

stanZdalnyMF

Bieżąca wersja programu: 1.1.3 z dnia 2025-10-22

Współpracuje z bazą danych programu motoFirma od wersji 1.23.752

Autor: Wojciech Pawłowski ©2025

Spis treści

1. Informacje ogólne	2
1.1. Tryb GET	2
1.2. Tryb POST	2
2. Instalacja programu	3
3. Konfiguracja	4
3.1. Sekcja [stan]	4
3.2. Sekcje definicji dostawców	4
3.3. Przechowywanie tokenów sesji	4
4. Typy obsługiwanych web service	5
4.1. typWS=asco	5
4.2. typWS=ascoWebAPI	6
4.3. typWS=autopartner	6
4.4. typWS=denckermann	6
4.5. typWS=mekonomen	6
4.6. typWS=falcon	7
4.7. typWS=hart	7
4.8. typWS=intercars	8
4.9. typWS=interteam	8
4.10. typWS=intervito	9
4.11. typWS=motofirma	9
4.12. typWS=motoleg	10
4.13. typWS=motoprofil	10
4.14. typWS=motorol	10
4.15. typWS=nextis	10
4.16. typWS=NextisErpAPI	11
4.17. typWS=polcar	12
4.18. typWS=vanking	12
5. Migracja ustawień z programu stanMotonet	12

1. Informacje ogólne

Program **stanZdalnyMF** umożliwia sprawdzanie stanu magazynowego towarów u wybranych dostawców. Współpracuje z programem **motoShop** oraz z programem **mSklep**, prezentując bieżącą dostępność towarów u dostawców. Dane pobierane są dynamicznie poprzez mechanizmy **WebService**.

Program **stanZdalnyMF** może pracować w dwóch trybach: **get** oraz **post**.

1.1. Tryb GET

Tryb GET jako argument posiada pojedynczy parametr **query url** i zwraca stan jednego towaru u jednego dostawcy, na przykład wywołanie:

```
https://mojadomena.pl/stan/stan.asp?m=FTROP570
```

zwróci informację o dostępności produktu *Filtron OP570* u dostawcy zdefiniowanego w sekcji **[m]** pliku **stanZdalnyMF.ini**.

1.2. Tryb POST

W trybie POST program **stanZdalnyMF** zwraca informację o dostępności u wybranego dostawcy całej listy produktów. Żądanie o stan odbywa się poprzez wysłanie formularza **html** zawierającego pole **input** o zdefiniowanej nazwie, w którym jako tekst, przechowywany jest **XML** lub **JSON** zawierający informacje zarówno o nazwie sekcji definiującej dostawcę, jak również listę produktów do sprawdzenia dostępności. Zamiast formularza można umieścić **XML** lub **JSON** w sekcji **BODY** żądania HTTP (patrz parametr **form**)

Przykład XML:

```
<z r="n7" s="5">
  <t k="VERTH1290.7" i="165234"/>
  <t k="VERTH3327.81J" i="165235"/>
  <t k="VERTH4495.83" i="165236"/>
</z>
```

Przykład JSON:

```
{
  "z": {
    "r": "n7",
    "s": "5",
    "t": [
      {
        "k": "VERTH1290.7",
        "i": "165234"
      },
      {
        "k": "VERTH3327.81J",
        "i": "165235"
      },
      {
        "k": "VERTH4495.83",
        "i": "165236"
      }
    ]
  }
}
```

Gdzie argument **r** określa sekcję, **s** oznacza maksymalny pokazywany stan, **k** określa kod towaru u tego dostawcy a argument **i** jest unikatowym identyfikatorem karty produktu w programie **mSklep**, do którego ma trafić informacja zwrotna.

W zwrotnym XML lub JSON parametr **k** zawiera informację o dostępności. Jeśli w sekcji **[stan]** parametr **stanyZerowe=0** a stan produktu u dostawcy jest zerowy, element **<t>** tego produktu jest usuwany z xml zwrotnego.

Przykład XML:

```
<z r="n7" s="5">
  <t k="5" i="165234"/>
  <t k="2" i="165235"/>
  <t k="5" i="165236"/>
</z>
```

Przykład JSON:

```
{
  "z": {
    "r": "n7",
    "s": "5",
    "t": [
      {
        "k": "5",
        "i": "165234"
      },
      {
        "k": "2",
        "i": "165235"
      },
      {
        "k": "5",
        "i": "165236"
      }
    ]
  }
}
```

2. Instalacja programu

1. Pliki składowe najnowszej wersji programu **stanZdalnyMF** należy pobrać z lokalizacji:

```
https://moto-firma.pl/pobieranie/stanZdalnyMF.zip
```

2. W dalszych punktach przyjęto, że katalogiem roboczym programu **stanZdalny** jest:

```
C:\inetpub\wwwroot\sklep\stan\
```

3. Zatrzymać usługi internetowe IIS.
4. Jeśli biblioteka **stanZdalnyMF.dll** była już wcześniej zarejestrowana, należy ją wyrejestrować poleceniem:

```
regsvr32 C:\inetpub\wwwroot\sklep\stan\stanZdalnyMF.dll /u
```

5. Pobrany plik archiwum **stanZdalnyMF.zip** należy rozpakować do katalogu roboczego. Zawiera on pliki:

```
stanZdalny.dll      stanZdalny.ini      stan.asp
```

6. Bibliotekę **stanZdalnyMF.dll** należy zarejestrować poleceniem:

```
regsvr32 C:\inetpub\wwwroot\sklep\stan\stanZdalnyMF.dll
```

7. Uruchomić usługi internetowe IIS.
8. Otworzyć plik **stanZdalnyMF.ini**, najlepiej za pomocą programu **Notepad++** w celu konfiguracji parametrów pracy.

3. Konfiguracja

Plik `stanZdalnyMF.ini` jest plikiem tekstowym (kodowanie ANSI). Zawiera wiersze, w których są sekcje oznakowane w formacie `[nazwa_sekcji]`. Po wystąpieniu nazwy sekcji wymagane są parametry w osobnych wierszach.

- Parametr składa się z nazwy, znaku = oraz wartości, np: `login=abcd123`
- Nazwy sekcji muszą być unikatowe.
- Nazwy parametrów w obrębie pojedynczej sekcji muszą być unikatowe.
- Wielkość liter w nazwach sekcji oraz w nazwach parametrów nie ma znaczenia.
- Wielkość liter w wartościach parametrów ma znaczenie.
- Każdy zapis musi znajdować się w pojedynczej linii tekstu.
- Linie tekstu oddzielane są znakami nowej linii (CRLF).
- Plik może zawierać puste wiersze w dowolnym miejscu.
- Jeśli wiersz rozpoczyna się znakiem ; (średnik) traktowany jest jako komentarz.

3.1. Sekcja [stan]

```
[stan]
klucz=klucz licencyjny
HTTP_REFERER=https://mojadomena.pl/sklep/
max=5
stanyZerowe=0 lub 1
form=nazwa pola formularza HTML dla trybu POST
json=0 lub 1
```

- `HTTP_REFERER` - parametr opcjonalny, zabezpiecza przed zadaniami http spoza strony macierzystej mSklep
- `max` - definiuje maksymalny prezentowany stan dostawców w trybie `GET`, dotyczy to także trybu `POST` jeśli w XML atrybut `s` nie zostanie podany, w trybie `GET` jeśli stan u dostawcy będzie większy niż zdefiniowany, zwracany tekst zostanie poprzedzony znakiem `>`, np: `>5`
- `stanyZerowe` - jeśli `stanyZerowe=0` stan zerowy dostawcy nie będzie zwracany, jeśli `stanyZerowe=1` stan zerowy dostawcy będzie zwracany
- `form` - parametr opcjonalny, poprawna nazwa pola formularza HTML zadanía `POST`
- `json` - jeśli `json=1` argumentem zadanía musi być kodowany w `JSON`, `json=0` argument zadanía musi być kodowany w XML

3.2. Sekcje definicji dostawców

Nazwy sekcji definicji dostawców mogą być dowolne. Mogą składać się z dowolnych liter i cyfr. Powinny rozpoczynać się literą.

W plikach składowych programu mSklep o nazwie: `zdalne.xml` lub `stany.xml` nazwy sekcji powinny być przekazywane jako atrybut o nazwie `parametr`.

W programie motoshop nazwy sekcji przekazywane są jak poprzednio w parametrach `zdalny` w sekcji `[db]`, `zdalnyX=rodzaj_indeksu|nazwa_magazynu|stan/stan.asp|x` (tu `x` na końcu jest nazwą sekcji).

Każda sekcja definicji dostawcy powinna zawierać parametr: `typWS` oraz inne parametry wymagane dla danego typu, np: `login`, `haslo`, `url`, `filia`.

3.3. Przechowywanie tokenów sesji

Niektóre rodzaje web service wymagają autoryzacji za pomocą tokenów sesji. Token sesji generowany jest po stronie dostawcy na podstawie `loginu` oraz `hasła` i służy do autoryzacji połączenia przez określony czas, np 5 godzin. Token ten pamiętany jest przez program `stanZdalnyMF` w rejestrze Windows pod nazwą zgodną z nazwą sekcji. Jeśli istnieje potrzeba sprawdzania stanów magazynowych tego samego dostawcy w różnych filiach, zalecane jest aby sprawdzenia stanów odbywały się z podaniem tego samego tokenu sesji.

Program `stanZdalnyMF` umożliwia pracę na wspólnym tokenie sesji dla różnych definicji web service. Sekcja definiująca takiego dostawcę powinna posiadać parametr `token` o wartości równej nazwie sekcji, dla której będzie pamiętany token sesji, np: `token=U7`, gdzie `u7` jest nazwą sekcji innej definicji tego samego dostawcy.

Praca na wspólnych tokenach sesji posiada zalety w postaci mniejszego obciążenia serwera dostawcy i w rezultacie szybszej obsługi każdego z ządań sprawdzania stanów.

4. Typy obsługiwanych web service

typWS	protokół	token	info
asco	SOAP XML		patrz Sekcja 4.1
ascoWebAPI	REST JSON	tak	patrz Sekcja 4.2
autopartner	REST JSON		patrz Sekcja 4.3 <i>Auto-Partner Bieruń</i>
denckermann	SOAP XML		patrz Sekcja 4.4
mekonomen	SOAP XML		patrz Sekcja 4.5 <i>Mekonomen, dawny Elit PL</i>
falcon	REST JSON	tak	patrz Sekcja 4.6
hart	REST JSON	tak	patrz Sekcja 4.7
intercars	REST JSON	tak	patrz Sekcja 4.8
interteam	SOAP XML		patrz Sekcja 4.9
intervito	SOAP XML		patrz Sekcja 4.10
motofirma	SOAP XML		patrz Sekcja 4.11
motoleg	XML		patrz Sekcja 4.12
motoprofil	SOAP XML		patrz Sekcja 4.13
motorol	SOAP XML		patrz Sekcja 4.14
nextis	SOAP XML		patrz Sekcja 4.15
nextisErpAPI	REST JSON	tak	patrz Sekcja 4.16
polcar	SOAP XML		patrz Sekcja 4.17
vanking	REST JSON		patrz Sekcja 4.18

4.1. typWS=asco

parametr	wartość	info
typWS	asco	
login	<i>Login</i>	
hasło	<i>Password</i>	
url	<i>url</i>	
filia	<i>filia</i>	np: 1, C

Przykładowe adresy url

dostawca	url
AJS Parts Ożarów Mazowiecki	http://ajsparts.pl:82/F20000openService.svc
Auto-France Wrocław	http://sklep.auto-france.com.pl:8082/F20000openService.svc
Auto-GH Świdnik	http://ows.auto-gh.pl/F20000openService.svc
JLK Kielce	http://ows.auto-gh.pl/F20000openService.svc
Mannol Polska Płock	http://sklep.mannolpolska.pl:8081/F20000openService.svc
MOTORES Jasionka	http://sklep.motores.pl:82/F20000openService.svc
Peace	http://212.180.231.199:88/F20000openService.svc

4.2. typWS=ascoWebAPI

parametr	wartość	info
typWS	ascoWebAPI	
login	Login	
hasło	Password	
url	url	
filia	filia	np: 1, C, 1+, znak + zwróci sumaryczny stan wszystkich filii dostawcy
token	sekcja	opcjonalne, podać nazwę sekcji, dla której będzie pamiętany token sesji

Przykładowe parametry url:

dostawca	url
Saga	https://webapi.sagaauto.com.pl/client-service/
Gordon	https://webapi.gordon.com.pl/client-service/

4.3. typWS=autopartner

parametr	wartość	info
typWS	autopartner	Auto-Partner Bieruń
login	clientCode	
hasło	clientPassword	
url	url	https://customerapi.autopartner.dev/CustomerAPI.svc/rest
filia	filia	symbol filii, opcjonalnie po znaku + symbole innych filii: 08, 08+12, 08+15+CE

4.4. typWS=denckermann

parametr	wartość	info
typWS	denckermann	Denckermann Warszawa
login	kodKlienta	
hasło	hasloKlienta	
hasłoWS	haslo	opcjonalne, 12Jk87BB77
url	url	patrz poniżej

Parametr url:

http://212.180.254.155:30001/WS_DENCKERMANN_WEB5_FASADA/DenckermannWebService.asmx

4.5. typWS=mekonomen

parametr	wartość	info
typWS	mekonomen	
login	kh_kod	
hasło	api_key	
url	url	https://soap-katalog.mekonomen.pl/ws/api

filia	<i>filia</i>	
-------	--------------	--

4.6. typWS=falcon

parametr	wartość	info
typWS	falcon	
login	<i>userLogin</i>	
hasło	<i>userPassword</i>	
url	<i>url</i>	
filia	<i>warehouseId</i>	patrz poniżej, dodatkowy znak + oznacza całkowitą dostępność
token	<i>sekcja</i>	opcjonalne, podać nazwę sekcji, dla której będzie pamiętany token sesji

Przykładowe parametry url:

dostawca	filia	url
ARGE PALIWA Kraków	K1	http://213.17.131.162:7891/ws
AS-PL Gdańsk	CE	http://188.252.14.130:7891/ws
AUTO EURO Lublin	CE	http://sklep.autoeuro.com.pl:7891/ws
Auto Land Olsztyn	SN	https://api.auto-land.pl:9891/ws
Auto-Partner Gdańsk	CE	http://terminal.autopartner.pl:7891/ws
AUTO-ZATOKA Wrocław	CE	http://zatokakaz.no-ip.org:7891/ws
David Vasco Bielsko-Biała	CE	http://91.192.196.162:7891/ws
Edpol Częstochowa	01	http://baza.edpol.pl:7891/ws
Hipol Warszawa	HW	http://webservice.hipol.pl:7891/ws
INTER PARTS Stawiguda	CE	http://dc.interparts.pl:7891/ws
KTD Warszawa	00, 02, WW	http://ktd.eu:7891/ws
Motogama Rząska	01	https://api.motogama.pl:8891/ws
Rodon Leszno	CE	http://main-t.rodon.pl:7891/ws
Tomala Bystrzyca Kłodzka	01	http://46.170.190.234:7891/ws

4.7. typWS=hart

parametr	wartość	info
typWS	hart	
login	<i>username</i>	
hasło	<i>password</i>	
url	<i>url</i>	https://restapi.hartphp.com.pl/v1/
filia	<i>filia</i>	patrz poniżej, dodatkowy znak + oznacza całkowitą dostępność
token	<i>sekcja</i>	opcjonalne, podać nazwę sekcji, dla której będzie pamiętany token sesji

Przykładowe parametry filia:

nazwa filii	filia
Centrala OPOLE - HUB	1

Wrocław	2
Bytom	3
Wieliczka	4
Poznań	8
Katowice	9
Łódź	10
Częstochowa	11
Zielona Góra	12
Kraków	13
Białystok	15
Targówek - Warszawa 1	16
Rzeszów	17
Bielsko-Biała	18
Warszawa 2	19
Radom	20
Psary	22
Bydgoszcz	23
Ostrava (CZ)	24
Brno (CZ)	25
Prešov (SK)	26
Koluszki - HUB	27
Szczecin	28
Gdańsk	29
Lublin	30
Nitra (SK)	31
Warszawa 3	32

4.8. typWS=intercars

parametr	wartość	info
typWS	intercars	
login	<i>login</i>	
hasło	<i>password</i>	
url	<i>url</i>	https://api.webapi.intercars.eu/ic/
filia	<i>location</i>	symbol filii dostawcy, np: KRZ, WRA, ..., dodatkowy znak + oznacza całkowitą dostępność
token	<i>sekcja</i>	opcjonalne, podać nazwę sekcji, dla której będzie pamiętany token sesji

4.9. typWS=interteam

parametr	wartość	info
typWS	interteam	
login	<i>id</i>	
hasło	<i>password</i>	

url	url	https://services.platforma.inter-team.com.pl/soap/WsProxy.svc
filia	branch	numer filii dostawcy, można dodać dodatkowe numery filii dostawcy oddzielając je znakiem +
certifikat	nazwa_certyfikatu	ścieżka_certyfikatu\nazwa_certyfikatu
error	0 lub 1	jeśli =1 to w trybie GET zwracany jest ewentualny komunikat błędu, w trybie POST komunikat nie będzie zwracany

Przykład parametru *certifikat*:

certifikat=LOCAL_MACHINE\TrustedPeople\services.platforma.inter-team.com.pl

Użycie parametr *error=1* może służyć do zdiagnozowania, czy certyfikat został ponownie zainstalowany na serwerze. Dotyczy to tylko żądań GET. W trybie produkcyjnym powinien zostać ustawiony na 0.

4.10. typWS=intervito

parametr	wartość	info
typWS	intervito	
login	auth_login	
hasło	auth_pass	
url	url	https://lsd.intervito.pl/ws/ws_1_6.php

4.11. typWS=motofirma

parametr	wartość	info
typWS	motofirma	
login	Login	
hasło	Hasło	
filia	Filia	numer filii dostawcy
url	url	

Przykładowe adresy url

dostawca	url
AG-PARTS Kozy k/Bielska Białej	http://109.95.151.118:82/servicemf.asmx
ARTER Bytom	https://arter.pl/ws/ServiceMF.asmx
AUTO CENTRUM LIPIEC Głogów	https://auto-czesci-lipiec.pl/wsmf/ServiceMF.asmx
Automix Oleśnica	http://automix.com.pl/webservice/ServiceMF.asmx
AUTO-PRO Kłodzko	http://sklep.auto-plus.pl/ws/ServiceMF.asmx
AutoPrimaPlus Wiązowna	https://autoprimaplus.hopto.org/ws/servicemf.asmx
Auto-Wojdyła Kędzierzyn-Koźle	https://auto-wojdyła.pl/ws/ServiceMF.asmx
Dyl Staszów	https://motoparts24.pl/webservice/ServiceMF.asmx
Dyl Tarnobrzeg	https://dyltarnobrzeg.pl/motoshop/ws/ServiceMF.asmx
GO-WEST Goczałkowice Zdrój	http://sklep.go-west.ig.pl/ws/ServiceMF.asmx
Inter-Auto Pułtusk	https://pultusk.interauto-czesci.pl/wsmf/ServiceMF.asmx
Inter-Auto Maków Mazowiecki	https://makow.interauto-czesci.pl/wsmf/ServiceMF.asmx
KOLCAR Pruszków	http://79.188.194.34/motoshop/ServiceMF/ServiceMF.asmx

Kris-Auto Radzyń Podlaski	https://sklep.krisauto.pl/ws/ServiceMF.asmx
MP Auto Tarnowskie Góry	http://85.11.77.20/ServiceMF/ServiceMF.asmx
Kris-Auto Radzyń Podlaski	https://sklep.wazcar.pl/ws/ServiceMF.asmx
X-Tech Tarnowskie Góry	https://x-techauto.pl/ws/ServiceMF.asmx

4.12. typWS=motoleg

parametr	wartość	info
typWS	motoleg	
login	uznr	
hasło	uzhaslo	
url	url	http://motoleg.konar.pl:7778/

4.13. typWS=motoprofil

parametr	wartość	info
typWS	motoprofil	
login	fiks	
hasło	nr_kontrahenta_motonet	
filia	Filia	opcjonalne, filia=W oznacza Warszawę
url	url	https://195.242.186.3/MotoBiznesWS/WSMotoKlient.asmx

4.14. typWS=motorol

parametr	wartość	info
typWS	motorol	
login	Login	
hasło	Password	
url	url	http://ezamowienia.motorol.pl:82/MotorolService.svc

4.15. typWS=nextis

parametr	wartość	info
typWS	nextis	
login	partnerLogin	
hasło	partnerPassword	
url	url	

Przykładowe parametry url:

Podane w tabeli prefiksy powinny poprzedzać symbol towaru w indeksach, np: dla produktu VERNET TH1290.7 wymagany indeks to: VERTH1290.7

dostawca	prefiks	url
Q4Y Dolz (DOL)	DOL	http://dolzpl.nextis.cz/Web2WSPublic.svc
Q4Y Misfat/Mecafilter	MIS, MEC	http://misfatpl.nextis.cz/web2wspublic.svc
Q4Y Protechnic	PRO	http://q4ypl.nextis.cz/Web2WSPublic.svc
Q4Y Sasic	SAS	http://sasicpl.nextis.cz/Web2WSPublic.svc
Q4Y Vernet	VER	http://vernetpl.nextis.cz/web2wspublic.svc

4.16. typWS=NextisErpAPI

parametr	wartość	info
typWS	NextisErpAPI	
login	<i>login</i>	
hasło	<i>password</i>	
brand	<i>brand</i>	opcjonalne, patrz opis poniżej
url	<i>url</i>	
token	<i>sekcja</i>	opcjonalne, podać nazwę sekcji, dla której będzie pamiętany token sesji

Przekazywanie informacji o brandzie.

WebService dostawcy wymaga przekazania informacji o nazwie brandu każdego ze sprawdzanych towarów.

Wywołanie GET

Informacja o nazwie brandu może zostać przekazana w metodzie GET za pomocą dodatkowego parametru p, np:

```
https://mojadomena.pl/stan/stan.asp?ne=6050200&p=DACO
```

Wywołanie POST

Przykład XML:

```
<z r="ne" s="5">
  <t k="604815" i="165234" p="DACO"/>
  <t k="605025" i="165235" p="DACO"/>
  <t k="PK4760" i="165236" p="DACO"/>
</z>
```

Przykład JSON:

```
{
  "z": {
    "r": "n7",
    "s": "5",
    "t": [
      {
        "k": "604815",
        "i": "165234",
        "p": "DACO"
      },
      {
        "k": "605025",
        "i": "165235",
        "p": "DACO"
      },
      {
        "k": "PK4760",
        "i": "165236",
        "p": "DACO"
      }
    ]
  }
}
```

```
}
}
```

Jeśli żądanie nie zawiera parametru `p`, jako nazwa brandu zostanie przekazany parametr `brand`.

Przykładowe adresy url

dostawca	url
Mastersport Gorlice	https://api.mastersportpl.nextis.cz/
Rzepka Kraków	https://api.rzepkapl.nextis.cz/

4.17. typWS=polcar

parametr	wartość	info
typWS	polcar	
login	<i>Login</i>	
hasło	<i>Password</i>	
url	<i>url</i>	https://dedal.polcar.com/Dystrybutorzy/Products.aspx

4.18. typWS=vanking

parametr	wartość	info
typWS	vanking	
login	<i>Login</i>	
url	<i>url</i>	https://api.vanking.pl:21443/api/production/v1/

5. Migracja ustawień z programu stanMotonet

Program stanMotonet posiadał plik konfiguracyjny o nazwie `stanMotonet.ini`. Zawarte w nim parametry można przepisać (skopiować) do pliku `stanZdalny.dll`, po czym należy uzupełnić i zmodyfikować w miarę potrzeb nazwy sekcji i parametry.

W pliku `stanZdalny.ini` należy dopisać w sekcji `[stan]` parametry:

```
[stan]
...
stanyZerowe=0 //0 dla POST lub 1 dla GET
form=s //niepotrzebne dla GET
json=0
```

Nazwę każdej pozostałej sekcji należy zmienić - jako nazwy sekcji należy przyjąć parametry wywołania - oraz dopisać w niej parametr `typWS`, np:

Przed zmianą:

```
[motoprofil]
fiks=xxxxx
motonet=yyyy
url=https://195.242.186.3/MotoBiznesWS/WSMotoKlient.aspx
```

Po zmianie:

```
[m1]
typWS=motoprofil
login=xxxxx
```

haslo=yyyy
url=https://195.242.186.3/MotoBiznesWS/WSMotoKlient.asmx

Nowe nazwy sekcji i parametry typWS konieczne dla zachowania starych parametrów wywołania:

stara nazwa sekcji	typWS	nowa nazwa sekcji
asco0 do asco9	asco	a0 do a9
falcon5 do falcon9	falcon	f5 do f9
motoprofil	motoprofil	m1
autopartner	autopartner	m4
polcar	polcar	m5
hart	hart	m9
interteam	interteam	n0
elite	mekonomen	n2
motofirma0 do motofirma9	motofirma	t0 do t9
motoleg	motoleg	n3
motorol	motorol	n7

Należy zweryfikować nazwy i wartości parametrów w każdej z sekcji zgodnie z niniejszą instrukcją.