

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

Adobe Dreamweaver CS3 z ASP, ColdFusion i PHP. Oficjalny podręcznik

Autor: Jeffrey Bardzell, Bob Flynn

Tłumaczenie: Marcin Rogóż

ISBN: 978-83-246-1509-4

Tytuł oryginału: [Adobe Dreamweaver CS3 with ASP, ColdFusion, and PHP: Training from the Source](#)

Format: 170x230, stron: 472



Wiedza o Adobe Dreamweaver CS3 prosto ze źródła

- Jaką technologię skryptową wybrać?
- W jaki sposób wykorzystywać skrypty z biblioteki Dreamweavera?
- Jak zbudować interfejs użytkownika w technologii AJAX?

Adobe Dreamweaver to narzędzie, które od wielu lat cieszy się uznaniem projektantów witryn internetowych. Doskonałe połączenie projektowania wizualnego z edycją kodu źródłowego spełnia oczekiwania zarówno początkujących, jak i zaawansowanych użytkowników. Najnowszy Dreamweaver, oznaczony symbolem CS3, to nie tylko edytor kodu HTML – to zintegrowane środowisko programistyczne, pozwalające na tworzenie dynamicznych witryn WWW z wykorzystaniem trzech technologii skryptowych – PHP, ASP i ColdFusion. Użytkownicy mają do dyspozycji bibliotekę gotowych skryptów oraz wiele narzędzi usprawniających i przyspieszających proces kreowania witryny.

„Adobe Dreamweaver CS3 z ASP, ColdFusion i PHP. Oficjalny podręcznik” to podręcznik przedstawiający zasady tworzenia dynamicznych witryn WWW, przygotowany przez instruktorów posiadających certyfikaty firmy Adobe. W kolejnych rozdziałach znajdziesz ćwiczenia prowadzące przez proces budowania dynamicznej witryny WWW z wykorzystaniem różnych technologii skryptowych. Nauczysz się tworzyć statyczne elementy witryny, mechanizmy nawigacji i weryfikacji danych. Przeczytasz o przesyłaniu danych pomiędzy stronami, komunikacji z bazami danych, uwierzytelnianiu użytkowników i tworzeniu interfejsów wykorzystujących AJAX. Znajdziesz również opis tworzenia modułów zarządzania treścią witryny.

- Przygotowanie struktury witryny i arkuszy stylów
- Wybór technologii i konfiguracja środowiska roboczego
- Przesyłanie danych pomiędzy stronami
- Wysyłanie poczty elektronicznej z formularzy na stronach WWW
- Przetwarzanie danych z formularzy
- Połączenia z bazami danych
- Budowanie interfejsu użytkownika z wykorzystaniem technologii AJAX
- Moduły wyszukiwawcze
- Uwierzytelnianie użytkowników
- Mechanizm zarządzania treścią witryny

Poznaj niezwykle narzędzie dla nowoczesnego webmastera!

Wydawnictwo Helion
ul. Kościuszki 1c
44-100 Gliwice
tel. 032 230 98 63
e-mail: helion@helion.pl



Spis treści

Wstęp	11
Wymagania	11
Lekcje	12
Projekt witryny	13
Struktura książki	13
W razie problemów	14
O tym wydaniu	15
Oficjalny podręcznik	16
Autoryzowane szkolenia i certyfikaty	17
Czego się nauczysz?	17
Minimalne wymagania systemowe — Windows	18
Minimalne wymagania systemowe — Macintosh	19
Lekcja 1. Przedstawiamy Newland Tours	21
Czego się nauczysz?	21
Przewidywany czas	21
Pliki lekcji	21
Definiowanie statycznej witryny	23
Tworzenie strony „Kontakt z nami”	29
Formatowanie strony „Kontakt z nami”	32
Ocena witryny. Kod	39
Ocena witryny. Procesy biznesowe	43
Widoki na przyszłość	49
Czego się nauczyłeś?	51
Lekcja 2. Przygotowanie statycznych stron HTML i CSS	53
Czego się nauczysz?	53
Przewidywany czas	53
Pliki lekcji	53
Automatyzowanie zmian za pomocą funkcji Find and Replace	55

Przedefiniowanie wyglądu elementów HTML za pomocą CSS	59
Tworzenie i stosowanie własnych stylów CSS	67
Tworzenie szablonu do wielokrotnego użytku	78
Zwiększanie dostępności za pomocą niewidocznej nawigacji	80
Czego się nauczyłeś?	84

Lekcja 3. Dynamiczne witryny WWW 85

Czego się nauczysz?	85
Przewidywany czas	85
Pliki lekcji	85
Podstawy dynamicznych witryn WWW	86
Wybór technologii serwerowych	90
Zmiana witryny Newland Tours na witrynę dynamiczną	93
Praca na serwerze lokalnym	94
Konfigurowanie środowiska lokalnego dla IIS/ASP	94
Konfigurowanie środowiska lokalnego dla ColdFusion	96
Konfigurowanie środowiska lokalnego dla Apache i PHP	98
Praca na serwerze zdalnym	104
Definiowanie witryny dynamicznej w programie Dreamweaver (wszyscy użytkownicy)	105
Budujemy prostą aplikację dynamiczną	110
Czego się nauczyłeś?	120

Lekcja 4. Przesyłanie danych między stronami 121

Czego się nauczysz?	121
Przewidywany czas	121
Pliki lekcji	121
Poznajemy protokół HTTP	123
Pobieranie danych z adresu URL	125
Wysyłanie danych za pomocą odnośników	131
Umieszczanie i odczytywanie cookies	135
Czego się nauczyłeś?	141

Lekcja 5. Wysyłanie wiadomości e-mail z formularza WWW 143

Czego się nauczysz?	143
Przewidywany czas	143
Pliki lekcji	143
Wprowadzenie do usługi pocztowej SMTP	145
Konfigurowanie systemu w celu wysyłania wiadomości e-mail przez SMTP	145
Konfigurowanie serwera IIS w celu wysyłania wiadomości e-mail (użytkownicy ASP)	146

Konfigurowanie ColdFusion w celu wysyłania wiadomości e-mail	148
Konfigurowanie PHP w celu wysyłania wiadomości e-mail	150
Pisanie kodu wysyłającego własności	153
Obiekty, metody i właściwości	153
Tworzenie formularzy WWW	159
Wysyłanie wartości formularza przez e-mail	163
Walidacja formularza po stronie klienta	165
Czego się nauczyłeś?	168
Lekcja 6. Budujemy kalkulator cen wycieczek	171
Czego się nauczysz?	171
Przewidywany czas	171
Pliki lekcji	171
Tworzenie stron	173
Budujemy formularz	175
Zbieranie, przetwarzanie i wyświetlanie danych	181
Walidacja po stronie serwera	187
Tworzenie obszarów warunkowych	191
Tworzenie i zastosowanie własnej klasy CSS	193
Czego się nauczyłeś?	195
Lekcja 7. Bazy danych w WWW	197
Czego się nauczysz?	197
Pliki lekcji	197
Przyspieszony kurs obsługi baz danych	198
Wprowadzenie do obiektów związanych z bazami danych	199
Relacje	201
Bazy danych w sieci WWW	203
Bezpieczeństwo baz danych a WWW	206
Instalacja bazy danych Newland Tours w MySQL	206
Przegląd bazy danych Newland	211
Nawiązywanie połączenia z bazą danych	215
Pobieranie zestawu rekordów i wyświetlanie informacji z bazy danych	224
Czego się nauczyłeś?	229
Lekcja 8. Kalkulator cen — ciąg dalszy	231
Czego się nauczysz?	231
Przewidywany czas	231

Pliki lekcji	231
Menu rozwijane wypełniane dynamicznie	232
Filtrowanie zestawów rekordów	237
Skrypt korzystający z prawdziwych danych	240
Dokumentowanie kodu za pomocą komentarzy	245
Czego się nauczyłeś?	248

Lekcja 9. Filtrowanie i wyświetlanie danych z wykorzystaniem techniki AJAX249

Czego się nauczysz?	249
Przewidywany czas	249
Pliki lekcji	249
Przesyłanie danych z użyciem techniki AJAX	251
Rozbudowa Dreamweavera za pomocą rozszerzeń	252
Przekształcanie zestawu rekordów w XML	254
Łączenie strony opisów z danymi XML	256
Tworzenie tabeli Spry	261
Tworzenie obszarów Spry	266
Wyświetlanie danych z użyciem Spry Accordion	268
Dodawanie obrazów i nazwy kraju	271
Pobieranie danych z tabel zewnętrznych przy użyciu SQL	274
Dostosowywanie wyglądu strony za pomocą formatowania i pozycjonowania CSS	278
Czego się nauczyłeś?	284

Lekcja 10. Tworzenie strony zawierającej opisy wyjazdów285

Czego się nauczysz?	285
Przewidywany czas	285
Pliki lekcji	285
Planowanie aplikacji	287
Tworzenie zestawów rekordów za pomocą operatora JOIN	290
Tworzenie opisów wyjazdów	295
Wstawianie obrazów i atrybutów alt	300
Implementowanie stronicowania zestawu rekordów	302
Przesyłanie danych do innych aplikacji	308
Czego się nauczyłeś?	317

Lekcja 11. Budowanie interfejsów wyszukiwania319

Czego się nauczysz?	319
Przewidywany czas	319
Pliki lekcji	319

Przygotowywanie strony wyszukiwania i tworzenie odnośnika „Wyszukaj wszystkie”	321
Wyszukiwanie według rejonu świata — tworzenie interfejsów	324
Poprawianie zapytania SQL oraz używanie komentarzy do kodu w celu testowania kodu i usuwania błędów	330
Kontrolowanie wyświetlania paska nawigacji	334
Wyszukiwanie według kraju — filtrowanie względem wartości pola formularza	338
Wybór zapytania SQL w zależności od zmiennych środowiskowych	344
Czego się nauczyłeś?	348
Lekcja 12. Uwierzytelnianie użytkowników	349
Czego się nauczysz?	349
Przewidywany czas	349
Pliki lekcji	349
Uwierzytelnianie użytkowników jako aplikacja sieci WWW	351
Tworzenie strony rejestracji	354
Tworzenie strony logowania	360
Tworzenie Application.cfc (tylko ColdFusion)	365
Ograniczanie dostępu do stron	367
Czego się nauczyłeś?	369
Lekcja 13. Zarządzanie treścią za pomocą formularzy	371
Czego się nauczysz?	371
Przewidywany czas	371
Pliki lekcji	371
Tworzenie części administracyjnej	374
Formatowanie zawartości przechowywanej w bazie danych	377
Tworzenie formularza	378
Zmienne sesji i ukryte pola wykorzystywane do śledzenia użytkowników	382
Wstawianie rekordu	383
Czego się nauczyłeś?	386
Lekcja 14. Tworzenie stron aktualizujących dane	387
Czego się nauczysz?	387
Przewidywany czas	387
Pliki lekcji	387
Przygotowywanie stron	389
Planowanie stron Master-Detail	390
Wprowadzenie aktualizacji na stronie szczegółowej	398
Czego się nauczyłeś?	407

Lekcja 15. Kodowanie prostego systemu zarządzania treścią (CMS)	409
Czego się nauczysz?	409
Przewidywany czas	409
Pliki lekcji	409
Przygotowywanie systemu zarządzania treścią (CMS)	411
Tworzenie formularza i wpisywanie kodu zestawu rekordów	415
Umożliwienie wstawiania rekordów	424
Tworzenie strony ogólnej	429
Tworzenie strony szczegółowej do aktualizowania rekordów	433
Dodawanie funkcji aktualizującej	442
Usuwanie rekordów	445
Aktualizacja odnośników na stronie administracyjnej	447
Co dalej?	447
Czego się nauczyłeś?	448
Skorowidz	449

3 Dynamiczne witryny WWW

Czego się nauczysz?

W czasie tej lekcji:

- poznasz podstawowe pojęcia związane z witrynami dynamicznymi,
- wybierzesz rodzaj serwera (ASP, Adobe ColdFusion lub PHP),
- skonfigurujesz komputer w celu uruchomienia serwera WWW współpracującego z wybranym rodzajem serwera aplikacyjnego (opcjonalnie),
- zmienisz konfigurację witryny Newland Tours tak, by opisywała ona witrynę dynamiczną,
- zbudujesz prostą aplikację dynamiczną.

Przewidywany czas

Lekcja zajmie około 60 minut.

Pliki lekcji

Pliki wyjściowe:

Lekcja03/Start/newland/about.html

Lekcja03/Start/newland/contact.html

Lekcja03/Start/newland/css/main.css

Lekcja03/Start/newland/generic_template.html

Lekcja03/Start/newland/index.html

Lekcja03/Start/newland/profiles.html

Lekcja03/Start/newland/tours.html

Pliki końcowe:*Lekcja03/Complete/newland-asp/about.asp**Lekcja03/Complete/newland-asp/contact.asp**Lekcja03/Complete/newland-asp/css/main.css**Lekcja03/Complete/newland-asp/generic_template.asp**Lekcja03/Complete/newland-asp/index.asp**Lekcja03/Complete/newland-asp/profiles.asp**Lekcja03/Complete/newland-asp/test_form.asp**Lekcja03/Complete/newland-asp/test_form_processor.asp**Lekcja03/Complete/newland-asp/tours.html*

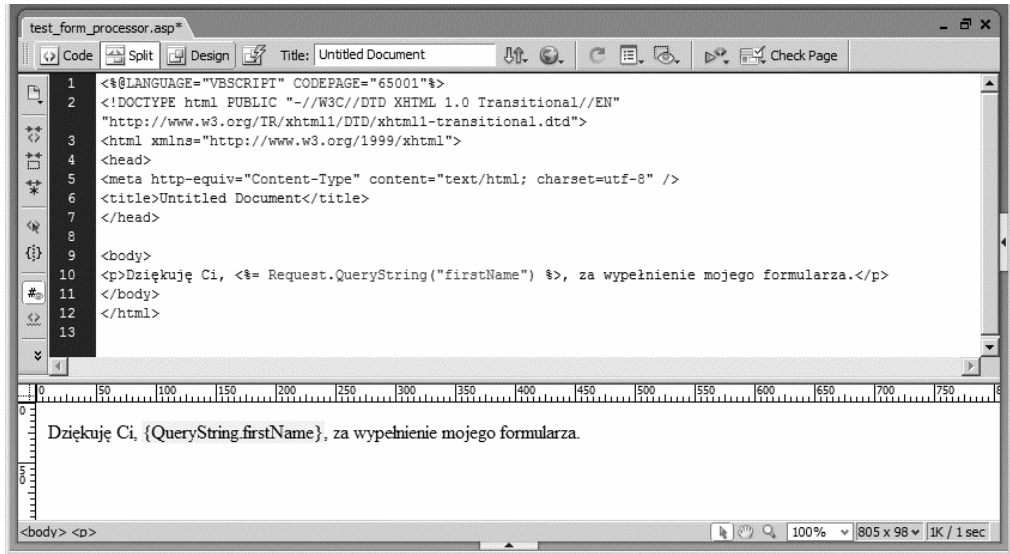
Jeśli używasz ColdFusion lub PHP i chcesz mieć dostęp do plików końcowych, skorzystaj z lustrzanego folderu (newland_cf lub newland_php) na załączonej płycie CD. Wszystkie pliki mają te same nazwy, zmieniono jedynie ich rozszerzenia (.cfm lub .php zamiast .asp).

Dokonałiśmy znacznego postępu w procesie poprawiania witryny Newland Tours. Utworzyliśmy znaczniki i opatrzyliśmy nimi nową stronę, utworzyliśmy niestandardowe style dla całej witryny, zbudowaliśmy szablon oraz poprawiliśmy dostępność witryny. Gdy oglądaliśmy witrynę w przeglądarce, może się wydawać, że te dwie zakończone lekcje nie były warte włożonego wysiłku. Ale wiemy, że zaszły tu ważniejsze zmiany: stworzyliśmy podstawę witryny, która będzie zgodna ze standardami i łatwa z utrzymaniu, w dodatku za pomocą technologii, która nie wyjdzie tak szybko z użycia.

Rozpoczynając tę lekcję, odrzucimy (w przeważającej części) tradycyjny model tworzenia statycznych witryn WWW i przejdziemy do tworzenia interaktywnych witryn działających w oparciu o bazy danych. Jednak zanim się tym zajmiemy, musimy poznać pewne zasadnicze pojęcia oraz kwestie natury technicznej. W tej lekcji poznamy zasady działania witryn dynamicznych i ich przeznaczenie, skonfigurujemy program Adobe Dreamweaver w celu tworzenia stron pobierających dane oraz utworzymy naszą pierwszą stronę korzystającą ze zmieniających się treści.

Podstawy dynamicznych witryn WWW

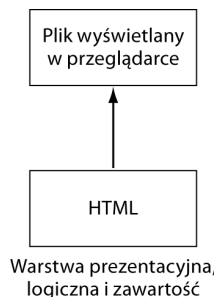
W poprzednich lekcjach poznaliśmy kilka pojęć o decydującym znaczeniu w tworzeniu dynamicznych witryn WWW. Jednym z nich jest oddzielenie warstwy logicznej od warstwy prezentacyjnej. Warstwa logiczna witryny na tym etapie obsługiwana jest wyłącznie przez XHTML, podczas gdy kaskadowe arkusze stylów (CSS) odpowiadają za prezentację. Zapoznaliśmy się też z łączeniem „w locie” dwóch różnych dokumentów (strony XHTML i CSS) w celu utworzenia czegoś odmiennego od tych dwóch dokumentów z osobną. Są to podstawowe pojęcia związane z tworzeniem dynamicznych stron WWW.



Tworzenie dynamicznych stron WWW wiąże się często z łączeniem zwykłego tekstu z miejscami na zmienne

Aby zrozumieć zachodzące tu interakcje oraz przygotować się do czekających nas zadań, poświęćmy chwilę na przeanalizowanie wzajemnych relacji między trzema głównymi źródłami informacji, składającymi się na każdą stronę WWW, czyli zawartością (tekst, rysunki itd.), warstwą logiczną (hierarchia dokumentu, np. nagłówki i zasadnicza treść dokumentu) oraz warstwą prezentacyjną (kolory, wielkość czcionek, położenie i inne efekty „kosmetyczne”).

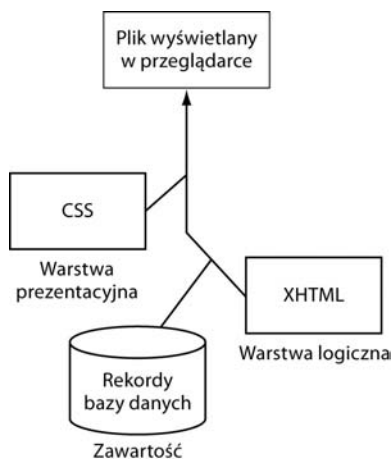
We wcześniejszych wersjach języka HTML tekst, znacznikowanie i kod prezentacyjny występują razem w jednym miejscu — w samym dokumencie HTML. Tym samym dokument znajdujący się na dysku twardym autora strony jest taki sam jak dokument wyświetlany w przeglądarce u osoby odwiedzającej witrynę. Ten prosty związek przedstawiony został na poniższym rysunku.



W wyniku zmian, jakie przeprowadziliśmy w lekcji 2., związek ten uległ zmianie: oddzieliliśmy wygląd dokumentu od jego warstwy logicznej i zawartości. Informacje o wyglądzie strony przechowywane są teraz w arkuszu CSS. Zawartość dokumentu znajduje się wewnątrz oznakowań XHTML, które tworzą również warstwę logiczną dokumentu. „Prawdziwa” strona powstaje dopiero po połączeniu dokumentów XHTML i CSS. Ten nowy związek przedstawiony został na poniższym rysunku.



W tej i kolejnych lekcjach uzupełnimy ten schemat o następną warstwę. Będzie to zmiana jeszcze bardziej zasadnicza niż przejście od HTML do XHTML i CSS. Dokładniej mówiąc, oddzielimy zawartość od warstwy logicznej przez dodanie do witryny zawartości z bazy danych. Spowoduje to, że wszystkie te trzy poziomy — warstwa prezentacyjna, warstwa logiczna oraz zawartość — staną się w dużym stopniu niezależne od siebie, co sprawi, że będziemy mogli dokonać radykalnych zmian w każdej z tych warstw bez konieczności wprowadzania zmian w pozostałych. Taka relacja — i oparta na niej podstawowa strategia zastosowana w pozostałej części książki — przedstawiona została na poniższym rysunku.



HTML nie potrafi oddzielić zawartości od warstwy logicznej dokumentu. Nawet w swojej 5. wersji (XHTML 1.0) HTML jest w zasadzie przeznaczony do znacznikowania zwykłego tekstu dokumentu. Nie może on przetwarzać skryptów, obliczać wyrażeń, wykonywać działań matematycznych, wchodzić w interakcję z bazą danych czy wysyłać informacji e-mailem do użytkownika. A oddzielenie warstwy logicznej od zawartości wymaga czasem każdej z tych operacji (i nie tylko). Aby wykonać te zadania, musimy nieco wspomóc HTML. I tu przychodzą z pomocą technologie serwerowe, takie jak Microsoft ASP, Adobe ColdFusion i PHP.



Język arkuszy stylów (przypominający CSS „na sterydach”) XSLT (ang. *eXtensible Stylesheet Language for Transformation*) może wykonywać pętle, przeprowadzać konwersję danych i wiele więcej. Jedną z jego najbardziej przydatnych cech jest zdolność do przekonwertowania dokumentu XML z jednego formatu (np. zastrzeżonego języka XML do użytku wewnętrznego) na inny format dokumentu XML (np. standardowy dokument XML, który można dzielić z innymi) i odwrotnie. Pomimo swojej przydatności XSLT nie jest tak silny i elastyczny, jak aplikacje po stronie serwera zbudowane przy użyciu ASP, ColdFusion czy PHP.

Technologie serwerowe, takie jak ASP, ColdFusion czy PHP (są też inne, na przykład JSP i ASP.NET), mogą wykonać takie zadania programistyczne, jak obliczanie wyrażeń, wykonywanie działań matematycznych i przetwarzanie danych pochodzących z różnych źródeł, włącznie z bazami danych, strukturalnymi plikami tekstowymi, a w niektórych przypadkach z danymi XML. Mają one również szczególne funkcje, dotyczące wyłącznie WWW, takie jak zdolność do zbierania danych wysyłanych przez użytkownika oraz kontrolowanie wysyłanych do niego informacji zwrotnych.

Ale tkwi w tym pewien haczyk. Przeglądarki są ograniczone do obsługi HTML, CSS i JavaScript — nie „rozumieją” skryptów serwera (mówiąc „skrypty serwera”, mam tu na myśli kod napisany w ASP, ColdFusion, PHP itd.). Wszystko, co serwer przesyła do przeglądarki, musi być zapisane w standardowym formacie HTML. Przed wysłaniem danych do przeglądarki wszystkie skrypty serwera muszą zostać uruchomione na serwerze i dać na wyjściu standardowy HTML.

Mówiąc prościej, aby zobaczyć stronę zawierającą zmieniające się dane, musimy ją uruchomić na serwerze zdolnym do przetworzenia kodu. Różni się to od standardowych stron HTML, które można wyświetlić bezpośrednio w przeglądarce, niezależnie od tego, czy są one wczytywane z serwera, czy też np. z lokalnego dysku. Gdy w przeglądarce Internet Explorer lub Netscape otworzymy dowolną stronę HTML z folderu *Lekcja03/Start*, zostanie ona wyświetlona tak, jak tego oczekiwaliśmy. Jeśli jednak przejdziemy do stron z folderu *Lekcja03/Complete*, odkryjemy, że strony się nie otwierają (lub otwierają się w programie Dreamweaver, zamiast w przeglądarce). Przeglądarka napotyka kod, którego nie „rozumie” i nie może otworzyć pliku. Z tego powodu w lekcji 1. oglądaliśmy końcową wersję witryny, znajdującą się pod adresem <http://www.bobflynn.info>, zamiast na płycie CD.

Witryny umieszczamy na serwerach internetowych. By jednak były one dostępne, musi być na nich uruchomione specjalne oprogramowanie: serwer WWW (czasem używa się też nazwy „serwer HTTP”). Najpopularniejsze serwery WWW to Apache, działający w systemach Unix (Linux), Mac OS X i Windows oraz stworzony przez Microsoft IIS (*Internet Information Services*), stosowany na serwerach internetowych działających pod kontrolą systemu Windows.

Oprócz serwera WWW prawdopodobnie będziemy też korzystać z innych serwerów dostarczających dane. Możemy użyć serwera baz danych, np. MySQL lub Microsoft SQL Server. Możemy również skorzystać z serwera aplikacji, który przetworzy skrypty serwera. Takim serwerem aplikacji jest ColdFusion. Serwer aplikacji przetwarzający skrypty ASP wbudowany jest w IIS, można więc powiedzieć, że IIS stanowi połączenie serwera WWW i serwera aplikacji. PHP jest serwerem aplikacji uruchamianym jako moduł serwera Apache.

Wybór technologii serwerowych

Jak już wiemy, istnieje kilka języków powszechnie stosowanych po stronie serwera. Nasuwa to pytanie (często stawiane przez osoby początkujące w tworzeniu witryn dynamicznych), jakiej technologii serwera należy użyć. Poniższa lista zawiera podsumowanie podstawowych funkcji oraz wad i zalet każdego z nich.

Active Server Pages (ASP): ASP to technologia Microsoftu wiążąca z sobą jego serwery IIS (*Internet Information Services*) dla Windows 2000 i XP z VBScript (*Visual Basic Script*) w celu tworzenia dynamicznych witryn WWW (możemy też użyć JScriptu — stworzonej przez Microsoft implementacji języka JavaScript). ASP jest darmowy i został wbudowany we wszystkie serwery IIS, co oznacza, że po dokonaniu niewielkiej konfiguracji większość użytkowników może tworzyć witryny ASP, nie ponosząc przy tym dodatkowych kosztów. Jednak dla osób, które mają niewielkie doświadczenie w programowaniu, język używany do programowania (VBScript) może wydawać się nieco zniechęcający. ASP jest aktualnie zastępowany bardzo promowanym przez Microsoft ASP.NET (patrz niżej).

ColdFusion: ColdFusion jest technologią serwerową oferowaną przez Adobe. Jego oparta na znacznikach składnia jest o wiele łatwiejsza w użyciu niż VBScript i wymaga wpisania mniejszej ilości kodu. Większość projektantów uważa tę technologię za najbardziej przystępną. ColdFusion jest ponadto potężnym językiem, który sprawia, że tworzenie dynamicznych witryn przebiega bardzo szybko. Ujemną stroną ColdFusion jest to, że nie jest on darmowy, choć zazwyczaj zwiększenie produktywności powoduje, że inwestycja szybko się zwraca. ColdFusion jest również wyjątkowo łatwy do skonfigurowania. W okresie, gdy pisałem tę książkę, nad silnikami ColdFusion pracowali również inni producenci, udostępniający wersje darmowe o otwartym kodzie źródłowym.

PHP: Skrót ten jest akronimem rekurencyjnym angielskiej nazwy *PHP Hypertext Preprocessor*. PHP to szybko rozwijająca się technologia serwerowa. Składa się na to wiele powodów. Jest to darmowe rozwiązanie open source, doskonale współpracujące z innymi znakomitymi produktami open source, w tym z serwerem WWW Apache i systemem zarządzania bazą danych MySQL. Pod względem trudności składnia języka PHP jest porównywalna ze składnią ASP — może nieco prostsza. W ostatnio wydanej wersji PHP 5 język ten został rozszerzony o bardziej obiektowe podejście oraz wykorzystanie XML, jednak podstawowe funkcjonalności i składnia są podobne do PHP 4. Dla naszych celów wystarczy któraś z tych wersji.

ASP.NET: ASP.NET, część platformy .NET odpowiadająca za obsługę WWW, to wszechstronna technologia, język o wielkich możliwościach, pozwalający szybko tworzyć projekty internetowe. Podobnie jak jego poprzednik, czyli ASP, daje się uruchomić na dowolnym serwerze IIS w Windows 2000 i późniejszych. Jednakże pod względem pojęciowym oraz architektury ASP.NET różni się od klasycznego ASP, ColdFusion czy PHP. Bez względu na to, czy znasz tylko HTML, czy też masz doświadczenie z JavaScript lub nawet ASP, efektywna praca z ASP.NET będzie wymagać przystosowania. ASP.NET wspiera różne języki, dwa najbardziej rozpowszechnione z nich to VisualBasic.NET oraz C#.

JSP (*Java Servlet Pages*): JSP to rozwiązanie oparte na języku Java, przeznaczone do tworzenia dynamicznych witryn WWW, wymagające do interpretacji kodu serwera Java (np. serwera J2EE). JSP jest szybki w działaniu i zapewnia imponujący czas reakcji. Jest to także narzędzie wyjątkowo potężne —

z pewnością najpotężniejsze do czasu pojawienia się .NET i z pewnością wystarczająco potężne, by bezpośrednio z .NET rywalizować. Jednak i w tym przypadku składnia tego języka może okazać się zniechęcająca dla osób, które dopiero rozpoczynają tworzenie dynamicznych witryn WWW.

Książka ta zawiera omówienie klasycznego ASP (zwanego dalej po prostu ASP), ColdFusion i PHP, jednakże nie jest poświęcona specjalnie ASP, ColdFusion czy PHP. Jest ona przeznaczona dla Czytelników będących nowicjuszami w zakresie pojęć i praktyki budowania bazodanowych, dynamicznych witryn WWW przy użyciu programu Dreamweaver CS3. Podczas lektury poznasz wiele zagadnień dotyczących kodu i pojęć związanych z pisanem programów, nauczysz się też wykorzystywać działania serwera do przyspieszania i ułatwiania sobie pracy. Będziesz miał dobre rozeznanie w możliwościach programu Dreamweaver, dowiesz się, jak wykorzystać połączenie kilku różnych technologii w celu tworzenia stron zawierających zmieniające się dane oraz jak zaplanować i zbudować witryny efektywnie korzystające z tych technologii. Nie staniesz się ekspertem w ASP, ColdFusion czy PHP, ale po przeczytaniu jakiejś książki dla zaawansowanych, omawiającej kod ASP, ColdFusion lub PHP, powinieneś być w stanie zrozumieć ją na tyle, aby podjąć się tworzenia ambitnych projektów WWW.

Po tym podsumowaniu plusów i minusów różnych technologii serwerowych powiem Ci coś w sekrecie. Twórcy WWW rzadko dokonują wyborów w oparciu o kryteria racjonalne, gdy muszą na przykład zdecydować, która technologia odpowiada ich potrzebom lepiej niż inne. Ja z pewnością rzadko miałem taką okazję. W rzeczywistości wybór podyktowany jest zwykle dostępnością danej technologii, naszym budżetem, technologiami zastosowanymi w istniejącej witrynie oraz umiejętnościami i doświadczeniem współpracowników. Prawdopodobnie nie będziesz miał luksusu nauki tylko jednej technologii, chyba że pracujesz dla jednej i tylko jednej organizacji i zamierzasz pozostać w niej przez bardzo długi czas. Początkowo uczyłem się jednocześnie ColdFusion i ASP, ponieważ oba były wymagane w różnych projektach, nad którymi pracowałem¹.

ASP, ColdFusion i PHP obok siebie. Strategia nauki

Nie przestrasz się perspektywą nauki wszystkich trzech języków jednocześnie. Prawda jest taka, że w większości przypadków, jeśli chcesz dodać blok kodu ASP w celu obsługi jakiejś funkcji, będziesz też musiał dodać równoznaczny blok kodu ColdFusion lub PHP, aby obsłużyć tę samą funkcję. A najtrudniejszej części nie stanowi tu składnia tego czy innego języka, ale raczej zrozumienie, jakie dane są dostępne, gdzie są dostępne, a także zadecydowanie, jak zmusić je do wykonania tego, co chcemy. Jeśli już to wiesz, składnia nie stanowi problemu.

Z tych powodów w książce tej korzystamy jednocześnie z ASP, ColdFusion i PHP. Choć nie potrzebujesz tworzyć tej samej witryny trzy razy, aby wykorzystać wszystkie trzy technologie serwerowe, powinieneś jednak podjąć wysiłek zrozumienia wszystkich trzech zestawów kodu. To znaczy, jeśli zdecydujesz się tworzyć w ColdFusion, możesz po prostu pominąć kod ASP i PHP. Poświęć chwilę na przyjrzenie się, w jaki sposób języki ASP i PHP wykonują te same działania, co kod ColdFusion. Zrozumienie, w jaki sposób wszystkie trzy bloki kodu wykonują te same zadania, znacznie przyspieszy biegłe opanowanie przez Ciebie programowania WWW.

¹ W polskich realiach technologia PHP odniosła zdecydowanie największy sukces. Znalezienie dostawcy internetu, który umożliwia korzystanie z ASP, jest dość trudne, choć możliwe. Natomiast ColdFusion to w Polsce technologia egzotyczna, stosowana sporadycznie.

Poniższe trzy fragmenty kodu spełniają tę samą funkcję: wysyłają (lub wyświetlają) wartość o nazwie "firstName" wpisaną przez użytkownika w polu formularza XHTML.

w ASP:

```
<p>Dziękuję, <% Response.Write(Request.Form("firstName")) %>, za wypełnienie  
mojego formularza.</p>
```

w ColdFusion:

```
<p>Dziękuję, <cfoutput>#form.firstName#</cfoutput>, za wypełnienie mojego  
formularza.</p>
```

w PHP:

```
<p>Dziękuję, <?php echo $_POST['firstName']; ?>, za wypełnienie mojego  
formularza.</p>
```

Przeanalizujemy podobieństwa tych trzech fragmentów kodu.

- Wszystkie używają specjalnego zestawu znaczników do wskazania oznaczeń kodu w języku przetwarzanym przez serwer. ASP używa `<% i %>`, ColdFusion stosuje `<cf[nazwa_znacznika]> i </cf[nazwa_znacznika]>`, a PHP wykorzystuje `<?php i ?>`.
- We wszystkich znajdziesz oznaczenia, wskazujące, że wysyłane są jakieś dane: ASP używa `Response.Write`, ColdFusion — `<cfoutput>`, a PHP — `echo`.
- Wszystkie jawnie odsyłają do nazwy parametru (`firstName`).
- Wszystkie określają, że chodzi o wartość pola formularza wysyłaną metodą POST (pola formularzy, co omówimy dalej, wysyłane są przy użyciu POST): ASP używa `Request.Form("firstName")`, ColdFusion — `#form.firstName#`, podczas gdy PHP — `$_POST['firstName']`.
- Żaden nie zawiera dodatkowego kodu, nieopisanego w tych czterech punktach.

Nie musisz uczyć się tego kodu na pamięć; nie będzie pytaniem do quizu. Z czasem nabędziesz praktyki. W tym momencie chodzi o pokazanie głębokiego podobieństwa pomiędzy tym, co wykonują te trzy fragmenty kodu: wszystkie żądają zmiennej formularza o nazwie `firstName` i umieszczają ją wewnątrz zwykłego kodu XHTML. Różnice występujące w tych trzech fragmentach kodu mają charakter czysto kosmetyczny: jest to kwestia składni. Najważniejsze jest tu przede wszystkim zrozumienie, że możemy przechwycić wartość wpisaną do formularza i przesłać ją z powrotem wplecioną do zwykłego kodu XHTML.

A zatem w całej książce prezentować będę wszystkie trzy zestawy kodu tuż obok siebie. We wszystkich przypadkach będę analizować czynności wykonywane przez dany kod, więc powinieneś dokładnie wiedzieć, co się dzieje. Musisz tylko przeczytać trzy zestawy kodów i przyjrzeć się, jak każdy z nich wykonuje w swój własny sposób funkcję, którą przedstawię w głównym tekście.

Ale zanim zagłębisz się po uszy w programowaniu, musisz najpierw skonfigurować swój system do pracy z dynamicznymi witrynami WWW.

Zmiana witryny Newland Tours na witrynę dynamiczną

Skonfigurowanie programu Dreamweaver do pracy z dynamicznymi witrynami WWW jest nieco bardziej skomplikowane niż jego konfiguracja do pracy z witrynami statycznymi. W obu przypadkach dokonujemy tego za pomocą okna dialogowego *Site Definition*. Powód, dla którego definiowanie dynamicznych witryn jest tak skomplikowane, znajduje się poza programem Dreamweaver. W celu tworzenia witryn dynamicznych potrzebujemy dostępu do serwera WWW (i uprawnień na nim) z aplikacją serwerową i (lub) serwerem bazy danych.

Dla wielu Czytelników taki tok pracy może być czymś nowym. Dawniej można było utworzyć witrynę lokalnie na swoim dysku twardym, a następnie — gdy była już gotowa do opublikowania — przesłać ją na serwer publikujący (lub publiczny). Podczas tworzenia witryn dynamicznych także możemy pracować na swoim dysku twardym, ale potrzebujemy również dostępu do serwera roboczego. Dopiero po ukończeniu pisanja witryny na serwerze roboczym można ją przesłać na publiczny serwer WWW.



Jedyna różnica między roboczym a zwykłym serwerem WWW polega na tym, że serwer roboczy nie jest udostępniony publicznie. Ale z technicznego punktu widzenia serwer roboczy jest identyczny ze zwykłym serwerem WWW. Przetwarzanie i przesyłanie kodu przebiega na nim w taki sam sposób.

Możemy podłączyć się do serwera na dwa różne sposoby: możemy uruchomić serwer na własnym komputerze i wszystko opracowywać właśnie na nim, możemy wykorzystać w tym celu komputer zdalny, zarówno dostępny w naszej wewnętrznej sieci, jak i zlokalizowany u dostawcy internetu serwer internetowy, dostępny przez FTP.

Jeśli chcemy pracować lokalnie, wówczas musimy poświęcić trochę czasu na skonfigurowanie swojego komputera (instrukcje poniżej). Praca zdalna nie wymaga wprawdzie żadnej konfiguracji naszego komputera, ale będziemy potrzebowali kilku informacji od administratora naszego serwera, aby Dreamweaver mógł współpracować z serwerem.

W zależności od dokonanej decyzji zapoznaj się z następującymi podrozdziałami:

- Jeśli tworzysz lokalnie, przeczytaj następny podrozdział, „Praca na serwerze lokalnym”.
- Jeśli tworzysz na komputerze zdalnym, przejdź od razu dalej, do podrozdziału „Praca na serwerze zdalnym”.
- Po zakończeniu lektury odpowiedniego podrozdziału, niezależnie od wybranej technologii serwerowej i przeprowadzonej konfiguracji, należy skonfigurować program Dreamweaver do pracy z serwerem i wybraną technologią serwerową; temat ten omówiony jest w podrozdziale „Definiowanie witryny dynamicznej w programie Dreamweaver (wszyscy użytkownicy)”.

Praca na serwerze lokalnym

Praca na serwerze lokalnym ma swoje plusy i minusy. Taki model pracy daje następujące korzyści:

- kontrola nad swoim własnym komputerem i niezależność — nie będziesz musiał kontaktować się z administratorem serwera,
- praca taka nie ma wymaga połączenia internetowego lub sieciowego,
- brak opóźnień przy logowaniu, autoryzacji i przesyłaniu danych przez sieć.

Główne wady pracy lokalnej są następujące:

- uruchamiając serwer, narażasz swój komputer na zagrożenia bezpieczeństwa, a im mniej wiesz na temat tego, co robisz, tym bardziej jesteś narażony na ataki, wirusy, włamania i robaki internetowe,
- jeśli napotkasz jakiś problem w konfiguracji lub coś pójdzie nie tak, jak oczekiwałeś, jesteś zwykle zdany na siebie.



Najlepszą metodą ochrony własnego systemu przed włamaniami, wirusami i robakami internetowymi jest regularne uruchamianie usługi Windows Update (Windows) lub Software Update (Mac OS) — co najmniej dwa razy w tygodniu — i instalowanie poprawek związanych z bezpieczeństwem. Jest to szczególnie ważne w przypadku użytkowników Windows, gdyż system ten znacznie częściej pada ofiarą złośliwych ataków. Usługę Windows Update znajdziemy w menu Start, natomiast Macintosh Software Update znajduje się w System Preferences.

Jeśli zdecydujesz się tworzyć witrynę Newland Tours lokalnie, musisz wybrać rodzaj serwera, z którego będziesz korzystać, oraz odpowiednio skonfigurować swój system. Kierując się poniższymi nagłówkami wyróżnionymi pogrubionym drukiem, wybierz instrukcje odpowiadające Twoim potrzebom. Po zakończeniu przejdź od razu do podrozdziału „Definiowanie witryny dynamicznej w programie Dreamweaver (wszyscy użytkownicy)”.



Użytkownicy systemu Mac OS X pracujący lokalnie mają do wyboru tylko dwie opcje: PHP z użyciem serwera Apache oraz ColdFusion. Użytkownicy Macintosh OS 9 oraz OS X, którzy chcą pracować w ASP, mogą to zrobić, ale nie lokalnie. Będą zmuszeni podłączyć się do zdalnego serwera ASP.

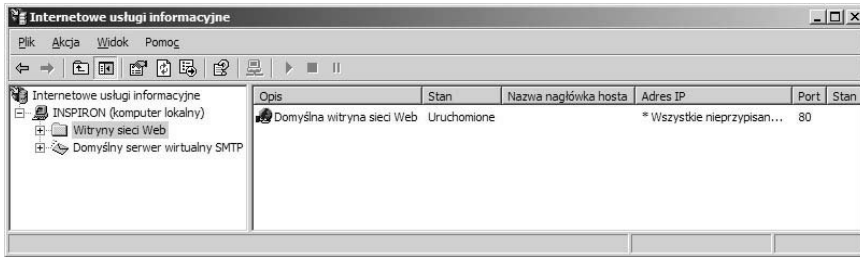
Konfigurowanie środowiska lokalnego dla IIS/ASP

Użytkownicy ASP muszą się upewnić, że w ich systemie jest zainstalowany i uruchomiony serwer IIS (*Internet Information Services*). IIS dostępny jest za darmo w Windows 2000 oraz XP Professional.



Użytkownicy Windows XP Home nie mają szczęścia: Microsoft oficjalnie oświadczył, że jego serwer WWW wymaga systemu Windows XP w wersji Professional.

W zależności od sposobu, w jaki został zainstalowany system Windows, być może mamy już zainstalowany i działający serwer IIS. Aby sprawdzić, czy IIS został zainstalowany, wejdź w *Panel sterowania/Narzędzia administracyjne* (aby zobaczyć tę opcję, użytkownicy Windows XP muszą przełączyć się do widoku klasycznego). Jeśli widoczna jest tam ikona o nazwie *Internetowe usługi informacyjne*, oznacza to, że serwer IIS został już zainstalowany. Aby sprawdzić, czy jest on uruchomiony, kliknij dwukrotnie ikonę i w lewej części okna dialogowego wybierz pozycję *Witryny sieci Web*. W prawym panelu powinna być widoczna pozycja *Domyślna witryna sieci Web*, a przy niej słowo *Uruchomione*. Jeśli napis brzmi *Zatrzymane*, wówczas kliknij przycisk *Uruchom element*, aby ją uruchomić.



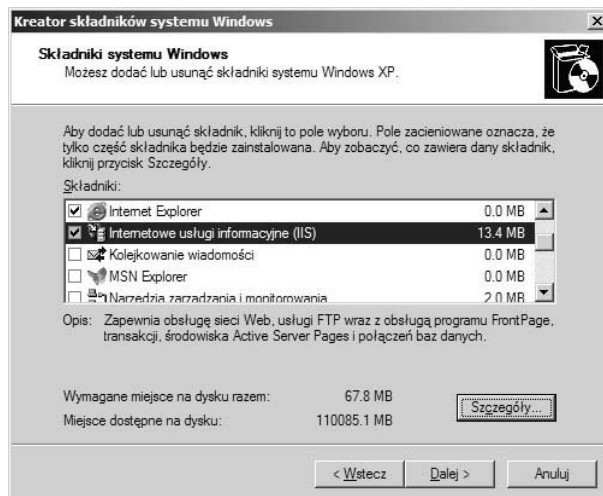
Aby zainstalować i uruchomić IIS, wykonaj następujące czynności:

1. W oknie dialogowym *Dodaj/Usuń programy* (dostępnym z *Panelu sterowania*) wybierz opcję *Dodaj/Usuń składniki systemu Windows*.

System będzie przez kilka chwil analizował nasz komputer w celu sprawdzenia, co zostało już zainstalowane. Po utworzeniu profilu pojawi się okno *Kreator składników systemu Windows*.

2. Na podanej liście zaznacz pozycję *Internetowe usługi informacyjne (IIS)* i kliknij *Dalej*.

Ustawienia domyślne powinny być odpowiednie dla naszych celów, więc nie jest tu konieczna dodatkowa konfiguracja. Po kliknięciu przycisku *Dalej* system Windows zainstaluje i uruchomi serwer IIS.



3. Aby upewnić się, że instalacja zakończyła się powodzeniem, powróć do *Panel sterowania/Narzędzia administracyjne* i sprawdź w konsoli *Internetowe usługi informacyjne*, czy *Domyslna witryna sieci Web* ma stan *Uruchomione*.

Konfigurowanie środowiska lokalnego dla ColdFusion

Lokalna konfiguracja programu ColdFusion jest bardzo łatwa dzięki jego instalatorowi.

1. Pobierz najnowszą wersję deweloperską ColdFusion, dostępną pod adresem <http://www.adobe.com/go/coldfusion>.

Po kilku chwilach rozpocznie się proces instalacji.

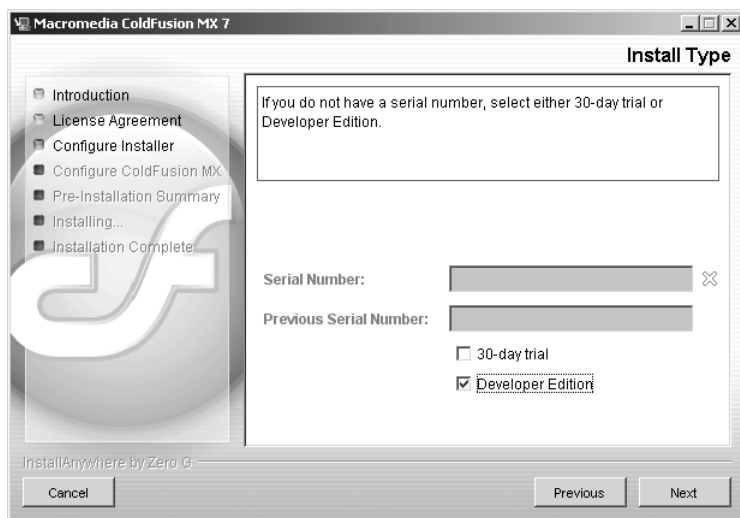
2. Przy wyborze języka przeczytaj wyświetloną informację, naciśnij *OK*, a następnie kliknij przycisk *Next* (dwukrotnie), aby przejść przez wstęp oraz zaakceptować umowę licencyjną.

Te dwa punkty zawierają ważne informacje, więc nie pomijajmy ich bez zapoznania się z nimi.

3. W oknie *Install Type (Rodzaj instalacji)* będzie wymagane wpisanie numeru seryjnego. Jeśli nie masz takiego numeru, zaznacz pole wyboru *Developer Edition* i kliknij *Next*.

Wersji deweloperskiej można używać za darmo przez dowolnie długi czas. Najważniejszym ograniczeniem jest to, że może ona być używana tylko na komputerze lokalnym. To znaczy, jeśli inny komputer w Twojej sieci podejmie próbę dostępu do strony WWW udostępnianej przez wersję Development Edition serwera ColdFusion, jego użytkownik zobaczy komunikat o błędzie, wskazujący na przekroczenie dopuszczalnej liczby adresów IP.

W przypadku instalowania wersji Enterprise Edition serwera ColdFusion musiałbyś podać otrzymany numer seryjny, a ograniczenie dotyczące liczby komputerów zostałoby zniesione.



4. W oknie *Installer Configuration (Konfiguracja instalacji)* zostaw wartość domyślną, *Server configuration (Konfiguracja serwerowa)*. Kliknij *Next*.

Dwie pozostałe opcje przeznaczone są do konfiguracji serwera ColdFusion uruchamianego na serwerze J2EE.

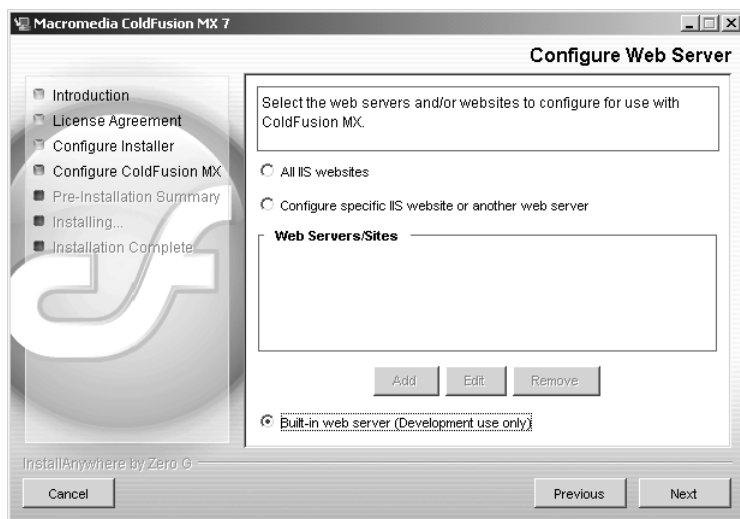
5. W oknie *Sub-component Installation (Instalacja podkomponentów)* ponownie kliknij *Next*, pozostawiając zaznaczone trzy pola wyboru.

Tutaj zaznaczamy do instalacji usługi składowe ColdFusion, jak również dodatkową dokumentację. W przypadku instalacji lokalnej, jeśli nie masz dobrych powodów, aby postąpić inaczej, dobrym pomysłem jest wykonanie kompletnej instalacji.

6. W oknie *Choose Install Directory (Wybierz katalog instalacyjny)* zaakceptuj wartość domyślną i kliknij przycisk *Next*.

W tym oknie możemy wskazać katalog, w którym zainstalowane zostaną pliki programu ColdFusion.

7. W oknie *Web Server Selection (Wybór serwera WWW)* wybierz *Built-In web server (Wbudowany serwer WWW)*, jeśli nie masz uruchomionego serwera WWW (np. IIS lub Apache), lub (jeśli masz już uruchomiony serwer WWW) wybierz *Configure web server connector for ColdFusion (Skonfiguruj złącze z serwerem WWW dla ColdFusion)* i sprawdź, czy Twój serwer jest wymieniony w polu *Web Servers/Sites (Serwery WWW/witryny)*.



ColdFusion jako serwer aplikacji nie jest przeznaczony do pełnienia roli serwera WWW. W prawdziwych witrynach publikacyjnych taką rolę pełni zwykle inny serwer, np. IIS czy Apache.

W środowisku produkcyjnym nie trzeba mieć serwera WWW z prawdziwego zdarzenia. Adobe dla celów produkcyjnych umożliwia wypełnianie roli serwera WWW przez ColdFusion, jeśli tak sobie zażyczymy. Aby aktywować tę funkcję, należy wybrać *Built-In web server*.

Jeśli masz już zainstalowany jakiś serwer WWW, np. IIS lub Apache, możesz pozwolić ColdFusion na podłączenie się do niego. Wówczas, gdy serwer WWW napotka kod ColdFusion, którego nie „rozumie”, będzie „wiedział”, że należy go przesłać do serwera aplikacji ColdFusion w celu jego przetworzenia.

Dokonany tu wybór ma wpływ na adres URL używany do wyświetlenia stron ColdFusion, co będzie miało znaczenie dla konfiguracji programu Dreamweaver w dalszej części lekcji.

8. Kontynuuj aż do zakończenia pracy kreatora. Od tego momentu dalsze postępowanie nie wymaga wyjaśnień.



W czasie instalacji będziesz musiał wybrać hasła administratora oraz hasła do usługi RDS (Remote Development Service). Nie zapomnij ich! Będą one potrzebne do konfiguracji serwera ColdFusion oraz do dostępu do ColdFusion z programu Dreamweaver podczas tworzenia stron, więc nie wyłączaj tej opcji i wprowadź takie hasło, które będziesz w stanie zapamiętać.

Proces instalacji może zająć kilka minut, po czym uruchomiony zostanie serwer ColdFusion.

Po zakończeniu instalacji otworzy się okno przeglądarki, z którego uzyskamy dostęp do aplikacji administratora ColdFusion. Aplikacja ta sama działa na serwerze ColdFusion. Musimy kilkakrotnie kliknąć *Next* i poczekać jeszcze parę minut, aż ColdFusion zakończy proces instalacji. Gdy znajdziemy się znów w aplikacji administratora, oznacza to, że proces instalacji dobiegł końca i możemy przystąpić do pracy.

Instalacja w systemie Mac OS X jest podobna do instalacji w systemie Windows.

Przy programowaniu lokalnym zalecam korzystanie z serwera wbudowanego. Jeżeli jednak wybierzesz Apache jako serwer WWW, w internecie znajdziesz sporo zasobów związanych z konfiguracją takiego zestawu, w tym między innymi *ACME Guide* (Apache/ColdFusion/MySQL/Eclipse), znajdujący się w witrynie <https://acidlabs.org>.

Konfigurowanie środowiska lokalnego dla Apache i PHP

W przeszłości konfigurowanie środowiska lokalnego przy zastosowaniu produktów open source, jakimi są Apache i PHP, było trudne i frustrujące dla nieobeznanych z systemami Linux i Unix użytkowników systemów Windows i Mac OS. Jednak sytuacja ulega poprawie i obecnie dostępnych jest kilka instalatorów WAMP (Windows/Apache/MySQL/PHP) dla systemu Windows, a ostatnio ukazał się również darmowy MAMP (Macintosh/Apache/MySQL/PHP). Są one idealne do pracy lokalnej i dla naszych celów.



Użytkownik Windows z uruchomionym na swoim komputerze serwerem IIS może ulec pokusie zainstalowania jedynie modułu PHP bezpośrednio w IIS, z całkowitym pominięciem instalacji Apache. Taki sposób jest oczywiście wykonalny, ale ma swoje wady: jedynie u nielicznych dostawców usług internetowych oferujących PHP jest ono uruchomione na serwerze IIS. PHP jest prawie niezmiennie łączone z Apache. Ponieważ na ogół pożądanym jest, aby środowisko robocze przypominało — tak bardzo jak to możliwe — ostateczne środowisko publikacyjne, warto zainstalować Apache. Ponadto dobrą praktyką będzie zapoznanie się ze środowiskiem Apache — jego strukturą uprawnień, poleceniami, interfejsem itp.

Instalacja Apache, MySQL i PHP w Windows

Istnieje wiele dobrych instalatorów WAMP, z których większość jest darmowa. Najlepszym, jaki udało mi się znaleźć, jest WAMP5.



W poniższych instrukcjach zakładam, że IIS nie jest uruchomiony jako lokalny serwer WWW. Konfiguracja i ustawienia są inne w sytuacji, gdy Apache nie jest jedynym serwerem WWW.

1. Przejdź do tego adresu URL, przeczytaj instrukcje i kliknij odnośnik *Download*, aby pobrać najnowszą wersję WAMP5: <http://wampserver.com>.

Gdy pisałem niniejszą książkę, najnowszą wersją była 1.7.0².

2. Kliknij dwukrotnie plik instalatora, który będzie nosił nazwę podobną do *wamp5_1.7.0.exe*, aby uruchomić kreator instalacji.

Plik instalatora znajduje się na dysku twardym, w katalogu, w którym zapisałeś go w poprzednim kroku.

3. Przeprowadź proces instalacji za pomocą kreatora. Akceptuj domyślne lokalizacje instalacji, chyba że chcesz umieścić pliki w innych miejscach.

Instalator WAMP5 jest łatwy, być może z wyjątkiem jednego etapu — ekranu *Auto Start*. Zwykle nie ustawiam automatycznego uruchamiania programów podczas uruchamiania systemu operacyjnego. To wydłuża czas uruchamiania systemu i wpływa na ogólną wydajność. Wolę uruchamiać je samodzielnie, gdy zaczynam programowanie. Kliknij przycisk *Next*, pozostawiając niezaznaczone pole opcji *Auto Start* (patrz pierwszy rysunek na następnej stronie).

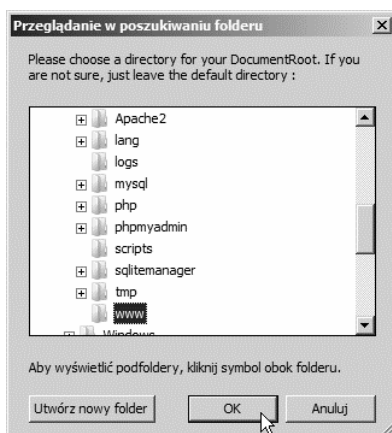
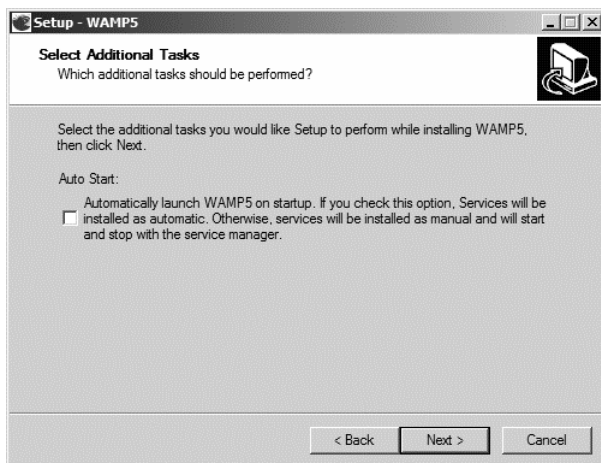
Następnie trzeba wskazać katalog głównego serwera WWW. Ponownie zalecam pozostawienie lokalizacji domyślnej (patrz drugi rysunek na następnej stronie).

4. Aby upewnić się, że instalacja serwera zakończyła się powodzeniem, otwórz przeglądarkę i przejdź do adresu <http://localhost>.

Powinna zostać wyświetlona strona zastępcza — patrz pierwszy rysunek na stronie 101.

² Obecnie najnowszą wersją jest WampServer 2 — *przyp. tłum.*

Właśnie zainstalowałeś Apache, PHP i bazę danych MySQL; możesz przystąpić do pisania kodu.

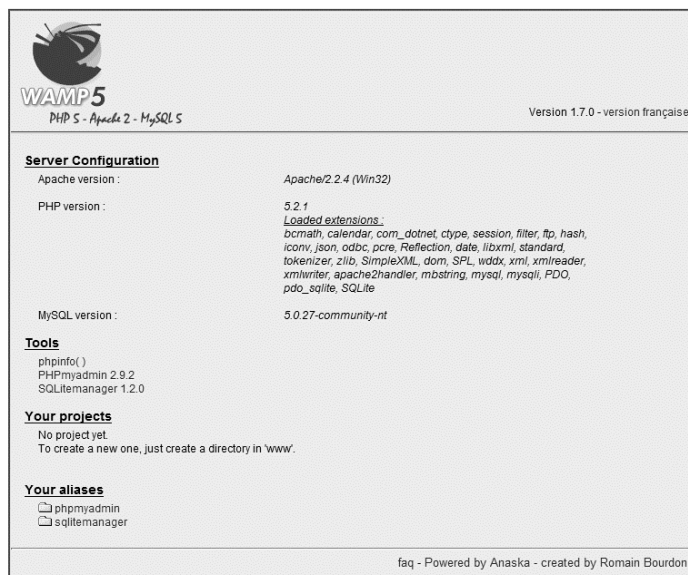


Instalacja Apache, MySQL i PHP w systemie Mac OS X

Do niedawna konfiguracja lokalnego środowiska MAMP (Macintosh/Apache/MySQL/PHP) była stosunkowo żmudna. Natura nie znosi próżni, co odnosi się również do świata techniki. Firma Living-e wydała świetny instalator MAMP, dostępny zarówno w wersji darmowej, jak i profesjonalnej. Wersja darmowa w pełni zaspokoi nasze potrzeby.



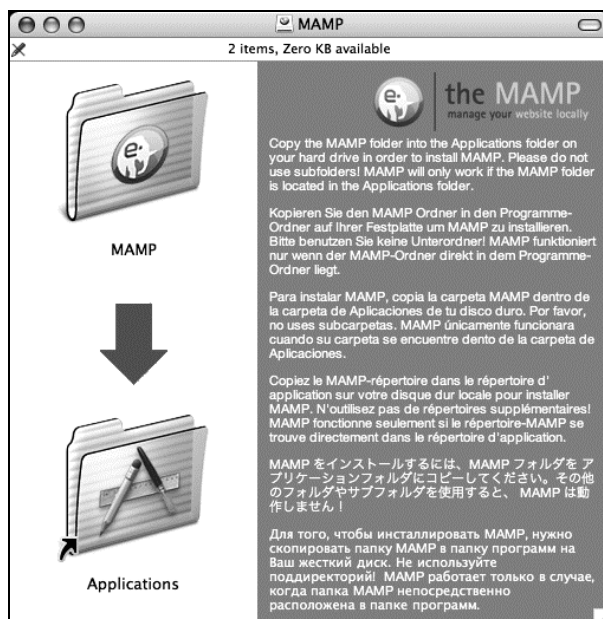
Firma Apple zamieściła w internecie przydatne informacje dotyczące konfiguracji systemów Mac OS dla celów publikowania WWW: <http://developer.apple.com/internet/macosx/intro.html>. W internecie dostępne są również inne ciekawe zasoby dotyczące tworzenia w środowisku MAMP. Można wymienić następujące: <http://phpmac.com>, <http://MacDevCenter.com> i <http://entropy.ch/software/macosx>.



1. Przejdź do adresu <http://www.mamp.info> i pobierz najnowszą wersję MAMP. Dla celów niniejszej książki nie będą potrzebne możliwości oferowane przez MAMP-Pro.

Gdy pisałem tę książkę, najnowsza wersja nosiła numer 1.4.1.

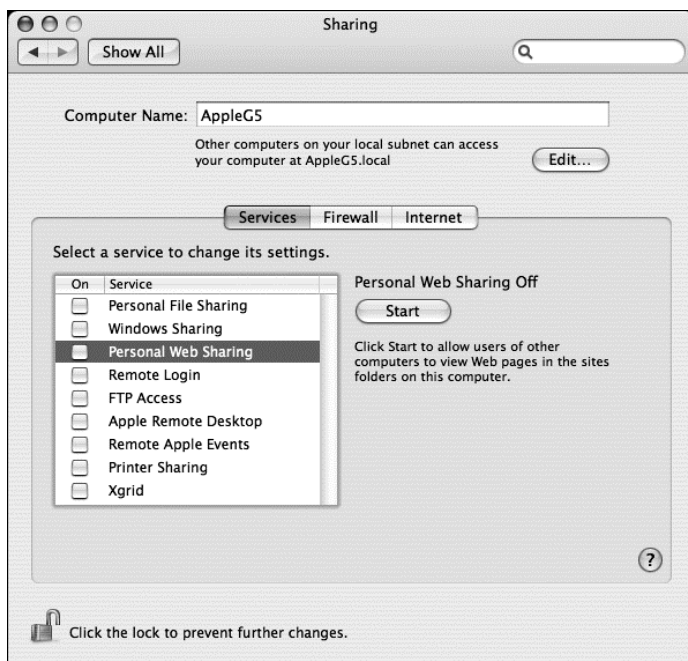
2. Wypakuj plik instalatora i kliknij dwukrotnie plik z rozszerzeniem *.dmg*, aby zamontować obraz dysku. Przeciągnij folder MAMP do folderu *Applications*.





5. Upewnij się, że w zakładce *Services (Usługi)* folderu *Sharing* opcja *Personal Web Sharing* nie jest aktywna. Jeżeli jest, zaznacz ją i kliknij przycisk *Stop*.

Ze względów bezpieczeństwa warto wyłączyć wszystkie usługi udostępniana, chyba że koniecznie potrzebujesz, aby któraś była włączona.



Praca na serwerze zdalnym

Nie każdy może skonfigurować serwer lokalny. Użytkownicy komputerów „Macintosh” nie mają oczywiście dostępu do ASP na serwerze IIS. Ponadto wielu użytkowników Windows i Mac OS nie ma zezwolenia na instalację i uruchomienie serwera WWW. Mimo że mój komputer jest chroniony przez zaporę uniwersyteckiego serwera, nie wolno mi uruchomić serwera. Powodem są kwestie bezpieczeństwa. Serwery WWW wymagają stałego nadzorowania, aby mogły poradzić sobie z wirusami i lukami w zabezpieczeniach, które stanowią zagrożenie dla sieci korporacyjnych i uniwersyteckich. Aby wyeliminować (lub przynajmniej zminimalizować) możliwości włamań naruszających cały system bezpieczeństwa, administratorzy często zabraniają użytkownikom instalowania serwerów WWW na ich własnych komputerach. Przy poprawnej konfiguracji większość prezentowanych wyżej narzędzi programistycznych nie powinna „wyeksponować” komputera w internecie. Jeżeli masz jakieś wątpliwości, skonsultuj się z administratorem sieci lub uruchamiaj te narzędzia tylko wtedy, gdy komputer jest odłączony od sieci.

Oczywiście, jeśli nie mamy dostępu do lokalnego serwera, musimy poszukać innego serwera roboczego. Może to być dedykowany serwer roboczy (z jakiego korzystam na uniwersytecie) albo katalog na naszym publicznym serwerze WWW. Do serwera możemy uzyskać dostęp przez sieć, jeśli mamy połączenie sieciowe z serwerem, lub za pomocą FTP. Tak czy inaczej, zanim będziemy mogli kontynuować i zdefiniować swoją witrynę w programie Dreamweaver, musimy najpierw otrzymać od administratora serwera informacje na temat ścieżki sieciowej lub FTP do serwera. Powinien to być serwer IIS (do pracy z ASP), musi mieć zainstalowany ColdFusion (do pracy z ColdFusion) lub Apache/IIS z modułem PHP (do pracy z PHP).

Oprócz konta oraz uprawnień do dodawania i usuwania plików i folderów wewnątrz tego konta będziemy też potrzebować jednej z poniższych informacji od administratora witryny.

- Musimy znać ścieżkę dostępu do folderu w sieci; ścieżka ta powinna wyglądać podobnie do poniższych:

```
\\webdev.duza_firma.pl\twoja_witryna\  
\\Nazwa_serwera\twoja_witryna\
```



Jeśli masz dostęp sieciowy do serwera, powinieneś zmapować dysk sieciowy do Twojego konta na tym serwerze.

- Musisz otrzymać informacje dotyczące dostępu do witryny przez FTP, włącznie z nazwą domenową (lub adresem IP) oraz nazwą użytkownika i hasłem dostępu do swojego konta na tym serwerze.

Powyższe informacje są wystarczające do uzyskania dostępu do tych folderów w celu przesłania tam zawartości. Ale będziemy też potrzebowali jakiegoś sposobu na jej przeglądanie. Przede wszystkim musimy znać adres URL prowadzący do treści umieszczonych przez nas na serwerze. Zwykle adres ten ma postać: http://webdev.duza_firma.pl/twoja_witryna/ lub http://Nazwa_serwera/twoja_witryna/. Gdy witryna zostanie przeniesiona do publikacji, adres URL (w tym przykładzie) zmieni się na http://www.duza_firma.pl/twoja_witryna/. Ważna jest tu znajomość kompletnego adresu URL, łącznie z <http://>. Taką informację możemy otrzymać wyłącznie od administratora serwera.

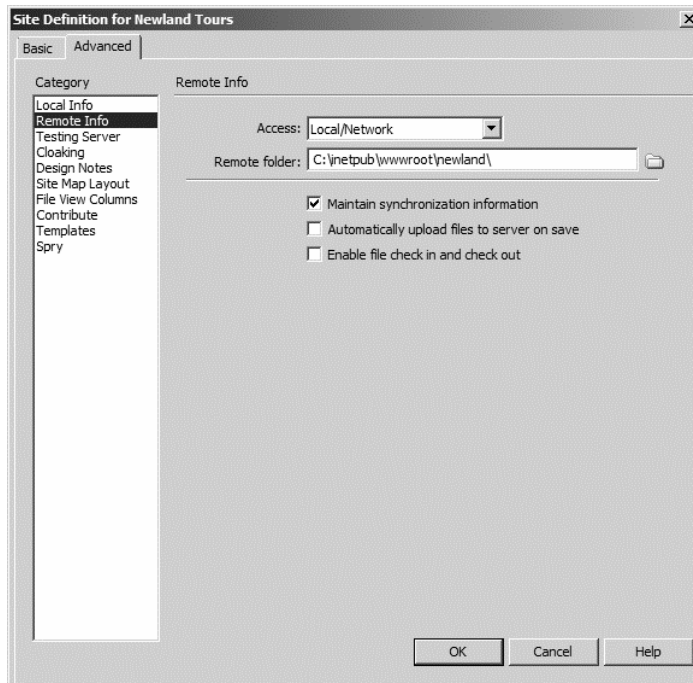
Definiowanie witryny dynamicznej w programie Dreamweaver (wszyscy użytkownicy)

Niezależnie od tego, który z powyższych podrozdziałów ma dla Ciebie zastosowanie, wykonaj poniższe kroki, by zdefiniować swoją witrynę w programie Dreamweaver. Przed rozpoczęciem musisz mieć dostęp do w pełni skonfigurowanego serwera WWW, wyposażonego w działający serwer aplikacji (z aktywnymi modułami).

1. W programie Dreamweaver, przy otwartej witrynie Newland Tours, wybierz z menu *Site/Manage Sites*. Upewnij się, że w oknie dialogowym *Manage Sites* wybrana jest pozycja *Newland Tours*, i kliknij *Edit* (*Edycja*).

Pamiętajmy, że witryna Newland Tours jest już zdefiniowana. Nie musimy zaczynać od początku. Wystarczy tylko dodać zdalny serwer i sprawdzić informacje serwera dla istniejącej witryny.

2. W oknie dialogowym *Site Definition for Newland Tours* kliknij zakładkę *Advanced*. Z listy *Category* (*Kategoria*) w lewej części okna wybierz pozycję *Remote info* (*Informacje o serwerze zdalnym*). Z menu rozwijanego *Access* (*Dostęp*) wybierz właściwą pozycję i wprowadź odpowiednie informacje w pola, które się pojawiają, korzystając z poniższych wskazówek.



Jeżeli pracujesz na komputerze z lokalnym serwerem IIS (ASP lub ColdFusion przez IIS), wybierz z menu rozwijanego *Access* opcję *Local/Network* (*Lokalny/Sieć*). Kliknij ikonę folderu obok pola *Remote folder* (*Folder zdalny*) i odśledź folder *C:\inetpub\wwwroot*. Kliknij przycisk *Utwórz nowy*

folder i utwórz folder o nazwie *newland*. Kliknij go dwukrotnie i naciśnij przycisk *Zapisz*, aby wybrać go jako folder zdalny.



Folder Inetpub/wwwroot jest głównym folderem WWW lokalnego serwera IIS. Gdy będziemy przeglądać swoją witrynę (<http://localhost/>), wyświetlane będą strony z tego folderu.

Jeśli pracujesz na komputerze z lokalnym serwerem WWW ColdFusion, wybierz w menu rozwijanym *Access* opcję *Local/Network*. Kliknij ikonę folderu obok pola *Remote folder (Folder zdalny)* i odśzukaj folder *C:\CFusion\MX7\wwwroot*. Kliknij przycisk *Utwórz nowy folder* i utwórz folder o nazwie *newland*. Kliknij go dwukrotnie i naciśnij przycisk *Zapisz*, aby wybrać go jako folder zdalny.



Folder CFusionMX7 jest głównym folderem lokalnego serwera ColdFusion. Gdy będziemy przeglądać swoją witrynę (<http://localhost:8500/>), wyświetlane będą strony z tego folderu. Należy również zauważyć, że w czasie, gdy pisałem tę książkę, dostępna była wersja beta ColdFusion 8. Jeżeli korzystasz z ColdFusion 8, a nie MX7, nazwy folderów mogą się różnić.

Jeśli pracujesz na komputerze z lokalnym serwerem Apache, zainstalowanym za pomocą skonfigurowanego powyżej instalatora W/MAMP, wybierz z menu rozwijanego *Access* opcję *Local/Network*. Kliknij ikonę folderu obok pola *Remote folder (Folder zdalny)* i odśzukaj folder *C:\wamp\www* (Windows) lub *HD:Applications:MAMP\htdocs* (Mac OS). Kliknij przycisk *Utwórz nowy folder* (Windows) lub *New Folder* (Mac OS) i utwórz folder o nazwie *newland*. Kliknij go dwukrotnie i naciśnij przycisk *Zapisz*, aby wybrać go jako folder zdalny.



Folder C:\wamp\www\ jest głównym folderem serwera Apache w Windows. Folder HD:Applications:MAMP\htdocs jest głównym folderem Apache w systemach Mac OS. Gdy będziemy przeglądać swoją witrynę (<http://localhost/> w Windows lub <http://localhost:8888> w MacOS), wyświetlane będą strony z tego folderu.

Jeśli pracujesz na komputerze mającym połączenie sieciowe z serwerem, wybierz z menu rozwijanego *Access* opcję *Local/Network* i odśzukaj swój folder na serwerze. Najprawdopodobniej pojawi się on na zmapowanym dysku sieciowym. Za pomocą okna dialogowego *Choose remote root folder* dodaj nowy folder o nazwie *newland* i wybierz go jako folder zdalny.

Jeśli pracujesz na komputerze, który ma dostęp do serwera przez FTP, upewnij się najpierw, że na Twoim koncie istnieje folder o nazwie *newland*. Wówczas w oknie dialogowym *Site Definition* programu Dreamweaver wybierz z menu rozwijanego *Access* opcję *FTP* i w pole *FTP host* wpisz właściwy adres IP serwera lub URL. W pole *Host directory* wpisz ścieżkę do folderu *newland*. Następnie wypełnij pola *Login (Użytkownik)* i *Password (Hasło)*. Gdy już zrobisz to wszystko, kliknij przycisk *Test*, aby upewnić się, że wszystko zostało poprawnie skonfigurowane.

3. Z listy *Category* w lewej części okna wybierz pozycję *Testing Server (Serwer testowy)*. Z menu *Server model (Model serwera)* wybierz *ASP VBScript*, *ColdFusion* lub *PHP MySQL*, w zależności od tego, z którego modelu serwera zdecydowałeś się korzystać. W menu *Access* (oraz we wszystkich opcjach, jakie pojawiają się poniżej) wprowadź te same informacje, które podałeś w poprzednim punkcie.



Nie wybieraj ASP JavaScript. Choć ogólnie rzecz biorąc, ten model serwera działa prawidłowo, to jest on niezgodny z większością kodu występującego w niniejszej książce.

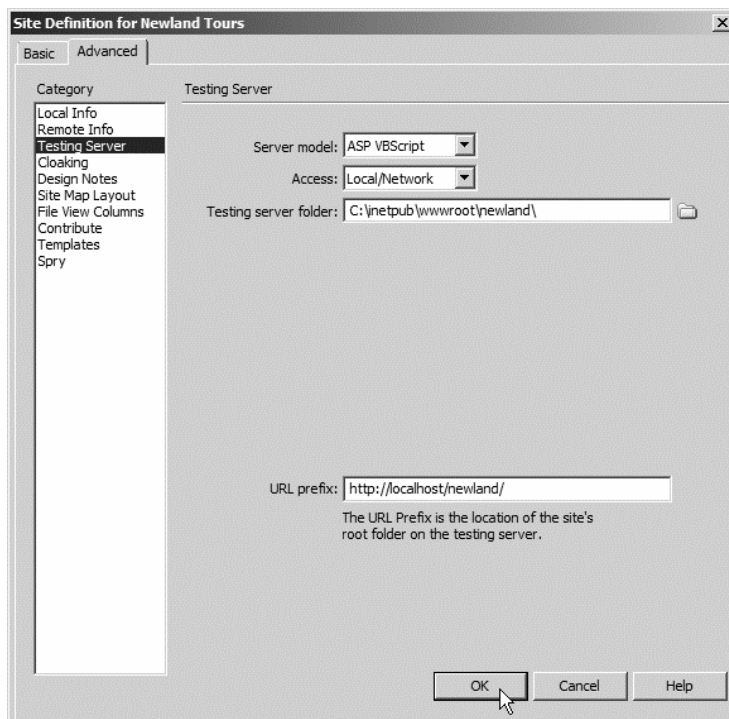
Witryna zdalna Newland Tours jest w gruncie rzeczy taka sama jak witryna na serwerze testowym. Różnica polega na tym, że witryna zdalna istnieje w celu umożliwienia programowi Dreamweaver zapisu plików we właściwym folderze, podczas gdy serwer testowy powoduje, że Dreamweaver może testować pliki po ich przetworzeniu przez serwer, dzięki czemu możemy sprawdzić poprawność ich działania.

4. W polu *URL prefix (Prefiks adresu URL)*, znajdującym się w dolnej części zakładki kategorii *Testing Server*, wpisz adres URL witryny.

Jeśli korzystasz z lokalnego IIS na swoim komputerze (wszyscy użytkownicy ASP i niektórzy z użytkowników ColdFusion), wpisz tu `http://localhost/newland/`.

Jeśli masz uruchomiony lokalnie serwer ColdFusion, działający niezależnie, bez IIS, wpisz tu `http://localhost:8500/newland/`.

Jeśli masz uruchomiony lokalnie serwer Apache w systemie Windows, wpisz tu `http://localhost/newland/`, natomiast w systemie MacOS wpisz `http://localhost:8888/newland`.





Jak wspomniano wcześniej, słowo „localhost” jest skrótem, który oznacza serwer WWW na komputerze lokalnym, i nakazuje pokazanie stron z głównego folderu serwera (jest to folder wwwroot w przypadku serwera IIS oraz niezależnie uruchomionego serwera ColdFusion lub htdocs w przypadku serwera Apache w MacOS i www w Windows), w którym to folderze utworzyliśmy (w poprzednim punkcie) folder newland.

Jeśli korzystasz ze zdalnego serwera (czy to przez sieć, czy to przez FTP), wpisz jego adres URL, podany przez administratora serwera. Prawdopodobnie będzie on miał postać: http://www.duza_firma.pl/newland/.

Tak czy inaczej adres musi rozpoczynać się od <http://> i nie może zawierać oznaczeń dysków (np. h:). Zwróćmy też uwagę, że w adresie tym używa się zwykłych ukośników (/), a nie lewych (\).



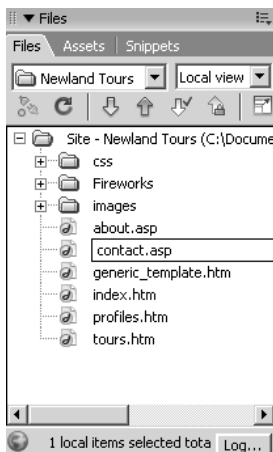
Jeśli na dalszym etapie lekcji witryna nie wyświetli się poprawnie, pole URL prefix jest pierwszym miejscem, w które powinniśmy zajrzeć. Jeśli informacja ta nie została wpisana poprawnie, nie będziemy w stanie wczytać witryny i zobaczyć jej w działaniu, nawet gdy składnia jest poprawna, a serwer i model serwera zostały poprawnie skonfigurowane i uruchomione.

5. Kliknij **OK**, aby zapisać dane i zamknąć okno dialogowe, a następnie kliknij **Done**, aby zamknąć okno dialogowe *Edit Sites*.

Witryna została zdefiniowana i przygotowana do pracy ze zmieniającymi się danymi.

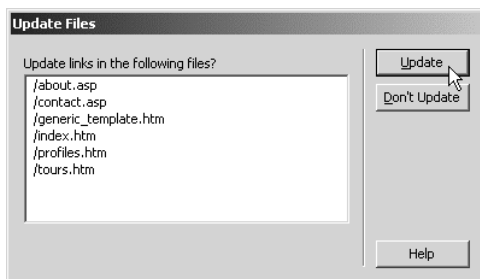
6. Kliknij po kolei prawym przyciskiem myszy (Windows) lub z wciśniętym klawiszem **Control** (Mac OS) każdy z plików HTML widocznych w palecie *Files*, wybierz z menu kontekstowego opcję *Edit/Rename (Edycja/Zmień nazwę)* i zmień odpowiednio rozszerzenia plików z *.html* na *.asp*, *.cfm* lub *.php*. Ilekroć pojawi się okno dialogowe *Update Files (Aktualizacja plików)*, kliknij *Update (Aktualizuj)*.

Przy aktualizacji do witryny dynamicznej wymagana jest zmiana rozszerzeń, gdyż za pomocą rozszerzeń serwer określa, czy strona zawiera jakiś szczególny kod, który należy przetworzyć.



Po zmianie rozszerzenia wszystkie odnośniki wskazujące na daną stronę są niewłaściwe.

Dreamweaver wychwytuje to i naprawi problem po wybraniu opcji *Update*.

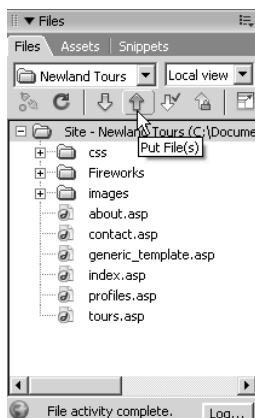


7. Kliknij folder najwyższego poziomu, a następnie kliknij przycisk *Put File(s)* (*Dodaj pliki*).

Spowoduje to przesłanie całej witryny do zdalnego folderu i serwera testowego.

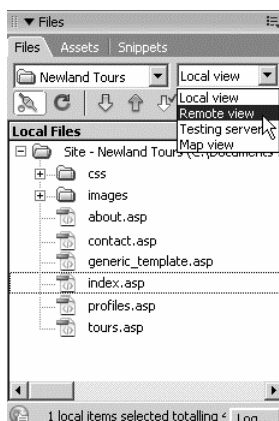


Jeśli pojawi się okno dialogowe z pytaniem, czy chcemy dodać całą witrynę, kliknij OK.



8. Przełącz panel *Files* na widok zdalny (*Remote view*). Kliknij raz, aby wybrać plik *index.asp*, *index.cfm* lub *index.php* w palecie *Files*, i naciśnij *F12* (Windows) lub *Option+F12* (Mac OS).

Naciśnięcie klawisza *F12* lub *Option+F12* testuje witrynę na serwerze. Wynik testu może przynieść albo dobre, albo złe wieści. Jeżeli ujrzymy stronę główną witryny, oznacza to, że została ona poprawnie skonfigurowana i jesteśmy gotowi do dalszej pracy. Jeśli natomiast otrzymamy komunikat o błędzie lub strona nie wyświetli się poprawnie, to znaczy, że coś poszło nie tak. Aby naprawić problem, zajrzyjmy ponownie w pole *URL prefix*. Sprawdźmy też za pomocą Eksploratora Windows lub Macintosh Finder, czy pliki rzeczywiście zostały przesłane do zdalnej witryny. Jeśli problem wciąż występuje, skontaktuj się z administratorem witryny, który powinien sobie z nim poradzić.



Adres URL wyświetlony w pasku adresowym



Budujemy prostą aplikację dynamiczną

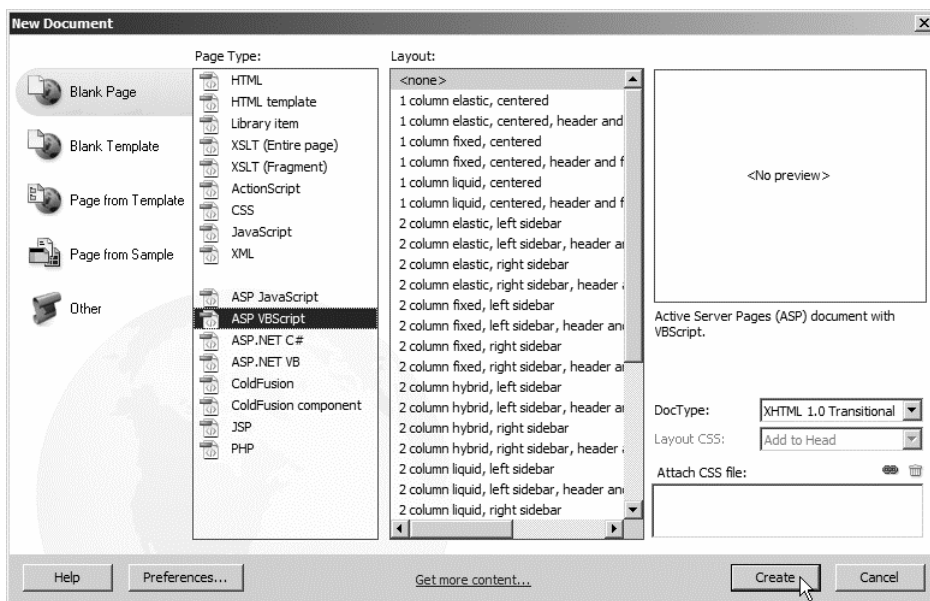
Przetestowaliśmy działanie witryny na zdalnym serwerze i — zakładając, że ujrzeliśmy w przeglądarce stronę `index.asp`, `index.cfm` lub `index.php` — mamy wszystko poprawnie skonfigurowane. A w witrynie Newland Tours nie ma jeszcze nawet jednego elementu zawierającego dane ulegające zmianie. Wykonaliśmy tyle pracy, a nie ujrzeliśmy jeszcze żadnych korzyści. W tym zadaniu utworzymy bardzo

prostą aplikację dynamiczną, dzięki której zobaczymy, co nas dalej czeka, zarówno pod względem pojęciowym (sposób działania witryn dynamicznych), jak i pod względem działania (sekwencje kroków niezbędnych do utworzenia danej funkcji).

Tworzymy stronę wprowadzania danych

Naszym zamiarem jest zbudowanie strony zawierającej formularz WWW z pytaniem o imię użytkownika. Po kliknięciu przycisku *Wyślij* użytkownik zostanie przekierowany na inną stronę, która wyświetli wpisane przez niego imię. Oczywiście, taka aplikacja nie stanowi szczytu możliwości ASP, ColdFusion czy PHP. Jej zadaniem jest wprowadzenie do budowania formularzy, posługiwania się danymi ulegającymi zmianom i rozróżniania między stroną serwera a klienta.

1. Przy otwartej witrynie Newland Tours wybierz *File/New (Plik/Nowy)*. W oknie dialogowym *New Document (Nowy dokument)*, z listy *Category* po lewej stronie, wybierz pozycję *Dynamic Page (Strona dynamiczna)*, a następnie w prawej kolumnie wskaż *ASP VBScript*, *ColdFusion* lub *PHP*, a na liście *Layout (Układ)* wybierz opcję *<none>*. Upewnij się, że menu *DocType (Rodzaj dokumentu)* ustawione jest na *XHTML 1.0 Transitional*. Kliknij przycisk *Create (Utwórz)*.



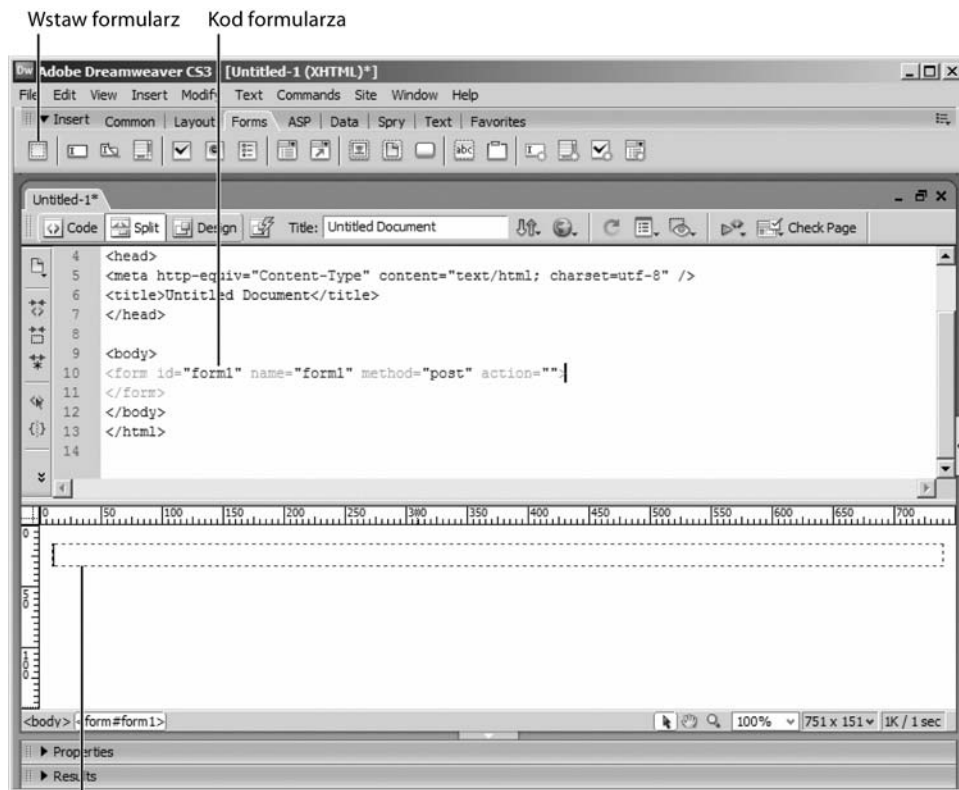
W tym kroku tworzymy nową stronę dynamiczną. Określając jej rodzaj, wskazujemy programowi Dreamweaver, jakiego języka ma użyć, gdy zastosujemy gotowe funkcje serwera, jakie rozszerzenie ma nadać przy zapisywaniu pliku i — w niektórych przypadkach — jaki dodatkowy kod ma zostać dopisany do nagłówków dokumentu.

Użytkownicy ASP zobaczą wiersz `<%@LANGUAGE="VBSCRIPT" CODEPAGE="65001"%>`. Ten wiersz określa, czy korzystamy z VBScript, czy też z JScript — obu języków można użyć z ASP. W ćwiczeniach w tej książce będziemy jednak pracować z VBScript. ColdFusion i PHP

nie posiadają kilku języków skryptowych, więc w kodzie PHP lub ColdFusion nie ma odpowiednika tego wiersza.

2. Kliknij gdziekolwiek w oknie projektu, z palety *Insert* wybierz kategorię *Forms* (Formularze) i kliknij przycisk *Form* (Formularz), aby wstawić formularz na stronę.

Właśnie wstawiliśmy na stronę prosty formularz.



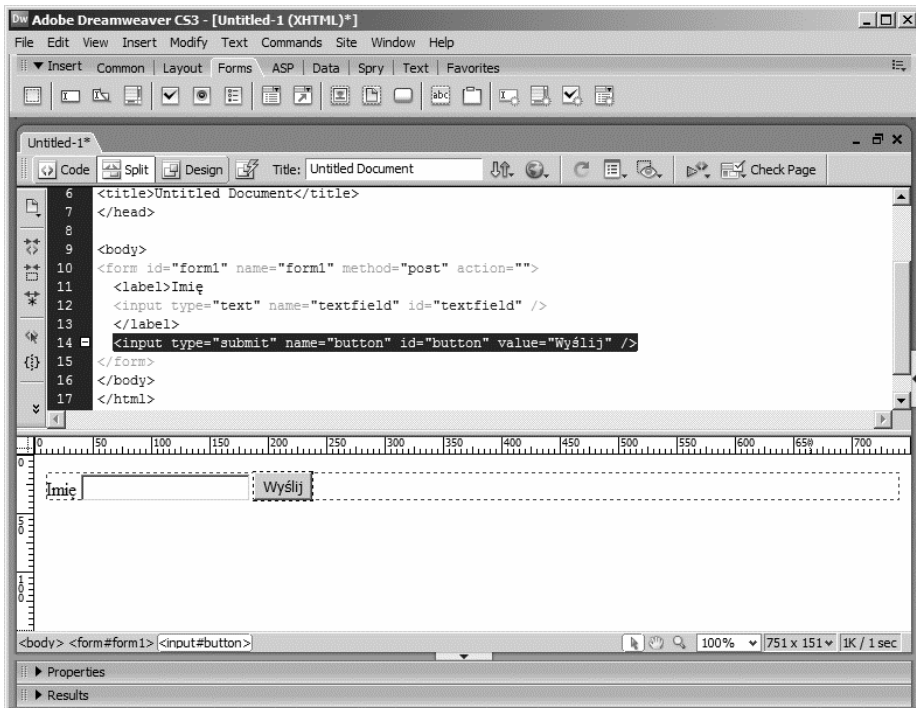
Przerywana linia wskazująca granice formularza

Przerywana linia w kolorze czerwonym wskazuje granice formularza. Nie będzie ona widoczna w przeglądarce, jej zadaniem jest pomoc w określeniu początku i końca formularza na stronie.

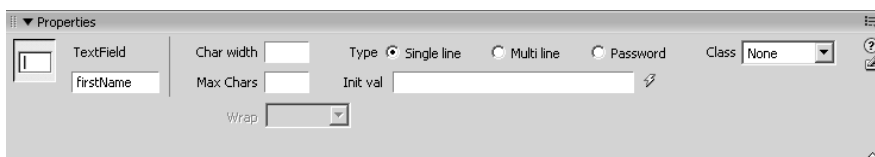
3. Kliknij przycisk *Text Field* (Pole tekstowe). W oknie dialogowym *Input Tag Accessibility Attributes* (Atrybuty dostępności znacznika input), w polu *Label* (Etykieta), wpisz *Imię* i kliknij *OK*. Kliknij przycisk *Button* (Przycisk) i w oknie *Insert Tag Accessibility Attributes* naciśnij *Cancel* (Anuluj). W oknie projektu kliknij prawym przyciskiem myszy i z menu kontekstowego wybierz *Label*. Wpisz tu *Wyślij*. Kliknij *OK*.

Tak oto dodaliśmy dwa elementy formularza: pole tekstowe, do którego użytkownik może wpisać tekst, oraz przycisk *Wyślij*.

Dreamweaver stał się bardziej aktywny w zachęcaniu twórców do projektowania zgodnego ze standardami WWW i wytycznymi dotyczącymi dostępności, jak widzieliśmy na przykładzie dwukrotnie pojawiającego się tu okna dialogowego dotyczącego atrybutów dostępności.



4. Kliknij pole tekstowe i w paletce *Properties* nadaj mu nazwę `firstName`, po czym wciśnij klawisz tabulacji lub *Enter/Return*.



Zaraz użyjemy tej nazwy w celu pobrania wartości w ASP, ColdFusion lub PHP. Zawsze nadawaj wszystkim polom formularza znaczące nazwy. Kod jest już i tak wystarczająco trudny do wpisania — nie pogarszajmy tego i nie zostawiamy tam nazw typu *Textfield1*, *Textfield2* i *Textfield3*, jakie domyślnie nadawane są przez program Dreamweaver.

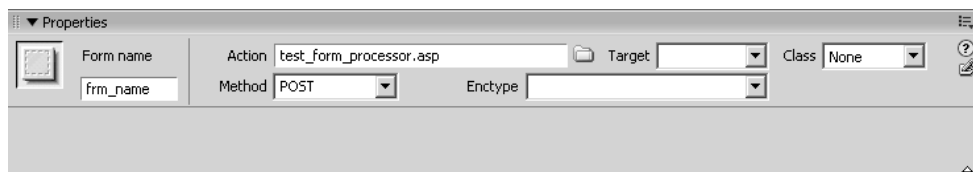
Klawisz tabulacji lub *Enter/Return* użyty został w celu zastosowania wartości wprowadzonej do palety *Properties*.

5. Kliknij `<form#form1>` na pasku wyboru znaczników, aby aktywować paletę *Properties* dla formularza. Nadaj formularzowi nazwę `frm_name`, a w pole *Action* (*Działanie*) wpisz `test_form_processor.asp` (bądź `.cfm` czy `.php`).

Pole *Action* wskazuje na stronę (lub inny element) zawierającą skrypt, który może przetworzyć dane formularza. Jest to zawsze adres URL. W tym przypadku pole wskazuje na nieistniejący adres, gdyż nie utworzyliśmy jeszcze strony `test_form_processor.asp` (bądź `.cfm` czy `.php`). Metoda wysłania powinna być ustawiona na wartość POST. W dalszej części książki wyjaśnię, co oznacza POST.



Odtąd będę zakładał, że sam dojdiesz do tego, jakie rozszerzenia są właściwe w Twoim przypadku. Wymienianie za każdym razem, gdy odwołuję się do pliku, słów „bądź `.cfm` czy `.php`” byłoby marnowaniem miejsca i obrażałoby Twoją inteligencję. Będę zawsze używał rozszerzenia `.asp`, tak więc jeśli korzystasz z ColdFusion, po prostu podstaw w to miejsce rozszerzenie `.cfm`, a jeśli używasz PHP, zamień rozszerzenie na `.php`.



- Wybierz z menu *File/Save As (Plik/Zapisz jako)* i nazwij plik `test_form.asp`.

Jest to plik jednorazowego użytku, który tworzymy tylko w celu przetestowania prostej cechy witryny dynamicznej. Do stron używanych dla celów testowych dodaje często przedrostek „test_”, w ten sposób po zakończeniu mogą je łatwo znaleźć i usunąć.

Tworzymy stronę wyjściową

Zakończyliśmy budowanie strony służącej do wprowadzania danych. Nadszedł czas, aby pokazać, w jaki sposób ASP lub ColdFusion mogą pobrać tę informację, wstawić ją do zwykłego kodu XHTML i zwrócić do przeglądarki klienta.

- Utwórz nową stronę dynamiczną.

Jeśli nie pamiętasz jak, zajrzyj do punktu 1. z poprzedniego zadania.

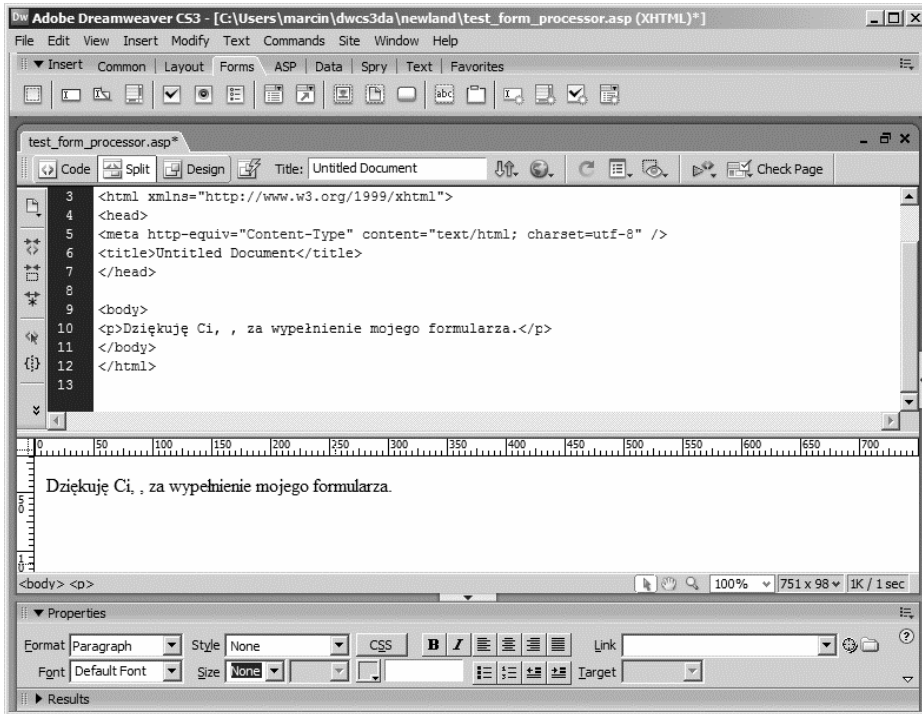
- Zapisz nowy plik jako `test_form_processor.asp`.

Często stosuję przyrostek „_processor” dla stron istniejących w celu przetwarzania pewnego rodzaju danych. Ta strona przetworzy dane wpisane przez użytkownika do formularza.

- W widoku projektu wpisz *Dziękuję Ci, ,* za wypełnienie mojego formularza. Z kursorem umieszczonym gdziekolwiek wewnątrz tego akapitu wybierz *Paragraph* z menu *Format*, w palecie *Properties*.

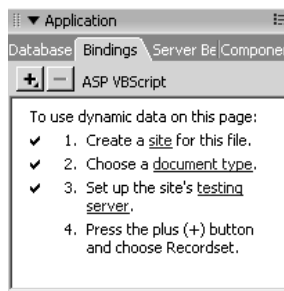
Ostatecznie tekst ten będzie brzmiał: *Dziękuję Ci, [imię wpisane przez użytkownika], za wypełnienie mojego formularza.* Większość zdania stanowi statyczny tekst. Elementem zmiennym będzie rzeczywista wartość oznaczająca imię, które zostanie pobrane z formularza.

Wybierając *Paragraph* jako format akapitu, objęliśmy ciąg tekstu znacznikami `<p></p>`.



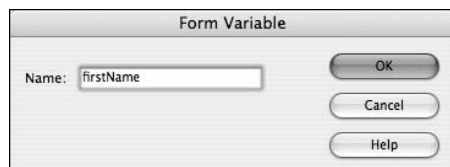
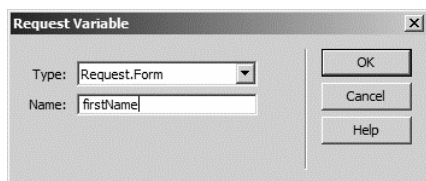
4. Umieść kursor między przecinkami, w miejscu na imię. Otwórz paletę *Bindings* (Wiązania), wybierając z menu *Windows/Bindings*.

Paleta *Bindings* używana jest do określenia wszystkich danych dostępnych na stronie. Dane przechowywane są zazwyczaj w formacie nazwa-wartość. W tym konkretnym przypadku nazwą jest `firstName`. Wartość jeszcze nie istnieje i nie będzie istnieć, dopóki ktoś nie wypełni formularza. Pamiętajmy też, że wartość ta wprowadzana jest ze strony *test_form.asp*. Oprócz formularzy mogą to być również inne źródła (zapoznamy się z tym później), włącznie z adresem URL, zestawem rekordów (danych otrzymanych z bazy danych), cookies itd. Tym razem jednak będzie to formularz.

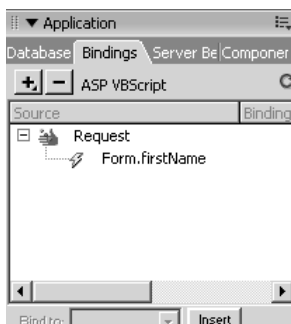


5. Kliknij przycisk **+**, aby dodać nowe powiązanie. Jeśli korzystasz z ASP, wybierz z menu *Request Variable* (*Zażądaj zmiennej*), a jeśli używasz ColdFusion lub PHP — wybierz tu *Form Variable* (*Zmienna odpowiadająca polu formularza*). W wyświetlonym oknie dialogowym w przypadku ASP z menu *Type* (*Rodzaj*) wybierz *Request.Form*, a w polu *Name* wpisz `firstName`; w przypadku ColdFusion lub PHP wpisz tylko `firstName` w pole *Name*. Kliknij **OK**.

Zrzut ekranu po lewej stronie przedstawia okno dialogowe *Request Variable*, widoczne u użytkowników ASP, natomiast drugi rysunek przedstawia okno dialogowe *Form Variable*, widoczne dla użytkowników ColdFusion i PHP.



Paleta *Bindings* zostanie zaktualizowana i będzie pokazywać zmienną `firstName`. Zrzut ekranu przedstawia wygląd palety *Bindings* w ASP. Różni się on nieznacznie od tego w ColdFusion i PHP (słowo *Request* zastąpione będzie słowem *Form*, a zamiast wyrażenia *Form.firstName* będzie tu *firstName*).



Pewnie zastanawiasz się, co właściwie wykonaliśmy. Jeśli spojrzymy w kod, zobaczymy, że dokument w ogóle się nie zmienił: jest to ten sam kod, co przed otwarciem palety *Bindings*. Za pomocą interfejsu graficznego programu Dreamweaver określiliśmy, w jaki sposób Dreamweaver ma wpisać blok kodu dynamicznego.

Na początku rozdziału wymieniałem trzy fragmenty kodu: po jednym dla ASP, ColdFusion i PHP. Kod w tych wierszach określał zmienną (`firstName`), jej pochodzenie (formularz) oraz przeznaczenie (wstawienie do XHTML). W paletce *Bindings* określiliśmy te elementy w sposób, który Dreamweaver może „zrozumieć” je i przetworzyć na postać kodu.

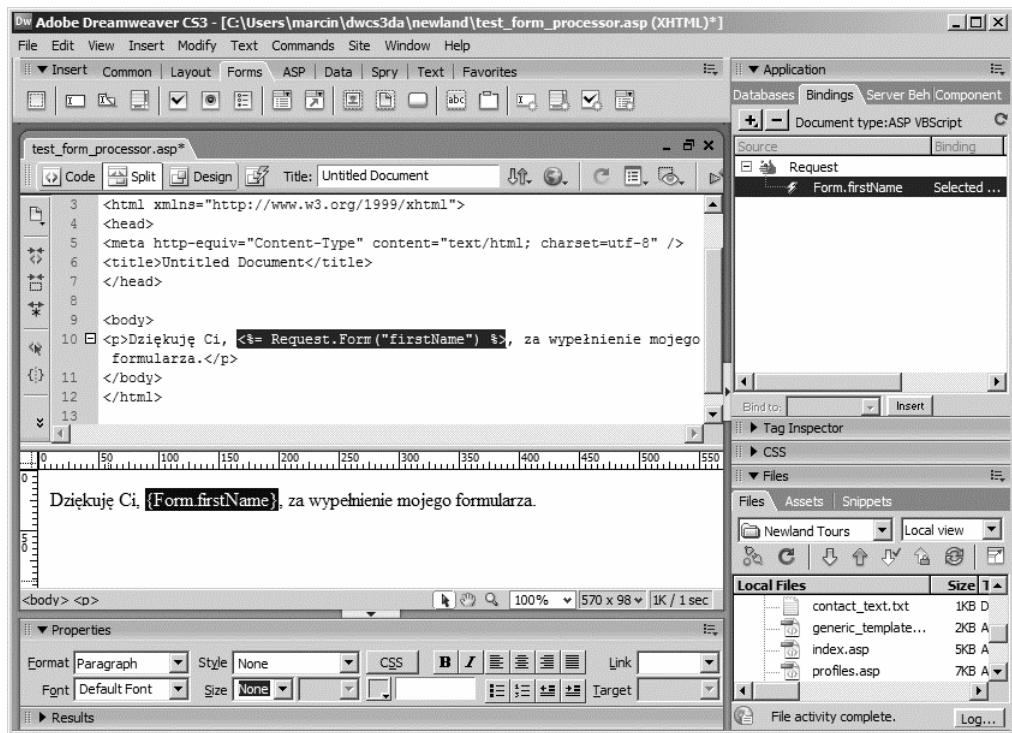
W przypadku ASP wybraliśmy zmienną *Request*. W ASP obiekt *Request* stosowany jest do pobrania informacji z podanej lokalizacji. W oknie dialogowym określiliśmy wtedy *Request.Form*, co nakazuje ASP „zajrzeć” do obiektu *Request* w poszukiwaniu wartości pola formularza. Na koniec podaliśmy nazwę tego pola. Nakreśliśmy programowi Dreamweaver (z ASP) plan działania, mający na celu znalezienie wartości pola `firstName`.

W ColdFusion i PHP określiliśmy pole formularza, co w tym przypadku jest wystarczające do odnalezienia właściwego miejsca (nie ma tu potrzeby określania obiektu *Request*). Następnie podaliśmy nazwę zmiennej. Podsumowując, ponownie dostarczyliśmy programowi Dreamweaver (z ColdFusion lub PHP) plan działania do znalezienia wartości pola `firstName`.

Jednak do tej pory wskazaliśmy programowi Dreamweaver jedynie to, *jak* znaleźć zmienną. Nie *zaządaliśmy* jej znalezienia ani nie nakazaliśmy wykonania na tej zmiennej żadnego działania.

6. Upewnij się, że w palecie *Bindings* wybrana została zmienna `Form.firstName (ASP)` lub `firstName` (ColdFusion lub PHP), i kliknij znajdujący się poniżej przycisk *Insert (Wstaw)*.

Na stronie, pomiędzy przecinkami, pojawi się niebieskie zaznaczenie `{Form.firstName}`. Zaznaczenie tekstu kolorem niebieskim oznacza w programie Dreamweaver obecność zawartości ulegającej zmianie. Przy wyświetlaniu strony w przeglądarce takiego zaznaczenia nie będzie. Nie wyświetli się również tekst `{Form.firstName}`, w zamian pojawi się tu imię użytkownika.



Choć `{Form.firstName}` wygląda jak kod, tak naprawdę jest to pseudokod. Wygląda on tak samo, niezależnie od rodzaju serwera. Możemy założyć, że Adobe używa pseudokodu do utworzenia ogólnego, opisowego języka, służącego do przekazania, co w rzeczywistości zawiera element ulegający zmianie. Założenie to jest słuszne, jeżeli tylko nie próbujemy zastosować tej składni w prawdziwym kodzie.

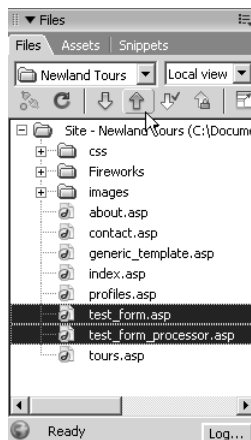
Jeśli zerkniemy w rzeczywisty kod skryptu, jaki został umieszczony na serwerze, zobaczymy, że został tam dodany ciąg `<%= Request.Form("firstName") %>` (ASP), `<cfoutput>#form.firstName#</cfoutput>` (ColdFusion) lub `<?php echo $_POST ['firstName']; ?>` (PHP). Jest to ten sam kod, który podałem wcześniej w tym rozdziale, z jednym małym wyjątkiem w przypadku ASP.

Możemy nakazać IIS przesłanie na wyjście wyrażenia przy użyciu obiektu `Response`. Najpopularniejszym zastosowaniem obiektu `Response` jest `Response.Write()`. Polecenie to nakazuje IIS wstawienie do dokumentu treści objętej nawiasami. Pomijając kilka niuansów, `Response.Write()` jest mniej więcej odpowiednikiem `<cfoutput>` lub `echo`. `Response.Write()` i jest na tyle popularne, że doczekało się skrótu. Gdy zobaczymy blok kodu ASP rozpoczynający się wyrażeniem `<%=` zamiast zwykłego `<%`, oznacza to `<% Response.Write()`. Innymi słowy, poniższe dwa wiersze wyrażają dokładnie to samo:

```
<% Response.Write(Request.Form("firstName")) %>
<%= Request.Form("firstName") %>
```

Podsumujmy czynności z dwóch ostatnich kroków. Wskazaliśmy programowi Dreamweaver (z ASP, ColdFusion lub PHP), jak znaleźć wartość pola `firstName`, używając w tym celu przycisku `+` na palecie *Bindings*. Następnie wstawiliśmy to powiązanie na stronę, co poinstruuje ASP, ColdFusion lub PHP, jakiej zmiennej mają użyć i co z nią mają zrobić (wstawić jej wartość do dokumentu).

7. Zapisz i zamknij wszystkie otwarte dokumenty. W paletce *Site*, trzymając wciśnięty klawisz *Shift*, zaznacz strony `test_form.asp` i `test_form_processor.asp`. Kliknij przycisk *Put File(s)* na pasku narzędzi w górnej części palety.



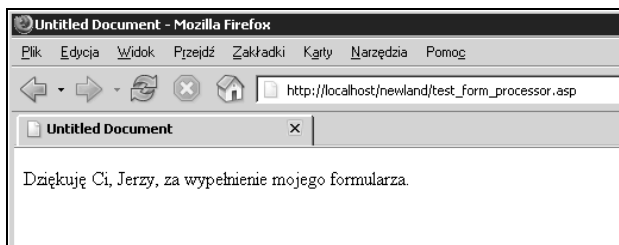
Nie możemy przetestować witryny, dopóki nie zostanie ona uruchomiona za pośrednictwem serwera, a nasz serwer nie jest naszą witryną lokalną. Żeby przetestować witrynę, musimy przesłać pliki na serwer. Jeżeli zostanie wyświetlone pytanie, czy przesłać pliki zależne, zaznacz pole opcji ustawiającej niewyświetlanie tego okna dialogowego i kliknij przycisk *No*, ponieważ nie mamy plików zależnych.



Jest to czynność, o której stale zapominam. Jeśli w czasie pracy pojawi się niespodziewany błąd, Twoim pierwszym krokiem powinno być sprawdzenie, czy wszystkie wymagane pliki zostały przesłane na serwer.

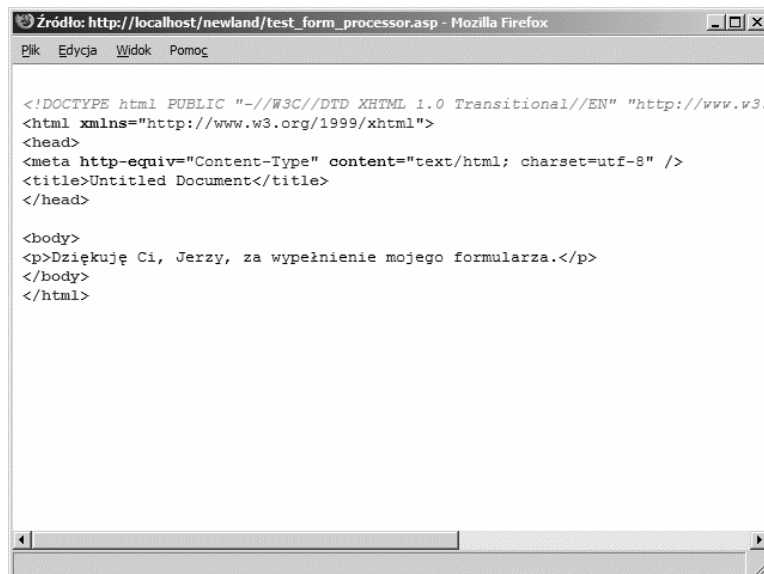
8. Zaznacz w palecie *Site* stronę *test_form.asp* i naciśnij klawisz *F12*, aby przetestować ją w przeglądarce. Po wczytaniu strony wpisz swoje imię w pole formularza i kliknij *Wyslij*.

Zostaniesz przekierowany na stronę *test_form_processor.asp*. Mam nadzieję, że (tak jak się spodziewałeś) na ekranie pojawiło się imię wpisane przez Ciebie do formularza.



9. We wciąż otwartym oknie przeglądarki wybierz z menu *Widok/Źródło strony* (lub odpowiadające temu polecenie w Twojej przeglądarce). Spójrz na wiersz objęty znacznikami `<p>`.

To interesujące. Kod dynamiczny został całkowicie usunięty! Strona ta zawiera statyczny kod XHTML. Nawet element zmienny (imię) wygląda tak, jakby był wpisany ręcznie do kodu. Ale oczywiście wiemy, że było inaczej.



Przeglądając kod wyjściowy, dochodzimy do zasadniczej kwestii. Strona, którą wpisujemy w programie Dreamweaver, różni się od strony widocznej w przeglądarce użytkownika, mimo że obie noszą tę samą nazwę (i wciąż oczywiście mają ze sobą wiele wspólnego).

Różnica między dwiema wersjami tej strony wynika z tego, że oryginalny kod ASP (ColdFusion, PHP) został przetworzony i usunięty, a jego wartości wyjściowe zostały dołączone do kodu XHTML.

Między dwiema wersjami strony występują również pewne podobieństwa: całość standardowego kodu XHTML znajdującego się w oryginalnej wersji, włącznie ze znacznikami `<body>` i `<p>`, oraz większość tekstu pozostały niezmiennione w wersji wyjściowej.

Czego się nauczyłeś?

W czasie tej lekcji:

- poznałeś związki między warstwą prezentacyjną, logiczną i zawartością (strony 86 – 89),
- poznałeś wady i zalety pięciu głównych technologii serwerowych (strony 90 – 92),
- skonfigurowałeś lokalny serwer WWW i serwer aplikacyjny (strony 93 – 104),
- zdefiniowałeś witrynę dynamiczną w programie Dreamweaver (strony 105 – 110),
- utworzyłeś formularz WWW (strony 111 – 114),
- utworzyłeś stronę zbierającą i wyświetlającą dane z formularza WWW (strony 114 – 120).