectionsList() | 9 |
| 4.1.3.4 checkIfConnectionExists() | 9 |
| 4.1.3.5 deleteConnection() | 10 |
| 4.1.3.6 deleteFromList() | 10 |
| 4.1.3.7 deleteWhenPerson() | 10 |
| 4.1.3.8 deleteWhenVehicle() | 12 |
| 4.1.3.9 findUserByVehicle() | 12 |
| 4.1.3.10 findVehicleByUser() | 12 |
| 4.1.3.11 print() | 13 |
| 4.1.3.12 showConnections() | 13 |
| 4.1.3.13 showUserCars() | 13 |
| 4.1.3.14 toFile() | 14 |
| 4.2 Dokumentacja klasy Data | 14 |
| 4.2.1 Opis szczegółowy | 14 |
| 4.2.2 Dokumentacja konstruktora i destruktora | 15 |
| 4.2.2.1 Data() [1/2] | 15 |
| 4.2.2.2 Data() [2/2] | 15 |
| 4.2.3 Dokumentacja funkcji składowych | 15 |
| 4.2.3.1 print() | 15 |
| 4.3 Dokumentacja klasy Login | 15 |
| 4.3.1 Opis szczegółowy | 16 |
| 4.3.2 Dokumentacja konstruktora i destruktora | 16 |
| 4.3.2.1 Login() [1/2] | 16 |
| 4.3.2.2 Login() [2/2] | 17 |
| 4.3.3 Dokumentacja funkcji składowych | 17 |

| | |
|---|----|
| 4.3.3.1 addFromFile() | 17 |
| 4.3.3.2 addToList() | 17 |
| 4.3.3.3 addUser() | 18 |
| 4.3.3.4 adminUserMenu() | 18 |
| 4.3.3.5 changeAnyPassword() | 18 |
| 4.3.3.6 changePassword() | 19 |
| 4.3.3.7 deleteFromBase() | 19 |
| 4.3.3.8 getIsAdmin() | 19 |
| 4.3.3.9 getPassword() | 19 |
| 4.3.3.10 getUserID() | 20 |
| 4.3.3.11 isIdInBase() | 20 |
| 4.3.3.12 isIdNumber() | 20 |
| 4.3.3.13 isUserInBase() | 20 |
| 4.3.3.14 normalUserMenu() | 21 |
| 4.3.3.15 searchForUser() | 21 |
| 4.3.3.16 setPassword() | 22 |
| 4.3.3.17 show() [1/2] | 22 |
| 4.3.3.18 show() [2/2] | 22 |
| 4.3.3.19 toFile() | 22 |
| 4.4 Dokumentacja klasy Menu | 23 |
| 4.4.1 Dokumentacja funkcji składowych | 23 |
| 4.4.1.1 begin() | 23 |
| 4.4.1.2 changePassword() | 24 |
| 4.4.1.3 changeUserPassword() | 24 |
| 4.4.1.4 checkIfContinue() | 24 |
| 4.4.1.5 checkWithPattern() | 25 |
| 4.4.1.6 deleteUser() | 25 |
| 4.4.1.7 fromWhereVehicle() | 25 |
| 4.4.1.8 guestOptions() | 26 |
| 4.4.1.9 login() | 26 |
| 4.4.1.10 loginOrGuest() | 26 |
| 4.4.1.11 userChoice() | 27 |
| 4.4.1.12 whichVehiclePersonHave() | 27 |
| 4.4.1.13 whoseVehicle() | 28 |
| 4.5 Dokumentacja klasy Person | 28 |
| 4.5.1 Opis szczegółowy | 29 |
| 4.5.2 Dokumentacja konstruktora i destruktora | 29 |
| 4.5.2.1 Person() [1/2] | 29 |
| 4.5.2.2 Person() [2/2] | 29 |
| 4.5.3 Dokumentacja funkcji składowych | 30 |
| 4.5.3.1 addFromFile() | 30 |
| 4.5.3.2 addNewPerson() | 30 |

| | |
|---|----|
| 4.5.3.3 addToPersonList() | 30 |
| 4.5.3.4 deleteFromList() | 31 |
| 4.5.3.5 deletePerson() | 31 |
| 4.5.3.6 getLastName() | 31 |
| 4.5.3.7 getName() | 31 |
| 4.5.3.8 isCityInBase() | 31 |
| 4.5.3.9 isGivenPersonInBase() | 32 |
| 4.5.3.10 isPersonInBase() | 32 |
| 4.5.3.11 print() | 32 |
| 4.5.3.12 printForUser() | 33 |
| 4.5.3.13 showPerson() | 33 |
| 4.5.3.14 showPersons() | 33 |
| 4.5.3.15 toFile() | 33 |
| 4.6 Dokumentacja klasy User | 34 |
| 4.6.1 Opis szczegółowy | 34 |
| 4.6.2 Dokumentacja konstruktora i destruktora | 34 |
| 4.6.2.1 User() | 34 |
| 4.6.3 Dokumentacja funkcji składowych | 34 |
| 4.6.3.1 getIsAdmin() | 34 |
| 4.6.3.2 guestMenu() | 34 |
| 4.7 Dokumentacja klasy Vehicle | 35 |
| 4.7.1 Opis szczegółowy | 35 |
| 4.7.2 Dokumentacja konstruktora i destruktora | 36 |
| 4.7.2.1 Vehicle() [1/2] | 36 |
| 4.7.2.2 Vehicle() [2/2] | 36 |
| 4.7.3 Dokumentacja funkcji składowych | 36 |
| 4.7.3.1 addFromFile() | 36 |
| 4.7.3.2 addNewVehicle() | 37 |
| 4.7.3.3 addToVehicleList() | 37 |
| 4.7.3.4 deleteFromList() | 37 |
| 4.7.3.5 deleteVehicle() | 38 |
| 4.7.3.6 getBrand() | 38 |
| 4.7.3.7 isVehicleInBase() | 38 |
| 4.7.3.8 print() | 38 |
| 4.7.3.9 printForUser() | 39 |
| 4.7.3.10 searchForVehicle() | 39 |
| 4.7.3.11 showUserVehicles() | 39 |
| 4.7.3.12 showVehicle() | 39 |
| 4.7.3.13 showVehicles() | 40 |
| 4.7.3.14 toFile() | 40 |

5 Dokumentacja plików

41

| | |
|--|-----------|
| 5.1 Dokumentacja pliku Connections.h | 41 |
| 5.2 Dokumentacja pliku Data.h | 41 |
| 5.3 Dokumentacja pliku Login.h | 41 |
| 5.4 Dokumentacja pliku Menu.h | 42 |
| 5.5 Dokumentacja pliku Person.h | 42 |
| 5.6 Dokumentacja pliku User.h | 42 |
| 5.7 Dokumentacja pliku Vehicle.h | 43 |
| Indeks | 45 |

Rozdział 1

Indeks hierarchiczny

1.1 Hierarchia klas

Ta lista dziedziczenia posortowana jest z grubsza, choć nie całkowicie, alfabetycznie:

| | |
|-----------------------|----|
| Data | 14 |
| Connections | 7 |
| Person | 28 |
| Vehicle | 35 |
| Login | 15 |
| Menu | 23 |
| User | 34 |

Rozdział 2

Indeks klas

2.1 Lista klas

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

| | |
|-----------------------------|----|
| Connections | 7 |
| Data | 14 |
| Login | 15 |
| Menu | 23 |
| Person | 28 |
| User | 34 |
| Vehicle | 35 |

Rozdział 3

Indeks plików

3.1 Lista plików

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

| | |
|-------------------------------|----|
| Connections.h | 41 |
| Data.h | 41 |
| Login.h | 41 |
| Menu.h | 42 |
| Person.h | 42 |
| User.h | 42 |
| Vehicle.h | 43 |

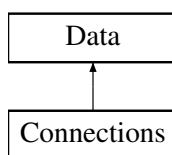
Rozdział 4

Dokumentacja klas

4.1 Dokumentacja klasy Connections

```
#include <Connections.h>
```

Diagram dziedziczenia dla Connections



Metody publiczne

- `Connections ()`
- `Connections (int personIdT, int vehicleIdT)`
- `void print ()`
- `void addToConnectionsList (list< Connections > &listOfContsT, Connections &connectionT)`
- `void addFromFile (list< Connections > &listOfContsT)`
- `void showConnections (list< Connections > listOfContsT)`
- `bool checkIfConnectionExists (list< Connections > &listOfContsT, int idP, int idV)`
- `void toFile (list< Connections > &listOfContsT)`
- `void addConnection (list< Connections > &listOfContsT, list< Vehicle > &listOfVehiclesT, list< Person > &listOfPersonsT)`
- `void deleteWhenVehicle (list< Connections > &listOfContsT, int idV)`
- `void deleteWhenPerson (list< Connections > &listOfContsT, int idP)`
- `void deleteFromList (list< Connections > &listOfContsT, int idP, int idV)`
- `void deleteConnection (list< Connections > &listOfContsT, list< Vehicle > &listOfVehiclesT, list< Person > &listOfPersonsT)`
- `void showUserCars (list< Connections > listOfContsT, list< Vehicle > listOfVehiclesT, Login userT)`
- `int findUserByVehicle (list< Connections > listOfContsT, int idV)`
- `void findVehicleByUser (list< Connections > listOfContsT, list< Vehicle > listOfVehiclesT, int isAdmin, int idP)`

Dodatkowe Dziedziczone Składowe

4.1.1 Opis szczegółowy

Klasa reprezentująca obiekt connections (połączenie między id pojazdu, a id właściciela), dziedziczy publicznie po klasie [Data](#)

4.1.2 Dokumentacja konstruktora i destruktora

4.1.2.1 Connections() [1/2]

```
Connections::Connections ( ) [inline]
```

Konstruktor domyślny korzystający z konstruktora klasy [Data](#)

4.1.2.2 Connections() [2/2]

```
Connections::Connections (
    int personIdT,
    int vehicleIdT ) [inline]
```

Konstruktor dwuargumentowy

Parametry

| | |
|-------------------|--------------------|
| <i>personIdT</i> | Zawiera id osoby |
| <i>vehicleIdT</i> | Zawiera id pojazdu |

4.1.3 Dokumentacja funkcji składowych

4.1.3.1 addConnection()

```
void Connections::addConnection (
    list< Connections > & listOfContsT,
    list< Vehicle > & listOfVehiclesT,
    list< Person > & listOfPersonsT )
```

Metoda dodająca nowe połączenie między osobą, a pojazdem

Parametry

| | |
|------------------------|--|
| <i>listOfContsT</i> | lista STL zawierająca obiekty klasy <code>Connections</code> |
| <i>listOfVehiclesT</i> | lista STL zawierająca obiekty klasy <code>Vehicle</code> |
| <i>listOfPersonsT</i> | lista STL zawierająca obiekty klasy <code>Person</code> |

4.1.3.2 `addFromFile()`

```
void Connections::addFromFile (
    list< Connections > & listOfContsT )
```

Metoda ta wczytuje dane z pliku txt, tworzy obiekt klasy `Connections`, a ostatecznie wywołuje metodę `addToConnectionsList()`

Parametry

| | |
|---------------------|--|
| <i>listOfContsT</i> | lista STL zawierająca obiekty klasy <code>Connections</code> |
|---------------------|--|

4.1.3.3 `addToConnectionsList()`

```
void Connections::addToConnectionsList (
    list< Connections > & listOfContsT,
    Connections & connectionT )
```

Metoda dodająca do listy obiekty klasy `Connections`

Parametry

| | |
|---------------------|--|
| <i>listOfContsT</i> | lista STL zawierająca obiekty klasy <code>Connections</code> |
| <i>connectionT</i> | Reprezentuje obiekt klasy <code>Connections</code> |

4.1.3.4 `checkIfConnectionExists()`

```
bool Connections::checkIfConnectionExists (
    list< Connections > & listOfContsT,
    int idP,
    int idV )
```

Metoda sprawdzająca czy istnieje już podane połączenie w liście

Parametry

| | |
|---------------------|---|
| <i>listOfContsT</i> | lista STL zawierająca obiekty klasy Connections |
| <i>idP</i> | Zawiera id osoby podane przez użytkownika |
| <i>idV</i> | Zawiera id pojazdu podane przez użytkownika |

4.1.3.5 deleteConnection()

```
void Connections::deleteConnection (
    list< Connections > & listOfContsT,
    list< Vehicle > & listOfVehiclesT,
    list< Person > & listOfPersonsT )
```

Metoda usuwająca połączenie między pojazdem, a osobą

Parametry

| | |
|------------------------|---|
| <i>listOfContsT</i> | lista STL zawierająca obiekty klasy Connections |
| <i>listOfVehiclesT</i> | lista STL zawierająca obiekty klasy Vehicle |
| <i>listOfPersonsT</i> | lista STL zawierająca obiekty klasy Person |

4.1.3.6 deleteFromList()

```
void Connections::deleteFromList (
    list< Connections > & listOfContsT,
    int idP,
    int idV )
```

Metoda usuwająca z listy podane połączenie

Parametry

| | |
|---------------------|---|
| <i>listOfContsT</i> | lista STL zawierająca obiekty klasy Connections |
| <i>idP</i> | Zawiera id osoby |
| <i>idV</i> | Zawiera id pojazdu |

4.1.3.7 deleteWhenPerson()

```
void Connections::deleteWhenPerson (
    list< Connections > & listOfContsT,
    int idP )
```


Metoda usuwająca połączenie gdy została usunięta osoba z bazy osób

Parametry

| | |
|---------------------|---|
| <i>listOfContsT</i> | lista STL zawierająca obiekty klasy Connections |
| <i>idP</i> | Zawiera numer id usuniętej osoby |

4.1.3.8 deleteWhenVehicle()

```
void Connections::deleteWhenVehicle (
    list< Connections > & listOfContsT,
    int idV )
```

Metoda usuwająca połączenie gdy został usunięty pojazd z bazy pojazdów

Parametry

| | |
|---------------------|---|
| <i>listOfContsT</i> | lista STL zawierająca obiekty klasy Connections |
| <i>idV</i> | Zawiera numer id usuniętego pojazdu |

4.1.3.9 findUserByVehicle()

```
int Connections::findUserByVehicle (
    list< Connections > listOfContsT,
    int idV )
```

Zwraca numer id osoby po numerze pojazdu

Parametry

| | |
|---------------------|---|
| <i>listOfContsT</i> | lista STL zawierająca obiekty klasy Connections |
| <i>idV</i> | Zawiera numer id pojazdu |

4.1.3.10 findVehicleByUser()

```
void Connections::findVehicleByUser (
    list< Connections > listOfContsT,
    list< Vehicle > listOfVehiclesT,
    int isAdmin,
    int idP )
```

Metoda służy do znalezienia numeru pojazdu po numerze numerze użytkownika

Parametry

| | |
|------------------------|---|
| <i>listOfContsT</i> | lista STL zawierająca obiekty klasy Connections |
| <i>listOfVehiclesT</i> | lista STL zawierająca obiekty klasy Vehicle |
| <i>isAdmin</i> | Zawiera informację czy użytkownik wywołujący tę metodę jest adminem |
| <i>idP</i> | Zawiera id osoby |

4.1.3.11 print()

```
void Connections::print ( ) [virtual]
```

Metoda przysłaniająca metodę wirtualną, wypisuje połączenia numerów id osób i pojazdów

Implementuje [Data](#).

4.1.3.12 showConnections()

```
void Connections::showConnections (
    list< Connections > listOfContsT )
```

Metoda wypisująca dane znajdujące się w liście za pomocą metody [print\(\)](#)

Parametry

| | |
|---------------------|---|
| <i>listOfContsT</i> | lista STL zawierająca obiekty klasy Connections |
|---------------------|---|

4.1.3.13 showUserCars()

```
void Connections::showUserCars (
    list< Connections > listOfContsT,
    list< Vehicle > listOfVehiclesT,
    Login userT )
```

Metoda ta służy do wyszukania numrów id pojazdów, które posiada aktualnie zalogowany użytkownik

Parametry

| | |
|------------------------|---|
| <i>listOfContsT</i> | lista STL zawierająca obiekty klasy Connections |
| <i>listOfVehiclesT</i> | lista STL zawierająca obiekty klasy Vehicle |
| <i>userT</i> | Reprezentuje obiekt klasy Login |

4.1.3.14 toFile()

```
void Connections::toFile (
    list< Connections > & listOfContsT )
```

Metoda wpisująca dane z listy do pliku txt

Parametry

| | |
|---------------------|---|
| <i>listOfContsT</i> | lista STL zawierająca obiekty klasy Connections |
|---------------------|---|

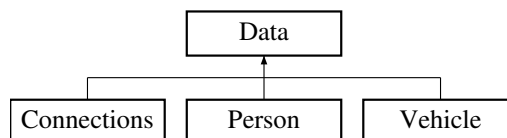
Dokumentacja dla tej klasy została wygenerowana z plików:

- [Connections.h](#)
- Connections.cpp

4.2 Dokumentacja klasy Data

```
#include <Data.h>
```

Diagram dziedziczenia dla Data



Metody publiczne

- [Data](#) ()
- [Data](#) (int idT)
- virtual void [print](#) ()=0

Atrybuty publiczne

- int [id](#)
Zawiera numer id.

4.2.1 Opis szczegółowy

Klasa, po której dziedziczą trzy klasy pochodne

4.2.2 Dokumentacja konstruktora i destruktora

4.2.2.1 Data() [1/2]

```
Data::Data ( ) [inline]
```

Konstruktor bezargumentowy

4.2.2.2 Data() [2/2]

```
Data::Data (
    int idT ) [inline]
```

Konstruktor jednoargumentowy

Parametry

| | |
|------------|------------------|
| <i>idT</i> | Zawiera numer id |
|------------|------------------|

4.2.3 Dokumentacja funkcji składowych

4.2.3.1 print()

```
virtual void Data::print ( ) [pure virtual]
```

Metoda wirtualna przysłonięta w klasach pochodnych

Implementowany w [Vehicle](#), [Person](#) i [Connections](#).

Dokumentacja dla tej klasy została wygenerowana z pliku:

- [Data.h](#)

4.3 Dokumentacja klasy Login

```
#include <Login.h>
```

Metody publiczne

- `Login ()`
- `Login (int userIDT, string passwordT, bool isAdminT)`
- `bool isIdNumber (string str)`
- `void show ()`
- `int getUserID () const`
- `string getPassword () const`
- `bool getIsAdmin () const`
- `void setPassword (string password)`
- `void addToList (Login &userT, list< Login > &listOfUserT)`
- `void addFromFile (list< Login > &listOfUsersT)`
- `void show (list< Login > listOfUserT)`
- `int searchForUser (Login &userT, list< Login > &listOfUsersT, int number, string password, list< Vehicle > &listOfVehiclesT, list< Person > &listOfPersonsT, Person &personsT)`
- `void normalUserMenu (Login &userT, list< Login > &listOfUsersT, list< Vehicle > &listOfVehiclesT, list< Person > &listOfPersonsT, Person &personsT)`
- `void changePassword (Login &userT, string password, list< Login > &listOfUsersT)`
- `void toFile (list< Login > &listOfUsersT, Login &userT)`
- `void adminUserMenu (Login &userT, list< Login > &listOfUsersT, list< Vehicle > &listOfVehiclesT, list< Person > &listOfPersonsT, Person &personsT)`
- `bool isUserInBase (list< Login > &listOfUsersT, int id)`
- `void changeAnyPassword (Login &userT, int id, string password, list< Login > &listOfUsersT)`
- `void deleteFromBase (Login &userT, int id, list< Login > &listOfUsersT)`
- `bool isIdInBase (list< Login > &listOfUsersT, int idT)`
- `void addUser (list< Login > &listOfUsersT)`

Atrybuty chronione

- `int userID`
Zawiera numer id osoby logującej się do bazy.
- `string userPassword`
Zawiera hasło osoby logującej się do bazy.
- `bool isAdmin`
Zawiera informację czy osoba logująca się jest adminem.

4.3.1 Opis szczegółowy

Klasa reprezentująca obiekt login

4.3.2 Dokumentacja konstruktora i destruktor

4.3.2.1 Login() [1/2]

```
Login::Login ( ) [inline]
```

Konstruktor domyślny

4.3.2.2 Login() [2/2]

```
Login::Login (
    int userIDT,
    string passwordT,
    bool isAdminT )
```

Konstruktor trójargumentowy

Parametry

| | |
|------------------|--|
| <i>userIDT</i> | Zawiera numer id osoby logującej |
| <i>passwordT</i> | Zawiera hasło osoby logującej |
| <i>isAdminT</i> | Zawiera informację czy dana osoba jest adminem |

4.3.3 Dokumentacja funkcji składowych

4.3.3.1 addFromFile()

```
void Login::addFromFile (
    list< Login > & listOfUsersT )
```

Metoda służąca do odczytywania danych z pliku txt, następnie tworząca obiekt klasy [Login](#) i przekazująca do metody [addToList\(\)](#)

Parametry

| | |
|---------------------|-----------|
| <i>listOfUsersT</i> | lista STL |
|---------------------|-----------|

4.3.3.2 addToList()

```
void Login::addToList (
    Login & userT,
    list< Login > & listOfUserT )
```

Metoda ta służy do dodania do listy STL obiektów klasy [Login](#)

Parametry

| | |
|--------------------|---|
| <i>userT</i> | Zawiera obiekt klasy Login |
| <i>listOfUserT</i> | lista STL do której dodawane są obiekty klasy Login |

4.3.3.3 addUser()

```
void Login::addUser (
    list< Login > & listOfUsersT )
```

Metoda służąca do dodania nowego użytkownika do bazy

Parametry

| | |
|---------------------|---|
| <i>listOfUsersT</i> | lista STL zawierająca obiekty klasy Login |
|---------------------|---|

4.3.3.4 adminUserMenu()

```
void Login::adminUserMenu (
    Login & userT,
    list< Login > & listOfUsersT,
    list< Vehicle > & listOfVehiclesT,
    list< Person > & listOfPersonsT,
    Person & personsT )
```

Metoda wyświetlająca menu dla admina, menu zawiera metody dla zwykłego użytkownika oraz specjalne tylko dla admina

Parametry

| | |
|------------------------|---|
| <i>userT</i> | Reprezentuje obiekt klasy Login |
| <i>listOfUsersT</i> | lista STL zawierająca obiekty klasy Login |
| <i>listOfVehiclesT</i> | lista STL zawierająca obiekty klasy Vehicle |
| <i>listOfPersonsT</i> | lista STL zawierająca obiekty klasy Person |
| <i>personsT</i> | Reprezentuje obiekt klasy Person |

4.3.3.5 changeAnyPassword()

```
void Login::changeAnyPassword (
    Login & userT,
    int id,
    string password,
    list< Login > & listOfUsersT )
```

Metoda zmieniająca hasło dowolnego użytkownika w bazie

Parametry

| | |
|---------------------|---|
| <i>userT</i> | Reprezentuje obiekt klasy Login |
| <i>id</i> | Zawiera numer id osoby dla której nastąpi zmiana hasła |
| <i>password</i> | Zawiera nowe hasło |
| <i>listOfUsersT</i> | lista STL zawierająca obiekty klasy Login |

4.3.3.6 changePassword()

```
void Login::changePassword (
    Login & userT,
    string password,
    list< Login > & listOfUsersT )
```

Metoda dostępna dla zwykłego użytkownika, która służy do zmiany hasła użytkownika

Parametry

| | |
|---------------------|---|
| <i>userT</i> | Reprezentuje obiekt klasy Login |
| <i>password</i> | Zawiera nowe hasło użytkownika |
| <i>listOfUsersT</i> | lista STL zawierająca obiekty klasy Login |

4.3.3.7 deleteFromBase()

```
void Login::deleteFromBase (
    Login & userT,
    int id,
    list< Login > & listOfUsersT )
```

Metoda służy do usunięcia podanej osoby z listy użytkowników

Parametry

| | |
|---------------------|---|
| <i>userT</i> | Reprezentuje obiekt klasy Login |
| <i>id</i> | Zawiera numer id osoby do usunięcia z bazy |
| <i>listOfUsersT</i> | lista STL zawierająca obiekty klasy Login |

4.3.3.8 getIsAdmin()

```
bool Login::getIsAdmin ( ) const [inline]
```

Metoda zwracająca informację o tym czy dany użytkownik jest administratorem

4.3.3.9 getPassword()

```
string Login::getPassword ( ) const [inline]
```

Metoda służy do zwrócenia hasła osób w bazie

4.3.3.10 getUserID()

```
int Login::getUserID ( ) const [inline]
```

Metoda służy do zwrócenia numeru id osób w bazie

4.3.3.11 isIdInBase()

```
bool Login::isIdInBase (
    list< Login > & listOfUsersT,
    int idT )
```

Metoda sprawdzająca czy podany numer id już jest zajęty przez któregoś z użytkowników w bazie

Parametry

| | |
|---------------------|---|
| <i>listOfUsersT</i> | lista STL zawierająca obiekty klasy Login |
| <i>idT</i> | Podany numer id |

4.3.3.12 isIdNumber()

```
bool Login::isIdNumber (
    string str )
```

Metoda sprawdzająca za pomocą wyrażeń regularnych czy podany wyraz jest liczbą sześciocyfrową

Parametry

| | |
|------------|---|
| <i>str</i> | Zawiera podany przez użytkownika numer w formie stringa |
|------------|---|

4.3.3.13 isUserInBase()

```
bool Login::isUserInBase (
    list< Login > & listOfUsersT,
    int id )
```

Metoda zwracająca prawdę jeśli podany użytkownik znajduje się w bazie, w przeciwnym wypadku zwraca fałsz

Parametry

| | |
|---------------------|---|
| <i>listOfUsersT</i> | lista STL zawierająca obiekty klasy Login |
| <i>id</i> | Zawiera numer id, który będzie porównywany do numerów id w liście |

4.3.3.14 normalUserMenu()

```
void Login::normalUserMenu (
    Login & userT,
    list< Login > & listOfUsersT,
    list< Vehicle > & listOfVehiclesT,
    list< Person > & listOfPersonsT,
    Person & personsT )
```

Metoda ta wyświetla menu dla użytkownika, który nie jest adminem

Parametry

| | |
|------------------------|---|
| <i>userT</i> | Reprezentuje obiekt klasy Login |
| <i>listOfUsersT</i> | lista STL zawierająca obiekty klasy Login |
| <i>listOfVehiclesT</i> | lista STL zawierająca obiekty klasy Vehicle |
| <i>listOfPersonsT</i> | lista STL zawierająca obiekty klasy Person |
| <i>personsT</i> | Reprezentuje obiekt klasy Person |

4.3.3.15 searchForUser()

```
int Login::searchForUser (
    Login & userT,
    list< Login > & listOfUsersT,
    int number,
    string password,
    list< Vehicle > & listOfVehiclesT,
    list< Person > & listOfPersonsT,
    Person & personsT )
```

Metoda ta służy do sprawdzenia czy logująca się osoba znajduje się w bazie, a następnie w zależności od tego czy jest adminem czy też nie wywołuje różne menu

Parametry

| | |
|------------------------|---|
| <i>userT</i> | Reprezentuje obiekt klasy Login |
| <i>listOfUsersT</i> | lista STL zawierająca obiekty klasy Login |
| <i>number</i> | Zawiera numer id podany podczas próby logowania |
| <i>password</i> | Zawiera hasło podane podczas próby logowania |
| <i>listOfVehiclesT</i> | lista STL zawierająca obiekty klasy Vehicle |
| <i>listOfPersonsT</i> | lista STL zawierająca obiekty klasy Person |
| <i>personsT</i> | Reprezentuje obiekt klasy Person |

4.3.3.16 setPassword()

```
void Login::setPassword (
    string password )
```

Metoda służąca do ustawienia hasła użytkownika na takie jakie podał

Parametry

| | |
|-----------------|--------------------------------|
| <i>password</i> | Zawiera nowe hasło użytkownika |
|-----------------|--------------------------------|

4.3.3.17 show() [1/2]

```
void Login::show ( )
```

Metoda służy do wypisania danych o osobach mających konto w bazie danych

4.3.3.18 show() [2/2]

```
void Login::show (
    list< Login > listOfUserT )
```

Metoda służy do wyświetlenia danych osób mających konto w bazie danych

Parametry

| | |
|--------------------|---|
| <i>listOfUserT</i> | lista STL w której przechowywane są obiekty reprezentujące osoby posiadające konto w bazie danych |
|--------------------|---|

4.3.3.19 toFile()

```
void Login::toFile (
    list< Login > & listOfUsersT,
    Login & userT )
```

Metoda ta służy do zapisania danych z listy do pliku txt

Parametry

| | |
|---------------------|---|
| <i>listOfUsersT</i> | lista STL zawierająca obiekty klasy Login |
| <i>userT</i> | Reprezentuje obiekt klasy Login |

Dokumentacja dla tej klasy została wygenerowana z plików:

- [Login.h](#)
- [Login.cpp](#)

4.4 Dokumentacja klasy Menu

Metody publiczne

- void [loginOrGuest](#) ([Login](#) &userT, list< [Login](#) > &listOfUserT, [User](#) &guestUser, list< [Vehicle](#) > &listOfVehiclesT, [Vehicle](#) &vehicleT, list< [Person](#) > &listOfPersonsT, [Person](#) &personsT)
- void [begin](#) ([Login](#) &userT, list< [Login](#) > &listOfUserT, list< [Vehicle](#) > &listOfVehiclesT, list< [Person](#) > &listOfPersonsT, [Person](#) &personsT)
- void [login](#) ([Login](#) &userT, list< [Login](#) > &listOfUserT, list< [Vehicle](#) > &listOfVehiclesT, list< [Person](#) > &listOfPersonsT, [Person](#) &personsT)
- void [guestOptions](#) ([User](#) guest)
- void [userChoice](#) ([Login](#) &userT, list< [Login](#) > &listOfUsersT, list< [Vehicle](#) > &listOfVehiclesT, list< [Person](#) > &listOfPersonsT, [Person](#) &personsT)
- void [changePassword](#) (list< [Login](#) > &listOfUsersT, [Login](#) userT)
- void [changeUserPassword](#) (list< [Login](#) > &listOfUsersT, [Login](#) userT)
- void [deleteUser](#) (list< [Login](#) > &listOfUsersT, [Login](#) userT)
- bool [checkWithPattern](#) (string str, regex pattern)
- void [whoseVehicle](#) (list< [Vehicle](#) > listOfVehiclesT, list< [Person](#) > listOfPersonsT, list< [Connections](#) > listOfContsT, [Login](#) userT)
- void [whichVehiclePersonHave](#) (list< [Vehicle](#) > listOfVehiclesT, list< [Person](#) > listOfPersonsT, list< [Connections](#) > listOfContsT, [Login](#) userT)
- void [fromWhereVehicle](#) (list< [Vehicle](#) > listOfVehiclesT, list< [Person](#) > listOfPersonsT, list< [Connections](#) > listOfContsT, [Login](#) userT)
- void [checkIfContinue](#) ([Login](#) &userT, list< [Login](#) > &listOfUsersT, list< [Vehicle](#) > &listOfVehiclesT, list< [Person](#) > &listOfPersonsT, [Person](#) &personsT)

4.4.1 Dokumentacja funkcji składowych

4.4.1.1 begin()

```
void Menu::begin (
    Login & userT,
    list< Login > & listOfUserT,
    list< Vehicle > & listOfVehiclesT,
    list< Person > & listOfPersonsT,
    Person & personsT )
```

Metoda ta służy do wywołania poszczególnych metod, aby wczytać dane z plików i przetrzymywać je w odpowiednich listach STL

Parametry

| | |
|------------------------|---|
| <i>userT</i> | Reprezentuje obiekt klasy Login |
| <i>listOfUserT</i> | lista STL zawierająca obiekty klasy Login |
| <i>listOfVehiclesT</i> | lista STL zawierająca obiekty klasy Vehicle |
| <i>listOfPersonsT</i> | lista STL zawierająca obiekty klasy Person |
| <i>personsT</i> | Reprezentuje obiekt klasy Person |

4.4.1.2 changePassword()

```
void Menu::changePassword (
    list< Login > & listOfUsersT,
    Login userT )
```

Metoda która odpowiada za wywołanie metody zmieniającej hasło użytkownika

Parametry

| | |
|---------------------|---|
| <i>listOfUsersT</i> | lista STL zawierająca obiekty klasy Login |
| <i>userT</i> | Reprezentuje obiekt klasy Login |

4.4.1.3 changeUserPassword()

```
void Menu::changeUserPassword (
    list< Login > & listOfUsersT,
    Login userT )
```

Metoda która odpowiada za wywołanie metody zmieniającej hasło dowolnego użytkownika

Parametry

| | |
|---------------------|---|
| <i>listOfUsersT</i> | lista STL zawierająca obiekty klasy Login |
| <i>userT</i> | Reprezentuje obiekt klasy Login |

4.4.1.4 checkIfContinue()

```
void Menu::checkIfContinue (
    Login & userT,
    list< Login > & listOfUsersT,
    list< Vehicle > & listOfVehiclesT,
    list< Person > & listOfPersonsT,
    Person & personsT )
```

Metoda służąca do sprawdzenia czy użytkownik chce kontynuować czy skończyć korzystanie z programu

Parametry

| | |
|------------------------|---|
| <i>userT</i> | Reprezentuje obiekt klasy Login |
| <i>listOfUsersT</i> | lista STL zawierająca obiekty klasy Login |
| <i>listOfVehiclesT</i> | lista STL zawierająca obiekty klasy Vehicle |
| <i>listOfPersonsT</i> | lista STL zawierająca obiekty klasy Person |
| <i>personsT</i> | Reprezentuje obiekt klasy Person |

4.4.1.5 checkWithPattern()

```
bool Menu::checkWithPattern (
    string str,
    regex pattern )
```

Uniwersalna metoda sprawdzająca za pomocą RegEx'a czy podany string jest zgodny z podanym wzorcem

Parametry

| | |
|----------------|---|
| <i>str</i> | Zawiera wyraz do sprawdzenia ze wzorcem |
| <i>pattern</i> | Zawiera wzorec, który jest porównywany ze stringiem |

4.4.1.6 deleteUser()

```
void Menu::deleteUser (
    list< Login > & listOfUsersT,
    Login userT )
```

Metoda odpowiada za wywołanie metody usuwającej użytkownika z bazy

Parametry

| | |
|---------------------|---|
| <i>listOfUsersT</i> | lista STL zawierająca obiekty klasy Login |
| <i>userT</i> | Reprezentuje obiekt klasy Login |

4.4.1.7 fromWhereVehicle()

```
void Menu::fromWhereVehicle (
    list< Vehicle > listOfVehiclesT,
    list< Person > listOfPersonsT,
    list< Connections > listOfContsT,
    Login userT )
```

Metoda ta służy do zobaczenia gdzie znajdują się pojazdy danej marki (na podstawie miejscowości zamieszkania właściciela)

Parametry

| | |
|------------------------|---|
| <i>listOfVehiclesT</i> | lista STL zawierająca obiekty klasy Vehicle |
| <i>listOfPersonsT</i> | lista STL zawierająca obiekty klasy Person |
| <i>listOfContsT</i> | lista STL zawierająca konekcje między osobami, a pojazdami |
| <i>userT</i> | Reprezentuje obiekt klasy Login |

4.4.1.8 guestOptions()

```
void Menu::guestOptions (
    User guest )
```

Metoda ta służy do wywoływania metod, które wybrał gość

Parametry

| | |
|--------------|---|
| <i>guest</i> | Reprezentuje obiekt klasy User (gość) |
|--------------|---|

4.4.1.9 login()

```
void Menu::login (
    Login & userT,
    list< Login > & listOfUserT,
    list< Vehicle > & listOfVehiclesT,
    list< Person > & listOfPersonsT,
    Person & personsT )
```

Metoda służy do logowania użytkownika jeśli wybrał logowanie w metodzie [loginOrGuest\(\)](#)

Parametry

| | |
|------------------------|---|
| <i>userT</i> | Reprezentuje obiekt klasy Login |
| <i>listOfUserT</i> | lista STL zawierająca obiekty klasy Login |
| <i>listOfVehiclesT</i> | lista STL zawierająca obiekty klasy Vehicle |
| <i>listOfPersonsT</i> | lista STL zawierająca obiekty klasy Person |
| <i>personsT</i> | Reprezentuje obiekt klasy Person |

4.4.1.10 loginOrGuest()

```
void Menu::loginOrGuest (
    Login & userT,
    list< Login > & listOfUserT,
    User & guestUser,
    list< Vehicle > & listOfVehiclesT,
    Vehicle & vehicleT,
    list< Person > & listOfPersonsT,
    Person & personsT )
```

Metoda dająca możliwość użytkownikowi zalogowania się bądź skorzystania z programu jako gość

Parametry

| | |
|------------------------|---|
| <i>userT</i> | Reprezentuje obiekt klasy Login |
| <i>listOfUserT</i> | lista STL zawierająca obiekty klasy Login |
| <i>guestUser</i> | Reprezentuje obiekt klasy User (gość) |
| <i>listOfVehiclesT</i> | lista STL zawierająca obiekty klasy Vehicle |
| <i>vehicleT</i> | Reprezentuje obiekt klasy Vehicle |
| <i>listOfPersonsT</i> | lista STL zawierająca obiekty klasy Person |
| <i>personsT</i> | Reprezentuje obiekt klasy Person |

4.4.1.11 userChoice()

```
void Menu::userChoice (
    Login & userT,
    list< Login > & listOfUsersT,
    list< Vehicle > & listOfVehiclesT,
    list< Person > & listOfPersonsT,
    Person & personsT )
```

Metoda ta służy do wywoływania metod, które wybrał użytkownik, bądź admin (dzięki zabezpieczeniom użytkownik nie jest w stanie wywołać metod przeznaczonych dla admina)

Parametry

| | |
|------------------------|---|
| <i>userT</i> | Reprezentuje obiekt klasy Login |
| <i>listOfUsersT</i> | lista STL zawierająca obiekty klasy Login |
| <i>listOfVehiclesT</i> | lista STL zawierająca obiekty klasy Vehicle |
| <i>listOfPersonsT</i> | lista STL zawierająca obiekty klasy Person |
| <i>personsT</i> | Reprezentuje obiekt klasy Person |

4.4.1.12 whichVehiclePersonHave()

```
void Menu::whichVehiclePersonHave (
    list< Vehicle > listOfVehiclesT,
    list< Person > listOfPersonsT,
    list< Connections > listOfContsT,
    Login userT )
```

Metoda służąca do zobaczenia jakie pojazdy posiada dana osoba

Parametry

| | |
|------------------------|---|
| <i>listOfVehiclesT</i> | lista STL zawierająca obiekty klasy Vehicle |
| <i>listOfPersonsT</i> | lista STL zawierająca obiekty klasy Person |
| <i>listOfContsT</i> | lista STL zawierająca konekcje między osobami, a pojazdami |
| <i>userT</i> | Reprezentuje obiekt klasy Login |

4.4.1.13 whoseVehicle()

```
void Menu::whoseVehicle (
    list< Vehicle > listOfVehiclesT,
    list< Person > listOfPersonsT,
    list< Connections > listOfContsT,
    Login userT )
```

Metoda służąca do zobaczenia kto jest właścicielem podanej marki pojazdu

Parametry

| | |
|------------------------|---|
| <i>listOfVehiclesT</i> | lista STL zawierająca obiekty klasy Vehicle |
| <i>listOfPersonsT</i> | lista STL zawierająca obiekty klasy Person |
| <i>listOfContsT</i> | lista STL zawierająca konekcje między osobami, a pojazdami |
| <i>userT</i> | Reprezentuje obiekt klasy Login |

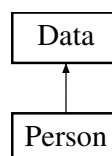
Dokumentacja dla tej klasy została wygenerowana z plików:

- [Menu.h](#)
- [Menu.cpp](#)

4.5 Dokumentacja klasy Person

```
#include <Person.h>
```

Diagram dziedziczenia dla Person



Metody publiczne

- [Person](#) ()
- [Person](#) (int idT, string nameT, string lastNameT, string mailT, int phoneNumberT, int birthYearT, string cityT, string streetT, string houseNumberT)
- string [getName](#) () const
- string [getLastName](#) () const
- void [print](#) ()
- void [printForUser](#) ()
- void [addToPersonList](#) (list< [Person](#) > &listOfPeronsT, [Person](#) &personsT)
- bool [isPersonInBase](#) (list< [Person](#) > &listOfPeronsT, int idT)
- void [addFromFile](#) (list< [Person](#) > &listOfPersonsT)

- void `showPersons` (list< `Person` > listOfPeronsT, int isAdmin)
- void `toFile` (list< `Person` > &listOfPeronsT)
- void `addNewPerson` (list< `Person` > &listOfPeronsT)
- void `deleteFromList` (list< `Person` > &listOfPeronsT, int idT)
- int `deletePerson` (list< `Person` > &listOfPeronsT)
- void `showPerson` (list< `Person` > listOfPeronsT, int isAdmin, int idP)
- int `isGivenPersonInBase` (list< `Person` > listOfPeronsT, string nameT, string lastNameT)
- int `isCityInBase` (list< `Person` > listOfPeronsT, string cityT)

Dodatkowe Dziedziczone Składowe

4.5.1 Opis szczegółowy

Klasa reprezentująca obiekt `Person`, dziedzicząca publicznie po klasie `Data`

4.5.2 Dokumentacja konstruktora i destruktor

4.5.2.1 `Person()` [1/2]

```
Person::Person ( ) [inline]
```

Konstruktor domyślny korzystający z konstruktora klasy dziedzicznej

4.5.2.2 `Person()` [2/2]

```
Person::Person (
    int idT,
    string nameT,
    string lastNameT,
    string mailT,
    int phoneNumberT,
    int birthYearT,
    string cityT,
    string streetT,
    string houseNumberT ) [inline]
```

Konstruktor dziewięcioargumentowy

Parametry

| | |
|---------------------------|-------------------------------|
| <code>idT</code> | Zawiera numer id osoby |
| <code>nameT</code> | Zawiera imię osoby |
| <code>lastNameT</code> | Zawiera nazwisko osoby |
| <code>mailT</code> | Zawiera adres mailowy |
| <code>phoneNumberT</code> | Zawiera numer telefonu |
| <code>birthYearT</code> | Zawiera rok urodzenia |
| <code>cityT</code> | Zawiera nazwę miejscowości |
| <code>streetT</code> | Zawiera nazwę ulicy |
| <code>houseNumberT</code> | Zawiera numer domu/mieszkania |

4.5.3 Dokumentacja funkcji składowych

4.5.3.1 addFromFile()

```
void Person::addFromFile (
    list< Person > & listOfPersonsT )
```

Metoda służąca do odczytywania danych z pliku txt, następnie tworząca obiekt klasy [Person](#) i przekazująca do metody [addToPersonList\(\)](#)

Parametry

| | |
|-----------------------|--|
| <i>listOfPersonsT</i> | lista STL zawierająca obiekty klasy Person |
|-----------------------|--|

4.5.3.2 addNewPerson()

```
void Person::addNewPerson (
    list< Person > & listOfPeronsT )
```

Metoda dodająca nową osobę do bazy

Parametry

| | |
|----------------------|--|
| <i>listOfPeronsT</i> | lista STL zawierająca obiekty klasy Person |
|----------------------|--|

4.5.3.3 addToPersonList()

```
void Person::addToPersonList (
    list< Person > & listOfPeronsT,
    Person & personsT )
```

Metoda dodająca do listy STL obiekty klasy [Person](#)

Parametry

| | |
|----------------------|--|
| <i>listOfPeronsT</i> | lista STL zawierająca obiekty klasy Person |
| <i>personsT</i> | Reprezentuje obiekt klasy Person |

4.5.3.4 deleteFromList()

```
void Person::deleteFromList (
    list< Person > & listOfPeronsT,
    int idT )
```

Metoda usuwająca z listy osobę o podanym numerze id

Parametry

| | |
|----------------------|--|
| <i>listOfPeronsT</i> | lista STL zawierająca obiekty klasy Person |
| <i>idT</i> | Numer id do usunięcia z listy |

4.5.3.5 deletePerson()

```
int Person::deletePerson (
    list< Person > & listOfPeronsT )
```

Metoda usuwająca osobę z bazy

Parametry

| | |
|----------------------|--|
| <i>listOfPeronsT</i> | lista STL zawierająca obiekty klasy Person |
|----------------------|--|

4.5.3.6 getLastName()

```
string Person::getLastName ( ) const [inline]
```

Metoda zwracająca nazwisko osoby

4.5.3.7 getName()

```
string Person::getName ( ) const [inline]
```

Metoda zwracająca imię osoby

4.5.3.8 isCityInBase()

```
int Person::isCityInBase (
    list< Person > listOfPeronsT,
    string cityT )
```

Metoda sprawdzająca czy istnieje taka miejscowość w bazie, jeśli tak to zwraca numer id osoby, która mieszka w tej miejscowości

Parametry

| | |
|----------------------|--|
| <i>listOfPeronsT</i> | lista STL zawierająca obiekty klasy Person |
| <i>cityT</i> | Zawiera nazwę miejscowości podane przez użytkownika |

4.5.3.9 isGivenPersonInBase()

```
int Person::isGivenPersonInBase (
    list< Person > listOfPeronsT,
    string nameT,
    string lastNameT )
```

Metoda sprawdzająca czy osoba o podanym imieniu i nazwisku znajduje się w bazie i zwracająca jej numer id

Parametry

| | |
|----------------------|--|
| <i>listOfPeronsT</i> | lista STL zawierająca obiekty klasy Person |
| <i>nameT</i> | Zawiera imię wpisane przez użytkownika |
| <i>lastNameT</i> | Zawiera nazwisko wpisane przez użytkownika |

4.5.3.10 isPersonInBase()

```
bool Person::isPersonInBase (
    list< Person > & listOfPeronsT,
    int idT )
```

Metoda sprawdzająca czy osoba o podanym numerze id znajduje się w bazie

Parametry

| | |
|----------------------|--|
| <i>listOfPeronsT</i> | lista STL zawierająca obiekty klasy Person |
| <i>idT</i> | Numer id, który jest sprawdzany czy znajduje się w liście osób |

4.5.3.11 print()

```
void Person::print ( ) [virtual]
```

Metoda przysłaniająca metodę wirtualną, służy do wyświetlenia pełnych informacji o osobie dla użytkownika

Implementuje [Data](#).

4.5.3.12 printForUser()

```
void Person::printForUser ( )
```

Metoda wypisująca imię, nazwisko oraz miejscowość osoby dla użytkownika

4.5.3.13 showPerson()

```
void Person::showPerson (
    list< Person > listOfPeronsT,
    int isAdmin,
    int idP )
```

Metoda wyświetlająca informacje o podanej osobie (pełne informacje są tylko ukazywane adminowi)

Parametry

| | |
|----------------------|---|
| <i>listOfPeronsT</i> | lista STL zawierająca obiekty klasy Person |
| <i>isAdmin</i> | Zawiera informację czy osoba wywołująca metodę jest adminem |
| <i>idP</i> | Zawiera id osoby, której informacje mają być wyświetlone |

4.5.3.14 showPersons()

```
void Person::showPersons (
    list< Person > listOfPeronsT,
    int isAdmin )
```

Metoda w zależności od tego czy użytkownik jest adminem wyświetla pełne bądź tylko ogólne informacje o osobach w bazie

Parametry

| | |
|----------------------|---|
| <i>listOfPeronsT</i> | lista STL zawierająca obiekty klasy Person |
| <i>isAdmin</i> | Zawiera informację czy użytkownik wywołujący tę metodę ma uprawnienia admina (1 - ma) |

4.5.3.15 toFile()

```
void Person::toFile (
    list< Person > & listOfPeronsT )
```

Metoda zapisująca do pliku txt zawartość listy person

Parametry

| | |
|----------------------------|--|
| <code>listOfPeronsT</code> | lista STL zawierająca obiekty klasy Person |
|----------------------------|--|

Dokumentacja dla tej klasy została wygenerowana z plików:

- [Person.h](#)
- [Person.cpp](#)

4.6 Dokumentacja klasy User

```
#include <User.h>
```

Metody publiczne

- [User](#) ()
- void [guestMenu](#) ([User](#) guest)
- bool [getIsAdmin](#) () const

4.6.1 Opis szczegółowy

Klasa reprezentująca obiekt user (gość)

4.6.2 Dokumentacja konstruktora i destruktora

4.6.2.1 User()

```
User::User ( ) [inline]
```

Konstruktor domyślny

4.6.3 Dokumentacja funkcji składowych

4.6.3.1 getIsAdmin()

```
bool User::getIsAdmin ( ) const [inline]
```

Metoda zwracająca zmienną isAdmin zawierającą informację czy gość jest administratorem

4.6.3.2 guestMenu()

```
void User::guestMenu (  
    User guest )
```

Metoda służąca do wyświetlenia menu gościa

Parametry

| | |
|--------------------|--|
| <code>guest</code> | Reprezentuje obiekt klasy User |
|--------------------|--|

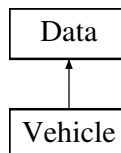
Dokumentacja dla tej klasy została wygenerowana z plików:

- [User.h](#)
- [User.cpp](#)

4.7 Dokumentacja klasy Vehicle

```
#include <Vehicle.h>
```

Diagram dziedziczenia dla Vehicle



Metody publiczne

- [Vehicle](#) ()
- [Vehicle](#) (int idT, string typeOfVehicleT, string brandT, string modelT, int power, string engineT, string colourT)
- string [getBrand](#) () const
- void [print](#) ()
- void [printForUser](#) ()
- void [addToVehicleList](#) (list< [Vehicle](#) > &listOfVehiclesT, [Vehicle](#) &vehicleT)
- void [addFromFile](#) (list< [Vehicle](#) > &listOfVehiclesT)
- void [showVehicles](#) (list< [Vehicle](#) > listOfVehiclesT, int isAdmin)
- void [showUserVehicles](#) (list< [Vehicle](#) > listOfVehiclesT, int idV)
- bool [isVehicleInBase](#) (list< [Vehicle](#) > &listOfVehiclesT, int idT)
- void [toFile](#) (list< [Vehicle](#) > &listOfVehiclesT)
- void [addNewVehicle](#) (list< [Vehicle](#) > &listOfVehiclesT)
- void [deleteFromList](#) (list< [Vehicle](#) > &listOfVehiclesT, int idT)
- int [deleteVehicle](#) (list< [Vehicle](#) > &listOfVehiclesT)
- int [searchForVehicle](#) (list< [Vehicle](#) > listOfVehiclesT, string brandT)
- void [showVehicle](#) (list< [Vehicle](#) > listOfVehiclesT, int isAdmin, int idV)

Dodatkowe Dziedziczone Składowe

4.7.1 Opis szczegółowy

Klasa reprezentująca obiekt [Vehicle](#), dziedzicząca publicznie po klasie [Data](#)

4.7.2 Dokumentacja konstruktora i destruktora

4.7.2.1 Vehicle() [1/2]

```
Vehicle::Vehicle ( ) [inline]
```

Konstruktor domyślny korzystający z konstruktora klasy [Data](#)

4.7.2.2 Vehicle() [2/2]

```
Vehicle::Vehicle (
    int idT,
    string typeOfVehicleT,
    string brandT,
    string modelT,
    int power,
    string engineT,
    string colourT ) [inline]
```

Konstruktor siedmioargumentowy

Parametry

| | |
|-----------------------|--------------------------|
| <i>idT</i> | Zawiera numer id pojazdu |
| <i>typeOfVehicleT</i> | Zawiera typ pojazdu |
| <i>brandT</i> | Zawiera markę pojazdu |
| <i>modelT</i> | Zawiera model pojazdu |
| <i>power</i> | Zawiera moc pojazdu |
| <i>engineT</i> | Zawiera układ silnika |
| <i>colourT</i> | Zawiera kolor pojazdu |

4.7.3 Dokumentacja funkcji składowych

4.7.3.1 addFromFile()

```
void Vehicle::addFromFile (
    list< Vehicle > & listOfVehiclesT )
```

Metoda służąca do odczytywania danych z pliku txt, następnie tworząca obiekt klasy [Vehicle](#) i przekazująca go do metody [addToVehicleList\(\)](#)

Parametry

| | |
|------------------------|---|
| <i>listOfVehiclesT</i> | lista STL zawierająca obiekty klasy Vehicle |
|------------------------|---|

4.7.3.2 addNewVehicle()

```
void Vehicle::addNewVehicle (
    list< Vehicle > & listOfVehiclesT )
```

Metoda dodająca nowy pojazd do bazy

Parametry

| | |
|------------------------|---|
| <i>listOfVehiclesT</i> | lista STL zawierająca obiekty klasy Vehicle |
|------------------------|---|

4.7.3.3 addToVehicleList()

```
void Vehicle::addToVehicleList (
    list< Vehicle > & listOfVehiclesT,
    Vehicle & vehicleT )
```

Metoda służy do dodawania obiektów klasy [Vehicle](#) do listy STL

Parametry

| | |
|------------------------|---|
| <i>listOfVehiclesT</i> | lista STL zawierająca obiekty klasy Vehicle |
| <i>vehicleT</i> | Reprezentuje obiekt klasy Vehicle |

4.7.3.4 deleteFromList()

```
void Vehicle::deleteFromList (
    list< Vehicle > & listOfVehiclesT,
    int idT )
```

Metoda usuwająca z listy pojazd o podanym numerze id

Parametry

| | |
|------------------------|---|
| <i>listOfVehiclesT</i> | lista STL zawierająca obiekty klasy Vehicle |
| <i>idT</i> | Numer id do usunięcia z listy |

4.7.3.5 deleteVehicle()

```
int Vehicle::deleteVehicle (
    list< Vehicle > & listOfVehiclesT )
```

Metoda usuwająca z bazy pojazdów dany pojazd

Parametry

| | |
|------------------------|---|
| <i>listOfVehiclesT</i> | lista STL zawierająca obiekty klasy Vehicle |
|------------------------|---|

4.7.3.6 getBrand()

```
string Vehicle::getBrand ( ) const [inline]
```

Metoda zwracająca markę pojazdu

4.7.3.7 isVehicleInBase()

```
bool Vehicle::isVehicleInBase (
    list< Vehicle > & listOfVehiclesT,
    int idT )
```

Metoda ta służy do sprawdzenia czy pojazd o podanym numerze id znajduje się w bazie

Parametry

| | |
|------------------------|---|
| <i>listOfVehiclesT</i> | lista STL zawierająca obiekty klasy Vehicle |
| <i>idT</i> | Zawiera numer id pojazdu podanego przez użytkownika |

4.7.3.8 print()

```
void Vehicle::print ( ) [virtual]
```

Metoda przysłaniająca metodę wirtualną, służy do wyświetlenia pełnych informacji o pojeździe

Implementuje [Data](#).

4.7.3.9 printForUser()

```
void Vehicle::printForUser ( )
```

Metoda wypisująca wszystkie dane pojazdu oprócz numeru id

4.7.3.10 searchForVehicle()

```
int Vehicle::searchForVehicle (
    list< Vehicle > listOfVehiclesT,
    string brandT )
```

Metoda ta zwraca numer id pojazdu, który posiada podaną markę

Parametry

| | |
|------------------------|---|
| <i>listOfVehiclesT</i> | lista STL zawierająca obiekty klasy Vehicle |
| <i>brandT</i> | Zawiera markę podaną przez użytkownika |

4.7.3.11 showUserVehicles()

```
void Vehicle::showUserVehicles (
    list< Vehicle > listOfVehiclesT,
    int idV )
```

Metoda służąca do wyświetlenia pełnych danych o pojazdach zalogowanego użytkownika

Parametry

| | |
|------------------------|---|
| <i>listOfVehiclesT</i> | lista STL zawierająca obiekty klasy Vehicle |
| <i>idV</i> | Zawiera id pojazdu, którego właścicielem jest użytkownik |

4.7.3.12 showVehicle()

```
void Vehicle::showVehicle (
    list< Vehicle > listOfVehiclesT,
    int isAdmin,
    int idV )
```

Metoda ta wyświetla informacje o pojeździe o podanym numerze id

Parametry

| | |
|------------------------|---|
| <i>listOfVehiclesT</i> | lista STL zawierająca obiekty klasy Vehicle |
| <i>isAdmin</i> | Zawiera informację o tym czy aktualny użytkownik jest adminem |
| <i>idV</i> | Zawiera numer id pojazdu, dla którego mają być wyświetlone dane |

4.7.3.13 showVehicles()

```
void Vehicle::showVehicles (
    list< Vehicle > listOfVehiclesT,
    int isAdmin )
```

Metoda ta w zależności od tego czy użytkownik jest adminem wywołuje metodę [print\(\)](#) bądź [printForUser\(\)](#) gdy nie jest adminem

Parametry

| | |
|------------------------|---|
| <i>listOfVehiclesT</i> | lista STL zawierająca obiekty klasy Vehicle |
| <i>isAdmin</i> | Zawiera informację o tym czy użytkownik jest adminem |

4.7.3.14 toFile()

```
void Vehicle::toFile (
    list< Vehicle > & listOfVehiclesT )
```

Metoda służąca do zapisu do pliku txt zawartości listy pojazdów

Parametry

| | |
|------------------------|---|
| <i>listOfVehiclesT</i> | lista STL zawierająca obiekty klasy Vehicle |
|------------------------|---|

Dokumentacja dla tej klasy została wygenerowana z plików:

- [Vehicle.h](#)
- [Vehicle.cpp](#)

Rozdział 5

Dokumentacja plików

5.1 Dokumentacja pliku Connections.h

```
#include <iostream>
#include <list>
#include "Login.h"
#include "Data.h"
#include "Person.h"
#include "Vehicle.h"
```

Komponenty

- class [Connections](#)

5.2 Dokumentacja pliku Data.h

```
#include <iostream>
```

Komponenty

- class [Data](#)

5.3 Dokumentacja pliku Login.h

```
#include <iostream>
#include <string>
#include <fstream>
#include <list>
#include "Data.h"
#include "Vehicle.h"
#include "Person.h"
```

Komponenty

- class [Login](#)

5.4 Dokumentacja pliku Menu.h

```
#include <iostream>
#include <string>
#include <list>
#include <regex>
#include "Login.h"
#include "User.h"
#include "Data.h"
#include "Vehicle.h"
#include "Person.h"
#include "Connections.h"
```

Komponenty

- class [Menu](#)

5.5 Dokumentacja pliku Person.h

```
#include <iostream>
#include <iterator>
#include <string>
#include <list>
#include "Data.h"
```

Komponenty

- class [Person](#)

5.6 Dokumentacja pliku User.h

```
#include <iostream>
#include <string>
#include <fstream>
#include <list>
#include "Login.h"
```

Komponenty

- class [User](#)

5.7 Dokumentacja pliku Vehicle.h

```
#include <iostream>
#include <iterator>
#include <string>
#include <list>
```

Komponenty

- class [Vehicle](#)

Indeks

- addConnection
 - Connections, [8](#)
- addFromFile
 - Connections, [9](#)
 - Login, [17](#)
 - Person, [30](#)
 - Vehicle, [36](#)
- addNewPerson
 - Person, [30](#)
- addNewVehicle
 - Vehicle, [37](#)
- addToConnectionsList
 - Connections, [9](#)
- addToList
 - Login, [17](#)
- addToPersonList
 - Person, [30](#)
- addToVehicleList
 - Vehicle, [37](#)
- addUser
 - Login, [17](#)
- adminUserMenu
 - Login, [18](#)
- begin
 - Menu, [23](#)
- changeAnyPassword
 - Login, [18](#)
- changePassword
 - Login, [19](#)
 - Menu, [24](#)
- changeUserPassword
 - Menu, [24](#)
- checkIfConnectionExists
 - Connections, [9](#)
- checkIfContinue
 - Menu, [24](#)
- checkWithPattern
 - Menu, [25](#)
- Connections, [7](#)
 - addConnection, [8](#)
 - addFromFile, [9](#)
 - addToConnectionsList, [9](#)
 - checkIfConnectionExists, [9](#)
 - Connections, [8](#)
 - deleteConnection, [10](#)
 - deleteFromList, [10](#)
 - deleteWhenPerson, [10](#)
 - deleteWhenVehicle, [12](#)
 - findUserByVehicle, [12](#)
 - findVehicleByUser, [12](#)
 - print, [13](#)
 - showConnections, [13](#)
 - showUserCars, [13](#)
 - toFile, [14](#)
- Connections.h, [41](#)
- Data, [14](#)
 - Data, [15](#)
 - print, [15](#)
- Data.h, [41](#)
- deleteConnection
 - Connections, [10](#)
- deleteFromBase
 - Login, [19](#)
- deleteFromList
 - Connections, [10](#)
 - Person, [30](#)
 - Vehicle, [37](#)
- deletePerson
 - Person, [31](#)
- deleteUser
 - Menu, [25](#)
- deleteVehicle
 - Vehicle, [38](#)
- deleteWhenPerson
 - Connections, [10](#)
- deleteWhenVehicle
 - Connections, [12](#)
- findUserByVehicle
 - Connections, [12](#)
- findVehicleByUser
 - Connections, [12](#)
- fromWhereVehicle
 - Menu, [25](#)
- getBrand
 - Vehicle, [38](#)
- getIsAdmin
 - Login, [19](#)
 - User, [34](#)
- getLastName
 - Person, [31](#)
- getName
 - Person, [31](#)
- getPassword
 - Login, [19](#)
- getUserID

- Login, 19
- guestMenu
 - User, 34
- guestOptions
 - Menu, 26
- isCityInBase
 - Person, 31
- isGivenPersonInBase
 - Person, 32
- isIdInBase
 - Login, 20
- isIdNumber
 - Login, 20
- isPersonInBase
 - Person, 32
- isUserInBase
 - Login, 20
- isVehicleInBase
 - Vehicle, 38
- Login, 15
 - addFromFile, 17
 - addToList, 17
 - addUser, 17
 - adminUserMenu, 18
 - changeAnyPassword, 18
 - changePassword, 19
 - deleteFromBase, 19
 - getIsAdmin, 19
 - getPassword, 19
 - getUserID, 19
 - isIdInBase, 20
 - isIdNumber, 20
 - isUserInBase, 20
 - Login, 16
 - normalUserMenu, 21
 - searchForUser, 21
 - setPassword, 21
 - show, 22
 - toFile, 22
- login
 - Menu, 26
- Login.h, 41
- loginOrGuest
 - Menu, 26
- Menu, 23
 - begin, 23
 - changePassword, 24
 - changeUserPassword, 24
 - checkIfContinue, 24
 - checkWithPattern, 25
 - deleteUser, 25
 - fromWhereVehicle, 25
 - guestOptions, 26
 - login, 26
 - loginOrGuest, 26
 - userChoice, 27
 - whichVehiclePersonHave, 27
 - whoseVehicle, 28
- Menu.h, 42
- normalUserMenu
 - Login, 21
- Person, 28
 - addFromFile, 30
 - addNewPerson, 30
 - addToList, 30
 - deleteFromList, 30
 - deletePerson, 31
 - getLastName, 31
 - getName, 31
 - isCityInBase, 31
 - isGivenPersonInBase, 32
 - isPersonInBase, 32
 - Person, 29
 - print, 32
 - printForUser, 32
 - showPerson, 33
 - showPersons, 33
 - toFile, 33
- Person.h, 42
- print
 - Connections, 13
 - Data, 15
 - Person, 32
 - Vehicle, 38
- printForUser
 - Person, 32
 - Vehicle, 38
- searchForUser
 - Login, 21
- searchForVehicle
 - Vehicle, 39
- setPassword
 - Login, 21
- show
 - Login, 22
- showConnections
 - Connections, 13
- showPerson
 - Person, 33
- showPersons
 - Person, 33
- showUserCars
 - Connections, 13
- showUserVehicles
 - Vehicle, 39
- showVehicle
 - Vehicle, 39
- showVehicles
 - Vehicle, 40
- toFile
 - Connections, 14

- Login, [22](#)
- Person, [33](#)
- Vehicle, [40](#)

- User, [34](#)
 - getIsAdmin, [34](#)
 - guestMenu, [34](#)
 - User, [34](#)
- User.h, [42](#)
- userChoice
 - Menu, [27](#)

- Vehicle, [35](#)
 - addFromFile, [36](#)
 - addNewVehicle, [37](#)
 - addToVehicleList, [37](#)
 - deleteFromList, [37](#)
 - deleteVehicle, [38](#)
 - getBrand, [38](#)
 - isVehicleInBase, [38](#)
 - print, [38](#)
 - printForUser, [38](#)
 - searchForVehicle, [39](#)
 - showUserVehicles, [39](#)
 - showVehicle, [39](#)
 - showVehicles, [40](#)
 - toFile, [40](#)
 - Vehicle, [36](#)
- Vehicle.h, [43](#)

- whichVehiclePersonHave
 - Menu, [27](#)
- whoseVehicle
 - Menu, [28](#)