

## IDŹ DO

PRZYKŁADOWY ROZDZIAŁ



SPIS TREŚCI

## KATALOG KSIĄŻEK

KATALOG ONLINE

ZAMÓW DRUKOWANY KATALOG

## TWÓJ KOSZYK

DODAJ DO KOSZYKA

## CENNIK I INFORMACJE

ZAMÓW INFORMACJE  
O NOWOŚCIACH

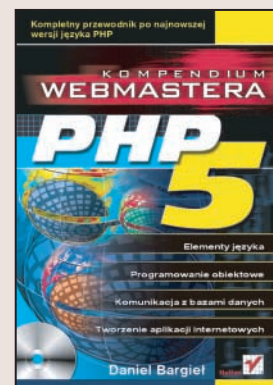
ZAMÓW CENNIK

## CZYTELNIA

FRAGMENTY KSIĄŻEK ONLINE

# PHP5. Kompendium webmastera

Autor: Daniel Bargieł  
ISBN: 83-7361-697-7  
Format: B5, stron: 576



Języka PHP nie trzeba już chyba przedstawiać żadnej osobie związanej z branżą internetową. Jeden z najpopularniejszych języków wykorzystywanych do tworzenia aplikacji internetowych rozwija się bardzo dynamicznie. W kolejnej jego wersji, oznaczonej numerem 5, pojawiło się wiele udoskonaleń rozszerzających bogate możliwości tej platformy. PHP5 jest językiem obiektowym, zawiera obsługę wyjątków i rozbudowane mechanizmy przetwarzania plików w formacie XML. Dzięki tym funkcjom znacznie zwiększył się zakres zastosowań PHP.

„PHP5. Kompendium webmastera” to książka opisująca najnowszą wersję tego środowiska. Przedstawia wszystko, co jest niezbędne do tworzenia aplikacji internetowych. Czytając ją, dowiesz się, jak zainstalować PHP na serwerze i stacji roboczej, poznasz elementy języka i zasady programowania obiektowego. Nauczysz się korzystać z baz danych i wykorzystywać w swoich aplikacjach pliki XML i mechanizmy szablonów Smarty. Stworzysz także przykładową aplikację internetową – tablicę ogłoszeń.

- Instalacja i konfiguracja PHP w Windows i Linuksie
- Zmienne, operatory i wyrażenia
- Tablice
- Konstrukcje warunkowe i pętle
- Elementy programowania obiektowego – klasy, obiekty, właściwości i metody
- Współpraca aplikacji PHP z bazami danych
- Obsługa formularzy i stosowanie wyrażeń regularnych
- Operacje na plikach i katalogach
- Wykorzystywanie plików w formacie XML
- Mechanizmy obsługi wyjątków
- Sesje, cookies i nagłówki HTTP
- Szablony Smarty

Jeśli chcesz poznać język PHP i napisać swoją pierwszą aplikację internetową, sięgnij po tę książkę.



---

# Spis treści

	<b>Wstęp</b> .....	9
	Czym jest PHP?.....	9
	Jak to działa?.....	11
	Dla kogo jest ta książka.....	12
	Co można znaleźć w książce.....	12
<b>Rozdział 1.</b>	<b>Instalacja i konfiguracja PHP</b> .....	<b>15</b>
	Instalacja w systemie Windows.....	15
	Instalacja PHP dla serwera IIS.....	16
	Instalacja PHP5 dla serwera Apache.....	17
	Instalacja w systemie Linux.....	19
	Konfigurowanie PHP.....	22
<b>Rozdział 2.</b>	<b>Podstawy programowania w PHP</b> .....	<b>25</b>
	Składnia.....	25
	Zmienne i stałe w języku PHP.....	30
	Nazwy zmiennych.....	30
	Przypisywanie oraz odczytywanie wartości zmiennych.....	31
	Typy proste zmiennych.....	35
	Definiowanie stałych.....	41
	Predefiniowane stałe PHP.....	42
	Operatory i wyrażenia.....	43
	Operatory i wyrażenia arytmetyczne.....	44
	Operator tekstowy.....	47
	Operatory binarne.....	48
	Operatory i wyrażenia porównania.....	49
	Operatory logiczne.....	50
	Konwersja typów.....	51
	Sterowanie przepływem programu.....	54
	Instrukcje if, else oraz else if.....	55
	Pętla warunkowa while.....	58
	Pętla warunkowa do ... while.....	59
	Pętla for.....	59

Instrukcja switch .....	61
Instrukcje pomocnicze .....	63
Tablice.....	64
Tablice jednowymiarowe .....	65
Tablice wielowymiarowe .....	67
Pętla foreach.....	69
Konwersja tablic do zmiennych typów prostych.....	70
Funkcje.....	71
Argumenty funkcji .....	72
Zwracanie wartości przez funkcję .....	75
Złożone definiowanie funkcji.....	76
Funkcje wbudowane w PHP .....	78
Zagnieżdżanie skryptów PHP.....	108
Zasięg zmiennych.....	112
Predefiniowane zmienne PHP .....	115
Zmienna \$_SERVER .....	116
Zmienna \$_GET.....	118
Referencje i ich zastosowanie.....	120
<b>Rozdział 3. Programowanie obiektowe .....</b>	<b>123</b>
Klasa, obiekt, właściwości i metody.....	123
Metody oraz właściwości klasy.....	126
Konstruktory i destruktory .....	134
Konstruktor klasy .....	134
Destruktor klasy .....	143
Dziedziczenie .....	146
Dostęp do metod i właściwości obiektu .....	151
Dostęp publiczny.....	151
Dostęp prywatny .....	152
Dostęp chroniony .....	154
Klasy i metody abstrakcyjne.....	159
Metody typu final.....	162
Rdzeń aplikacji.....	162
Korzystanie z interfejsów .....	166
Powielanie obiektów .....	171
Funkcja __autoload .....	175
<b>Rozdział 4. Komunikacja z bazą danych.....</b>	<b>177</b>
Konfigurowanie PHP .....	177
System Windows.....	178
System Linux .....	179
Klienci baz danych .....	182
Query Analyzer dla MS SQL Server.....	182
DBManager dla MySQL oraz PostgreSQL .....	187
Podstawy SQL.....	193
Tworzenie i usuwanie bazy danych.....	194
Tworzenie i usuwanie tabel w bazie danych .....	196
Wprowadzanie informacji do tabeli oraz pobieranie informacji z tabeli .....	197
Aktualizacja informacji oraz usuwanie informacji z tabeli .....	201
Tworzenie połączenia z bazą danych z poziomu PHP.....	202
MySQL .....	202
PostgreSQL 7 .....	227
MS SQL Server 2000 .....	240
Rdzeń aplikacji.....	250

<b>Rozdział 5.</b>	<b>Obsługa formularzy</b>	<b>257</b>
	Kod HTML formularza	258
	Pola formularza	259
	Odbieranie danych przesyłanych przez formularz HTML	265
	Metody GET oraz POST przesyłania danych	265
	Odbieranie danych z formularza	266
	Walidacja danych z formularza z wykorzystaniem wyrażeń regularnych	271
	Czym są wyrażenia regularne?	271
	Wyrażenia regularne w PHP	271
	Metaznaki wyrażeń regularnych	272
	Wyrażenia regularne walidujące adresy WWW oraz adresy e-mail	278
	Wydajność	279
	Funkcje PHP do obsługi wyrażeń regularnych	281
	Odszukiwanie dopasowań. Funkcja preg_match()	282
	Dzielenie ciągu przeszukiwanego. Funkcja preg_split()	282
	Tworzenie formularza walidującego gromadzone dane	284
<b>Rozdział 6.</b>	<b>Operacje na plikach</b>	<b>287</b>
	Otwarcie i zamknięcie pliku — funkcje fopen() oraz fclose()	287
	Odczytywanie zawartości pliku	290
	Czytanie pliku wiersz po wierszu	290
	Czytanie pojedynczych znaków — funkcja fgetc()	300
	Czytanie określonej liczby znaków — funkcja fread()	301
	Czytanie całego pliku	302
	Zapisywanie informacji do pliku	306
	Funkcja fwrite()	306
	Funkcja file_put_contents()	309
	Poruszanie się po pliku	310
	Funkcja rewind()	310
	Funkcja fseek()	311
	Funkcja ftell()	313
	Usuwanie plików	313
	Zbieranie informacji o plikach	314
	Sprawdzanie, czy plik istnieje — funkcja file_exists()	314
	Plik czy katalog? — funkcje is_file() oraz is_dir()	315
	Statystyki plików	316
	Sprawdzanie i modyfikacja uprawnień do plików i katalogów	319
	Funkcja is_readable()	319
	Funkcja is_writable()	320
	Funkcja is_executable()	320
	Zmiana nazwy plików i katalogów oraz kopiowanie plików	322
	Zmiana nazwy plików i katalogów — funkcja rename()	323
	Kopiowanie plików — funkcja copy()	323
	Operacje na katalogach	324
	Otwieranie i zamykanie katalogów	324
	Odczytywanie zawartości katalogu	325
	Tworzenie i usuwanie katalogów	331
	Sprawdzanie ilości wykorzystanego miejsca na dysku	332
	Rdzeń aplikacji — klasa Debugger	335
	Przesyłanie plików za pomocą formularzy	338
	Formularz wysyłania plików	338
	Zmienna \$FILES	339
	Kopiowanie przesłanych plików	340
	Buforowanie informacji o plikach	344

<b>Rozdział 7.</b>	<b>Programowanie wykorzystujące standard XML</b> .....	<b>345</b>
	Czym są dokumenty XML .....	345
	Obiektowy model dokumentu XML (DOM XML).....	349
	Przetwarzanie dokumentów XML za pomocą rozszerzenia PHP o nazwie SimpleXML.....	350
	Przetwarzanie dokumentów XML za pomocą klasy DOMDocument.....	361
	Wyrażenia XPath.....	396
	Tworzenie wyrażeń XPath w PHP .....	397
	Składnia XPath.....	397
<b>Rozdział 8.</b>	<b>Obsługa błędów</b> .....	<b>403</b>
	Grupy i typy błędów w PHP.....	404
	Błędy inicjacji PHP .....	405
	Błędy kompilacji .....	406
	Błędy czasu wykonania.....	407
	Błędy użytkownika.....	408
	Kolejność zgłaszania błędów .....	409
	Raportowanie i wyświetlanie błędów w PHP.....	410
	Raportowanie błędów.....	410
	Wyświetlanie błędów .....	412
	Zapisywanie błędów do pliku dziennika zdarzeń .....	413
	Przechwytywanie błędów z poziomu PHP .....	414
	Operator kontroli błędów — operator @ .....	417
	Wyjątki.....	417
	Klasa Exception .....	418
	Zgłaszanie i przechwytywanie wyjątków.....	420
	Klasy potomne klasy Exception .....	424
	Rdzeń aplikacji.....	425
<b>Rozdział 9.</b>	<b>Sesje, cookies oraz obsługa nagłówków HTTP</b> .....	<b>435</b>
	Czym jest sesja.....	436
	Jak działa mechanizm sesji .....	436
	Tworzenie skryptów korzystających z mechanizmu sesji .....	437
	Uruchamianie sesji .....	437
	Uzyskiwanie identyfikatora sesji po stronie klienta .....	439
	Tworzenie zmiennych sesji .....	439
	Niszczenie zmiennych sesyjnych i niszczenie sesji.....	441
	Czas trwania sesji.....	441
	Pliki sesji.....	443
	Prosty skrypt autoryzacji.....	444
	Cookies.....	446
	Tworzenie cookies .....	446
	Usuwanie cookie.....	449
	Nagłówki HTTP .....	449
	Rdzeń aplikacji.....	451
<b>Rozdział 10.</b>	<b>System szablonów Smarty</b> .....	<b>459</b>
	Instalacja systemu szablonów Smarty .....	459
	Struktura katalogowa .....	462
	Parametry konfiguracyjne .....	463

Metody obiektów klasy Smarty.....	467
Tworzenie zmiennych w szablonach TPL — metody assign() oraz assign_by_ref().....	467
Usuwanie utworzonych zmiennych z szablonów TPL — metody clear_all_assign() oraz clear_assign().....	469
Metoda clear_compiled_tpl().....	469
Tworzenie szablonów.....	470
Zasady tworzenia szablonów TPL.....	470
Zmienne.....	471
Zarezerwowana zmienna \$smarty.....	473
Modyfikatory zmiennych.....	476
Funkcje wbudowane.....	486
<b>Rozdział 11. Aplikacja internetowa — Tablica ogłoszeń.....</b>	<b>493</b>
Funkcjonalność aplikacji.....	493
Projekt aplikacji.....	494
Projekt bazy danych.....	494
Struktura katalogowa.....	497
Interfejs programowy aplikacji.....	497
Interfejs graficzny aplikacji.....	505
Co można poprawić.....	510
Instalacja i uruchomienie aplikacji.....	512
Instalacja bazy danych.....	512
Instalacja plików bazy danych i konfiguracja aplikacji.....	513
Problemy.....	514
Użytkownicy i dane aplikacji.....	515
<b>Dodatek A Instalacja i konfiguracja serwerów WWW.....</b>	<b>517</b>
Instalacja i konfiguracja serwerów IIS i Apache w systemie Windows.....	517
Instalacja i konfiguracja serwera IIS.....	517
Instalacja i konfiguracja serwera Apache.....	523
Instalacja i konfiguracja serwera Apache 2 w systemie Linux.....	527
<b>Dodatek B Instalacja i konfiguracja wybranych baz danych.....</b>	<b>531</b>
Instalacja i konfiguracja bazy danych MySQL.....	531
Instalacja MySQL w środowisku Windows.....	532
Użytkownicy bazy danych MySQL.....	535
Instalacja MySQL w środowisku Linux.....	536
Instalacja i konfiguracja bazy danych PostgreSQL.....	537
Instalacja bazy danych PostgreSQL w systemie Windows.....	537
Konfiguracja bazy danych PostgreSQL.....	539
Instalacja bazy danych PostgreSQL w systemie Linux.....	540
Instalacja serwera Microsoft SQL Server 2000.....	543
<b>Dodatek C Dyrektywy konfiguracyjne.....</b>	<b>547</b>
Funkcje operujące na wartościach dyrektyw.....	547
Dyrektywy ogólne.....	550
Obsługa plików.....	551
Obsługa i raportowanie błędów.....	551
Obsługa sesji.....	552
<b>Skorowidz.....</b>	<b>555</b>

# 4.

---

## Komunikacja z bazą danych

Niemal w każdej większej aplikacji internetowej istnieje komunikacja pomiędzy wykonywanymi skryptami oraz bazą danych, która zawiera informacje wykorzystywane przez aplikację WWW.

PHP5 daje programiście możliwość komunikowania się z wieloma typami baz danych. W tym rozdziale skupimy się jednak na trzech z nich: *MySQL* w wersji 4, *PostgreSQL* w wersji 7 oraz *Microsoft SQL Server (MS SQL Server)* w wersji 2000.

Wszystkie wymienione bazy danych można zainstalować w systemie Windows, natomiast w systemie Linux mogą zostać zainstalowane bazy *MySQL* oraz *PostgreSQL*.



### Uwaga

---

W rozdziale zakładamy, że serwery wspomnianych baz danych zostały zainstalowane i skonfigurowane. Informacja o tym, jak zainstalować oraz skonfigurować każde ze wspomnianych środowisk bazodanowych, znajduje się w dodatku B „Instalacja i konfiguracja wybranych baz danych”.

## Konfigurowanie PHP

PHP5 w przeciwieństwie do PHP4 nie posiada wbudowanych funkcji obsługujących połączenia z bazami danych, dlatego też musimy sprawić, aby interpreter PHP dołączył odpowiednie moduły.

Omawiane w tym podrozdziale procedury konfiguracyjne zakładają, że serwery baz danych zostały zainstalowane zgodnie z instrukcjami przedstawionymi w dodatku B „Instalacja i konfiguracja wybranych baz danych”.

# System Windows

## MySQL

Aby interpreter PHP zainstalowany w systemie Windows mógł komunikować się z bazą danych *MySQL*, w pliku *php.ini* należy ustawić dyrektywę `extension=php_mysql.dll`.

Uruchomienie tego rozszerzenia powoduje, że interpreter poszukuje biblioteki *libmysql.dll*, która stanowi interfejs klienta bazy danych *MySQL*, oraz biblioteki *php\_mysql.dll*. Biblioteki te znajdują się na CD-ROM-ie dołączonym do książki, w katalogu *Instalacja*. Powinny one się znaleźć w katalogu, na który wskazuje dyrektywa `extension_dir` w pliku *php.ini*.



### Uwaga

Więcej o pliku *php.ini* można przeczytać w dodatku C „Dyrektywy konfiguracyjne”.

Aby przekonać się, czy interpreter PHP załadował rozszerzenie pozwalające na komunikację z serwerem *MySQL*, wystarczy uruchomić skrypt:

```
<?php
mysql_connect();
?>
```

Jeżeli wynikiem wykonania skryptu będzie komunikat:

```
Fatal error: Call to undefined function mysql_connect() in skrypt.php on line 2
```

oznacza to, że rozszerzenie nie zostało załadowane, ponieważ interpreter nie rozpoznaje funkcji `mysql_connect()`.

## PostgreSQL

Aby interpreter PHP zainstalowany w systemie Windows mógł komunikować się z bazą danych *PostgreSQL*, w pliku *php.ini* należy uaktywnić dyrektywę `extension=php_pgsql.dll`.

Biblioteka *php\_pgsql.dll*, która jest ładowanym rozszerzeniem, znajduje się na CD-ROM-ie dołączonym do książki, w katalogu *Instalacja*. Powinna ona znaleźć się w katalogu wskazywanym przez dyrektywę `extension_dir` z pliku *php.ini*.



### Uwaga

Więcej o pliku *php.ini* można przeczytać w dodatku C „Dyrektywy konfiguracyjne”.

Aby przekonać się, czy interpreter PHP załadował rozszerzenie pozwalające na komunikację z serwerem *PostgreSQL*, wystarczy uruchomić skrypt:

```
<?php
pg_connect('');
?>
```



Jeżeli wynikiem wykonania skryptu będzie tekst:

```
Fatal error: Call to undefined function pg_connect() in skrypt.php on line 2
```

oznacza to, że rozszerzenie nie zostało załadowane, ponieważ interpreter nie rozpoznaje funkcji `pg_connect()`.

## MS SQL Server

Aby interpreter PHP zainstalowany w systemie Windows mógł komunikować się z bazą danych *MS SQL Server*, w pliku *php.ini* należy uaktywnić dyrektywę `extension=php_mssql.dll`.

Niezbędne są dwie biblioteki obecne w systemie: *ntwdblib.dll* oraz *php\_mssql.dll*. Obie znajdują się na CD-ROM-ie dołączonym do książki, w katalogu *Instalacja*. Biblioteka *ntwdblib.dll* powinna znaleźć się w katalogu `/winnt/system32/` (jeżeli jeszcze jej tam nie ma), natomiast biblioteka *php\_mssql.dll* powinna znaleźć się w katalogu wskazywanym przez dyrektywę `extension_dir` z pliku *php.ini*.



### Uwaga

Więcej o pliku *php.ini* można przeczytać w dodatku C „Dyrektywy konfiguracyjne”.

Aby przekonać się, czy interpreter PHP załadował rozszerzenie pozwalające na komunikację z serwerem *MS SQL Server*, wystarczy uruchomić skrypt:

```
<?php
mssql_connect();
?>
```

Jeżeli wynikiem wykonania skryptu będzie tekst:

```
Fatal error: Call to undefined function mssql_connect() in skrypt.php on line 2
```

oznacza to, że rozszerzenie nie zostało załadowane, ponieważ interpreter nie rozpoznaje funkcji `mssql_connect()`.

## System Linux

Konfiguracja PHP 5.0.2 pracującego w systemie Linux do współpracy z serwerami baz danych *MySQL*, *PostgreSQL* oraz *MS SQL Server* jest nieco bardziej skomplikowana niż w systemie Windows, ponieważ wymaga odpowiedniej kompilacji PHP.

## MySQL

Aby interpreter PHP stanowiący moduł serwera Apache 2 w systemie Linux mógł komunikować się z serwerem bazy danych *MySQL*, PHP musi zostać skompilowany z opcją `--with-mysql`.

Jeżeli w systemie Linux zainstalowaliśmy serwer bazy danych *MySQL* zgodnie z instrukcją zawartą w dodatku B „Instalacja i konfiguracja wybranych baz danych”, to PHP 5.0.2 dla serwera Apache 2 powinniśmy skompilować w następujący sposób:

```
shell> ./configure --with-apxs2=/usr/local/apache2/bin/apxs
--with-mysql=/usr/local/mysql
--with-mysql-sock=/tmp/mysql.sock

shell> make
shell> make install
```

Aby sprawdzić, czy proces instalacji PHP z obsługą bazy danych *MySQL* powiódł się, wystarczy wykonać prosty skrypt:

```
<?php
mysql_connect();
?>
```



### Uwaga

Więcej o procesie kompilacji i instalacji PHP w systemie Linux można przeczytać w rozdziale 1. „Instalacja i konfiguracja PHP”.

Jeżeli wynikiem wykonania skryptu będzie komunikat

```
Fatal error: Call to undefined function mysql_connect() in skrypt.php on line 2
```

oznacza to, że rozszerzenie nie zostało załadowane, ponieważ interpreter nie rozpoznaje funkcji `mysql_connect()`.

## PostgreSQL

Aby interpreter PHP stanowiący moduł serwera Apache 2 w systemie Linux mógł komunikować się z serwerem bazy danych *PostgreSQL*, PHP musi zostać skompilowany z opcją `--with-pgsql`.

Jeżeli w systemie Linux zainstalowaliśmy serwer bazy danych *PostgreSQL* zgodnie z instrukcją zawartą w dodatku B „Instalacja i konfiguracja wybranych baz danych”, to PHP 5.0.2 dla serwera Apache 2 powinniśmy skompilować w następujący sposób:

```
shell> ./configure --with-apxs2=/usr/local/apache2/bin/apxs
--with-pgsql=/usr/local/pgsql

shell> make
shell> make install
```

Aby sprawdzić, czy proces instalacji PHP z obsługą bazy danych *PostgreSQL* powiódł się, wystarczy wykonać prosty skrypt:

```
<?php
pg_connect();
?>
```



### Uwaga

Więcej o procesie kompilacji i instalacji PHP w systemie Linux można przeczytać w rozdziale 1. „Instalacja i konfiguracja PHP”.

Jeżeli wynikiem wykonania skryptu będzie komunikat:

```
Fatal error: Call to undefined function pg_connect() in skrypt.php on line 2
```

oznacza to, że rozszerzenie nie zostało załadowane, ponieważ interpreter nie rozpoznaje funkcji `pg_connect()`.

## MS SQL Server

Aby interpreter PHP stanowiący moduł serwera Apache 2 w systemie Linux mógł komunikować się z serwerem *MS SQL Server* (który musi być zainstalowany na jakiejś stacji z systemem Windows), należy wykonać dwupoziomą procedurę instalacyjną.

Po pierwsze, należy zainstalować w systemie bibliotekę *FreeTDS*, której pliki źródłowe znajdują się na CD-ROM-ie dołączonym do książki, w katalogu *Instalacja/freetds-0.62.4*

Po przejściu do katalogu, w którym znajdują się pliki źródłowe, należy skonfigurować proces kompilacji, skompilować bibliotekę oraz zainstalować ją w systemie:

```
shell> ./configure --prefix=/usr/local/freetds --enable-msdblib
shell> make
shell> make install
```

Po zainstalowaniu biblioteki *FreeTDS* możemy przystąpić do drugiej fazy, a mianowicie do kompilacji PHP, tak aby posiadał obsługę komunikacji z bazą danych *MS SQL Server*:

```
shell> ./configure --with-apxs2=/usr/local/apache2/bin/apxs --with-mssql=/usr/local/freetds
shell> make
shell> make install
```

Dyrektywa `--with-mssql` musi wskazywać na katalog, w którym została zainstalowana biblioteka *FreeTDS*.

Od tej pory możemy tworzyć skrypty PHP, które będą komunikować się z serwerem bazy danych *MS SQL Server* z poziomu Linuksa.

Aby przekonać się, czy interpreter PHP załadował rozszerzenie pozwalające na komunikację z serwerem *MS SQL Server*, wystarczy uruchomić skrypt:

```
<?php
mssql_connect();
?>
```

Jeżeli wynikiem wykonania skryptu będzie tekst:

```
Fatal error: Call to undefined function mssql_connect() in skrypt.php on line 2
```

oznacza to, że rozszerzenie nie zostało załadowane, ponieważ interpreter nie rozpoznaje funkcji `mssql_connect()`.