

Mariusz Duka

Programowanie
dla początkujących
w **50** lekcjach

PHP8 *i* SQL

Wydanie 2

Helion 

Wszelkie prawa zastrzeżone. Nieautoryzowane rozpowszechnianie całości lub fragmentu niniejszej publikacji w jakiegokolwiek postaci jest zabronione. Wykonywanie kopii metodą kserograficzną, fotograficzną, a także kopiowanie książki na nośniku filmowym, magnetycznym lub innym powoduje naruszenie praw autorskich niniejszej publikacji.

Wszystkie znaki występujące w tekście są zastrzeżonymi znakami firmowymi bądź towarowymi ich właścicieli.

Autor oraz wydawca dołożyli wszelkich starań, by zawarte w tej książce informacje były kompletne i rzetelne. Nie biorą jednak żadnej odpowiedzialności ani za ich wykorzystanie, ani za związane z tym ewentualne naruszenie praw patentowych lub autorskich. Autor oraz wydawca nie ponoszą również żadnej odpowiedzialności za ewentualne szkody wynikłe z wykorzystania informacji zawartych w książce.

Redaktor prowadzący: Małgorzata Kulik

Projekt okładki: Studio Gravite/Olsztyn
Obarek, Pokoński, Pazdrijowski, Zaprucki

Grafika na okładce została wykorzystana za zgodą AdobeStock.com.

Helion S.A.
ul. Kościuszki 1c, 44-100 Gliwice
tel. 32 230 98 63
e-mail: helion@helion.pl
WWW: helion.pl (księgarnia internetowa, katalog książek)

Drogi Czytelniku!
Jeżeli chcesz ocenić tę książkę, zajrzyj pod adres
helion.pl/user/opinie/php8pr
Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

ISBN: 978-83-289-2953-1

Copyright © Helion S.A. 2026

Printed in Poland.

- [Kup książkę](#)
- [Poleć książkę](#)
- [Oceń książkę](#)

- [Księgarnia internetowa](#)
- [Lubię to! » Nasza społeczność](#)

Spis treści

	Przedmowa	7
LEKCJA 01	Co nowego w PHP 8	15
LEKCJA 02	Konfiguracja	31
LEKCJA 03	Klasyka, czyli pierwszy program	47
LEKCJA 04	Komentarz mile widziany	53
LEKCJA 05	Co oferują PHP i MySQL	56
LEKCJA 06	PHP Composer	65
LEKCJA 07	Zmienne i stałe	73
LEKCJA 08	Typy wyliczeniowe	78
LEKCJA 09	Łańcuchy znaków	82
LEKCJA 10	Co, jeśli? Konstrukcje warunkowe	87
LEKCJA 11	Co, jeśli? Inaczej też się da	93
LEKCJA 12	Co, jeśli? Trochę logiki	97
LEKCJA 13	Znajdź coś w tekście	104
LEKCJA 14	Kombinacje z tekstem	111
LEKCJA 15	Operacje na liczbach	115
LEKCJA 16	Trochę matmy dla opornych	122
LEKCJA 17	Więcej matmy dla ambitnych	126
LEKCJA 18	Nareszcie własna funkcja	131
LEKCJA 19	Funkcje mogą coś zwracać	136
LEKCJA 20	Funkcja a zasięg zmiennych	141
LEKCJA 21	Pętla, czyli cykliczne powtarzanie instrukcji	144
LEKCJA 22	Pętla w akcji	148
LEKCJA 23	Tablice i klucze	152
LEKCJA 24	Wymiary tablic	159

LEKCJA 25	Odczyt i zapis plików	168
LEKCJA 26	Sytuacje wyjątkowe	179
LEKCJA 27	Dołącz kod źródłowy z pliku	186
LEKCJA 28	Data i czas	191
LEKCJA 29	Zrób coś z klasą	199
LEKCJA 30	Jest czy ma? Oto jest pytanie	207
LEKCJA 31	Trochę więcej o klasach	213
LEKCJA 32	Testowanie kodu z PHPUnit	222
LEKCJA 33	Dokumentowanie kodu	231
LEKCJA 34	Pierwsza baza danych	237
LEKCJA 35	Proste tabele z danymi	251
LEKCJA 36	Bezpieczeństwo w SQL-u	265
LEKCJA 37	Arytmetyka w SQL-u	275
LEKCJA 38	Rozmówki klienta z serwerem	283
LEKCJA 39	Serwer webowy w PHP	291
LEKCJA 40	HTML dla zielonych	298
LEKCJA 41	MVC — prosty sposób na porządek w kodzie	308
LEKCJA 42	Budujemy prosty szkielet MVC	312
LEKCJA 43	Logika i dane w MVC	317
LEKCJA 44	Sesje i ciasteczka	321
LEKCJA 45	Czas na pierwszy projekt	327
LEKCJA 46	Dwa światy, logika i widok	333
LEKCJA 47	Uruchomienie pierwszej strony	341
LEKCJA 48	Kontakt mile widziany	358
LEKCJA 49	A może trochę rozrywki?	366
LEKCJA 50	Trochę porad na zakończenie	375
	Skorowidz	378

Przedmowa

Minęło już kilka lat od chwili, gdy wydawnictwo Helion opublikowało moją ostatnią książkę na temat podstaw programowania w języku PHP 8. W międzyczasie język przeszedł wiele zmian, pojawiły się kolejne wersje zwiększające wydajność, bezpieczeństwo i elastyczność kodu. Wielu programistom na co dzień wykorzystującym język PHP nowe funkcjonalności powinny ułatwić precyzyjne pisanie kodu przy jednoczesnym zwiększeniu jego wydajności. Wiele emocji nadal wzbudza mechanizm JIT (ang. *Just In Time*), który służy do kompilowania programu do postaci kodu maszynowego przed jego wykonaniem. Zgodnie z zapowiedziami twórców języka mechanizm JIT ma zapewnić znaczne przyspieszenie wykonywania programu w porównaniu z tradycyjną metodą, czyli programem wykonywanym przez interpreter. W wersji PHP 8.4 mechanizm ten otrzymał istotną aktualizację, dzięki której zużywa mniej pamięci i mniej obciąża procesor przy jednoczesnym zwiększeniu wydajności.

Niniejsza książka, podobnie jak poprzednia¹, ma w prosty sposób nauczyć Cię programować w języku PHP, który już od wielu lat wykorzystuje się do projektowania stron internetowych oraz budowania skomplikowanych aplikacji działających po stronie serwera. O potężnych możliwościach PHP niech świadczy to, że używają go tacy giganci, jak Google, Facebook, Yahoo! czy Wikipedia. Warto dodać, że zgodnie ze statystykami serwisu W3Techs² prawie 75% stron internetowych na świecie działa na serwerach obsługujących język PHP. I choć nie wszystkie serwery udostępniają taką informację, to i tak oznacza to, że prawie 8 na 10 stron internetowych wykorzystuje język PHP. Jestem przekonany, że tak wysoki poziom będzie się długo utrzymywał, zwłaszcza że twórcy języka cały czas pracują nad jego rozwojem.

Od prawie 30 lat PHP dynamicznie się rozwija — powstają kolejne, wydajniejsze wersje, wzbogacone o nowe funkcje. Pod koniec lat 90. ubiegłego wieku programiści do budowy serwisów internetowych często stosowali język Perl, czyli skrypty CGI.

¹ M. Duka, *PHP8 i SQL. Programowanie dla początkujących w 43 lekcjach*, Helion, Gliwice 2021.

² W3Techs, <https://w3techs.com/technologies/details/pl-php> [dostęp: 30.06.2025].

Korzystano też z technologii SSI (ang. *Server Side Includes*), a więc mechanizmu skryptowego służącego do dynamicznego generowania stron WWW. Rewolucja nastąpiła, kiedy został udostępniony i spopularyzował się język PHP 3. I tak to się zaczęło.

PHP/FI, PHP 3, PHP 4, PHP 5, PHP 7, PHP 8 to historia rozwoju wspaniałego języka programowania, będąca wynikiem wyłożonej pracy wielu programistów z całego świata. PHP nadaje się nie tylko do programowania stron internetowych, ale również do budowy skomplikowanych aplikacji użytkowych, opartych na mnogości dostępnych rozszerzeń i modułów. Już od wersji PHP 7.4 istotnie poprawiono wydajność przetwarzania kodu przy jednoczesnym zmniejszeniu zapotrzebowania zasobów systemu. PHP 8 z mechanizmem JIT jest po prostu o wiele szybszy od swoich poprzedników, co ma znaczenie choćby dla firm hostingowych, które mogą oferować usługi na jeszcze wyższym poziomie. Programiści mogą wykorzystywać PHP do ogromnej liczby zastosowań, a dostępność wielu dodatkowych bibliotek liczbę tę znacznie zwiększa.

Począwszy od wersji PHP 7.2, udoskonalone w wersji PHP 8, dostępne są dla programistów wieloplatformowa biblioteka Sodium oraz nowy algorytm haszujący Argon2, co sprawia, że język PHP stał się bezpiecznym środowiskiem implementującym najnowsze rozwiązania kryptografii. PHP udostępnia bibliotekę OpenSSL, którą programista może wykorzystać do budowy bezpiecznego systemu opartego na kryptografii krzywych eliptycznych. Fascynacja możliwościami, jakie oferuje język PHP, skłoniła mnie do napisania publikacji naukowej³, w której przedstawiłem wyniki badań związanych ze wskazaniem optymalnej krzywej eliptycznej do budowy kryptosystemu.

PHP od zawsze był językiem dynamicznie rozwijanym, a kolejne wersje wprowadzają nowe funkcje, optymalizacje i ulepszenia, które czynią go jeszcze bardziej wszechstronnym. Gdy wprowadzono PHP 8, programiści zyskali dostęp do narzędzi, które nie tylko ułatwiają pracę, ale także pozwalają pisać kod bardziej czytelny, bezpieczny i wydajny. Każda wersja to kolejny krok w kierunku nowoczesnych rozwiązań, a historia ostatnich lat pokazuje, jak bardzo PHP ewoluował.

Wdrażanie nowych funkcjonalności języka w już istniejących projektach zwykle zajmuje programistom dużo czasu, natomiast nowe projekty wykorzystują już pełną moc PHP 8. Wersja PHP 9 jest w planach i najprawdopodobniej pojawi się po wersji 8.5 i 8.6, z pewnością wnosząc nowe możliwości dla programistów, dlatego nie ma

³ M. Duka, *Elliptic-curve cryptography (ECC) and Argon2 algorithm in PHP using OpenSSL and Sodium libraries*, „IAPGOŚ” 2020, 10 (3), s. 91 – 94, <https://doi.org/10.35784/iapgos.897>.

co czekać, po prostu już teraz rozpocznij naukę programowania w PHP. Wiedzę wyniesioną z lektury tej książki będziesz mógł przez następne lata rozwijać i stosować w wielu swoich projektach.

Książka ta ma Ci uświadomić, że nauka programowania wcale nie jest taka trudna, wymaga jedynie chęci, cierpliwości, determinacji i czasu. Przerobienie wszystkich lekcji będzie dobrą podstawą do tego, abyś w przyszłości mógł sięgnąć po bardziej zaawansowane tematycznie książki, dotyczące nie tylko PHP, lecz również języków SQL, HTML, CSS i JavaScript.

Zawartość książki

Książkę podzieliłem na praktyczne lekcje i uzupełniłem o wskazówki, ćwiczenia do samodzielnego wykonania i odpowiedzi na pytania, które mogą się pojawić. Kod zamieszczony w książce oznaczyłem dodatkową informacją o wersji PHP, która jest wymagana do prawidłowego działania.

Informacje, na które powinieneś zwrócić uwagę, oznaczyłem takim stylem.

W pierwszej kolejności przygotujesz środowisko programistyczne, w którym będziesz pisał swoje pierwsze programy. Na rynku dostępnych jest wiele ciekawych rozwiązań, my zastanowimy się nad wyborem pomiędzy Eclipse, Apache NetBeans, Visual Studio Code i Notepad++.

Następnie krok po kroku dowiesz się, jak wygląda składnia i struktura języka PHP, w tym nowe elementy PHP 8, a także nauczysz się używać języka SQL do obsługi baz danych SQLite i MySQL. W następnych lekcjach zapoznasz się ze sposobem prawidłowego łączenia kodu PHP i HTML, a na zakończenie wykorzystasz popularny framework Bootstrap oraz bibliotekę Smarty do budowy swojej pierwszej strony internetowej.

Ćwiczenia praktyczne

Każda lekcja zawiera ćwiczenia, które powinieneś przerobić samodzielnie. Do ich rozwiązania wykorzystasz już zdobyte wiadomości. Uzupełnieniem są ćwiczenia dodatkowe, które wymagają samodzielnego poszerzenia wiedzy. Wszystkie przykłady staraj się przepisywać, a nie kopiować i wklejać. Ma to ogromne znaczenie, gdyż tylko w taki sposób nabierzesz prawidłowego nawyku kodowania w PHP.

Jak zapewne wiesz, matematyka to królowa nauk, dlatego w ćwiczeniach praktycznych pojawiają się również zagadnienia związane z logiką, algebrą czy geometrią. Mam nadzieję, że podstawy matematyki są Ci znane i chętnie zastosujesz je w programowaniu.

Pamiętaj, tylko praktyka czyni mistrza!

W jakim środowisku programować?

Bardzo dobre pytanie! Każdy programista ma swoją teorię na ten temat i nie ukrywam, że ja również. Tak naprawdę każdy stara się wykorzystywać najwygodniejsze dla siebie narzędzie i jest święcie przekonany, że jest ono najlepsze. Na przestrzeni ostatnich lat do programowania wykorzystywałem wiele różnych narzędzi, jednak najczęściej ograniczałem się do kilku moich ulubionych, mianowicie Eclipse, Apache NetBeans i Notepad++. Na uwagę zasługuje również Visual Studio Code, które świetnie nadaje się do kodowania.

Dobry programista powinien umiejętnie używać dowolnego narzędzia umożliwiającego zrealizowanie konkretnego zadania. Niewielkie programy często szybciej się pisze za pomocą prostego edytora tekstowego, natomiast aktualizacja istotnego elementu w dużym projekcie, nad którym pracuje wiele osób, musi być wykonana w profesjonalnym środowisku programistycznym.

W niniejszej książce opisałem procedurę instalacji i konfiguracji środowisk programistycznych Eclipse, Apache NetBeans i Visual Studio Code. Warto nauczyć się wykorzystywać je do kodowania, gdyż są to środowiska bardzo popularne wśród programistów i umożliwiają programowanie w wielu językach, takich jak Java, C/C++, Python, JavaScript, Ruby oraz PHP, a przy tym, co najważniejsze, są bezpłatne i cały czas rozwijane. Przyswojenie obsługi tych narzędzi z pewnością ułatwi Ci naukę innych języków i pozwoli bez większych problemów tworzyć interesujące i nowoczesne projekty.

Zachęcam Cię, abyś wykorzystał do nauki programowania również swój ulubiony edytor tekstowy. Polecam Notepad++; nabierzesz wówczas wprawy w poruszaniu się w konsoli. Bardzo często korzystam z tego edytora. W pierwszej lekcji pokażę Ci, jak go w prosty sposób zainstalować.

Dobrą wiadomością jest to, że wszystkie programy zawarte w tej książce były przetestowane w każdym z wyżej wymienionych narzędzi programistycznych, a kody źródłowe można pobrać ze strony wydawnictwa Helion.

Dla kogo przeznaczona jest ta książka?

Książka przeznaczona jest dla początkujących programistów, jak również dla tych, którzy chcieliby odświeżyć swoją wiedzę. W porównaniu do mojej poprzedniej pozycji o programowaniu w języku PHP część przykładów w tej książce wybiega poza jego podstawy, aby zmotywować Cię do jeszcze większego wysiłku.

Książka przyda się również osobom prowadzącym warsztaty z informatyki, chcącym uzupełnić swoje zajęcia. Przy każdej lekcji znajdziesz informację o czasie, w jakim powinienś sobie z nią poradzić. Nie martw się, jeśli go przekroczysz, i tak nikt tego nie sprawdzi.

PHP jest bardzo rozbudowanym językiem programowania. Programy mają często po kilka lub nawet kilkanaście tysięcy linii kodu i wykorzystują ogromną liczbę funkcji, bibliotek i modułów. Nie musisz się jednak martwić — w tej książce ograniczyłem się do podstaw, abyś mógł krok po kroku rozwijać swoje umiejętności.

Czy dasz radę?

Pewnie zastanawiasz się, czy dasz radę i czy uda Ci się pomyślnie zrealizować wszystkie lekcje. Zastanawiasz się, jak masz to zrobić, przecież nie masz o tym zielonego pojęcia! Czytaj i ćwicz, a już po pierwszych lekcjach zauważysz postęp!

Moja historia z programowaniem zaczęła się pod koniec lat 80. ubiegłego wieku, kiedy jedynym językiem programowania, jaki wówczas był dostępny na moim komputerze, był język maszynowy. Do dyspozycji miałem książkę w języku niemieckim (sic!) i kartka po kartce uczyłem się programować. Tak zapoznałem się z asemblerem, moim pierwszym językiem programowania. Z innymi językami było już znacznie łatwiej. Miło wspominać Turbo Pascal firmy Borland i pierwsze programy w Perlu.

Do dzisiaj mam zachowane swoje pierwsze programy napisane w PHP. Są już od dawna pełnoletnie!

Dzisiaj masz o wiele większe możliwości nauki programowania, niż miałem ja. Właśnie dlatego jestem przekonany, że dasz sobie świetnie radę!

Po co się uczyć programowania, skoro jest AI?

Często to słyszę, jednak polecam samodzielną naukę programowania. Oczywiście, wykorzystanie sztucznej inteligencji znacznie ułatwia pisanie kodu, ale programistom. I to jest bardzo ważne dla Ciebie, zwłaszcza jeśli jesteś dopiero na początku swojej przygody z programowaniem. Pomijam fakt, że kod generowany przez AI może zawierać błędy, być przestarzały lub po prostu nie działać, tak jak tego oczekujesz. Bez wiedzy o składni języka PHP trudno będzie Ci zweryfikować, czy dane rozwiązanie jest optymalne, czy jego użycie nie spowoduje więcej problemów niż korzyści.

W mojej ocenie istotne jest to, że pisanie kodu to nie tylko kopiowanie gotowych fragmentów. Zrozumienie, jak działa kod, pozwala go optymalizować, dostosowywać do konkretnych wymagań i rozwiązywać nietypowe problemy. AI może pomóc, ale nie zastąpi myślenia i kreatywności programisty. Pamiętaj, że sztuczna inteligencja może podpowiadać rozwiązania, ale to Ty decydujesz, czy są one najlepsze. Wiele firm stara się zatrudniać programistów, którzy umieją samodzielnie pracować z kodem i rozumieją jego działanie. Nawet jeśli ChatGPT potrafi napisać kod, ktoś musi nim zarządzać, testować go, optymalizować i utrzymywać w dłuższym okresie.

Samodzielną pracę to inwestycja w siebie, zawsze o tym pamiętaj!

Czy na PHP można dobrze zarabiać?

To pytanie często się pojawia na wielu forach dla programistów i jest stawiane przez osoby dopiero co zaczynające przygodę z kodowaniem. Odpowiedź jest jak zwykle taka sama — „to zależy”.

PHP jest skorelowany z technologią webową i w tym kierunku powinieneś podążać, stawiając właśnie na ten język. Nie będę ukrywał, że znajomość języków HTML5, JavaScript, CSS (w tym Sass) i SQL będzie pomocna przy realizacji własnych projektów, które będziesz mógł dobrze sprzedać. Bardzo często popełniany jest błąd, jakim jest porównywanie wprost zarobków programistów Javy, C#, C++, Pythona i PHP, bez uwzględniania ich środowiska pracy. W mojej ocenie dobry programista znajdzie wiele sposobów na to, aby dobrze zmaterializować swój potencjał, zatrudniając się w nowoczesnej firmie technologicznej czy zakładając własną działalność.

Programista PHP znający dodatkowo języki HTML5, JS, CSS i SQL ma ogromne możliwości na rynku usług związanych z projektowaniem stron internetowych. Możesz

śmiało założyć jednoosobową działalność i szukać klientów wśród wielu milionów firm na całym świecie. Znajomość języka angielskiego z pewnością to ułatwi.

Strony internetowe to ledwie jeden aspekt wykorzystania języka PHP. W związku z coraz częstszym wykorzystywaniem systemów zarządzania treścią (ang. *Content Management System*, CMS) do budowy stron pojawiła się też dodatkowa możliwość zarobkowania, a mianowicie pisanie rozszerzeń (wtyczek) do takich systemów, jak choćby Wordpress, Drupal czy Moodle. Firmy oferujące takie usługi z reguły nie są zbyt duże, często zatrudniają jednego lub dwóch programistów.

A teraz trochę liczb — nowoczesny moduł do WordPressa o nazwie The7⁴, z którego chętnie korzystam, kosztuje 39 dolarów za jedną licencję. Do dzisiaj sprzedano ponad 320 tysięcy sztuk. Przez ostatnie 4 lata nabywców znalazło prawie 100 tysięcy licencji! Kolejny przykład to nakładka do popularnego systemu e-learningowego Moodle o nazwie Lambda⁵. Koszt to 59 dolarów za licencję — do dzisiaj sprzedano prawie 11 tysięcy sztuk, co oznacza wzrost sprzedaży o 3 tysiące licencji w ciągu ostatnich 4 lat. Weź do ręki kalkulator i policz, jakie to są kwoty! I teraz najlepsze! Zrobiwszy raz udaną wtyczkę czy nakładkę, przez lata będziesz mógł ją rozwijać i czerpać zyski z jej sprzedaży.

Niech do rozpoczęcia nauki programowania w PHP przekona Cię również to, że sam zajmuję się zawodowo PHP od ponad 25 lat, czyli od jego początków, i ciągle jest coś nowego i inspirującego do zrealizowania.

Osiągnięcie sukcesu i satysfakcji z pracy jest ściśle związane z nabywanym na przestrzeni wielu lat doświadczeniem. To właśnie zdobyte doświadczenie ułatwi Ci znalezienie świetnej pracy czy założenie własnej firmy. Powodzenia!

Uprzedzając Twoje pytania

Niniejszą książkę starałem się przygotować dla Ciebie najlepiej, jak potrafię. Wszystko po to, aby zawarte w niej informacje były aktualne i, co najważniejsze, praktyczne. Moim celem jest to, aby każdy Czytelnik był zadowolony z zakupu mojej książki i polecił ją innym. Jednak wiem, że każdy ma swoje oczekiwania i nie zawsze są one zaspokajane. Przygotowując materiał do niniejszej książki, analizowałem komentarze i oczekiwania zamieszczone przez Czytelników na forach internetowych.

⁴ Envato Market, <https://themeforest.net/item/the7/5556590> [dostęp: 30.06.2025].

⁵ Envato Market, <https://themeforest.net/item/lambda/9442816> [dostęp: 30.06.2025].

Sprawdziłem również opinie na temat moich poprzednich książek i cieszę się, że większość wypowiedzi jest pozytywna, jednak moją uwagę zawsze przykuwają te, których autorzy opisują występujące problemy. Na część z nich nie mam jako autor wpływu, dlatego postaram się odpowiedzieć na pytania, które mogą się pojawić podczas lektury tej książki:

- *Gdzie są kody źródłowe?* — wszystkie kody powinny być dostępne na stronie internetowej wydawnictwa, a jeśli ich tam nie ma, napisz do mnie.
- *Kod nie działa, i co teraz?* — wszystkie kody zostały sprawdzone, jednak zawsze coś może pójść nie tak, zatem napisz do mnie, a postaram się pomóc.
- *Książka jest za łatwa lub za trudna* — to jest jedno z najtrudniejszych wyzwań dla autora, bo jak napisać książkę dla każdego? W tym miejscu chciałbym wspomnieć o chwili, kiedy wiele lat temu zapoznawałem się z niemiecką instrukcją do monitora. Na całej stronie znajdował się dokładny opis, jak go uruchomić, łącznie z kątem położenia palca wskazującego i siłą nacisku na włącznik. W swojej książce starałem się Ci tego oszczędzić.
- *Podobne informacje i kody można znaleźć w sieci* — oczywiście, że tak! W sieci znajdziesz wiele poradników i listingów, z których śmiało możesz korzystać. Jednak kłopotem zawsze jest odpowiedni dobór źródeł i budowanie dziesiątek zakładek w przeglądarce internetowej. W mojej książce informacje i kody źródłowe zostały dokładnie sprawdzone i przetestowane. Pisząc kody źródłowe, starałem się umieszczać w nich jak najwięcej komentarzy, aby były one dla Ciebie zrozumiałe. Mam nadzieję, że mi się to udało.

Podziękowania

Chciałbym podziękować swojej żonie Idrze za wsparcie i motywację. Dzięki niej praca nad książką była ogromnie satysfakcjonująca.

Chciałbym także podziękować wydawnictwu Helion, a w szczególności Pani Małgorzacie Kulik za świetną i owocną współpracę.

Dziękuję Czytelnikom za cenne uwagi i komentarze dotyczące moich ostatnich książek o programowaniu w języku PHP, wydanych przez wydawnictwo Helion. Wszystkich Czytelników, którzy chcieliby się ze mną skontaktować, zapraszam na moją stronę internetową: <https://duka.pl>.

Bardzo Wam dziękuję.



Co, jeśli? Konstrukcje warunkowe

W języku PHP istnieją funkcje warunkowe, bez których programowanie najprawdopodobniej straciłoby sens. Istotą funkcji warunkowych jest dokonanie wyboru w zależności od spełnienia określonego warunku. Przykładem funkcji warunkowej może być wyświetlenie konkretnej oceny, zależnie od liczby punktów uzyskanych w teście. W konstrukcjach warunkowych będziesz korzystał z operatorów porównania, aby sprawdzić, czy dwie zmienne są ze sobą równe, czy różne, lub która ze zmiennych jest mniejsza bądź większa. Operatory porównania przedstawia poniższa tabela.

Operator	Co sprawdza?
==	czy zmienne są sobie równe co do wartości
!=	czy zmienne są różne co do wartości
===	czy zmienne są identyczne
!==	czy zmienne nie są identyczne
>	czy zmienna z lewej strony jest większa od zmiennej z prawej strony
>=	czy zmienna z lewej strony jest większa od zmiennej z prawej strony bądź jej równa
<	czy zmienna z prawej strony jest większa od zmiennej z lewej strony
<=	czy zmienna z prawej strony jest większa od zmiennej z lewej strony bądź jej równa

Przykładowa konstrukcja warunkowa może mieć postać:

```

01 // kod programu, np. deklaracja zmiennych $a i $b
02 if ($a > $b)
03 { // pierwszy warunek
04     print "a jest większe od b";
05 }
06 elseif ($a < $b)
07 { // drugi warunek
08     print "a jest mniejsze od b";
09 }
10 else

```

```

11 { //żaden warunek nie jest spełniony
12     print "a jest równe b";
13 }
14 // dalszy ciąg programu

```

Jak działa powyższy kod?

1. Sprawdź w linii 02, czy zmienna \$a jest większa od zmiennej \$b; jeśli tak, wykonaj kod w wierszu 04 pomiędzy nawiasami klamrowymi (linie 03 i 05), a następnie opuść konstrukcję warunkową i wykonaj kod, począwszy od wiersza 14.
2. Jeśli pierwszy warunek nie został spełniony, sprawdź kolejny w linii 06, czyli czy zmienna \$a jest mniejsza od zmiennej \$b. Jeśli warunek jest spełniony, wykonaj kod w wierszu 08 pomiędzy nawiasami klamrowymi (linie 07 i 09), po czym opuść konstrukcję warunkową i wykonaj kod, począwszy od wiersza 14.
3. W związku z tym, że żaden z warunków nie został spełniony, wykonaj kod w wierszu 12 pomiędzy nawiasami klamrowymi (linie 11 i 13), a następnie opuść konstrukcję warunkową i wykonaj kod, począwszy od linii 14.

Pamiętaj, że w konstrukcji warunkowej wykonywany jest kod znajdujący się pomiędzy nawiasami klamrowymi { i }.

Ćwiczenie 1. Konstrukcja if

Najprostszą konstrukcją warunkową jest if, która służy do wykonania kodu ujętego w nawiasy klamrowe tylko wtedy, kiedy określony operatorem warunek jest spełniony (listing 10.1).

LISTING 10.1	PHP 7/8	code_10_01.php
01	<?php	
02	/* definicja zmiennych */	
03	\$a = 25;	
04	\$b = 12;	
05		
06	/* konstrukcja warunkowa */	
07	if (\$a > \$b)	
08	{	
09	print "zmienna a jest większa od zmiennej b";	
10	}	
\$ php code_10_01.php ↵		KONSOLA
zmienna a jest większa od zmiennej b		

Jak działa powyższy kod?

1. W liniach 03 i 04 zdefiniowane są zmienne \$a i \$b.
2. W wierszu 07 pojawia się konstrukcja warunkowa if, która sprawdza, czy zmienna \$a jest większa od zmiennej \$b.
3. Jeśli tak, wykonywany jest kod programu zawarty pomiędzy nawiasami klamrowymi { i }, czyli pomiędzy 08 a 10 wierszem.

Ćwiczenie 2. Konstrukcja if-else

Następną konstrukcją warunkową jest if-else, która służy do wykonania odpowiedniego kodu w zależności od tego, czy warunek określony operatorem jest spełniony (listing 10.2).

LISTING 10.2	PHP 7/8	code_10_02.php
01	<?php	
02	/* definicja zmiennych */	
03	\$a = 12;	
04	\$b = 25;	
05		
06	/* konstrukcja warunkowa */	
07	if (\$a > \$b)	
08	{	
09	print "zmienna a jest większa od zmiennej b";	
10	}	
11	else	
12	{	
13	print "zmienna a jest równa lub mniejsza od zmiennej b";	
14	}	
\$ php code_10_02.php ↵		KONSOLA
zmienna a jest równa lub mniejsza od zmiennej b		

Jak działa powyższy kod?

1. W liniach 03 i 04 zdefiniowane są zmienne \$a i \$b.
2. W wierszu 07 pojawia się konstrukcja warunkowa if, która sprawdza, czy zmienna \$a jest większa od zmiennej \$b.
3. Jeśli tak, wykonywany jest kod programu zawarty pomiędzy nawiasami klamrowymi { i }, czyli pomiędzy wierszami 08 a 10.

4. Linia 11 informuje, że w razie niespełnienia warunku określonego w linii 07 zostanie wykonany kod pomiędzy wierszami 12 a 14.

Ćwiczenie 3. Konstrukcja if-elseif-else

Bardziej złożoną konstrukcją warunkową jest if-elseif-else, która wykorzystywana jest do obsługi bardziej rozbudowanych warunków (listing 10.3).

LISTING 10.3	PHP 7/8	code_10_03.php
01	<?php	
02	/* definicja zmiennych */	
03	\$a = 18;	
04	\$b = 18;	
05		
06	/* konstrukcja warunkowa */	
07	if (\$a > \$b)	
08	{	
09	print "zmienna a jest większa od zmiennej b";	
10	}	
11	elseif (\$a == \$b)	
12	{	
13	print "zmienna a jest równa zmiennej b";	
14	}	
15	else	
16	{	
17	print "zmienna a jest mniejsza od zmiennej b";	
18	}	
\$ php code_10_03.php ↵		KONSOLA
zmienna a jest równa zmiennej b		

Jak działa powyższy kod?

1. W liniach 03 i 04 zdefiniowane są zmienne \$a i \$b.
2. W wierszu 07 pojawia się konstrukcja warunkowa if, która sprawdza, czy zmienna \$a jest większa od zmiennej \$b.
3. Jeśli tak, wykonywany jest kod programu zawarty pomiędzy nawiasami klamrowymi { i }, czyli pomiędzy wierszami 08 a 10.
4. W linii 11 pojawia się następna konstrukcja warunkowa, która sprawdza, czy zmienna \$a jest równa zmiennej \$b.

5. Jeśli tak, wykonywany jest kod programu zawarty pomiędzy nawiasami klamrowymi { i }, czyli pomiędzy wierszami 12 a 14.
6. Linia 15 informuje, że w razie niespełnienia warunku określonego w wierszach 07 i 11 zostanie wykonany kod pomiędzy liniami 16 a 18.

Ćwiczenie 4. Operator trójargumentowy

Operator trójargumentowy (?:) wykorzystasz do sprawdzenia prostych warunków (listing 10.4).

LISTING 10.4	PHP 7/8	code_10_04.php
01	<?php	
02	\$wiek = 34;	
03		
04	// przypisanie zmiennej określonej wartości w zależności od spełnienia warunku	
05	\$status = (\$wiek >= 18) ? 'jest' : 'nie jest';	
06		
07	print "Mateusz \$status pełnoletni.";	
\$ php code_10_04.php ↵		KONSOLA
Mateusz jest pełnoletni.		

Jak działa powyższy kod?

1. W linii 02 zdefiniowana jest zmienna \$wiek.
2. W wierszu 05 do sprawdzenia warunku, czy zmienna \$wiek jest większa bądź równa 18, został użyty trójargumentowy operator warunkowy ?.

Można oczywiście wykorzystać operator warunkowy if; wówczas kod w linii 05 będzie wyglądał tak:

```
if ($wiek >= 18)
{
    $status = 'jest';
}
else
{
    $status = 'nie jest';
}
```

Oczywiście, obie metody są poprawne, choć najczęściej korzystam z operatora trójargumentowego dla prostych warunków.

Ćwiczenia dodatkowe

1. W każdym ćwiczeniu wstaw różne wartości zmiennych i przetestuj działanie programu.

Wskazówki

Zawsze pamiętaj, że operator porównania to dwa znaki równości `==`, a nie jeden. Początkujący programiści bardzo często popełniają ten błąd.

Operator porównania `===` (trzy znaki równości) służy do sprawdzenia, czy zmienne są równe co do wartości i typu. W następnych lekcjach dowiesz się, jak reprezentowane są liczby całkowite i rzeczywiste w języku PHP.

PROGRAM PARTNERSKI

— GRUPY HELION —

- 
1. ZAREJESTRUJ SIĘ
 2. PREZENTUJ KSIĄŻKI
 3. ZBIERAJ PROWIZJĘ

Zmień swoją stronę WWW w działający bankomat!

Dowiedz się więcej i dołącz już dzisiaj!

<http://program-partnerski.helion.pl>

GRUPA
Helion

50 razy PHP

Książka, po którą właśnie sięgasz, została napisana w jednym celu: by pomóc Ci nauczyć się programowania w jednym z najpopularniejszych języków programowania na świecie. PHP — bo o nim mowa — wraz z przypisaną do niego bazą danych SQL jest używany zarówno do projektowania stron internetowych, jak i do budowy skomplikowanych aplikacji działających po stronie serwera. Szacuje się, że 8 na 10 stron internetowych jest opartych na serwerze obsługującym język PHP.

PHP od zawsze był dynamicznie rozwijany. W jego kolejnych wersjach wprowadzane są nowe funkcje, optymalizacje i ulepszenia, które czynią ten język jeszcze bardziej wszechstronnym. Wraz z premierą wersji sygnowanej numerem 8 programiści uzyskali dostęp do narzędzi, które nie tylko ułatwiają pracę, ale także pozwalają pisać czytelniejszy, bezpieczniejszy i wydajniejszy kod. To właśnie na tej najnowszej wersji języka PHP skupiamy się w tym podręczniku.

W ramach 50 lekcji między innymi:

- Zapoznasz się z językiem PHP, bazą danych SQL i ze środowiskiem pracy
- Opanujesz podstawy programowania w PHP i konwencje stosowane w kodzie
- Przyswoisz takie pojęcia jak stałe i łańcuchy znakowe
- Poznasz instrukcje sterujące, a także sposoby ich używania
- Będziesz pracować na tekstach, używać operatorów i funkcji matematycznych
- Zastosujesz po raz pierwszy funkcję, pętlę, sięgniesz po tablice i pliki
- Dowiesz się, czym jest programowanie zorientowane obiektowo
- Skorzystasz z bazy danych SQL i kodu HTML
- Przetestujesz swój kod w PHPUnit
- Popracujesz ze wzorcem architektonicznym MVC
- Zaznajomisz się z nowościami wprowadzonymi w najnowszej wersji PHP 8

Helion 	KOD KORZYŚCI Sięgnij po więcej! ▶ 
 helion.pl  HELION S.A. ul. Kościuszki 1c 44-100 Gliwice tel.: 32 230 98 63 helion@helion.pl	ISBN 978-83-289-2953-1  9 788328 929531
Cena: 89,00 zł	

