

# **HERMES**

## **„MULTI IN/OUT DOKUMENT”**

### ***MIOD***

Opis systemu

wer. 1.0.0.9

**Rzeszów 2011**

## Spis treści

1. Wstęp .....	5
2. Konfiguracja .....	5
3. Uruchamianie .....	8
4. Przygotowanie skryptów .....	9
4.1. Zmienne globalne .....	9
4.2. Funkcje udostępniane przez HermesNG.....	11
HNG_DialogOtworz .....	11
HNG_DialogZapisz .....	13
HNG_Komunikat .....	15
HNG_WprowadzParametr .....	16
HNG_WybierzParametr .....	17
HNG_Powiadom .....	18
HNG_ZnajdzTowar .....	19
HNG_ZnajdzTowarEx .....	20
HNG_CzyscDokument .....	21
HNG_DopiszDoPrzyjecia .....	22
HNG_DopiszDoZamowieniaODB .....	24
HNG_DopiszDoRozchodu .....	26
HNG_ZnajdzDostawce .....	28
HNG_ZnajdzOdbiorce .....	30
HNG_RozpoczniejDokument .....	32
HNG_DokZewnExists .....	34
HNG_DodajKartoteke .....	35
HNG_Info .....	38
5. Import/Export treści dokumentu .....	39

5.1. Wymiana treści przez pliki .....	39
5.1.1. Format plików importu .....	39
5.1.1.1. Format przyjęcia.....	39
5.1.2. Format plików eksportu.....	40
5.2. Wymiana treści za pomocą funkcji „udostępnionych” .....	40
6. Środowisko PHP .....	40
6.1. Moduły standardowo dostarczane wraz z HermesNG .....	40
6.1.1. Rozszerzenia dynamiczne – DLL .....	40
6.1.2. Klasy/pliki PHP .....	40

## ZMIANY

### wer. 1.0.0.1

- Wprowadzono parametr *MIOD\_PATH*
- MIOD dostępny w „zamówienia do dostawców”

### wer. 1.0.0.2

- Wprowadzono funkcję *HNG\_WprowadzParametr*

### wer. 1.0.0.3

- Wprowadzono funkcję *HNG\_Powiadom*
- MIOD dostępny w opcji *Magazyn -> Kartoteka*
- Dodano parametr *DOK\_SPOSOB*

### wer. 1.0.0.4

- Wprowadzono funkcję *HNG\_WybierzParametr*
- Wprowadzono funkcję *HNG\_ZnajdzDostawce*
- Wprowadzono funkcję *HNG\_ZnajdzOdbiorce*
- Wprowadzono funkcję *HNG\_RozpocznijDokument*
- Wprowadzono funkcję *HNG\_DopiszDoZamowieniaODB*

### wer. 1.0.0.5

- Wprowadzono funkcję *HNG\_DokZewnExistsr*

### wer. 1.0.0.6

- Wprowadzono funkcję *HNG\_DodajKartoteke*
- Zmodyfikowano funkcję *HNG\_ZnajdzTowar*
- Zmodyfikowano funkcję *HNG\_ZnajdzTowarEx*

### wer. 1.0.0.7

- Zmodyfikowano funkcję *HNG\_DodajKartoteke*
- Zmodyfikowano funkcję *HNG\_CzyszcDokument*

### wer. 1.0.0.8

- Wprowadzono funkcję *HNG\_Info*

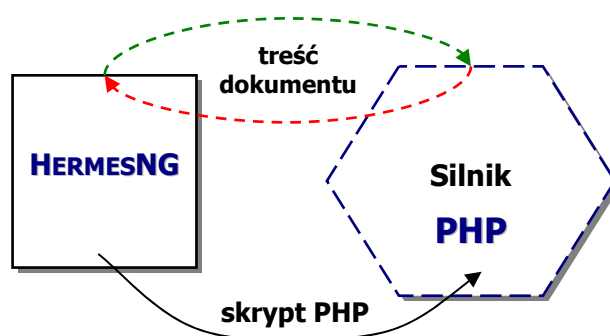
### wer. 1.0.0.9

- Dodanie nowego parametru globalnego 'DS.\_RECNO'
- Rozszerzenie modułu wgrywania skryptów o zakres danych na jakich dany skrypt ma działać.

## 1. WSTĘP

Celem modułu „Multi In/Out Dokument” (MIOD), jest umożliwienie importu/eksportu treści dokumentów do i z HermesNG. MIOD umożliwia tworzenie odpowiednich interfejsów, bez konieczności modyfikacji samego Hermes’a. Moduł został oparty na popularnym języku skryptowym PHP.

W celu importu/eksportu treści dokumentu, HermesNG uruchamia wcześniej przygotowany, zewnętrzny skrypt PHP, którego zadaniem jest konwersja pliku w dowolnym formacie, na format „zrozumiały” przez HermesNG.



schemat ideowy wymiany MIOD

Moduł MIOD umożliwia import/eksport treści dla:

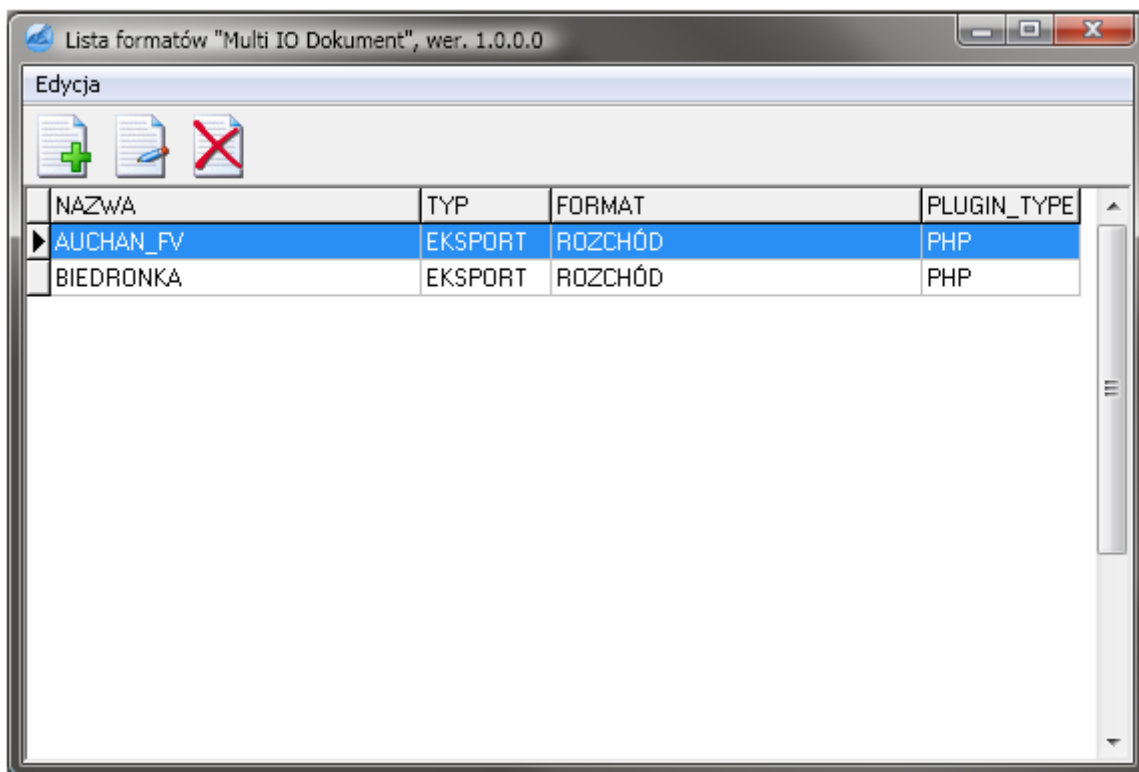
- dokumentów:
  - przyjąć,
  - rozchodów,
  - zamówień do dostawców,
  - zamówień od odbiorców\*,
  - dokumentów rozliczeń (wpłaty/wypłaty)
- listy magazynowej

## 2. KONFIGURACJA

Przed wykorzystaniem przygotowanych skryptów MIOD, konieczna jest ich konfiguracja. Opcje konfiguracyjne modułu MIOD znajdują się w opcji *Inne-> Konfiguracja-> Specjalne-> Wymiana MIOD*.

---

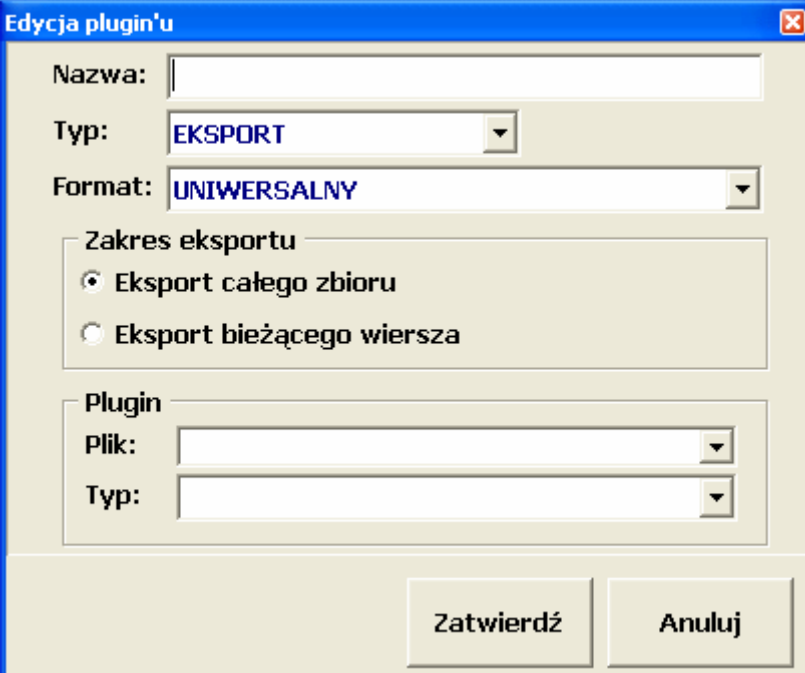
\* Funkcja niedostępna w wersji 1.0.0.3 modułu



**lista zdefiniowanych modułów MIOD**

Konfiguracja ma na celu:

- wprowadzenie nazwy, umożliwiającej użytkownikowi łatwe rozróżnienie dostępnych skryptów MIOD,
- określenie typu – import lub eksport,
- określenie zakresu eksportowanych danych z programu wywołujących dany skrypt,
- wskazanie przeznaczenia skryptu (jaki typ dokumentów),  
wczytanie do systemu skryptu PHP z dysku lokalnego.



Edycja plugin'u

Nazwa:

Typ:

Format:

Zakres eksportu

☒ Eksport całego zbioru

☐ Eksport bieżącego wiersza

Plugin

Plik:

Typ:

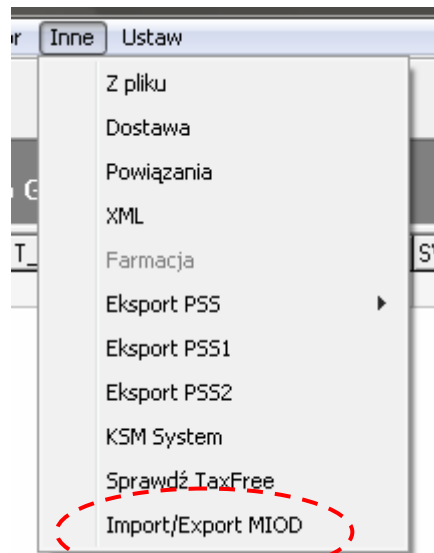
Zatwierdź Anuluj

### konfiguracja modułu MIOD

Po wprowadzeniu konfiguracji, skrypt modułu MIOD wczytywany jest do centralnej bazy danych i dostępny do użycia na dowolnym stanowisku z HermesNG.

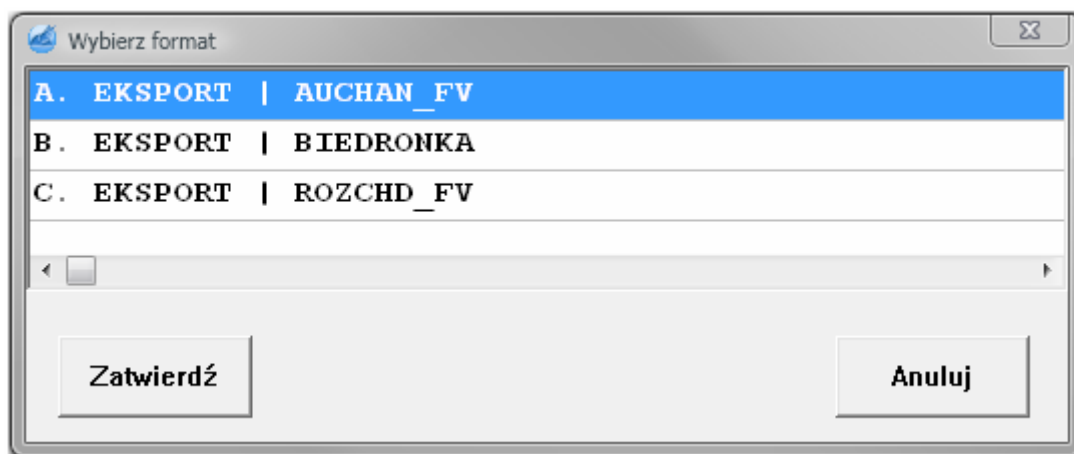
### 3. URUCHAMIANIE

Uruchomienie skonfigurowanych skryptów MIOD, wykonuje się z poziomu edycji treści dokumentu, w opcji *Inne-> Import/Export MIOD\**.



**wywołanie modułu MIOD**

Po uruchomieniu opcji, zostanie wyświetlona dostępna, skonfigurowana lista modułów MIOD, które mogą być użyte dla aktualnego dokumentu.



**lista dostępnych dla dokumentu skryptów MIOD**

Na liście dostępna jest informacja o typie akcji, jaka zostanie podjęta (import lub eksport) oraz nazwa modułu MIOD, umożliwiającą rozróżnienie dostępnych skryptów.

Po wybraniu skryptu, zostanie on pobrany z centralnej bazy danych na lokalne stanowisko i uruchomiony. Prawidłowe zakończenie pracy modułu MIOD, zostanie potwierdzone odpowiednim komunikatem.

---

\* opcja wspólna dla każdego typu dokumentu, chyba że wskazano inaczej



## 4. PRZYGOTOWANIE SKRYPTÓW

Skrypty MIOD tworzone są w języku skryptowym PHP. Nie jest konieczne instalowanie silnika PHP na końcówce klienckiej lub oddzielnego serwera. HermesNG zawiera wszystkie niezbędne moduły, do uruchomienia środowiska PHP. Standardowo jest to wersja 5.2.

Tworząc skrypt, programista ma dostęp do wszystkich standardowych funkcji PHP oraz dodatkowo do opcji oferowanych przez moduły: *php\_dbase.dll*, *php\_interbase.dll*, *php\_zip.dll*. Ponad to, istnieje możliwość rozszerzenia lokalnego środowiska PHP, o dodatkowe moduły współpracujące z PHP (patrz 6. *Środowisko PHP*).

Skrypt PHP musi:

- ✓ rozpocząć się znacznikiem `<?php`
- ✓ kończyć znacznikiem `?>`
- ✓ używać strony kodowej Windows

### 4.1. Zmienne globalne

HermesNG uruchamiając skrypt PHP, rejestruje w środowisku zmienne globalne, umożliwiając modułowi MIOD, odczytanie podstawowych danych np. o dokumencie, kontrahencie oraz zapewniając przekazywanie informacji między skryptem a programem.

**Rejestrowane zmienne globalne:**

DOTYCZY	NAZWA ZMIENNEJ	OPIS
moduł MIOD	<b>MIOD_FILEFROMHERM</b>	nazwa pliku z treścią dokumentu, przekazywanego z programu do skryptu MIOD (PHP)
	<b>MIOD_FILEFORHERM</b>	nazwa pliku z treścią dokumentu, przekazywanego ze skryptu MIOD do programu
	<b>MIOD_PATH</b>	lokalny folder, w którym znajdują się pliki środowiska PHP (dll, php). Wykorzystując biblioteki dodatkowe, np. <i>PHPMailer</i> lub inne, dołączając plik php za pomocą klauzul <i>require/include</i> należy użyć wskazane folderu.
program	<b>PRG_MAGAZYN</b>	oznaczenie aktywnego magazynu
baza danych	<b>DB_LOKAL</b>	lokalizacja podłączonej bazy danych w programie; nazwę użytkownika oraz hasło dostępu do serwera, na którym znajduje się baza danych, należy uzyskać od administratora systemu
operator	<b>OPR_ID</b>	nr (ID) zalogowanego operatora HermesNG
	<b>OPR_OPERATOR</b>	nazwa zalogowanego operatora HermesNG

Bieżący dokument	DOK_DOKUMENT	symbol edytowanego dokumentu; <b>UWAGA!</b> Jeżeli skrypt uruchamiany jest przed zapisaniem dokumentu w HermesNG, <u>symbol może ulec zmianie</u> ; Jeżeli symbol dokumentu, na potrzeby pracy skryptu jest znaczący, skrypt należy uruchomić <u>PO zapisaniu</u> dokumentu w HermesNG.
	DOK_DOKZEWN	symbol dokumentu zewnętrznego, powiązanego z edytowanym dokumentem
	DOK_DATA	data dokumentu
	DOK_TERMIN	termin płatności dokumentu
	DOK_SPOSOB	sposób płatności dokumentu
	DOK_KON_ID	numer (ID) kontrahenta powiązanego z dokumentem
	DOK_UWAGI	uwagi dokumentu
bieżący kontrahent	KON_ID	numer (ID) kontrahenta; unikalny identyfikator, jednoznacznie wskazujący kontrahenta w systemie
	KON_PLATNIK	nr (ID) płatnika kontrahenta
	KON_NAZWA	nazwa kontrahenta (50 znaków)
	KON_CDNAZWA	druga część nazwy kontrahenta (50 znaków)
	KON_ADRES	adres kontrahenta; zgodny z wpisami w bazie HermesNG
	KON_NIP	NIP kontrahenta
źródło danych	DS_RECNO	określa numer rekordu, na którym ustawiony jest kursor

## 4.2. Funkcje udostępniane przez HermesNG

HermesNG uruchamiając i konfiguruje silnik PHP, dodaje do środowiska funkcje umożliwiające sterowanie pracą programu lub systemu, wprost ze skryptu PHP.

### HNG\_DialogOtworz

- otwarcie standardowego okna windows, umożliwiającego użytkownikowi wybranie pliku do otwarcia

#### Opis

```
string HNG_DialogOtworz(string $sCaption, string $sDefName,  
                        string $sFilter)
```

Funkcja zwraca nazwę oraz ścieżkę wybranego przez użytkownika pliku.

#### Parametry

##### *sCaption*

Tekst w nagłówku okna jaki zostanie wyświetlony.

##### *sDefName*

Domyślna, proponowana nazwa pliku.

##### *sFilter*

Dostępne rozszerzenia (filtry) plików.

Każdy filtr (typ) składa się z dwóch części:

- opisowej – widocznej dla użytkownika na liście dostępnych plików, np. „Pliki XML (.xml)”
- definicji filtru – sposób w jaki system będzie filtrować widoczne pliki, np. „\*.xml”

Części filtru oddzielane są znakiem | , np. Pliki XML (\*.xml) | \*.xml

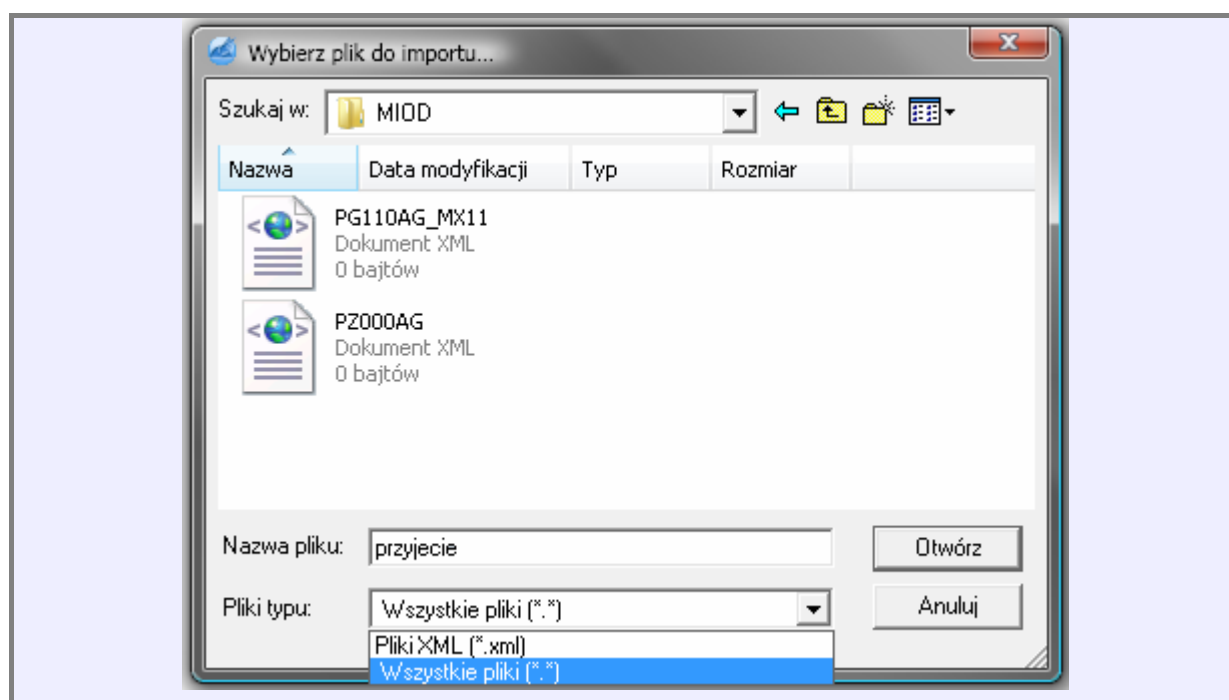
W przypadku kilku filtrów, każdy z nich również należy rozdzielić znakiem | .

np. Pliki XML (\*.xml) | \*.xml | Wszystkie pliki (\*.\*) | \*.\*

#### Przykład

```
<?php  
  
$nazwa_pliku = HNG_DialogOtworz("Wybierz plik do importu...",  
                                "przyjecie.xml",  
                                "Pliki XML (*.xml) | *.xml | Wszystkie pliki (*.*) | *.*");  
  
?>
```

Wynik działania:



## HNG\_DialogZapisz

- otwarcie standardowego okna windows, umożliwiającego użytkownikowi wybranie pliku do zapisu

### Opis

```
string HNG_DialogZapisz (  string $sCaption, string $sDefName,  
                           string $sFilter)
```

Funkcja zwraca nazwę oraz ścieżkę wybranego przez użytkownika pliku do zapisu.

Funkcja sprawdza czy wskazany plik istnieje i ostrzega operatora przez próbą nadpisania pliku.

### Parametry

#### *sCaption*

Tekst w nagłówku okna jaki zostanie wyświetlony.

#### *sDefName*

Domyślna, proponowana nazwa pliku.

#### *sFilter*

Dostępne rozszerzenia (filtry) plików.

Każdy filtr (typ) składa się z dwóch części:

- opisowej – widocznej dla użytkownika na liście dostępnych plików, np. „Pliki XML (.xml)”
- definicji filtru – sposób w jaki system będzie filtrować widoczne pliki, np. „\*.xml”

Części filtru oddzielane są znakiem | , np. Pliki XML (\*.xml) | \*.xml

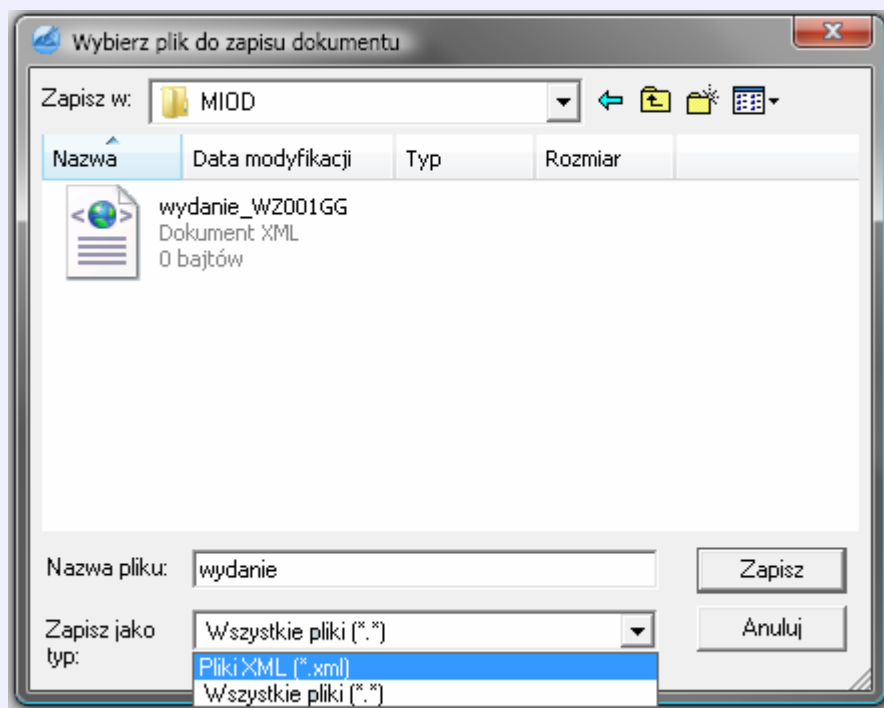
W przypadku kilku filtrów, każdy z nich również należy rozdzielić znakiem | .

np. Pliki XML (\*.xml) | \*.xml | Wszystkie pliki (\*.\*) | \*.\*

**Przykład**

```
<?php  
  
$nazwa_pliku = HNG_DialogZapisz ("Wybierz plik do zapisu dokumentu",  
"wydanie.xml",  
"Pliki XML (*.xml) | *.xml | Wszystkie pliki (*.*) | *.*");  
  
?>
```

Wynik działania:



## HNG\_Komunikat

– wyświetlenie komunikatu z wybranym tekstem.

### Opis

```
void HNG_Komunikat (string $sMsg, string $sCaption)
```

Funkcja wyświetla okno z komunikatem.

### Parametry

#### *sMsg*

Tekst wyświetlanego komunikatu.

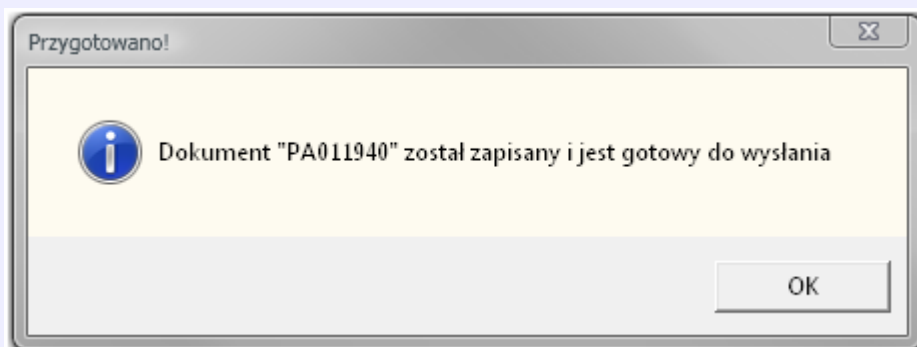
#### *sCaption*

Tytuł okna z komunikatem.

### Przykład

```
<?php  
HNG_Komunikat(  
"Dokument \"\". $DOK_DOKUMENT. \"\" został zapisany i jest gotowy do  
wysłania",  
"Przygotowano!");  
?>
```

Wynik działania:



## HNG\_WprowadzParametr

– wyświetlenie komunikatu z prośbą o wpisanie przez operatora wartości.

### Opis

```
string HNG_WprowadzParametr (string $sCaption, string $sPrompt,  
[string $sTekst])
```

Funkcja zwraca wartość wpisaną przez użytkownika

Funkcja wyświetla okno z komunikatem.

### Parametry

#### *sCaption*

Tytuł okna.

#### *sPrompt*

Tekst komunikatu.

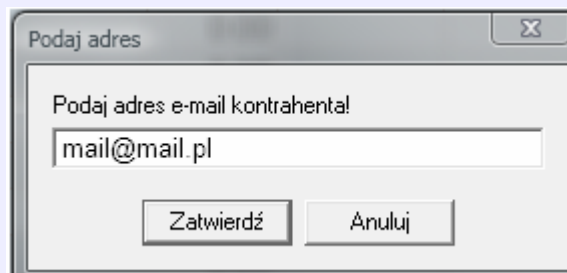
#### *sTekst*

Tekst wpisany do pola edycyjnego.

### Przykład

```
<?php  
$eml=HNG_WprowadzParametr('Podaj adres',  
                           'Podaj adres e-mail kontrahenta!',  
                           'mail@mail.pl'  
);  
?>
```

Wynik działania:





## HNG\_WybierzParametr

– wyświetlenie okna, w którym użytkownik będzie miał możliwość wskazania odpowiedniego parametru.

### Opis

```
string HNG_WybierzParametr ( string $sCaption, string $sPrompt,  
                             string $sInputValue)
```

Funkcja zwraca wartość wpisaną przez użytkownika

Funkcja wyświetla okno z komunikatem.

### Parametry

#### *sCaption*

Tytuł okna.

#### *sPrompt*

Tekst komunikatu.

#### *sInputValue*

Wartości, które mają pojawić się na liście rozwijalnej.

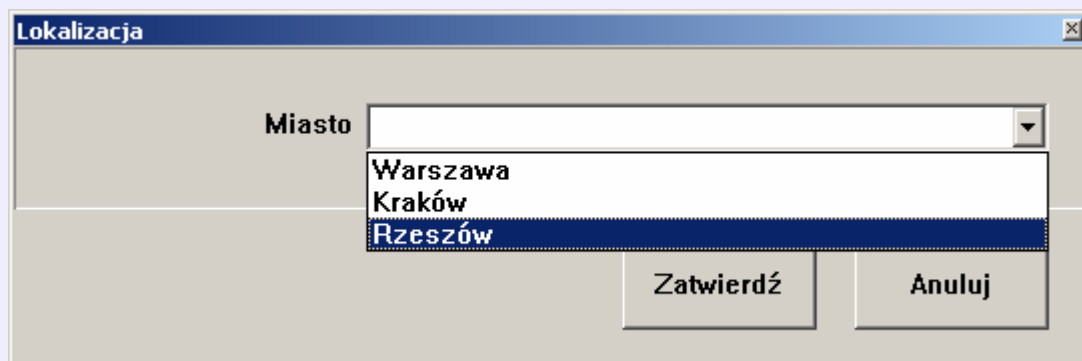
#### **Uwaga!**

Przekazywane wartości **muszą być** rozdzielone znakiem `;`

### Przykład

```
<?php  
$miasto = HNG_WybierzParametr(  
    'Lokalizacja',  
    'Miasto',  
    'Warszawa;Kraków;Rzeszów');  
?>
```

Wynik działania:



## HNG\_Powiadom

– dodanie powiadomienia, do listy informacyjnej o stanie wykonania skryptu MIOD, wyświetlanej w HermesNG.

### Opis

```
void HNG_Powiadom (string $sMsg)
```

Dodaje powiadomienie do listy ze statusem wykonania modułu MIOD.

### Parametry

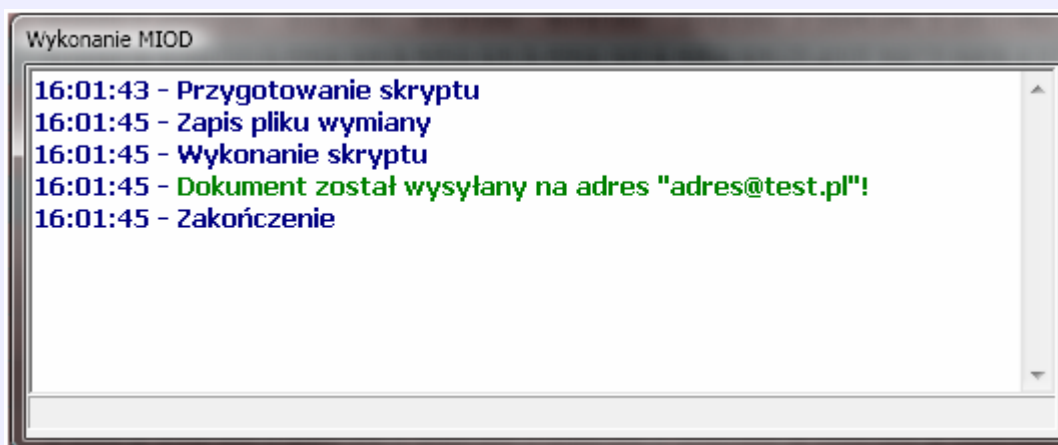
*sMsg*

Tekst powiadomienia.

### Przykład

```
<?php
    HNG_Powiadom("Dokument został wysyłany na adres \"adres@test.pl\"!");
?>
```

Wynik działania:



## HNG\_ZnajdzTowar

– odnalezienie pozycji magazynowej, na podstawie danych podstawowych kartoteki.

### Opis

```
integer HNG_ZnajdzTowar ( string $sArtykul, string $sIndeks,  
                        string $sKreski, boolean $bSilent)
```

Funkcja na podstawie indeksu oraz kodu kreskowego, odnajduje w bieżącym magazynie kartotekę magazynową. W przypadku braku kartoteki lub występowania kilku kartotek z tym samym indeksem lub kodem kreskowym, użytkownik zostanie poproszony o wybranie kartoteki z listy.

Funkcja zwraca unikalny identyfikator (POZ) kartoteki magazynowej.

### Parametry

#### *sArtykul*

Nazwa artykułu. Niewykorzystywana przy szukaniu, wyświetlana użytkownikowi, w celu identyfikacji kartoteki.

#### *sIndeks*

Indeks artykułu. Wykorzystywany podczas wyszukiwania kartoteki. Długość max 12 znaków.

#### *sKreski*

Kod kreskowy (EAN13) artykułu. Wykorzystywany podczas wyszukiwania kartoteki, jako pole uzupełniające w przypadku gdy indeks nie został odnaleziony. Długość max 13 znaków.

#### *bSilent*

Jeżeli jest ustawione na 'true', wtedy operatorowi nie ukazuje się okno z zawartością magazynu w przypadku, gdy system nie odnalazł wskazanego artykułu.

### Przykład

```
<?php  
$POZ=HNG_ZnajdzTowar("TOWAR TESTOWY1","IND0123","9012349012348");  
?>
```

## HNG\_ZnajdzTowarEx

– odnalezienie pozycji magazynowej, na podstawie indeksu zewnętrznego (dostawcy/odbiorcy).

### Opis

```
integer HNG_ZnajdzTowarEx( string $sArtykul, string $sIndZewn,  
                           integer $iKon, string $sKonTyp,  
                           boolean $bSilent)
```

Funkcja na podstawie indeksu zewnętrznego, powiązanego z dostawcą lub odbiorcą odnajduje w bieżącym magazynie kartotekę magazynową. W przypadku braku kartoteki lub występowania kilku kartotek z tym samym indeksem, użytkownik zostanie poproszony o wybranie kartoteki z listy.

Funkcja zwraca unikalny identyfikator (POZ) kartoteki magazynowej.

### Parametry

#### *sArtykul*

Nazwa artykułu. Niewykorzystywana przy szukaniu, wyświetlana użytkownikowi, w celu identyfikacji kartoteki.

#### *sIndZewn*

Indeks zewnętrzny artykułu. Wykorzystywany podczas wyszukiwania kartoteki. Długość max 20 znaków.

#### *iKon*

Identyfikator (KON) kontrahenta, którego indeksy powiązane zostaną przeszukane.

#### *sKonTyp*

Typ kontrahenta:

**O** – odbiorca

**D** – dostawca

#### *bSilent*

Jeżeli jest ustawione na 'true', wtedy operatorowi nie ukazuje się okno z zawartością magazynu w przypadku, gdy system nie odnalazł wskazanego artykułu.

### Przykład

```
<?php  
$POZ=HNG_ZnajdzTowarEx("TOWAR TESTOWY1","XYZ/123/WK0",1,"O");  
?>
```

## HNG\_CzyscDokument

– usuwa treść bieżącego dokumentu.

### Opis

```
void HNG_CzyscDokument (boolean $bSilent)
```

Funkcja usuwa treść bieżącego dokumentu. Przed usunięciem pozycji, użytkownik zostanie poproszony o potwierdzenie operacji.

### Parametry

#### *bSilent*

Jeżeli jest ustawione na 'true', wtedy operator nie musi potwierdzić operacji wyczyszczenia treści dokumentu, w przeciwnym wypadku pojawi się okno, w którym trzeba będzie dokonać wyboru dalszego postępowania algorytmu.

### Przykład

```
<?php  
  
HNG_CzyscDokument (true) ;  
  
?>
```

## HNG\_DopiszDoPrzyjecia

– dopisanie pozycji do treści dokumentu przyjęcia.

### Opis

```
integer HNG_DopiszDoPrzyjecia (integer $iPoz,  
                                string $sArtykul, string $sIndeks,  
                                float $fIlosc, float $fCenaZakup,  
                                float $fOdchylenie, string $sVat,  
                                string $sSeria, string $sDataWazn,  
                                string $sKreski, float $fRecykling)
```

Funkcja dodaje do treści dokumentu przyjęcia nową pozycję.

Funkcja zwraca identyfikator kartoteki (POZ), w przypadku błędu zwracana jest wartość zero (false).

### Parametry

#### *iPoz*

Unikalny identyfikator kartoteki. Może być przekazany jako 0 (zero), wtedy kartoteka zostanie odnaleziona na podstawie indeksu i kodu kreskowego (podobnie jak w przypadku *HNG\_ZnajdzTowar*).

#### *sArtykul*

Nazwa artykułu.

#### *sIndeks*

Indeks artykułu. Wykorzystywany podczas wyszukiwania kartoteki, w przypadku, gdy *\$iPoz=0*.

#### *fIlosc*

Przyjmowana ilość.

#### *fCenaZakup*

Cena zakupu przyjmowanej pozycji.

#### *fOdchylenie*

Odchylenia (różnica między ceną zakupu na dok. oryginalnym a ceną zakupu realizowaną) dla pozycji.

#### *sVat*

Stawka VAT z jaką przyjmowana jest pozycja.

#### *sSeria*

Nr serii/nr unikatowy/atest przyjmowanej partii

#### *sDataWazn*

Data ważności przyjmowanej partii.

***sKreski***

Kod kreskowy przyjmowanej partii. Wykorzystywany podczas wyszukiwania kartoteki, w przypadku, gdy *\$iPoz=0*.

***fRecykling***

Wartość kosztów recyklingu partii towaru.

**Przykład 1**

```
<?php
$POZ= HNG_DopiszDoPrzyjecia(1891,
                                "TOWAR TESTOWY1",
                                "IND0123",
                                5.00,
                                12.01,
                                0.00,
                                "22.00",
                                "",
                                "20090101",
                                "9012349012348",
                                0.00);

?>
```

**Przykład 2**

```
<?php
HNG_CzyscDokument();
$POZ=HNG_ZnajdzTowar("TOWAR TESTOWY1","IND0123","9012349012348");

if ($POZ!=0)
{
    $POZ=HNG_DopiszDoPrzyjecia( $POZ,
                                "TOWAR TESTOWY1",
                                "IND0123",
                                5.00,
                                12.01,
                                0.00,
                                "22.00",
                                "",
                                "20090101",
                                "9012349012348",
                                0.00);

    if ($POZ!=0)
    {
        HNG_Komunikat("Dopisano pozycję do dokumentu!", "Dopisano!");
    }
}

?>
```

## HNG\_DopiszDoZamowieniaODB

– dopisanie pozycji do treści dokumentu zamówienia od odbiorcy.

### Opis

```
integer HNG_DopiszDoZamowieniaODB (  
    integer $iPoz,  
    string $sArtykul,  
    string $sIndeks,  
    string $sKreski  
    float $fIlosc,  
    float $fCenaNetto,  
    string $sIndeksZewn)
```

Funkcja dodaje do treści dokumentu zamówienia nową pozycję.

Funkcja zwraca identyfikator kartoteki (POZ), w przypadku błędu zwracana jest wartość zero (false).

### Parametry

#### *iPoz*

Unikalny identyfikator kartoteki. Może być przekazany jako 0 (zero), wtedy kartoteka zostanie odnaleziona na podstawie indeksu i kodu kreskowego (podobnie jak w przypadku *HNG\_ZnajdzTowar*).

#### *sArtykul*

Nazwa artykułu.

#### *sIndeks*

Indeks artykułu. Wykorzystywany podczas wyszukiwania kartoteki, w przypadku, gdy *\$iPoz=0*.

#### *sKreski*

Kod kreskowy artykułu.

#### *fIlosc*

Ilość zamawianej pozycji.

#### *fCenaNetto*

Cena netto zamawianej pozycji (chwilowo nie wykorzystywana)

#### *sIndeksZewn*

Indeks zewnętrzny artykułu.



**Przykład 1**

```
<?php
$POZ = HNG_DopiszDoZamowieniaODB(
    1891,
    "TOWAR TESTOWY1",
    "IND0123",
    "5901597860433",
    12,
    2.55,
    "");
?>
```

## HNG\_DopiszDoRozchodu

– dopisanie pozycji do treści dokumentu zamówienia od odbiorcy.

### Opis

```
integer HNG_DopiszDoRozchodu (  
    integer $iPoz,  
    string $sArtykul,  
    string $sIndeks,  
    string $sKreski  
    float $fIlosc,  
    float $fCenaNetto,  
    string $sIndeksZewn)
```

Funkcja dodaje do treści dokumentu rozchodu nową pozycję.

Funkcja zwraca identyfikator kartoteki (POZ), w przypadku błędu zwracana jest wartość zero (false).

### Parametry

#### *iPoz*

Unikalny identyfikator kartoteki. Może być przekazany jako 0 (zero), wtedy kartoteka zostanie odnaleziona na podstawie indeksu i kodu kreskowego (podobnie jak w przypadku *HNG\_ZnajdzTowar*).

#### *sArtykul*

Nazwa artykułu.

#### *sIndeks*

Indeks artykułu. Wykorzystywany podczas wyszukiwania kartoteki, w przypadku, gdy *\$iPoz=0*.

#### *sKreski*

Kod kreskowy artykułu.

#### *fIlosc*

Ilość pozycji.

#### *fCenaNetto*

Cena netto zamawianej pozycji.

#### *sIndeksZewn*

Indeks zewnętrzny artykułu.

**Przykład 1**

```
<?php
$POZ = HNG_DopiszDoRozchodu(
    1891,
    "TOWAR TESTOWY1",
    "IND0123",
    "5901597860433",
    12,
    2.55,
    "");
?>
```

## HNG\_ZnajdzDostawce

– odnalezienie identyfikatora dostawcy na podstawie danych podstawowych.

### Opis

```
integer HNG_ZnajdzDostawce (  
    integer $iKon,  
    string $sNazwa,  
    string $sKod,  
    string $sMiasto  
    string $sUlica,  
    string $sNIP,  
    string $sCecha)
```

Funkcja na podstawie przekazanych informacji, odnajduje kartotekę dostawcy. W przypadku braku kartoteki lub występowania kilku kartotek spełniających zadane kryterium, użytkownik zostanie poproszony o wybranie dostawcy z listy.

Funkcja zwraca identyfikator dostawcy, w przypadku błędu zwracana jest wartość zero (false).

### Parametry

#### *iKon*

Unikalny identyfikator dostawcy. Może być przekazany jako 0 (zero), wtedy dostawca zostanie odnaleziona na podstawie pozostałych informacji.

#### *sNazwa*

Nazwa dostawcy.

#### *sKod*

Kod pocztowy dostawcy

#### *sMiasto*

Miasto dostawcy.

#### *sUlica*

Ulica dostawcy.

#### *sNIP*

NIP dostawcy.

#### *sCecha*

Cecha dostawcy.

**Przykład 1**

```
<?php
$iKon = HNG_ZnajdzDostawce(
    0,
    "TEST",
    "35-123",
    "Rzeszów",
    "SPORTOWA",
    "800-110-10-20",
    "");
?>
```

## HNG\_ZnajdzOdbiorce

– odnalezienie identyfikatora odbiorcy na podstawie danych podstawowych.

### Opis

```
integer HNG_ZnajdzOdbiorce (  
    integer $iKon,  
    string $sNazwa,  
    string $sKod,  
    string $sMiasto  
    string $sUlica,  
    string $sNIP,  
    string $sCecha)
```

Funkcja na podstawie przekazanych informacji, odnajduje kartotekę odbiorcy. W przypadku braku kartoteki lub występowania kilku kartotek spełniających zadane kryterium, użytkownik zostanie poproszony o wybranie odbiorcy z listy.

Funkcja zwraca identyfikator odbiorcy, w przypadku błędu zwracana jest wartość zero (false).

### Parametry

#### *iKon*

Unikalny identyfikator odbiorcy. Może być przekazany jako 0 (zero), wtedy odbiorca zostanie odnaleziona na podstawie pozostałych informacji.

#### *sNazwa*

Nazwa odbiorcy.

#### *sKod*

Kod pocztowy odbiorcy.

#### *sMiasto*

Miasto odbiorcy.

#### *sUlica*

Ulica odbiorcy.

#### *sNIP*

NIP odbiorcy.

#### *sCecha*

Cecha odbiorcy.

**Przykład 1**

```
<?php
$iKon = HNG_ZnajdzOdbiorce(
    0,
    "TEST",
    "35-123",
    "Rzeszów",
    "SPORTOWA",
    "800-110-10-20",
    "");
?>
```

## HNG\_Rozpoczni jDokument

– przekazuje podstawowe informacji wypełniane podczas tworzenia nagłówka dokumentu. (Może być używana zarówno po stronie rozchodu jak i przychodu).

### Opis

```
integer HNG_Rozpoczni jDokument (  
    string $sDokument,  
    integer $iKonID,  
    string $sData,  
    string $sTermin  
    string $sDokZewn,  
    string $sUwagi)
```

Funkcja przekazuje do programu podstawowe informacje, które zostaną umieszczone w nagłówku dokumentu.

Funkcja zwraca symbol dokumentu, w przypadku błędu zwracana jest wartość pusta.

### Parametry

#### *sDokument*

Symbol dokumentu jaki ma zostać wprowadzony, jeżeli parametr ten będzie pusty to symbol zostanie wygenerowany automatycznie.

#### *sKonID*

Identyfikator kontrahenta, na którego będzie wystawiany dokument

#### *sData*

Data dokumentu

#### *sTermin*

Termin dokumentu

#### *sDokZewn*

Symbol dokumentu zewnętrznego

#### *sUwagi*

Uwagi



**Przykład 1**

```
<?php
$sDokSymbol = HNG_RozpoczniejDokument(
    "",
    123,
    "2009.09.14",
    "2009.09.14",
    "FV00/11/22",
    "brak uwag");
?>
```

## HNG\_DokZewnExists

– Sprawdza czy w systemie istnieje dokument, który posiada powiązanie z dokumentem zewnętrznym.

### Opis

```
integer HNG_DokZewnExists(  
    string $sDokZewn,  
    string $sTyp)
```

Funkcja zwraca '1', jeżeli istnieje dokument powiązany z danym dokumentem zewnętrznym lub '0', jeżeli takiego dokumentu nie ma.

### Parametry

#### *sDokZewn*

Symbol dokumentu zewnętrznego.

#### *sTyp*

Typ:  
'O' = odbiorca  
'D' = dostawca

### Przykład 1

```
<?php  
$iExist = HNG_DokZewnExists(  
    "Fv00/11/22",  
    "O");  
?>
```

## HNG\_DodajKartoteke

– umożliwia dodanie do systemu nowej kartoteki, przekazując do odpowiedniego okna w programie podstawowe informacje o nowej kartotece.

### Opis

```
integer HNG_DodajKartoteke (  
    string $sArtykuł,  
    string $sIndeks,  
    string $sKod_Kreskowy,  
    char $cTyp,  
    string $sPKWiU,  
    string $sVat  
    string $sJM  
    float $fOPK  
    string $sJM2  
    float $fCenaNettoZakup  
    float $fCenaNetto  
    float $fCenaBrutto  
    float $fWaga  
    float $fObjetosc  
    string $sLogo  
    string $sProducent  
    boolean $bPokazListe)
```

Funkcja przekazuje do programu podstawowe informacje, które zostaną umieszczone w kartotece artykułu.

Funkcja zwraca identyfikator pozycji lub '0', jeżeli pozycja nie została dodana.

### Parametry

#### *sArtykuł*

Nazwa nowego artykułu.

#### *sIndeks*

Indeks nowego artykułu.

#### *sKod\_Kreskowy*

Kod kreskowy nowego artykułu.

#### *cTyp*

Typ artykułu:

- T – towar,
- U – usługa,
- O – opakowanie,

#### *sPKWiU*

Symbol PKWiU.

**sVat**

Wartość podatku (numeryczna np.: 23, 8).

**sJM**

Jednostka miar.

**fOPK**

Ilość pozycji w opakowaniu.

**sJM2**

Druga jednostka miar.

**fCenaNettoZakup**

Cena netto zakupu nowej pozycji.

**fCenaNetto**

Cena netto sprzedaży nowej pozycji.

**fCenaBrutto**

Cena brutto sprzedaży nowej pozycji.

**fWaga**

Waga nowej pozycji.

**fObjetosc**

Objętość nowej pozycji.

**sLogo**

Logo.

**sProducent**

Producent.

**bPokazListe**

Wprowadzenie wartości **true** powodować będzie pojawienie się okna listy magazynowej, w której będzie możliwe dodanie nowej kartoteki lub wskazanie już istniejącej kartoteki, na którą ma być dokonane przyjęcie.  
Wprowadzenie wartości **false** powodować będzie automatyczne pojawienie się okna do wprowadzenia nowej pozycji magazynowej.

**Uwaga!!!**

W związku z istnieniem trzech parametrów odnośnie ceny artykułu (**'fCenaNettoZakup'**, **'fCenaNetto'**, **'fCenaBrutto'**) wprowadzony został następujący mechanizm ustalania ceny artykułu na kartotece.

- a) w przypadku, gdy **'fCenaNetto'** lub **'fCenaBrutto'** równa jest **'-1'**, to ceny sprzedaży zostaną ustawione na **'0'**;
- b) jeżeli **'fCenaNetto'** równa jest **'0'**, wtedy wyliczona będzie z **'fCenaBrutto'**;
- c) jeżeli **'fCenaBrutto'** równa jest **'0'**, wtedy wyliczona będzie z **'fCenaNetto'**;
- d) jeżeli **'fCenaNetto'** i **'fCenaBrutto'** będą równe **'0'** wtedy zostaną wyliczone z **'fCenaNettoZakup'**;

**Przykład 1**

```
<?php
$artId = HNG_DodajKartoteke(
    "Artykuł testowy",
    "IND_TEST",
    "5900306002089",
    "T",
    "",
    "23",
    "OPK",
    "10",
    "BT",
    15.55,
    20,
    24.6,
    0.5,
    0.75,
    "LOGO",
    "PRODUCENT",
    true);
?>
```

## HNG\_Info

– umożliwia wyświetlenie tabeli zawierającej dane z pliku DBF oraz określenia, które z rekordów w niej zawartych mają być w dalszej części skryptu przetwarzane.

### Opis

```
boolean HNG_Info(  
    string $sCaption,  
    string $sSubCaption,  
    string $sDbfFilePath)
```

Funkcja zwraca 'true', jeżeli użytkownik zatwierdził informacje zamieszczone w nowym oknie lub 'false' jeżeli nie zostały one przez niego zatwierdzone.

### Parametry

#### *sCaption*

Nagłówek okna informacji.

#### *sSubCaption*

Dodatkowe informacje wyświetlane w oknie.

#### *sDbfFilePath*

Ścieżka do pliku DBF, który będzie prezentowany w danym oknie.

### Uwaga!!!

Obecna wersja wymaga, aby w przekazywanym pliku DBF istniała kolumna 'WYBOR' wypełniona wartością 'TAK'. Jest ona używana do określania, które rekordy mają być przetwarzane w dalszej części skryptu.

## 5. IMPORT/EXPORT TREŚCI DOKUMENTU

Treść dokumentu może być importowana lub eksportowana, za pomocą plików pośredniczących DBF lub poprzez funkcje udostępniane przez HermesNG (np. *HNG\_DopiszDoPrzyjecia*).

### 5.1. Wymiana treści przez pliki

Stosując wymianę treści dokumentów za pomocą plików, w trakcie importu, HermesNG odczytuje lokalny plik DBF przygotowany przez skrypt MIOD; w trakcie eksportu HermesNG przygotowuje plik DBF, przekazując go do skryptu MIOD.

Lokalizacja plików DBF, z treścią dokumentu, przekazywana jest przez parametry globalne MIOD (patrz 4.1. *Zmienne globalne*).

- **MIOD\_FILEFORHERM** – lokalizacja pliku importu danych; skrypt PHP, przygotowując DBF'a z treścią dokumentu, musi ustawiać ten parametr,
- **MIOD\_FILEFROMHERM** – lokalizacja pliku eksportu danych; skrypt PHP powinien odczytać parametr w celu otwarcia pliku z treścią dokumentu.

#### 5.1.1. Format plików importu

Pliki importu treści do HermesNG, w zależności od przeznaczenia (przyjęcie, rozchód) mają różne formaty.

##### 5.1.1.1. Format przyjęcia

Nazwa pola	Typ	Opis
POZ	int	unikalny identyfikator pozycji w magazynie HermesNG; w przypadku pustego pola lub wartości 0 (zero), system podejmie próbę odnalezienia pozycji magazynowej, wg pól: <i>indeks, kreski, ind_zewn</i>
ARTYKUL	char(50)	nazwa artykułu
INDEKS	char(12)	indeks artykułu w bazie HermesNG
KRESKI	char(13)	kod kreskowy przyjmowanej partii towaru
JM	char(3)	jednostka miary
ILOSC	float	przyjmowana ilość
CENA_ZAKUP	float	cena zakupu przyjmowanej partii
VAT	char(5)	stawka VAT w jakiej przyjmowana jest pozycja
ODCHYLENIE	float	odchylenie w cenie zakupu dla przyjmowanej pozycji
DATA_WAZN	char(10)	data ważności przyjmowanej partii
SERIA	char(15)	seria/atest partii
IND_ZEWN	char(30)	indeks zewnętrzny, powiązany z aktywnym dostawcą, wykorzystywany przy wyszukiwaniu kartoteki w magazynie

### 5.1.2. Format plików eksportu

Pliki eksportu posiadają wszystkie pola, dostępne w HermesNG, podczas edycji dokumentu.

## 5.2. Wymiana treści za pomocą funkcji „udostępnionych”

Funkcje „udostępnione” umożliwiają skryptom PHP, bezpośrednie kierowanie pracą HermesNG, np. funkcja *HNG\_DopiszDoPrzyjecia* dodaje do treści dokumentu nową pozycję, bez konieczności użycia plików pośredniczących.

Aby HermesNG, pomijał plik importu, należy parametr *MIOD\_FILEFORHERM* ustawić na wartość „NIE DOTYCZY”.

## 6. ŚRODOWISKO PHP

Import/eksport danych MIOD, wykorzystuje skrypty pisane w języku PHP. Nie jest konieczna dodatkowa instalacja środowiska PHP na stanowisku gdzie pracuje HermesNG. Przed wykonaniem skryptu PHP, HermesNG sprawdza czy dostępne są wszystkie pliki i biblioteki wymagane do uruchomienia PHP.

Pliki silnika PHP, znajdują się w „\folder Hermes\PHP\”. Domyślne środowisko PHP, skonfigurowane jest do używania modułów dostępnych w dystrybucji *HermesNG* (sekcja *[Dynamic Extensions]* w *php.ini*). Jeżeli w trakcie tworzenia skryptu, zajdzie konieczność użycia funkcji z innych modułów, należy odpowiedni pliki wgrać do ww. folderu oraz odpowiednio zmodyfikować plik *php.ini* (wg instrukcji do PHP).

Aby przywrócić standardową instalację PHP wraz z domyślnymi ustawieniami, jaka dostępna jest w HermesNG, należy usunąć folder PHP\ z katalogu, w którym zainstalowany jest Hermes.

### 6.1. Moduły standardowo dostarczane wraz z HermesNG

#### 6.1.1. Rozszerzenia dynamiczne – DLL

- php\_dbase.dll
- php\_interbase.dll
- php\_zip.dll

#### 6.1.2. Klasy/pliki PHP

- *PHPMailer* – więcej informacji na <http://phpmailer.sourceforge.net>