

Co to jest PHP? Otóż PHP jest systemem oraz językiem programowania przeznaczonym do tworzenia i generowania dynamicznych stron WWW, coraz częściej nazywanych po prostu aplikacjami WWW. Przyjmuje się, że nazwa PHP to akronim rekursywny — PHP: Hypertext Preprocessor, choć pewne źródła podają, że pierwotnie był to skrót od jednej z pierwszych nazw tego narzędzia — Personal HomePage Tools. Rdzeniem PHP jest silnik Zend, wykonujący i nadzorujący wszystkie najistotniejsze operacje związane z przetwarzaniem kodu zawartego w skryptach PHP. PHP integruje się z określonym serwerem WWW (w tym kursie będzie to najczęściej popularny serwer Apache) jako moduł serwera lub skrypt CGI. Kod skryptu, na podstawie którego ma być wygenerowana strona WWW, osadzony jest najczęściej wewnątrz tej strony.

Tradycja nakazuje, aby szablony stron WWW z osadzonym kodem PHP nazywać skryptami, więc w tym kursie też będziemy stosować to określenie.

Kiedy klient (przeglądarka) zażąda od serwera strony, która ma zostać wygenerowana na podstawie skryptu PHP, serwer pobiera z dysku (lub pamięci) plik zawierający żądany skrypt i przekazuje go interpreterowi PHP (interpreter zwany jest także czasem aparatem wykonawczym). Interpreter PHP odnajduje wewnątrz skryptu kod PHP i stara się go wykonać. Jeżeli kod nie zawiera żadnego błędu i interpreter jest w stanie go wykonać bez żadnych problemów, serwer WWW otrzymuje z powrotem przygotowany kod — tym razem w języku (X)HTML, gotowy do wysłania do przeglądarki. Proces ten pokazany został na rysunku 2.1.

Rysunek 2.1. Przetwarzanie skryptów PHP (1 — żądanie strony WWW generowanej dynamicznie, 2, 3 — pobranie pliku strony z dysku, 4 — jeżeli plik zawiera skrypt PHP, serwer przekazuje go interpreterowi PHP, 5 — interpreter, po wykonaniu kodu PHP, przekazuje serwerowi wygenerowaną stronę WWW, 6 — serwer przekazuje klientowi gotową stronę WWW)

Interpreter PHP może być instalowany jako moduł serwera lub jako skrypt CGI. W przypadku instalacji jako moduł serwera trzeba mieć na uwadze przede wszystkim wersję serwera — np. osobny moduł jest dla Apache'a w wersji 1.3, a inny dla wersji 2.0 czy 2.2. PHP jako moduł serwera gwarantuje szybkie i bezpieczne wykonywanie kodu strony. W przypadku PHP jako skryptu CGI mamy co prawda większą uniwersalność i niezależność w stosunku do używanego serwera WWW, ale też większe ryzyko ze względu na bezpieczeństwo systemu i danych. Obecnie nie zaleca się używania interpretera PHP jako skryptu CGI.

Początkowo PHP przeznaczony był do przetwarzania formularzy HTML, jednak szybko ewoluował i stał się bardzo wydajnym i wszechstronnym narzędziem tworzenia aplikacji WWW (choć obecnie tak naprawdę jest to język uniwersalny i może być używany do rozmaitych zastosowań, niekoniecznie związanych z WWW).

PHP — jak się okazuje — to nie tylko język i środowisko do tworzenia aplikacji WWW. Możemy się o tym przekonać, jeśli zwrócimy uwagę na projekt PHP-GTK (<http://gtk.php.net/>) mający na celu rozwój środowiska programowania samodzielnych, niezależnych od systemu aplikacji z graficznym interfejsem użytkownika.

Spośród wielu cech, które powodują, że popularność PHP ciągle wzrasta, można wymienić przede wszystkim obsługę wymiany danych z różnymi systemami baz danych, dynamiczne generowanie grafiki, dokumentów PDF czy animacji Flash, bardzo dobrą obsługę protokołu HTTP i prostotę tworzenia i wdrażania aplikacji WWW. Jako język programowania PHP posiada bardzo prostą składnię — szczególnie dla tych wszystkich, którzy programowali już np. w C czy w Perlu. Dodatkowo w wersji 5. pojawiła się możliwość tworzenia prawdziwie obiektowych aplikacji, nadal jednak programista może pisać klasyczne skrypty PHP — obiektowość nie jest narzucona. Jak już wspomniano wcześniej, PHP jest projektem otwartym, co oznacza, że aby go używać, wystarczy pobrać źródła lub binaria (wersje skompilowane dla danej platformy sprzętowo-systemowej) ze strony <http://www.php.net>, zainstalować (jeżeli jeszcze tego nie zrobiliśmy) serwer WWW — np. Apache — i już możemy cieszyć się z platformy do tworzenia aplikacji WWW.