

Po prostu HTML 4 (wydanie II)

Autor: Elizabeth Castro

Tłumaczenie: Maciej Gdula, Andrzej Kierzkowski,

Wojciech Oliwa, Piotr Rajca, Zofia Sobocińska

ISBN: 83-7197-325-X

Format: B5, 408 stron

Data wydania: 07/2000

Cena książki: 39.00 zł



Wydawnictwo Helion
ul. Chopina 6, 44-100 Gliwice, POLAND
telefon: (32) 230-98-63, 231-22-19
fax: (32) 230-98-63 w.10
mail: helion@helion.com.pl



Ogromne zainteresowanie czytelników, pochlebne recenzje, a także nowe możliwości języka HTML skłoniły autorkę do uzupełnienia wiadomości zawartych w poprzednim wydaniu tej książki. "Po prostu HTML 4. Wydanie II" została opracowana z myślą o Czytelniku, który chciałby nauczyć się języka HTML. Jest ona doskonałym przewodnikiem dla początkujących. Przystępny, bogato ilustrowany wykład poprowadzi Cię, krok po kroku, przez wszystkie zagadnienia języka HTML 4, prezentując kolejne i wskazując właściwe czynności.

-  [Zobacz przykładowy rozdział](#)
-  [Spis treści](#)
-  [Jeżeli znasz tę książkę oceń ją](#)
-  [Aktualny cennik książek e-mailem](#)
-  [Książki i "3D" Online](#)
-  [Informacje o nowościach e-mailem](#)
-  [Zamów najnowszy katalog](#)
-  [Zobacz opis dołączonego pliku](#)

© Helion 2000

Co umożliwiają formularze?

- Uzyskanie informacji zwrotnej.
- Stworzenie książki gości.
- Przeprowadzanie ankiet.
- Podglądanie, kto wizytuje Twoje strony.
- Samoobsługę.
- I wiele więcej!

Jak do tej pory wszystko, czego nauczyłeś się o HTML-u pozwalało Ci jedynie na przekazywanie informacji Twoim gościom. W tym rozdziale nauczysz się definiować formularze, które umożliwiają przekazywanie informacji Tobie.

Na formularze składają się dwie zasadnicze części: struktura, czy też powłoka, zawierająca pola, etykiety, przyciski widziane przez użytkownika na stronie i, miejmy nadzieję, wypełniane oraz obsługujący skrypt, który odbiera informację i przetwarza ją do postaci, którą możesz odczytać lub przeliczyć.

Definiowanie struktury formularza jest zupełnie proste i podobne do tworzenia innych fragmentów strony WWW. Możesz wstawiać pola tekstowe, pola haseł, przyciski radiowe, pola wyboru, menu, bloki tekstu, a nawet aktywne ilustracje. Każdemu elementowi nadajesz nazwę, która spełnia rolę identyfikatora danych w trakcie obróbki. Konstruowanie formularzy zostało omówione na stronach 195-212.

Przetwarzanie danych z formularza jest trochę bardziej skomplikowane. Podstawowe narzędzie, czyli skrypty CGI, zazwyczaj pisane są w języku Perl lub innym języku programowania. Co prawda, programowanie w języku Perl, ba, nawet wyjaśnienie, jak wykorzystać istniejące skrypty, wykracza poza ramy tej książki, tym niemniej opisałam kilka gotowych skryptów CGI, które pomogą Ci zacząć (patrz strona 193).

Jeśli wszystko to wydaje Ci się zniechęcające lub Twój dostawca Internetu nie zezwala na wykorzystywanie skryptów CGI, to możesz zdecydować się na przesyłanie danych z formularzy pocztą elektroniczną (patrz strona 196), skorzystać z publicznych serwerów obsługujących formularze opisanych na stronie 197.

Skrypty CGI

Nazwa skrypt CGI może sprawić, że osoby nie będące programistami, szybko odłożą tę książkę i zapomną o formularzach. Ale zaczekajcie. Co to jest CGI? Co to jest skrypt? Nie jest to aż tak trudne, jak się wydaje. Po pierwsze, słowo skrypt zastępuje słowo program, taki jak Microsoft Word czy Adobe Photoshop. Oczywiście skrypty, które wykorzystujesz do obróbki danych z formularzy są o niebo prostsze od komercyjnych aplikacji kosztujących setki dolarów, lecz funkcjonują w podobny sposób.

CGI, skrót od Common Gateway Interface – Uniwersalny Interfejs Komunikacyjny, jest po prostu ujednoliconą metodą przesyłania informacji pomiędzy serwerem, a skryptem. Podsumowując: skrypt CGI to program (zazwyczaj napisany w języku programowania nazywanym się Perl) porozumiewający się z serwerem w standardowy (CGI) sposób.

Językiem najczęściej używanym do pisania skryptów CGI jest Perl. Dzieje się tak, gdyż jest to język łatwy do przeniesienia z jednej platformy na drugą, doskonale nadaje się do przekształcania danych w komunikaty zrozumiałe dla człowieka, a oprócz tego ma reputację odlotowego języka programowania (naprawdę!). Programiści korzystający z Perl uwielbiają przechwalać się, jak to w jednej linijce kodu, i to zapisanej na tysiąc różnych sposobów, mogą zrobić niemal wszystko.

Jednak do tworzenia skryptów CGI możesz używać także innych języków programowania: C++, tcl, VisulaBasic-a, a nawet języka AppletScript i skryptów wsadowych systemu MS-DOS. Jeśli znacz którykolwiek z tych języków, to użyj właśnie jego, jeśli nie – skorzystaj z Perla-a.

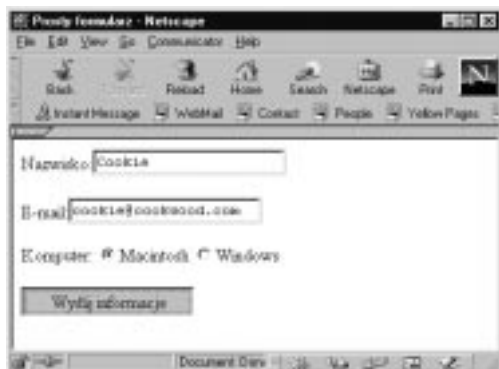
Programiści korzystający z języka Perl mają jedną wspólną cechę – lubią się dzielić swoimi programami. Wszędzie na WWW możesz znaleźć tysiące gotowych programów napisanych w tym języku (patrz strona 192). Wiele z nich jest dostępnych za darmo, inne niestety nie.



Rys. 11.1. Oto prosty formularz zawierający dwa pola tekstowe, jeden zestaw przycisków radiowych i przycisk dostarczający. Słowa Nazwisko, E-mail, Komputer, Macintosh i Windows to etykiety. Nie mają one wpływu na dane, które zbierane są w inny sposób



Rys. 11.2. A oto kod źródłowy formularza z rysunku 11.1. Na razie zwróć uwagę na atrybuty NAME i VALUE (nauczysz się definiować elementy formularza w dalszej części tego rozdziału). Zauważ, po pierwsze, że wszystkie elementy mają atrybut NAME, a nie wszystkie atrybut VALUE. Atrybut VALUE definiuje dane przesyłane z tego elementu do serwera. Niektóre elementy umożliwiają użytkownikowi wpisanie dowolnych wartości (np. pola tekstowe), a inne nie (np. przyciski radiowe). Elementy formularza, które się zaznacza lub wybiera, muszą mieć zdefiniowaną wartość za pomocą atrybutu VALUE



Rys. 11.3. Kiedy użytkownik wprowadzi informacje do pól tekstowych i wybierze przycisk radiowy, pary nazwa-wartość są gotowe. Kliknięcie przycisku dostarczającego (nazwany tu Wyślij informacje) prześle pary nazwa-wartość do skryptu CGI na serwerze

Etykieta	Nazwa	Wartość
Nazwisko	nazwisko_goscia	Cookie
E-mail	email_goscia	Cookie@ cookwood.com
Komputer	komputer	Mac
-	submit	Wyślij informacje

Rys.11.4. Oto bieżące pary nazwa-wartości, które zostaną przesłane po naciśnięciu przycisku dostarczającego (rysunek 11.3). Nie pomył etykiety z nazwą elementu. Zauważ też, że wartość pierwszych dwóch pól odpowiada temu, co wpisał użytkownik. Wartość przycisku radiowego – Mac (nazwa), nie Macintosh (etykieta) – została ustalona przeze mnie, autora strony WWW (patrz rysunek 11.2), a użytkownik jedynie kliknął przycisk (nic nie wpisywał)

```
http://www.cookwood.com/cgi-bin/
cgi.script.url?nazwisko_goscia=
Cookie&email_goscia=
Cookie@cookwood.com&komputer=
Mac&submit=Wyślij+informacje
```

Rys.11.5. Jeśli wykorzystujesz metodę GET, to dane zostaną przesłane do skryptu CGI w powyższy sposób. Zauważ, że łańcuch jest oddzielony od adresu URL pytajnikiem (?). Następnie każda nazwa jest połączona ze swoją wartością znakiem równości (=). A w końcu, znak & rozdziela pary nazwa-wartość

Co robią skrypty CGI?

Każdy element Twojego formularza posiada *nazwę* i *wartość*. Nazwa identyfikuje przesyłane dane. Może to być np. *nazwisko_goscia*. Wartością są dane (powiedzmy *Castro*), które mogą być wpisywane przez Ciebie, projektanta strony WWW, lub przez Twojego gościa (**rysunki 11.1 i 11.2**). Gdy użytkownik kliknie przycisk dostarczający (lub aktywną ilustrację – patrz strona 196), do serwera przesyłane są pary nazwa-wartość każdego elementu formularza (może to być np. *nazwisko_goscia=Castro*). Skrypt CGI odbiera wszystkie pary nazwa-wartość i segreguje je tak, by człowiek (lub baza danych) mógł je odczytać i zrozumieć.

Skrypt CGI ma zazwyczaj dwie funkcje. Pierwszą z nich jest pobranie wszystkich par nazwa-wartość i rozdzielenie ich na niezależne elementy. Drugą funkcją jest wykonanie na tych informacjach jakichś operacji – wyświetlenie ich, pomnożenie wartości pól, przesłanie wiadomości z potwierdzeniem pocztą elektroniczną, zapisanie w pliku na serwerze lub zrobienie czegośkolwiek innego. Istnieją skrypty CGI tworzące książki gości, obsługujące pogawędki, liczniki, gry, wysyłające kartki pocztowe, wybierające losowo obrazki, współpracujące z bazami danych oraz wiele wiele innych.

Bezpieczeństwo

Zanim się rozemocjonujesz, powinieneś się dowiedzieć, że skrypty CGI mogą „udostępnić” serwer intruzom. To z tego powodu wielu dostawców Internetu nie zezwala swoim użytkownikom na korzystanie ze skryptów CGI. Jeśli jest też tak w Twoim przypadku, to alternatywą może być wykorzystanie skryptu z wyspecjalizowanego serwera, tak jak to opisałam na stronie 197.

Zaś jeżeli Twój dostawca Internetu zezwala na uruchamianie skryptów CGI, to powinieneś poczytać na temat zabezpieczeń. Możesz zacząć od strony <http://www.go2net.com/poeple/paulp/cgi-security/>, aby dowiedzieć się, jakie skrypty CGI mogą okazać się zdraдлиwe.

Jeśli Twój dostawca Internetu zgadza się, byś korzystał ze skryptów CGI, następnym krokiem powinno być zdobycie skryptu. Możesz rozpocząć do skopiowania skryptów dołączonych do tej książki (patrz strona 193). Możesz także napisać własny skrypt lub zaadaptować do swoich potrzeb jeden z tysięcy skryptów dostępnych na WWW. Niektóre z nich są dostępne za darmo, inne wymagają jakiejś formy wynagrodzenia programisty. Choć na WWW istnieje setki miejsc, gdzie można znaleźć skrypty CGI, ja osobiście preferuję cztery z nich.

CGI Resource Index publikowany przez Matt's Scripts Archive, Inc. (patrz poniżej) zawiera setki połączeń do przeróżnych skryptów CGI napisanych w Perl-u, dokumentacji, książek, artykułów prasowych, programistów oraz propozycji pracy. Witrynę tę znajdziesz pod adresem www.cgi-resources.com (patrz **rys. 11.6**).

Witryna, znana także pod nazwą MSA, jest jednym z najbardziej popularnych magazynów skryptów na WWW. Możesz ją znaleźć pod adresem www.worldwidemart.com/scripts/. Witryna ta stworzona została przez nastolatka – Matta Wrighta – i zawiera wiele przydatnych i bezpłatnie dostępnych skryptów CGI napisanych w Perl-u.

Innym sławnym źródłem skryptów CGI napisanych w Perl-u jest witryna Extropia.com, stworzona przez Selenę Sol oraz Gunthera Birznieksa. Choć ostatnio witryna ta nabiera bardziej komercyjnego wyglądu, to jednak wciąż można na niej znaleźć wiele ciekawych skryptów do wykorzystania. Autorzy oferują bezpłatną pomoc przy korzystaniu z nich. Witrynę tę znajdziesz pod adresem www.extoripa.com.



Rys. 11.6. Na witrynie CGI Resource Index znajdziesz ponad 1600 gotowych do skopiowania i uruchomienia skryptów CGI. Niektóre z nich są darmowe, inne nie. Witrynę tę znajdziesz pod adresem <http://www.cgi-resources.com>

Choć ta witryna autorstwa Darryla Burgdorfa niewątpliwie nie jest najbardziej popularna, ani nie udostępnia największej ilości skryptów, to bez wątpienia zawiera najlepiej udokumentowaną i najłatwiejszą do implementacji grupę skryptów CGI napisanych w Perl-u. Znajdziesz ją pod adresem www.awsd.com/scripts/.



Rys. 11.7. Oto jedna z wersji niesłyszanie istotnego skryptu służącego do przetwarzania danych przesyłanych z formularza. Cały kod skryptu jest istotny, może za wyjątkiem ostatnich czterech linii, które służą do wyświetlania nadesłanych informacji w postaci prostej strony WWW

Wykorzystanie skryptów dołączonych do tej książki

Do niniejszej książki dołączyłam grupę kilku bardzo prostych skryptów CGI, które możesz wykorzystać do obsługi danych przesyłanych przez użytkowników do Twojej witryny. Skrypty te zostały zaprojektowane po to, aby pomóc Ci zrozumieć, jak działają formularze i umożliwić Ci korzystanie z nich. Aby choć trochę skorzystać z tych skryptów, będziesz się musiał trochę znać na Perlu (lub mieć ochotę do poznania tego języka). Jeśli bardziej Cię interesuje uzyskanie danych z formularzy, to prawdopodobnie powinieneś skorzystać z wysyłania danych pocztą elektroniczną (patrz strona 196).

Aby wykorzystać skrypty dołączone do tej książki:

1. Z serwera FTP Wydawnictwa HELION skopiuj plik <ftp://ftp.helion.com/ksiazki/pphtml4cgi.zip>, a następnie rozpakuj go.
2. Przeczytaj i wykonaj instrukcje zamieszczone na następnej stronie, w sekcji pt. „Przygotowywanie skryptów do użycia”.
3. W formularzu stwórz odwołanie do skryptu (umieszczonego na serwerze), tak jak opisałam to w kroku 2. na stronie 195.
4. Poczekać na użytkowników, którzy odwiedzą Twą witrynę, wypełnią formularz i klikną przycisk „Wyślij”.

✓ Rada

- Jeśli chciałbyś zdobyć więcej skryptów o większych możliwościach, to zajrzyj na stronę 192, sekcja pt. „Zdobywanie skryptów”.

Przygotowanie skryptu do użycia

Nie ważne, czy napisałeś skrypt sam, czy wykorzystasz napisany przez kogoś innego, musisz wykonać kilka czynności, aby przygotować go do użycia.

Adaptowanie cudzych skryptów

Jeśli chcesz korzystać ze skryptu Perl ściągniętego np. z Sieci, musisz otworzyć go i zobaczyć, jakie zmienne oraz ścieżki dostępu są w nim wykorzystywane. Pewnie będziesz musiał niektóre z nich zmienić, by dostosować je do Twojego przypadku.

Przeniesienie skryptów na serwer

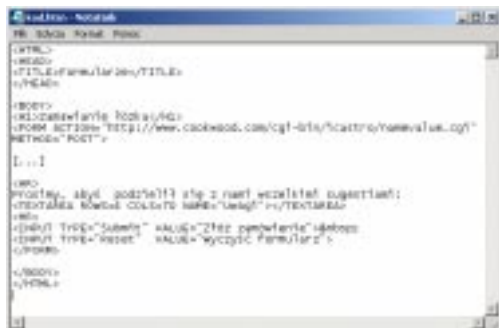
Następnym krokiem jest przekopiowanie skryptu na serwer, zazwyczaj za pomocą programu ftp np. Fetch lub WS_FTP (patrz strona 349). Niektóre serwery wymagają, by wszystkie skrypty znajdowały się w katalogu cgi-bin. Inne udostępniają każdemu użytkownikowi osobiste katalogi cgi-bin. Jeszcze inne pozwalają na przechowywanie skryptów CGI, gdzie tylko zechcesz, jeśli tylko nadasz im w celu identyfikacji odpowiednie rozszerzenie. Zapytaj swojego dostawcę Internetu.

Prawa dostępu

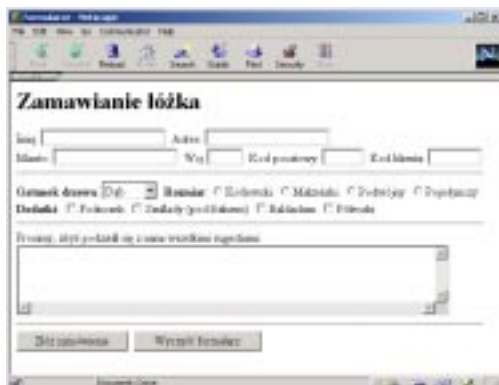
Jeśli Twoja strona znajduje się na serwerze Unixowym, to będziesz musiał wykorzystać program chmod, aby skrypty CGI udostępnić i zaznaczyć jako wykonywalne. Będziesz mógł zamiast tego użyć funkcji programu ftp, o ile ją udostępnia (np. WS-FTP, CuteFtp).

Dołączenie do swojego formularza!

Tak więc, masz swój skrypt CGI na serwerze gotowy do użycia. Ostatnią rzeczą, która pozostała do zrobienia to dołączenie do formularza (patrz strona 195).



Rys. 11.8. Każdy formularz składa się z trzech części: znacznika **FORM**, elementów formularza, w których użytkownicy podają lub wybierają informacje, oraz przycisku przesyłającego, który umożliwia przekazanie podanych w formularzu informacji na serwer (może to też być aktywny obrazek)



Rys. 11.9. Formularz stanowi doskonały sposób na pobranie informacji od użytkowników

Tworzenie formularza

Każdy formularz składa się z trzech ważnych elementów: znacznika **FORM** – zawierającego adres URL skryptu CGI, służącego do obsługi przesyłanych danych – elementów formularza (takich jak pola tekstowe i listy) oraz z przycisku powodującego przesłanie danych podanych w formularzu.

Aby stworzyć formularz:

1. Wpisz **<FORM METHOD=POST>**.
2. Wpisz **ACTION="skrypt.url"**, gdzie *skrypt.url* jest położeniem skryptu używanego do obsługi formularza na serwerze (patrz strony 190-194).
3. Stwórz zawartość formularza, zgodnie z informacjami podanymi na stronach 196-218. Nie zapomnij umieścić na formularzu przycisku **SUBMIT** (patrz strona 208) lub aktywnego obrazka (strona 212).
4. Zakończ formularz wpisując **</FORM>**.

✓ Rady

- Aby użytkownik mógł wysłać dane podane w formularzu, będziesz musiał umieścić na nim bądź to przycisku **SUBMIT** (jeśli formularz zawiera pola, przycisku lub inne elementy, które użytkownik może wypełnić), bądź aktywny obrazek. Więcej informacji na temat przycisków **SUBMIT** znajdziesz na stronie 208, pod hasłem „Przycisk dostarczający”. Więcej informacji na temat aktywnych obrazków znajdziesz na stronie 212 – „Aktywne obrazki”.
- W celu bardziej precyzyjnego rozmieszczenia elementów formularza, możesz wykorzystać tabele. Więcej informacji na ich temat znajdziesz w rozdziale 9.
- Do przesłania danych z formularza możesz użyć także metody **GET**. Jednak metoda ta ogranicza wielkość informacji, jakie można przesyłać, dlatego polecałabym stosowanie metody **POST**.

Przesyłanie danych pocztą elektroniczną

Jeśli nie masz ochoty na bawienie się skryptami CGI i możesz pogodzić się z otrzymywaniem danych, które nie będą idealnie sformatowane (lub wstępnie przetworzone przez skrypt), to możesz skorzystać z możliwości przesyłania danych z formularza pocztą elektroniczną pod wskazany adres.

Aby przesłać dane pocztą elektroniczną:

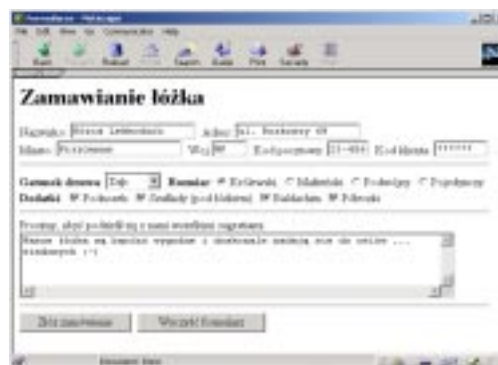
1. Wpisz **<FORM METHOD=POST**.
2. Wpisz **ENCTYPE="text/plain"**, aby przesyłane dane były odpowiednio zakodowane.
3. Wpisz **ACTION="mailto:adres@witryna.com.pl"**, gdzie *adres@witryna.com.pl* jest Twoim adresem poczty elektronicznej, pod który mają być przesyłane dane.
4. Wpisz **>**.
5. Stwórz zawartość formularza, tak jak to opisałam na stronach 198-218.
6. Wpisz **</FORM>**.

✓ Rady

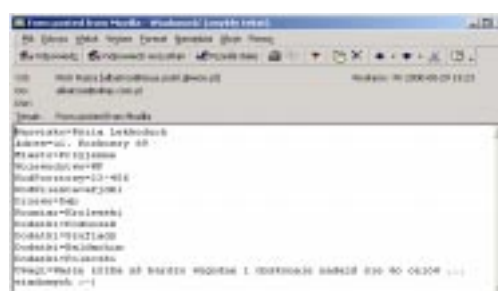
- Ta technika może być wykorzystywana jedynie w przeglądarkach w wersjach 4 lub późniejszych.
- Kiedy użytkownik kliknie przycisk SUBMIT, to zostanie wyświetlony komunikat ostrzegający, że ich adres e-mail będzie przesłany wraz z danymi oraz że dane nie będą w żaden sposób zaszyfrowane. Aby dane zostały przesłane, użytkownicy muszą kliknąć przycisk OK.
- Więcej informacji na temat modyfikacji tematu wiadomości, znajdziesz na stronie 317, w sekcji zatytułowanej „Sprytne połączenia do poczty”.
- Zawartość pola Od wiadomości zależy od informacji, jakie użytkownik podał we właściwościach przeglądarki. Informacja ta może nie być prawdziwa.



Rys. 11.10. Jedną z metod ominięcia problemów związanych z wykorzystywaniem skryptów CGI jest przesyłanie danych z formularza pocztą elektroniczną



Rys. 11.11. Formularz, który zostanie przesłany pocztą elektroniczną wygląda dokładnie tak samo, jak formularze obsługiwane przez skrypty CGI



Rys. 11.12. Dane zostają zapisane w treści wiadomości poczty elektronicznej. Zauważ, że domyślna linia tematu zależy od przeglądarki używanej przez użytkownika (w tym przypadku był to Netscape, o nazwie kodowej Mozilla). W polu Od powinien być podany adres poczty elektronicznej nadawcy. W polu Do zostanie wpisany adres, który podałeś w znaczniku FORM



Rys. 11.13. Oto domowa strona Response-O-Matic (www.reponse-o-matic.com). Jest to jeden z lepszych serwisów obsługujących formularze, jakie widziałam



Rys. 11.14. Na początek Response-O-Matic zada kilka pytań, a następnie stworzy szablon pokazany na rysunku 11.15



Rys. 11.15. Zachowaj ten szablon na twardym dysku i zmodyfikuj, jeśli to będzie potrzebne (nie zmieniaj ukrytych pól!)

Wykorzystanie serwisów obsługujących formularze

Kolejną opcją dla osób, które nie chcą lub nie mogą skorzystać ze skryptów CGI, są serwisy obsługujące. Istnieje kilka takich firm, które tworzą formularze, umożliwiają użytkownikom dostęp do skryptów CGI, które obsługują te formularze, lub przetwarzają je bezpośrednio i przesyłają wyniki pocztą elektroniczną. Zapłatą za usługi tego typu jest zazwyczaj umieszczanie różnego typu reklam.

Aby skorzystać z serwisu obsługującego:

1. Odszukaj serwis obsługujący przy pomocy wyszukiwarki Yahoo! lub AtlaVista.
2. Przeglądaj ich miejsce, by znaleźć odpowiedzi na następujące pytania:

Kto tworzy formularze: Ty czy oni?

Co musisz zrobić dla nich w zamian za obsługę formularzy?

Czy zezwalają na wykorzystanie formularzy w celach komercyjnych, czy tylko w prywatnych?

3. Aby uruchomić formularz postępuj zgodnie z ich instrukcjami. Przeczytaj poprzednie strony tego rozdziału, aby dowiedzieć się, jak definiować elementy formularza.

✓ Rady

- Większość serwisów obsługujących przesyła Ci zebrane informacje przez e-mail.
- Serwisy obsługujące różnią się znacznie między sobą. Choć w zasadzie wszystkie z nich przetwarzają formularze w zamian za reklamę, to jednak niektóre z nich są bardziej natrętne od innych. Dla przykładu, witryna Response-o-Matic wymaga wyświetlenie swojej reklamy oraz adresu e-mail wyłącznie na stronie z podziękowaniami, a nie na samym formularzu. Oznacza to, że będziesz mógł zaprojektować formularz, tak aby dokładnie pasował do reszty witryny.

Pola tekstowe

Pola tekstowe zawierają jedną linię dowolnego tekstu i zazwyczaj są wykorzystywane do wpisania nazwiska, adresu i innych podobnych informacji.

Zdefiniowanie pola tekstowego

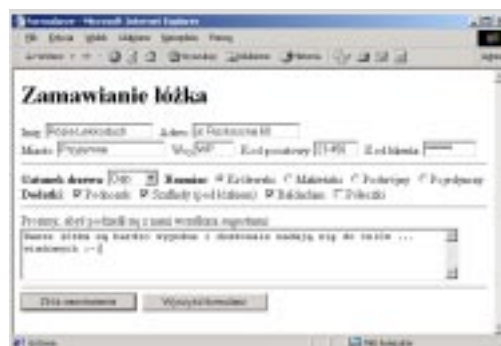
1. Jeśli chcesz, to opisuj pole tekstowe podając jego etykietę, np. **Nazwisko:**.
2. Napisz **<INPUT TYPE="text"**.
3. Nadaj polu nazwę, pisząc **NAME="nazwa"**, gdzie *nazwa* to tekst, który będzie służył do identyfikacji wprowadzonych przez użytkownika danych na serwerze i w skrypcie.
4. Jeśli chcesz, to wpisz **VALUE="wartość"**, gdzie *wartość* to dane, jakie będą początkowo wyświetlone w polu i które zostaną przesłane na serwer, jeśli użytkownik nie wpisze w polu niczego innego.
5. Jeżeli chcesz, zdefiniuj rozmiar pola wpisując **SIZE="n"**; zastąp *n* szerokością pola wyrażoną w znakach.
6. Jeśli chcesz, wpisz **MAXLENGTH="n"**, zastępując *n* maksymalną liczbą znaków, które można wprowadzić w polu.
7. Napisz znak **>**.

✓ Rada

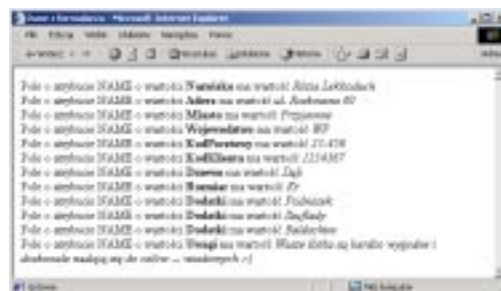
- Nawet jeśli nie wpisze w polu żadnej wartości oraz jeśli nie zostanie określona wartość domyślna pola, to i tak odpowiadający mu atrybut NAME zostanie przesłany na serwer (wraz z niezdefiniowaną, pustą wartością atrybutu VALUE).
- Domyślna wartość atrybutu SIZE wynosi 20. Użytkownik może wpisać więcej znaków niż mieści się w ramce, ale nie więcej niż określono atrybutem MAXLENGTH. Do wpisywania większych ilości tekstu, znacznie lepiej nadają się obszary tekstowe (patrz strona 200).



Rys. 11.16. Choć określenie wartości atrybutu NAME dla każdego pola tekstowego jest niezbędne, to atrybut VALUE można stosować, wtedy gdy chcesz określić domyślną wartość pola



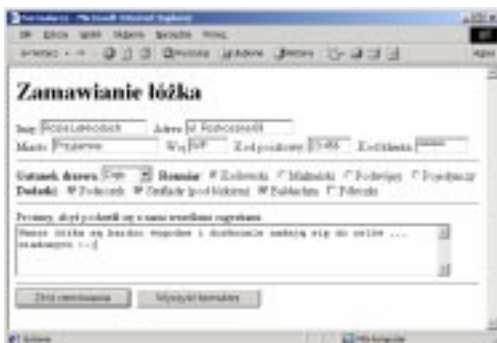
Rys. 11.17. Pola tekstowe mogą mieć różną wielkość, aby można w nich podawać różnego typu informacje



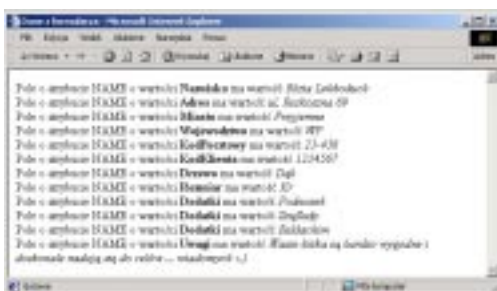
Rys. 11.18. Bardzo ważne jest nadawanie polom tekstowym opisowych nazw (używając do tego atrybutu NAME). Dzięki temu będziesz wiedział, jakie informacje otrzymujesz



Rys. 11.19. Atrybut NAME identyfikuje hasło, gdy dane z formularza są przetwarzane przez skrypt CGI



Rys. 11.20. Gdy użytkownik wpisuje hasło w formularzu, to jest ono „zastaniane” gwiazdkami



Rys. 11.21. Po przetworzeniu przez skrypt obsługujący dane z formularza, zawartość pola hasło przybiera formę normalnego tekstu. Pola tego typu nie zapewniają wysokiego poziomu bezpieczeństwa (patrz przedostatnia rada)

Pole hasła

Jedyną różnicą pomiędzy polem hasła a normalnym polem tekstowym, polega na tym, że znaki wpisywane w polu hasła są wyświetlane jako gwiazdki. Dane nie są jednak szyfrowane przy przesyłaniu na serwer.

Zdefiniowanie pola hasła

1. Jeśli chcesz, opisz pole hasła. Na przykład napisz **Wprowadź hasło:**.
2. Wpisz **<INPUT TYPE="password">**.
3. Wpisz **NAME="nazwa"**, gdzie *nazwa* będzie tekstem służącym do identyfikacji danych na serwerze (i w skrypcie).
4. Jeśli chcesz, zdefiniuj rozmiar pola wpisując **SIZE="n"**, zastępując *n* szerokością pola wyrażoną w znakach.
5. Jeżeli chcesz, napisz **MAXLENGTH="n"**, zastępując *n* maksymalną liczbą znaków, które można wprowadzić w tym polu.
6. Napisz znak >.

✓ Rady

- Nawet jeśli w polu hasła nie zostaną podane żadne informacje, to i tak jego nazwa zostanie przesłana na serwer (z pustą, niezdefiniowaną wartością atrybutu VALUE).
- Także w tym typie pól można określić ich domyślną wartość (patrz punkt 4. na stronie 198). Jednak jest to sprzeczne z ideą i przeznaczeniem pól hasła.
- Jedynym rodzajem zabezpieczenia, jakie oferuje pole hasła, to zabezpieczenie przed osobami postronnymi, zaglądającymi przez ramię użytkownikowi wpisującemu hasło. Ponieważ dane nie są kodowane na czas przesyłania do serwera, średnio zaawansowany cracker może bez większego wysiłku odczytać hasło.
- Informacje dotyczące zabezpieczania stron hasłem znajdziesz na stronie 315.

Obszary tekstowe

Czasami chcesz dać użytkownikowi więcej miejsca na odpowiedź. W odróżnieniu od pól tekstowych (patrz strona 198) obszary tekstowe mogą zajmować obszar całej strony, poza tym ich objętość może się powiększać w miarę wpisywania do nich nowych informacji. Pola tego typu doskonale nadają się do wpisywania pytań oraz komentarzy.

Zdefiniowanie bloku tekstu

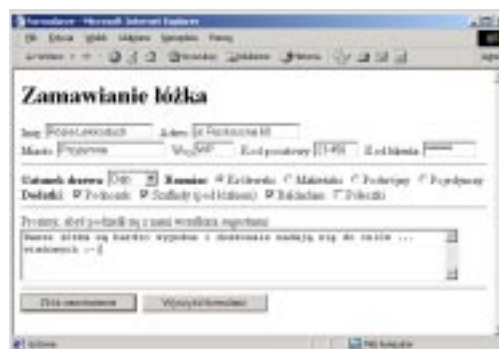
1. Jeśli chcesz, to wpisz tekst informujący o przeznaczeniu obszaru tekstowego.
2. Napisz **<TEXTAREA>**.
3. Wpisz nazwę pola wyboru: **NAME= "nazwa"**, gdzie *nazwa* będzie tekstem służącym do identyfikacji danych na serwerze (i w skrypcie).
4. Jeżeli chcesz, napisz **ROWS="n"**, gdzie *n* określa, na ile wierszy ma być wysokie pole tekstu. Domyślną wartością są 4 wiersze.
5. Jeśli chcesz, wpisz **COLS="n"**, gdzie *n* jest szerokością pola tekstu wyrażoną w znakach. Domyślną wartością jest 40 znaków.
6. Jeśli chcesz, by w trakcie wprowadzania tekstu linie były automatycznie dostosowywane do marginesów – napisz **WRAP**.
7. Napisz znak **>**.
8. Ewentualnie wprowadź domyślny tekst. W tym tekście nie wolno Ci używać jakichkolwiek znaczników HTML.
9. Wpisz **</TEXTAREA>**.

✓ Rady

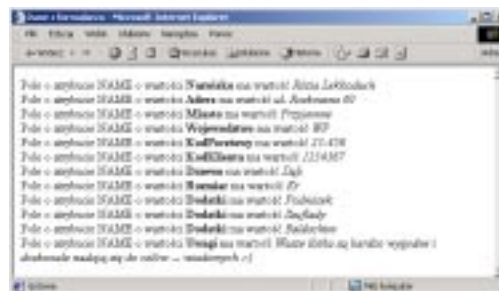
- W przypadku obszarów tekstowych wykorzystanie atrybutu **VALUE** nie ma sensu.
- Użytkownik może w bloku tekstowym wprowadzić do 32700 znaków. Jeśli będzie to potrzebne, pojawiają się paski przewijania.



Rys. 11.22. W znaczniku **TEXTAREA** nie używa się atrybutu **VALUE**. Domyślną wartość obszarów tekstowych podaje się wpisując ją pomiędzy otwierającym i zamykającym znacznikiem **TEXTAREA** (tak jak pokazano na rysunku)



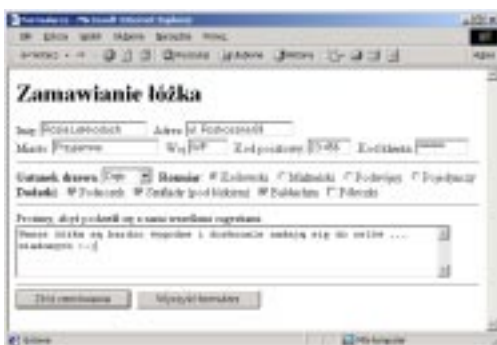
Rys. 11.23. Domyślny tekst można usunąć wprowadzając na jego miejsce nowy



Rys. 11.24. Obszary tekstowe doskonale nadają się do wypisywania dłuższych komentarzy i sugestii. Zazwyczaj stosuje się je w książkach gości oraz biuletynach informacyjnych



Rys. 11.25. W przypadku przycisków opcji atrybut **NAME** pełni podwójną rolę: kojarzy on dany przycisk z grupą przycisków oraz identyfikuje wartość po jej przesłaniu do skryptu. Kluczowe znaczenie ma atrybut **VALUE**, gdyż w przypadku przycisków opcji, użytkownik nie ma możliwości podania wartości



Rys. 11.26. Przyciski opcji tworzone są przy użyciu znaczników HTML. Natomiast ich etykiety (np. „Królewski”, „Mały” itp.) podawane są jako normalny tekst



Rys. 11.27. Zauważ, że na serwer jest przesyłana wartość atrybutu **VALUE** („Kr”), a nie tekst wyświetlany na stronie („Królewski”)

Przyciski opcji

Czy pamiętasz stare radio samochodowe z dużymi, czarnymi, plastikowymi przyciskami? Wcisnąłeś jeden przycisk, słuchałeś fal długich, wcisnąłeś drugi – słuchałeś UKF-u. Nigdy nie były wciśnięte dwa przyciski jednocześnie. Przyciski radiowe (ang. *radio buttons*) na formularzu zachowują się identycznie.

Definiowanie przycisków

1. Jeśli chcesz, to opisz w sekcji FORM grupę przycisków radiowych. Na przykład napisz **Wybierz jedną z następujących możliwości:**.
2. Wpisz **<INPUT TYPE="radio"**.
3. Wpisz **NAME="grupa"**. Grupa jest identyfikatorem danej grupy przycisków radiowych. Tylko jeden przycisk z grupy grupa może być w danej chwili zaznaczony.
4. Musisz zdefiniować wartość, inną dla każdego przycisku w grupie grupa. Wpisz więc **VALUE="wartość"**, gdzie *wartość* to słowo, skrót, kilka znaków itp., które zostaną przesłane do serwera jako dane, gdy przycisk będzie zaznaczony.
5. Jeżeli chcesz, aby przycisk był domyślnie zaznaczony po odebraniu strony WWW przez użytkownika wpisz **CHECKED**.
6. Napisz znak **>**.
7. Wprowadź tekst, który opisze użytkownikowi przycisk radiowy. Często jest to samo słowo lub zwrot, które wpisałeś dla VALUE, ale nie musi być.
8. Powtórz kroki od 2. do 7. dla wszystkich przycisków w grupie grupa przycisków.

✓ Rada

- Możesz przypisać przyciskom radiowym klawisze skrótu. Więcej informacji znajdziesz na stronie 201.

Pola wyboru

Podczas gdy przyciski radiowe pozwalają użytkownikowi na wybranie tylko jednej opcji z kilku w grupie, to pola wyboru (ang. *checkboxes*) są od siebie niezależne i użytkownik może zaznaczyć tyle pól wyboru, ile zechce. Podobnie jak w przypadku przycisków radiowych, tak i nazwa pola wyboru służy do zebrania pól w grupy.

Zdefiniowanie pól wyboru

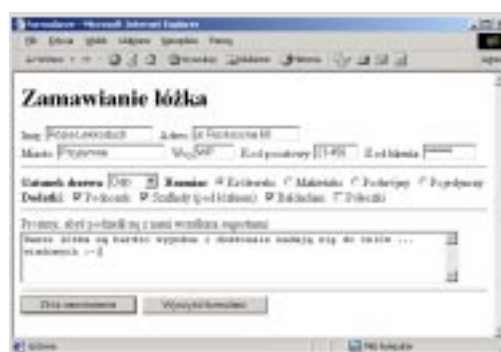
1. W sekcji FORM dokumentu HTML wprowadź tekst (jeśli chcesz) opisujący grupę pól wyboru. Na przykład napisz **Wybierz wszystkie odpowiadające Ci możliwości:**.
2. Wpisz `<INPUT TYPE="checkbox">`. Uwaga – nie ma spacji w słowie *checkbox*!
3. Nadaj polu nazwę pisząc `NAME="nazwa"`, gdzie *nazwa* to identyfikator danej grupy pól wyboru.
4. Zdefiniuj wartość dla każdego pola wyboru wpisując `VALUE="wartość"`. *Wartość* będzie przesłana do serwera, tylko gdy pole zostanie zaznaczone przez użytkownika albo przez Ciebie – patrz krok 5.
5. Jeżeli chcesz, wpisz **CHECKED**, aby przycisk był zaznaczony po odebraniu strony WWW przez użytkownika. I Ty jako projektant strony, i użytkownik możecie zaznaczyć tyle pól, ile chcecie.
6. Napisz znak `>`.
7. Wprowadź tekst, który opisze użytkownikowi pole wyboru. Często jest to ten sam tekst, który wpisałeś dla VALUE, ale równie dobrze może to być inny tekst.
8. Powtórz kroki od 2. do 7. dla wszystkich pól wyboru.

✓ Rada

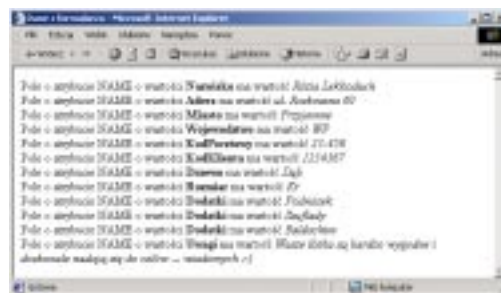
- Możesz przypisać każdemu polu wyboru w grupie własny klawisz skrót. Więcej informacji znajdziesz na stronie 201.



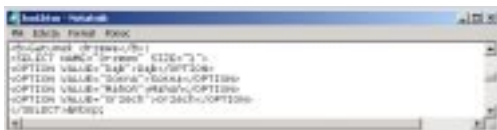
Rys. 11.28. Tekst podawany jako opis pola nie musi odpowiadać jego wartości ani nazwie. Jest to możliwe dzięki temu, że tekst identyfikuje pole w przeglądarce, natomiast wartość atrybutu VALUE – w skrypcie



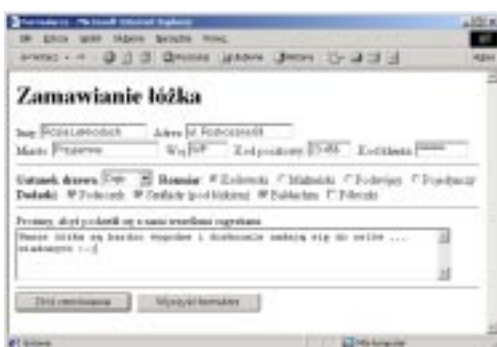
Rys. 11.29. Użytkownik może zaznaczyć tyle pól wyboru, ile chce. Wartość każdego zaznaczonego pola zostanie przesłana do skryptu wraz z nazwą grupy pól



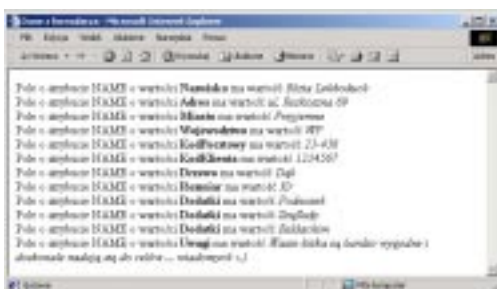
Rys. 11.30. Ponieważ użytkownik zaznaczył trzy pola wyboru, na serwer przesyłane są wszystkie trzy wartości (wartości – nie tekst widoczny na stronie przy polu). Ten skrypt wyświetla każde pole osobno



Rys. 11.31. Menu tworzy się przy użyciu dwóch znaczników HTML – `SELECT` oraz `OPTION`. Wspólny dla wszystkich opcji atrybut `NAME` określa się w znaczniku `SELECT`, natomiast poszczególne atrybuty `VALUE` – w znacznikach `OPTION`.



Rys. 11.32. Jeśli w menu nie określisz atrybutu `SIZE`, użytkownik nie będzie mógł niczego z niego wybrać. W menu domyślnie wybierana jest pierwsza opcja lub opcja, w której użyto atrybutu `SELECTED` (patrz krok 8.)



Rys. 11.33. Zauważ, że na serwer jest przesyłana wartość atrybutu `NAME`, a nie etykieta widoczna na stronie

Menu

Menu są doskonałym narzędziem pozwalającym na przedstawienie grupy opcji, z których użytkownik może wybrać jedną lub kilka.

Zdefiniowanie menu

1. W sekcji FORM dokumentu HTML opisz, jeśli chcesz, menu.
2. Napisz `<SELECT>`.
3. Nazwij menu wpisując `NAME="nazwa"`. Gdy dane będą wysyłane do serwera, to *nazwa* będzie identyfikować dane wybrane w menu.
4. Wpisz `SIZE="n"`, gdzie *n* to liczba pozycji menu, które będą widoczne po otwarciu strony.
5. Jeżeli chcesz, napisz `MULTIPLE`, by umożliwić użytkownikowi zaznaczenie więcej niż jednej pozycji menu.
6. Napisz znak `>` kończący znacznik `SELECT`.
7. Napisz `<OPTION>`.
8. Napisz `SELECTED`, jeśli chcesz, by pozycja była domyślnie wybrana.
9. Wpisz `VALUE="wartość"`, gdzie *wartość* identyfikuje dane otrzymywane przez serwer.
10. Napisz `>`.
11. Wprowadź tekst, który chcesz, by pojawił się w menu.
12. Powtórz kroki od 7. do 11. dla wszystkich pozycji menu.
13. Napisz `</SELECT>`.

✓ Rady

- Jeśli w kroku 4. dodasz atrybut `SIZE`, to menu będzie wyglądało raczej jak lista i nie będzie miało żadnego wybranego elementu (chyba że użyjesz atrybutu `SELECTED`).
- Jeśli wartość atrybutu `SIZE` jest większa od ilości opcji w menu, to użytkownik może usunąć zaznaczenie wszystkich opcji klikając na pustym obszarze poniżej nich.

Jeżeli definiujesz szczególnie duże menu z wieloma pozycjami, może dobrze byłoby pogrupować opcje w kategorie i umieścić je w podmenu.

Aby zdefiniować menu zawierające podmenu:

1. Stwórz menu wykorzystując metodę ze strony 190.
2. Przed pierwszym znacznikiem OPTION pierwszej grupy pozycji, które chcesz umieścić w podmenu, wpisz **<OPTGROUP>**.
3. Wpisz **LABEL="tytuł">**, gdzie *tytuł* to nagłówek podmenu.
4. Przed pierwszym znacznikiem OPTION pierwszej grupy pozycji, które chcesz umieścić w podmenu, wpisz **<OPTGROUP>**.
5. Po ostatnim znaczniku OPTION w grupie wpisz **</OPTGROUP>**.
6. Powtórz kroki od 2. do 4. dla każdego podmenu.

✓ Rady

- Teoretycznie, aby zdefiniować tekst menu, możesz do znacznika OPTION dodać **LABEL="nazwa_pozycji"**, ale mimo że jest to częścią standardu HTML 4, ani Explorer, ani Netscape nie wykorzystują tej informacji właściwie. Jeśli nie ma atrybutu LABEL, przeglądarki automatycznie wykorzystują tekst wpisany za znacznikiem OPTION (tak jak opisałam w kroku 11. na stronie 203).
- Podmenu, choć należą do oficjalnej specyfikacji standardu HTML 4, nie są jeszcze obsługiwane, ani przez Internet Explorera, ani przez Netscape'a.



Rys. 11.34. Każde podmenu ma tytuł, określony atrybutem LABEL w znaczniku OPTGROUP, i kilka pozycji zdefiniowanych znacznikiem OPTION i zwykłym tekstem



Rys. 11.35. Aby umożliwić użytkownikowi przesyłanie plików, trzeba spełnić dwa zasadnicze warunki. Po pierwsze, użyć metody POST w znaczniku FORM. Po drugie, nie zapomnieć o określeniu ENCTYPE w znaczniku INPUT



Rys. 11.36. Definiując pole przesyłu pliku tworzysz zarówno pole, w którym użytkownik może wprowadzić ścieżkę dostępu i nazwę pliku, jak i przycisk Przeglądaj (tak by gość mógł wykorzystać okno dialogowe Otwórz)

Umożliwienie użytkownikowi przesyłania plików

Jeśli informacja, którą chcesz zbierać za pomocą formularzy ma skomplikowaną postać, to może lepiej będzie, by użytkownik przysyłał na Twój serwer cały plik.

Aby zezwolić użytkownikowi na przesłanie pliku:

1. Wpisz **<FORM METHOD=POST ACTION="upload.cgi">**, gdzie *upload.cgi* jest adresem URL skryptu CGI służącego do przetwarzania przesyłanych plików. Większość skryptów służących do przetwarzania danych z formularzy nie będzie do tego celu wystarczająca.
2. Wpisz **ENCTYPE="multipart/form-data"**. Atrybut ten zapewni, że plik zostanie przesłany w odpowiednim formacie.
3. Wprowadź opis dla pola przesyłu pliku, tak aby użytkownik wiedział co z nim zrobić. Wpisz coś w rodzaju **Jaki plik chcesz przesłać?**
4. Wpisz **<INPUT TYPE="file">**, aby stworzyć pole przesyłu pliku i przycisk Przeglądaj (Browse).
5. Napisz **NAME="tytuł"**, gdzie *nazwa* jest dla serwera identyfikatorem przesyłanego pliku.
6. Jeśli chcesz wpisz **SIZE="n"**, gdzie *n* to szerokość (w znakach) pola, w którym gość wprowadza nazwę i ścieżkę dostępu do pliku.
7. Napisz kończący znak **>**.
8. W zwyczajny sposób dokończ formularz.

✓ Rada

- Atrybut SIZE jest opcjonalny, ale ponieważ większość ścieżek dostępu i nazw jest dość długa, to dobrze jest wybrać szerokość 40 czy 50 znaków. Domyślnie jest 20.
- Do obsługi formularzy umożliwiających przekazywanie plików nie możesz używać metody GET.
- Skrypty CGI służące do obsługi przekazywanych plików możesz znaleźć pod adresem www.cgi-resources.com.

Kilka uwag o polach ukrytych

Formularze HTML pozwalają na tworzenie pól specjalnego typu, które choć niewidoczne na ekranie, stanowią jednak część formularza. Na pierwszy rzut oka można by sądzić, że pola takie są całkowicie nieprzydatne – jeśli użytkownicy ich nie widzą, to jak mają w nich podać jakieś dane? Odpowiedź jest prosta – użytkownicy nie podają informacji w tych polach. Zamiast tego, to Ty będziesz zapisywał w tych polach informacje uzyskane z formularzy, wypełnionych wcześniej, łącząc je z informacjami podanymi przez użytkownika w bieżącym formularzu.

Wyobraź sobie, że w pierwszym etapie zbierania informacji prosisz użytkownika o podanie imienia i nazwiska, adresu oraz numeru telefonu. Następnie chcesz skierować użytkownika na stronę zawierającą katalog, na której może wybrać i zamówić meble. Zamiast zmuszać użytkowników do ponownego podawania informacji personalnych, możesz użyć skryptu CGI do pobrania tych informacji i wygenerowania w drugim formularzu pól ukrytych, w których informacje te zostaną zapisane. Dzięki temu, kiedy będziesz przetwarzał drugi formularz, to wszystkie pola – i te zawierające zamówione produkty, jak i te z informacjami personalnymi – zostaną przetworzone.

Nie daj się zmylić słowu „ukryte”. Choć pola tego typu faktycznie nie są wyświetlane w przeglądarce, to jednak stanowią integralną część kodu strony (tylko dzięki temu skrypt CGI może pobrać ich wartości), a zatem są też doskonale widoczne dla osób, które wyświetlą kod źródłowy strony (patrz strona 314).



Rys. 11.37. Fragment kodu HTML użytego do stworzenia formularza pokazuje, jak można definiować pola ukryte. Pisanie takiego kodu samemu nie ma większego sensu, dlatego dosyć niechętnie przedstawiam ten przykład. Pola ukryte powinny bowiem być generowane przez skrypty CGI

Pola ukryte

Choć rzadko kiedy sam będziesz dodawał pola ukryte do kodu formularza, to jednak aby pisać skrypty CGI tworzące takie pola, będziesz musiał wiedzieć, jak to robić.

Zdefiniowanie elementu ukrytego

1. Wpisz **<INPUT TYPE="hidden"**.
2. Napisz **NAME="nazwa"**, gdzie *nazwa* to identyfikator (nazwa) przechowywanych danych.
3. Napisz **VALUE="wartość"**, zastępując *wartość* informacją, którą chcesz przechować.
4. Wpisz znak **>**.

✓ Rady

- Pola ukryte nie są wyświetlane w przeglądarce, dlatego też nie ma znaczenia, w którym miejscu formularza zostaną umieszczone. Wystarczy, że zapiszesz je pomiędzy otwierającym i zamykającym znacznikiem FORM.
- Jeśli wartości atrybutów NAME oraz VALUE składają się wyłącznie ze znaków alfanumerycznych (tzn. nie zawierają odstępów, ani żadnych innych „śmiesznych” znaków), to nie musisz ich zapisywać w cudzysłowach. Cudzysłowy mają specjalne znaczenie w skryptach pisanych w języku Perl i dlatego, aby straciły to znaczenie, należy je poprzedzać znakiem odwrotnego ukośnika. Z tego względu, jeśli to tylko możliwe, nie powinieneś zapisywać wartości pól ukrytych w cudzysłowach.
- Aby stworzyć element dostarczający dane wraz z resztą formularza po kliknięciu przycisku Dostarcz, ale który jest widoczny dla użytkownika, zdefiniuj zwykły element i dodaj atrybut READONLY (patrz strona 218).

Przycisk dostarczający

Wszystkie wpisane przez użytkownika informacje nie mają dla Ciebie żadnej wartości, dopóki użytkownik nie wyśle tych danych do serwera. Dlatego zawsze na formularzu powinieneś umieścić przycisk SUBMIT, by użytkownik mógł dostarczyć Ci wprowadzone dane. (Jeśli w formularzu chcesz użyć aktywnych obrazków, to przeczytaj informacje przedstawione na stronie 212).

Stworzenie przycisku dostarczającego

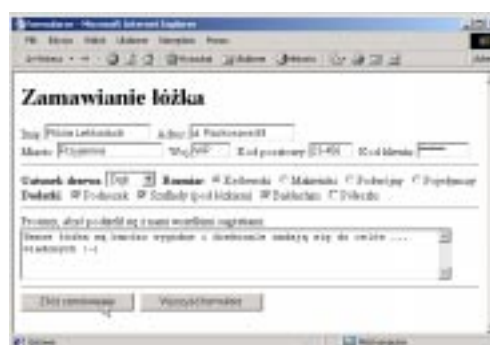
1. Wpisz **<INPUT TYPE="submit"**.
2. Jeśli chcesz, wpisz **VALUE= "nazwaprzycisku"**, gdzie *nazwaprzycisku* to napis, który wyświetlony zostanie na przycisku. Domyślnie wyświetlany jest napis *Submit Query*.
3. Wpisz znak **>**.

✓ Rady

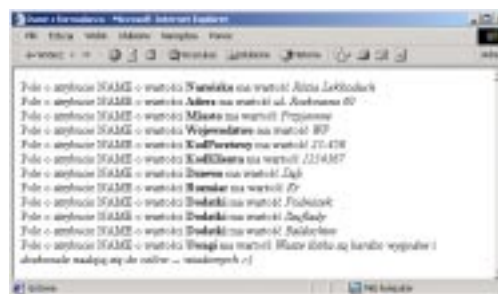
- Jeśli nie określisz atrybutu VALUE przycisku dostarczającego, to zostanie na nim wyświetlony napis „Submit Query”.
- Para nazwa-wartość dla przycisku dostarczającego jest przekazywana na serwer wyłącznie wtedy, gdy określiłeś atrybut NAME przycisku. A zatem, jeśli pominiesz ten atrybut, nie będziesz musiał przejmować się dodatkowymi, zupełnie niepotrzebnymi danymi.
- Z drugiej strony, możesz stworzyć kilka przycisków dostarczających, dla każdego z nich zdefiniować zarówno atrybut NAME, jak i VALUE, i napisać skrypt CGI reagujący na to, który przycisk został naciśnięty. Oczywiście do tego samego celu można wykorzystać przyciski opcji, co będzie znacznie prostsze.



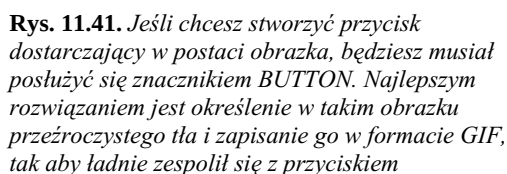
Rys. 11.38. Jeśli w znaczniku tworzącym przycisk nie podasz wartości atrybutu NAME, to para nazwa-wartość nie zostanie przesłana do skryptu. To dobry pomysł, wzięwszy pod uwagę fakt, że informacje o przycisku dostarczającym nie są Ci wcale potrzebne



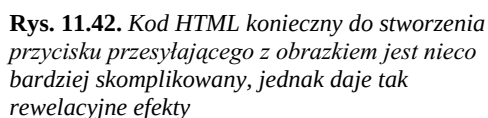
Rys. 11.39. Podstawową funkcją przycisku dostarczającego jest uruchomienie skryptu CGI, który posłuży do przetworzenia wartości pozostałych pól formularza. Możesz określić zawartość przycisku, podając ją w atrybucie VALUE. (Słowa „Złóż zamówienie” są bez wątpienia znacznie bardziej zrozumiałe dla użytkowników, niż domyślne „Submit Query”)



Rys. 11.40. Jeśli w przycisku nie określono wartości atrybutu NAME, to wartość atrybutu VALUE nie zostanie przesłana do skryptu. Ale czy ktoś jej do czegoś potrzebuje?



Rys. 11.41. Jeśli chcesz stworzyć przycisk dostarczający w postaci obrazka, będziesz musiał posłużyć się znacznikiem `BUTTON`. Najlepszym rozwiązaniem jest określenie w takim obrazku przeźroczystego tła i zapisanie go w formacie GIF, tak aby ładnie zespolił się z przyciskiem



Rys. 11.42. Kod HTML konieczny do stworzenia przycisku przesyłającego z obrazkiem jest nieco bardziej skomplikowany, jednak daje tak rewelacyjne efekty

Aby stworzyć ilustrowany przycisk dostarczający

1. Wpisz **<BUTTON NAME="submit" VALUE="submit" TYPE="submit".**
2. Jeśli chcesz, napisz **STYLE="font: 24pt Arial CE; background:yellow"** (lub coś podobnego), aby zmienić wygląd tekstu przycisku (patrz strona 261).
3. Wpisz **>**.
4. Wprowadź (lub nie) tekst, który ma się pojawić na przycisku po lewej stronie obrazka.
5. Wpisz **<IMG SRC="obrazek.gif"**, gdzie *obrazek.gif* to nazwa ilustracji, która ma się pojawić na przycisku.
6. Jeśli chcesz, wprowadź inne atrybuty obrazka.
7. Wpisz **>**, aby zakończyć definicję obrazka.
8. Wprowadź (lub nie) tekst, który ma się pojawić na przycisku po prawej stronie obrazka.
9. Napisz **</BUTTON>**.

✓ Rady

- Znacznik `BUTTON` możesz też wykorzystać do definiowania przycisków dostarczających bez ilustracji. Po prostu pomiń kroki od 5. do 7.
- Więcej informacji o tworzeniu przycisków za pomocą skryptów znajdziesz na stronie 300.
- Obecnie tylko Explorer 4 dla Windows obsługuje znacznik `BUTTON`, mimo że jest on częścią standardu HTML 4.

Wymazywanie formularza

Gdyby ludzie za pierwszym razem bezbłędnie wypełniali formularze, to na ołówkach nie byłoby gumek, a klawiatura nie zawierałaby klawisza BACKSPACE. Przycisk RESET umożliwia użytkownikowi rozpoczęcie wypełniania formularza od nowa.

Naciśnięcie tego przycisku powoduje wyczyszczenie wszystkich elementów formularza i przywrócenie im wartości domyślnych, określonych przez Ciebie w definicji formularza.

Stworzenie przycisku RESET

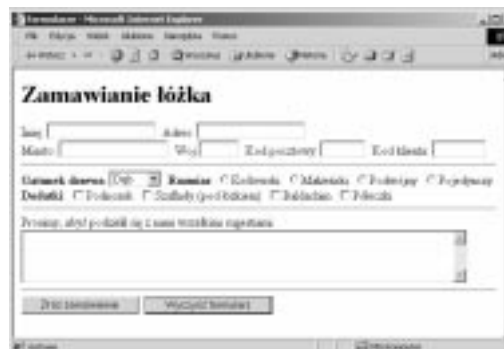
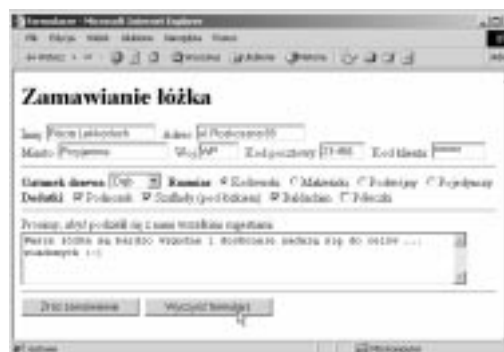
1. Wpisz `<INPUT TYPE="reset"`.
2. Jeżeli chcesz, wpisz `VALUE="nazwaprzycisku"`, gdzie *nazwaprzycisku* to napis, który zostanie wyświetlony na przycisku. Domyślnie wyświetlany jest napis *Reset*.
3. Wpisz znak `>`.

✓ Rada

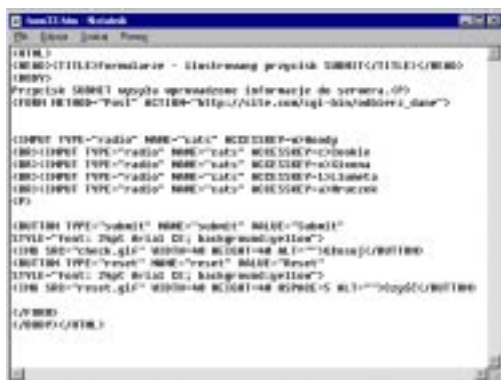
- Para nazwa-wartość dla przycisku RESET jest wysyłana do skryptu wyłącznie wtedy, gdy zostanie zdefiniowana wartość atrybutu NAME. Dlatego, jeśli pominiesz ten atrybut, to w skrypcie nie będziesz musiał zwracać sobie głowy obsługą zupełnie niepotrzebnych danych (które w przypadku tego przycisku przyjmują zazwyczaj postać „wyczyść formularz” lub czegoś podobnego).



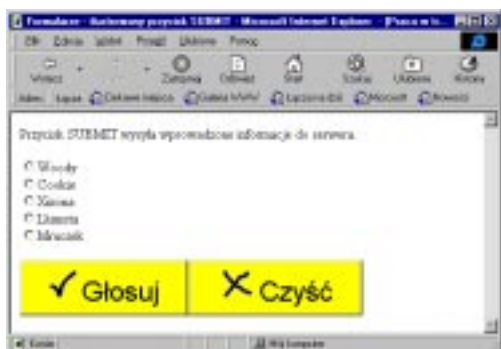
Rys. 11.43. Przy użyciu atrybutu *VALUE* możesz określić tekst, który zostanie wyświetlony na przycisku RESET



Rys. 11.44. Jeśli użytkownik kliknie przycisk RESET, to we wszystkich polach formularza zostaną wpisane ich wartości domyślne



Rys. 11.45. Upewnij się, że nadałeś atrybutowi **TYPE** wartość **reset**. W przeciwnym wypadku przycisk do niczego się nie przyda



Rys. 11.46. Możesz zilustrować przycisk resetujący, za pomocą znacznika **BUTTON** (i pewnie stworzysz ładniejszą ilustrację ode mnie)

HTML 4, jak do tej pory tylko w Explorerze, umożliwia dodawanie ilustracji, zmianę czcionki, a nawet koloru tła przycisku resetującego.

Aby stworzyć ilustrowany przycisk resetujący:

1. Wpisz **<BUTTON NAME="reset" VALUE="reset" TYPE="reset"**.
2. Jeśli chcesz, napisz **STYLE="font: 24pt Arial CE; background:yellow"** (lub coś podobnego), aby zmienić wygląd tekstu na przycisku (patrz strona 261).
3. Wpisz **>**.
4. Wprowadź (lub nie) tekst, który ma się pojawić na przycisku po lewej stronie obrazka.
5. Wpisz **<IMG SRC="obrazek.gif"**, gdzie **obrazek.gif** to nazwa ilustracji, która ma się pojawić na przycisku.
6. Jeśli chcesz, wprowadź inne atrybuty obrazka.
7. Wpisz **>**, aby zakończyć definicję obrazka.
8. Wprowadź (lub nie) tekst, który ma się pojawić na przycisku po prawej stronie obrazka.
9. Napisz **</BUTTON>**.

✓ Rady

- Znacznik **BUTTON** możesz też wykorzystać do definiowania przycisków resetujących bez ilustracji. Po prostu pomiń kroki od 5. do 7.
- Więcej informacji o tworzeniu przycisków za pomocą skryptów znajdziesz na stronie 300.
- Obecnie tylko Explorer 4 dla Windows obsługuje znacznik **BUTTON**, mimo że jest on częścią standardu HTML 4.

Przycisk RESET

Aktywne obrazki

Istnieje możliwość użycia obrazka – tak zwanego *aktywnego obrazka* – jako połączenia elementu umożliwiającego wprowadzenie danych oraz przycisku dostarczającego. Prócz przesłania danych, wpisanych w pozostałych polach formularza, kliknięcie na takim obrazku powoduje także przesłanie na serwer współrzędnych położenia wskaźnika myszy, określonych za pomocą dwóch par nazwa-wartość. W parach tych nazwy generowane są poprzez dodanie wyrażeń „x” oraz „y” do wartości atrybutu NAME. Wartości tych par określają poziome i pionowe położenie wskaźnika myszy w obszarze obrazka.

Wykorzystanie aktywnego obrazka

1. Stwórz obrazek i zapisz go w formacie GIF lub JPEG (patrz strona 57).
2. Wpisz `<INPUT TYPE="image".`
3. Napisz `SRC="adres_url_obrazka"`, gdzie *adres_url_obrazka* wskazuje, gdzie na serwerze znajduje się plik z obrazkiem.
4. Wpisz `NAME="nazwa"`. Gdy użytkownik kliknie obrazek, to współrzędne x i y kursora myszy zostaną wpisane do zmiennej *nazwa* i wysłane do serwera.
5. Wpisz znak `>` kończący definiowanie aktywnego obrazka.

✓ Rady

- Gdy użytkownik kliknie aktywny obrazek, wszystkie informacje wpisane do formularza są wysyłane do serwera. Dlatego dobrym zwyczajem jest poinformowanie użytkownika, jak i kiedy posłużyć się obrazkiem, a także umieszczenie obrazka na końcu formularza, żeby użytkownik najpierw wypełnił inne pola, a dopiero później kliknął obrazek i wysłał dane.
- Możesz zaprojektować cały kwestionariusz za pomocą obrazków. Kolejne pytanie (aktywny obrazek) pojawi się po kliknięciu i wysłaniu do serwera odpowiedzi



Rys. 11.47. Nie musisz umieszczać przycisku *SUBMIT* na formularzu zawierającym aktywny obrazek, ponieważ kliknięcie obrazka też wysła dane do serwera



Rys. 11.48. Gdy użytkownik kliknie obrazek, do serwera zostaną przesłane bieżące współrzędne kursora myszy



Rys. 11.49. Przeglądarka dodaje kropkę oraz x do nazwy atrybutu *NAME* i używa tej nazwy do identyfikacji poziomej współrzędnej miejsca, w którym kliknął użytkownik. W podobny sposób obsługiwana jest współrzędna pionowa

na aktualne pytanie.



Rys. 11.50. Znacznik **FIELDSET** doskonale nadaje się do podzielenia formularza na mniejsze, bardziej zrozumiałe porcje



Rys. 11.51. Zestawy pól są otoczone cienką linią, a tytuł pojawia się w prawym lub lewym, górnym rogu

Organizacja elementów formularzy

W przypadku zbierania za pomocą formularza wielu informacji można pogrupować powiązane ze sobą elementy, aby ułatwić wypełnianie formularza. Im łatwiej jest użytkownikowi zrozumieć formularz, tym bardziej prawdopodobne jest, że wypełni go poprawnie.

Aby pogrupować elementy formularza:

1. Za znacznikiem **FORM**, ale przed definicjami elementów, które chcesz włączyć do pierwszej grupy, napisz **<FIELDSET>**.
2. Wpisz **<LEGEND>**.
3. Jeśli chcesz, wpisz **ALIGN=kierunek**, gdzie **kierunek** to **top** (góra), **bottom** (dół), **left** (w lewo) lub **right** (w prawo).
4. Wpisz **>**.
5. Wprowadź tekst legendy (tytułu grupy).
6. Napisz **</LEGEND>**.
7. Zdefiniuj elementy formularza, które mają należeć do pierwszej grupy. Więcej informacji znajdziesz na stronach 198-212).
8. Napisz **</FIELDSET>**, aby zakończyć pierwszą grupę elementów.
9. Powtórz kroki od 1. do 8. dla każdej grupy elementów.

✓ Rady

- Nie musisz wpisywać legendy. Aby z niej zrezygnować, pominiń kroki od 2. do 6. Właściwie, w ogóle nie musisz grupować elementów na formularzu. Jest to użyteczne narzędzie, ale zupełnie opcjonalne.
- W trakcie pisania książki tylko IE 4 dla Windows rozpoznawał definicję zestawów pól. I potrafił wyrównywać legendę tylko do lewej lub prawej strony.

Formalne nadanie etykiety elementowi formularza

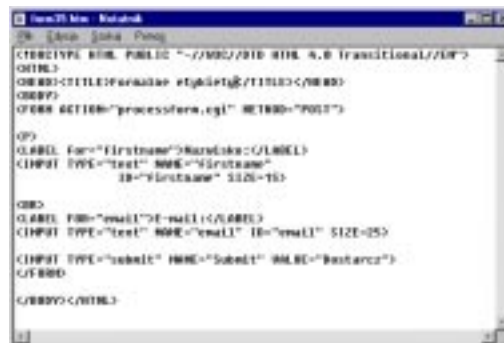
Jak zauważyłeś, informacja opisująca, wyświetlana obok elementów formularza, była do tej pory zwykłym tekstem. Mogłeś np. wpisać „Imię” przed polem, w którym użytkownik ma wpisać nazwisko. HTML 4 udostępnia metodę zaznaczania etykiet, dzięki czemu możesz formalnie połączyć etykietę z odpowiednim elementem oraz wykorzystywać je w skryptach.

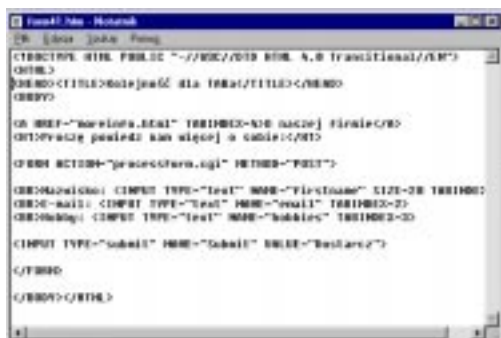
Aby formalnie nadać etykietę elementowi formularza:

1. Wpisz **<LABEL>**.
2. Napisz **FOR="nazwa"**, gdzie *nazwa* jest wartością atrybutu ID odpowiedniego elementu.
3. Wprowadź zawartość etykiety.
4. Napisz **</LABEL>**.

✓ Rady

- Aby wskazać znacznikiem LABEL dany element formularza, musisz zdefiniować atrybut ID w znaczniku tego elementu. Np.: `<INPUT TYPE=text SIZE=15 ID=nazwisko>`. Po szczegóły zerknij na stronę 254.
- Etykiety są częścią HTML-a 4, ale w trakcie wydawania tej książki, ani Explorer, ani Netscape nie obsługiwały ich. A szczerze mówiąc, sprawiają więcej kłopotów niż są tego warte. Osobiście z nich nie skorzystam.
- Bardziej efektywna technika określania nazw elementów opiera się na wykorzystaniu atrybutu TITLE. Więcej informacji na ten temat znajdziesz na stronie, w sekcji zatytułowanej „Nadawanie tytułów elementom stron WWW” na stronie 324.





Rys. 11.54. Możesz dodać atrybut **TABINDEX** do: definicji połączeń, elementów formularzy i map interpretowanych u klienta



Rys. 11.55. Gdy rozpoczynasz stronę formularza od definicji połączenia, to zapewne zechcesz zmienić kolejność dla klawisza **TAB**, tak by najpierw udostępniane było pierwsze pole, a nie połączenie

Określenie kolejności dla klawisza TAB

Za pomocą klawisza TAB użytkownicy mogą poruszać się po elementach formularza od góry do dołu. W zależności od konkretnej struktury formularza możesz zmienić tę kolejność, tak aby użytkownik wypełnił wszystkie pola danej grupy przed przejściem do następnej.

Aby określić kolejność dla klawisza TAB

Dopisz do definicji elementu formularza atrybut **TABINDEX=n**, gdzie *n* to liczba określająca kolejność dla TAB-a.

✓ Rady

- **TABINDEX** może przybierać wartości od 0 do 32767.
- Domyślna kolejność dla TAB-a zależy od kolejności elementów w dokumencie HTML. Jeśli zmieniasz kolejność, to elementy dostępne są od najniższego do najwyższego numeru **TABINDEX**.
- W formularzu kolejność dla TAB-a możesz określać dla: pól tekstowych, pól haseł, pól wyboru, przycisków radiowych, bloków tekstu, menu i przycisków.
- Możesz przypisać numer **TABINDEX** mapie interpretowanej u użytkownika i połączeniom. Po więcej informacji sięgnij na stronę 129 i 132.
- No dobrze, nie mogę Cię okłamywać: pierwsze naciśnięcie TAB wybiera adres aktualnej strony WWW, nawet wtedy gdy pasek z adresem jest ukryty. Dopiero drugie naciśnięcie TAB wybiera element z najniższym numerem **TABINDEX** na danej stronie.

Definiowanie klawiszy skrótów

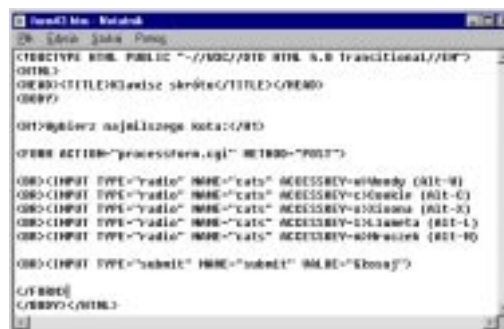
Wspniała nową cechą HTML-a 4 jest możliwość definiowania klawiszy skrótów do różnych fragmentów Twojej strony, w tym do elementów formularzy. Naciśnięcie klawisza skrótu powoduje wybranie danego elementu do wprowadzania danych (np. w przypadku pól tekstowych) lub zaznaczenie elementu (np. przycisku radiowego).

Aby zdefiniować klawisz skrótu do elementu formularza:

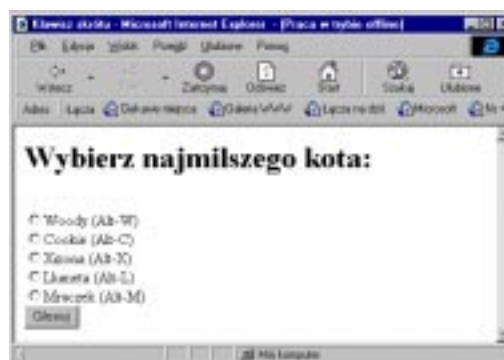
1. Wewnątrz znacznika definiującego element wpisz **ACCESSKEY="**.
2. Wpisz klawisz skrótu (dowolną literę lub cyfrę).
3. Napisz kończący znak **"**.
4. Jeśli chcesz, dodaj informację o klawiszu skrótu do tekstu, tak by goście wiedzieli o jego istnieniu.

✓ Rady

- Jedynie w Internet Explorerze 4 i 5 przeznaczonym dla systemów Windows udało mi się stworzyć poprawnie działające klawisze skrótów.
- W celu skorzystania z klawisza skrótu w Windows, użytkownik musi nacisnąć Alt i wybraną literę. Klawisze skrótów nie działają na Macintoshach. Gdy będą działać, użytkownik przypuszczalnie będzie musiał nacisnąć klawisz Command.
- Definiowane przez Ciebie klawisze skrótu mogą unieważniać klawisze skrótu przeglądarki. Jeśli wykorzystasz popularny klawisz skrótu (np. S od Save – Zachowaj), to możesz rozsierdzić swoich gości. Miej na uwadze, że przynajmniej w Windows najważniejsze klawisze skrótu przeglądarek wykorzystują Control, a nie Alt.



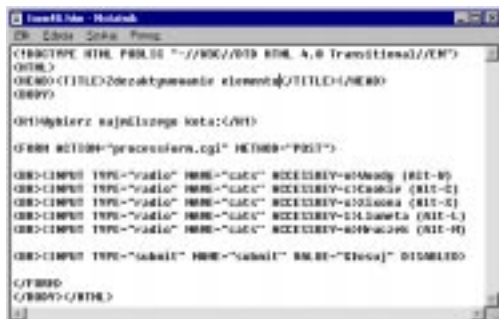
Rys. 11.56. Definiujesz klawisz skrótu przez dodanie atrybutu **ACCESSKEY** do definicji elementu formularza



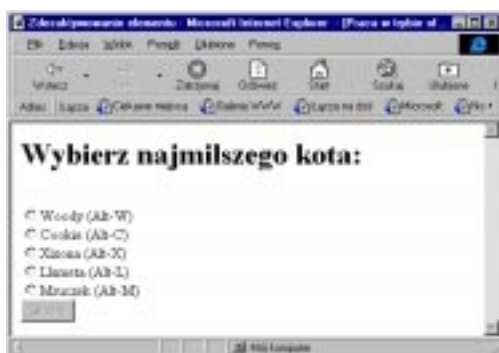
Rys. 11.57. Kiedy gość ogląda stronę, żaden element nie jest wybrany. Dobrym pomysłem jest dopisanie tekstu informującego gości o istnieniu klawiszy skrótów



Rys. 11.58. Po naciśnięciu klawisza skrótu (w przykładzie Alt-M), wybrany i zaznaczony zostanie przycisk radiowy Mruczek. W przypadku pola tekstowego, czy też bloku tekstu, kursor zostanie przesunięty do pola, a gość będzie mógł wprowadzić tekst



Rys. 11.59. Możesz dodać atrybut **DISABLED** do dowolnego elementu formularza, ale prawdopodobnie największy sens ma to w przypadku znacznika **INPUT** definiującego przycisk dostarczający



Rys. 11.60. W przykładzie przycisk **Głosuj** jest wycieniowany, ponieważ użytkownik nie wybrał jeszcze żadnego przycisku radiowego. Do uaktywnienia przycisku dostarczającego niezbędny jest skrypt

Uczynienie elementu formularza nieaktywnym

Czasami chcesz zabronić gościom wykorzystanie pewnych fragmentów formularza. Na przykład chcesz, by przycisk dostarczający nie był aktywny, dopóki wszystkie żądane pola formularza nie zostaną wypełnione.

Aby element formularza był nieaktywny

W znaczniku **FORM** wpisz **DISABLED**.

✓ Rady

- Jedynym sposobem na zmianę zawartości elementu, który nie jest aktywny, jest wykorzystanie skryptu. Po więcej informacji sięgnij do rozdziału 17. Będziesz także potrzebował pewnego doświadczenia w korzystaniu z języka JavaScript.
- Jeśli uczynisz element formularza nieaktywnym, jego klawisz skrótu również nie będzie aktywny. Po więcej informacji zerknij na stronę 216.

Uniemożliwienie zmian

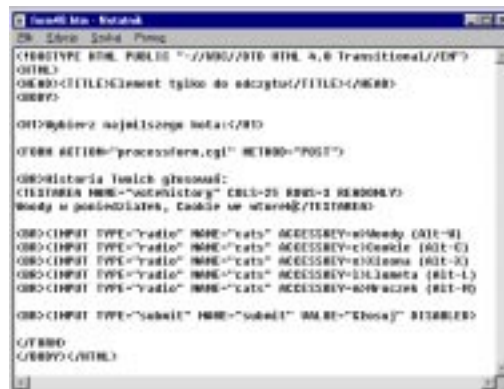
Czasami konieczne okazać się może automatyczne wypełnienie elementu formularza i uniemożliwienie użytkownikowi dokonywania zmian w tym elemencie. Np. możesz żądać od gościa potwierdzenia pewnych danych albo – w trakcie powtórnego zbierania danych – wyświetlać poprzednio przesłane dane. Takie elementy zaznaczasz jako „tylko do odczytu”.

Aby uniemożliwić użytkownikowi dokonywanie zmian

W znaczniku definiującym element wpisz **READONLY**.

✓ Rady

- Atrybut READONLY możesz używać w odniesieniu do: pól tekstowych, pól haseł, pól wyboru, przycisków radiowych i bloków tekstu.
- Użycie atrybutu READONLY przypomina ukrycie elementu, bez ukrywania. Więcej na ten temat znajdziesz na stronie 206.



Rys. 11.61. Dodaj atrybut READONLY do definicji dowolnego elementu formularza, którego zawartości użytkownik nie powinien zmieniać



Rys. 11.62. W przykładzie, w bloku tekstu „tylko do odczytu”, wyświetlany jest sposób głosowania użytkownika w przeszłości. Użytkownik może to obejrzeć i zagłosować ponownie

Skrypt Pik-Netu i Heliona

Chcielibyśmy polecić Ci skrypt, opracowany przez gliwicką firmę PIK-NET i Wydawnictwo Helion, specjalnie dla potrzeb użytkowników nie mających możliwości umieszczania programów CGI na serwerze. Jego wykorzystanie jest bardzo proste:

1. Wpisz **<FORM ACTION="http://www.pik-net.pl/cgi-bin/poczta" METHOD="GET">**.
2. Aby określić, na jaki adres mają być wysyłane dane z Twojego formularza, wpisz:
<INPUT TYPE="hidden" NAME="adresdoskryptu" value="adres@nazwa.twojego.serwera",
gdzie *adres@nazwa.twojego.serwera* to adres, na który ma być wysyłana zawartość formularza (czyli najczęściej Twój adres e-mailowy).
3. Wpisz **<INPUT TYPE="hidden" NAME="subjectdoskryptu" VALUE="Temat listu">**. *Temat listu* określa, jaka wartość będzie się pojawiać w polu *Subject* listów, które będziesz otrzymywał.
4. Wpisz pozostałą treść zaprojektowanego uprzednio formularza. Nie wolno Ci zdublować nazw pól *adresdoskryptu* i *subjectdoskryptu*.
5. Nie zapomnij o umieszczeniu przycisków SUBMIT i RESET (strony 208-211).
6. Przed udostępnieniem formularza użytkownikom, koniecznie przetestuj, czy wszystko działa poprawnie. W razie kłopotów połącz się ze stroną opisującą skrypt. Możesz ją znaleźć pod adresem: <http://www.pik-net.pl/poczta/>.