

# 7. Projektowanie stron www cz.1

(HTML – informacje ogólne)

<b>7.1 Projektowanie stron WWW – informacje ogólne .....</b>	<b>2</b>
W jakim celu tworzymy stronę internetową?.....	2
Strona powinna być prosta .....	2
Nie używajmy zbyt zaawansowanej technologii .....	2
Strona powinna być lekka .....	3
Umieszczajmy strony na wiarygodnych serwerach .....	4
Podsumowanie.....	4
<b>7.2 Czym jest HTML?.....</b>	<b>5</b>
Gdzie można znaleźć przykłady? .....	6
Jak się tworzy i zapisuje dokumenty HTML-owe?.....	6
<b>7.3 Struktura dokumentu HTML.....</b>	<b>7</b>
Znaczniki .....	8
<b>7.4 Budujemy pierwszą stronę WWW. ....</b>	<b>11</b>
Deklarowanie sposobu kodowania polskich znaków diakrytycznych .....	13
Wprowadzanie tekstu. ....	14

## **7.1 Projektowanie stron WWW – informacje ogólne**

### ***W jakim celu tworzymy stronę internetową?***

- ✓ dlaczego chcemy stworzyć własną stronę WWW?
- ✓ kto ma ją odwiedzać, czyli dla kogo to właściwie robimy?
- ✓ czy ma to być twórczość prywatna czy też zawodowa?
- ✓ jakie treści chcemy na niej zaprezentować światu?

### ***Strona powinna być prosta***

Jeżeli pozeoglujemy trochę po Internecie, zauważymy, że najbardziej nasycone efektami są amatorskie strony stworzone chyba tylko dla samych autorów.

Wszelkie serwisy notujące tysiące odwiedzin w ciągu godziny mają bardzo prostą, przejrzystą strukturę. Pamiętajmy zawsze, że serwis jest dla kogoś innego, a nie dla nas. Jeżeli więc sami dysponujemy IQ na poziomie 200 oraz doskonałą orientacją w prezentowanym zagadnieniu, niekoniecznie odwiedzający nasz serwis cyberpodróżnik będzie równie genialny.

Jeżeli struktura strony będzie niezrozumiała i trudna w nawigacji, to po prostu niewiele osób będzie chciało nas odwiedzić. Pamiętajmy, że Internet jest olbrzymi i dysponuje gigantyczną ofertą, a więc konkurencja jest duża.

### ***Nie używajmy zbyt zaawansowanej technologii***

Internet nie jest statyczny. Standardy bardzo szybko przestają być nimi ze względu na stale rozwijające się technologie i narzędzia. Niektórzy twórcy cyberprzestrzeni uwielbiają nowinki. Jeżeli więc pojawiła się na rynku nowa przeglądarka poprawnie interpretująca jakiś dziwaczny fragment języka Java, to cała strona zostaje tak zmieniona, żeby wykorzystać te nowe możliwości. Niestety, efekt jest zwykle odmienny od zamierzonego — zamiast podziwu i okrzyków zachwyty spotkamy się co

najwyżej z obojętnością innych internautów lub wręcz z krytyką.

Pamiętajmy, że wśród użytkowników Internetu tylko niewielka część należy do grona owych miłośników nowoczesności.

Zdecydowana większość to zwykli ludzie, traktujący sieć jako narzędzie. Niewielu ma więc ochotę instalować każdą nową nakładkę czy przeglądarkę zaraz po jej ukazaniu się na rynku. Nie każdy ma wreszcie odpowiedni sprzęt, żeby pracować na przykład przy rozdzielczości ekranu 1280x1024.

W Internecie można spotkać takie komunikaty: „zoptymalizowano dla rozdzielczości 1280x1024”.

Takie same uwagi znajdują zastosowanie w odniesieniu do najrozmaitszych wtyczek (*Plug-In*). Należy z ostrożnością traktować nawet tak standardowe elementy, jak ożywiający stronę i czyniący ją niebywale „ruchomą” Flash. Profesjonalne strony mają zwykle oddzielne wersje, bardziej i mniej zaawansowane technologicznie.

W skrócie można powiedzieć, że trzeba być postępowym i nie stronić od nowości. Jednak z umiarem! Nowość, która ma dopiero kilka miesięcy, jest jeszcze zbyt świeża. Musi dojrzeć, programiści powinni usunąć „dziury” i niedociągnięcia.

Jeżeli chcemy, by nasze strony były często i chętnie odwiedzane, powinny być wykonane w miarę możliwości w prosty sposób, bez zbędnych dziwactw i super nowości.

### **Strona powinna być lekka**

Pod tym pojęciem kryją się dwie sprawy. Po pierwsze, powinniśmy dbać o estetyczną stronę naszej pracy, tak by nie przeładować jej niepotrzebnymi elementami.

Po drugie — bardziej istotnym zagadnieniem jest „kilobajtowa waga” strony. Im większa, tym gorzej. Pamiętajmy, że im strona jest większa, tym więcej czasu zajmuje jej wczytanie. Tak więc jeżeli przesadzimy z ilością na przykład zajmujących dużo pamięci elementów graficznych, to może się okazać, że czas pobierania strony będzie nie do zaakceptowania

dla większości internautów. Efekt będzie taki sam jak w poprzednich przypadkach — spadek liczby odwiedzin. Oczywiście na czas wczytywania strony mają wpływ jakość serwera i jego łącza z resztą Internetu. Na to mamy wpływ, podejmując decyzję o umieszczeniu strony na tym czy innym serwerze. Nie mamy jednak najmniejszego wpływu na jakość łącza potencjalnie odwiedzającego naszą stronę użytkownika. Tak, więc zasada, jaką powinniśmy się kierować, to zminimalizowanie wielkości naszej strony w taki sposób, by jednak umieścić na niej wszelkie niezbędne elementy. Jednym słowem — złoty środek.

### ***Umieszczamy strony na wiarygodnych serwerach***

Wybór darmowych możliwości umieszczania stron w Internecie jest bardzo duży. W zasadzie każdy portal internetowy i serwisy poważniejszych firm masowych czy mediów dają takie możliwości. Zwróćmy uwagę na parametry oferowanych kont - czyli m.in. maksymalną wielkość pamięci przypadającej na jednego użytkownika.

Możemy również uczynić swój własny komputer serwerem i na nim umieścić swoje strony. Jest to jednak rozwiązanie trudne, drogie i nie jest zalecane dla początkujących.

### ***Podsumowanie***

Tak więc w skrócie proces przygotowywania naszych własnych stron WWW będzie wyglądał tak:

- 1) Zastanawiamy się, czy chcemy tworzyć własne strony www.
- 2) Zastanawiamy się, dla kogo je tworzymy i jaką rolę powinny spełniać.
- 3) Kompletujemy oprogramowanie i narzędzia niezbędne do tworzenia stron.
- 4) Przystępujemy do tworzenia stron na własnym komputerze, używając dostępnych narzędzi.
- 5) Testujemy strony.

- 6) Dokonujemy analizy możliwości umieszczenia naszych przyszłych stron na serwerze i udostępnienia ich światu.
- 7) Rejestrujemy nasze konto na wybranym serwerze.
- 8) Umieszczamy strony na serwerze.
- 9) Rejestrujemy strony w kilku wyszukiwarkach.
- 10) Cieszymy się efektem.
- 11) Aktualizujemy stworzone strony.
- 12) Cieszymy się efektem.
- 13) Powracamy do punktu 11.

## 7.2 Czym jest HTML?

To, co widzimy na ekranie swojej przeglądarki, jest obrazem, którego treść i formę określono w specjalnym pliku. Jest to zwykły plik tekstowy, którego treść zapisano w specjalnym języku zwanym HTML (z języka angielskiego *HyperText Markup Language*).

Plik ten jest wczytywany przez przeglądarkę (na przykład Internet Explorer, Netscape), dzięki czemu jest ona w stanie sformatować obraz w taki sposób, w jaki życzył sobie tego autor.

Jednym słowem, plik HTML jest jakby zbiorem rozkazów, które wykonuje przeglądarka. Rozkazy te określają, jakie obrazki mają być wczytane, jakie tło ma być podłożone pod tekst, jakie napisy mają wystąpić i w których częściach ekranu oraz jak mają być sformatowane itd. — słowem dokładnie wszystko to, co ma się znaleźć na stronie.

Nie ma tu jednak informacji o tym, jak dokładnie ma wyglądać każdy szczegół graficzny. W pliku HTML znajdują się tylko informacje, że obrazek o danej nazwie powinien być wczytany i umieszczony w danym miejscu. Jest to więc rodzaj czystego tekstu opatrzonego komentarzami typu: „ten fragment pogrubić”, „tutaj dodać rysunek” czy „ten tekst zapisać wielkimi literami” itp.

*Oczywiście przeglądarki nie potrafią zrozumieć poleceń zapisanych w powyższy sposób. Ze względu na ich ograniczony zakres percepcji trzeba było stworzyć specjalny język, który jednoznacznie określałby nasze*

życzenia — tym językiem jest właśnie HTML.

W zasadzie wszystko jest banalnie proste — trzeba tylko poznać składnię i słowa kluczowe tego języka.

### **Gdzie można znaleźć przykłady?**

Najlepiej uczy się na przykładach — to oczywista prawda. Pozostaje tylko jeden problem — gdzie ich szukać? Na szczęście sama sieć znowu przychodzi nam z pomocą. Przykładów jest mnóstwo — całe miliony. Każda strona dostępna w sieci jest takim przykładem do przestudiowania i wyciągnięcia wniosków.

Wystarczy po wczytaniu danej strony z menu głównego wybrać [Widok/Źródło](#). Na ekranie pojawi się kod źródłowy aktualnie przeglądanej strony. Jeżeli uznamy ją za godną uwagi, wówczas każemy ją przeglądarce zapisać na naszym dysku [Plik/Zapisz jako](#) i już możemy eksperymentować.

Nie trzeba nic samemu wpisywać, wystarczy otworzyć plik i dowolnie modyfikować, dostosowując do naszych potrzeb i preferencji. Pamiętajmy jednak, że zapiszemy w ten sposób tylko i wyłącznie sam plik HTML-owy, bez jakichkolwiek elementów dodatkowych, czyli np. grafiki.

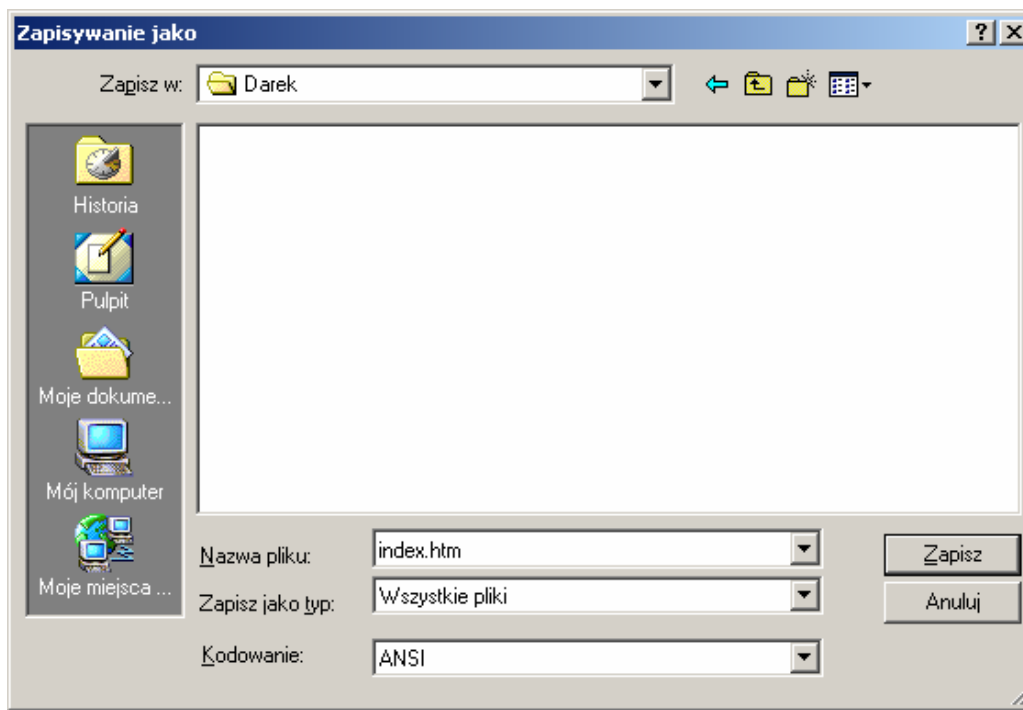
### **Jak się tworzy i zapisuje dokumenty HTML-owe?**

Powstało wiele programów ułatwiających pisanie dokumentów HTML-owych. Niektóre są lepsze, niektóre gorsze. Zamiast wpisywać śmieszne znaczki, lepiej klikać odpowiednie przyciski — powie zwolennik edytorów hipertekstowych. Prawda jest jednak taka, że stosują je w większości tylko początkujący, i to tylko w pierwszych chwilach sieciowego szaleństwa. Jak zgodnie twierdzą znawcy przedmiotu, nie ma jak pisanie wprost w HTML-u — przynajmniej wiadomo, jaki będzie efekt. Może do tego posłużyć standardowy notatnik.

Tak więc będąc np. w Notatniku, po napisaniu dokumentu HTML-owego wybieramy z menu [Plik/Zapisz jako](#). Pojawi się okienko dialogowe w standardzie Windows. Teraz jako typ pliku wybieramy „wszystkie pliki” i

nadajemy mu nazwę, która powinna mieć rozszerzenie *.html* lub *.htm*.

Jest to absolutnie konieczne, by dać znać przeglądarce, że ma do czynienia z dokumentem HTML-owym. W przeciwnym razie po jego wczytaniu na ekranie nie pojawi się nasza strona, tylko jej kod źródłowy.



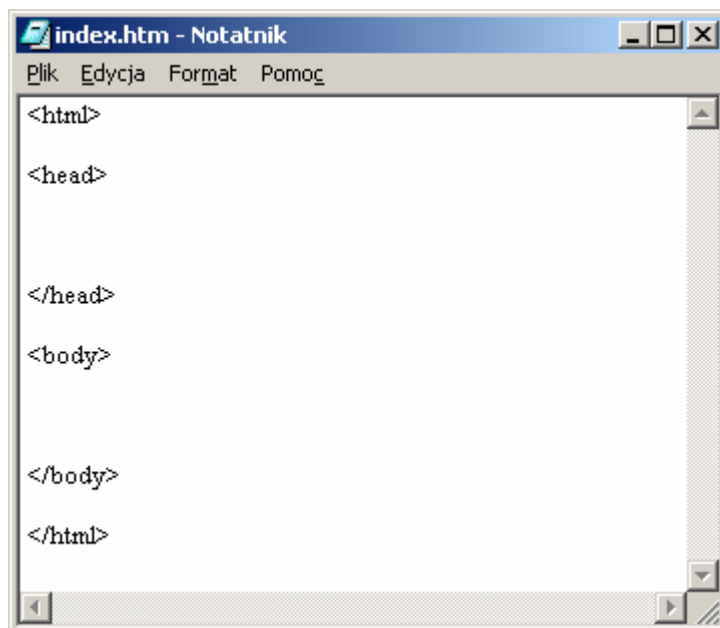
## 7.3 Struktura dokumentu HTML.

Tekstowa postać pliku HTML jest jego największą zaletą. Umożliwia to jego edycję praktycznie każdym najprostszym edytorem tekstu – choćby takim jak standardowy edytor tekstu w Windows – Notatnik.

Dopiero przeglądarka WWW dokonuje przeobrażenia tekstowego opisu wyglądu i zawartości strony, zawartego w pliku HTML, na postać graficzną.

Plik HTML zawiera szereg poleceń tak zwanych znaczników składających się na elementy strony. Znacznik jest to fragment tekstu zamknięty wewnątrz znaków nawiasów ostrokątnych „<” oraz „>”.

Poniżej znajduje się przykład najprostszej strony WWW.



```
index.htm - Notatnik
Plik  Edycja  Format  Pomoc

<html>

<head>

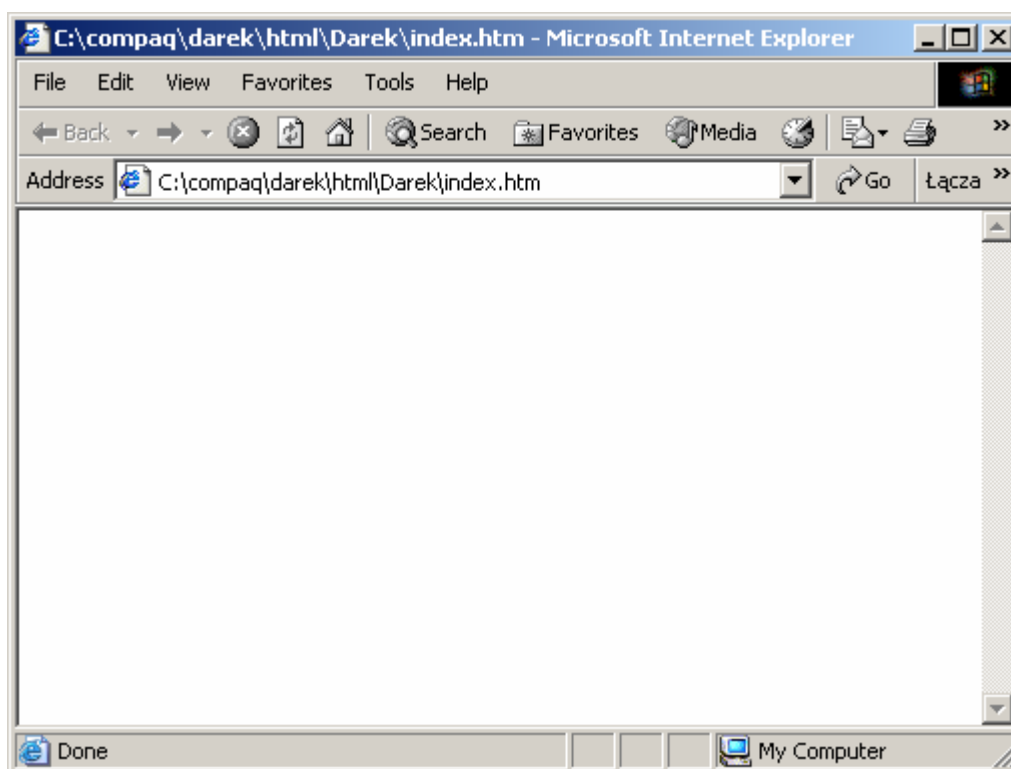
</head>

<body>

</body>

</html>
```

**Rysunek 1 Kod źródłowy najprostszej strony**



**Rysunek 2 Tak wygląda najprostsza strona po wczytaniu przez przeglądarkę**

## **Znaczniki**

Jak już wspomniano wcześniej znacznik jest to fragment tekstu zamknięty wewnątrz znaków nawiasów ostrokątnych „<” oraz „>”.

Elementy języka HTML składają się zawsze z dwóch znaczników:



- ✓ znacznika otwierającego `<ZNACZNIK>`, np. `<head>`
- ✓ znacznika zamykającego `</ZNACZNIK>`, np. `</head>`

W języku HTML nie ma znaczenia czy znacznik jest pisany wielką literą `<ZNACZNIK>`, czy małą `<znacznik>`.

Obie formy są prawidłowe, lecz ze względów estetycznych należy przyjąć jedną z nich i konsekwentnie ją stosować. Jest to zabieg czysto kosmetyczny, ułatwiający przeglądanie i poprawianie kodu HTML.

Strona WWW jest dokumentem HTML, który składa się zazwyczaj z dwóch części:

- nagłówka (`head`);
- i treści (`body`)

Aby komputer wiedział, z czym ma do czynienia, konieczne jest użycie odpowiedniego znacznika.

Znaczniki `<html>` i `</html>` informują, że wszystko, co znajduje się między nimi stanowi dokument HTML.

Pierwszym elementem strony jest nagłówek. Jego ramy wyznacza para znaczników `<head>` i `</head>`. Między tymi znacznikami umieszczany jest kod definiujący tytuł strony oraz inne elementy związane z ustawieniami danej strony. Kolejnym elementem w tym przykładzie jest para znaczników `<body>`, `</body>`. Jest to podstawowa część strony, w której umieszcza się wszystkie inne elementy: treść, obrazy, łącza itd.

Znaczniki HTML poza samą, zamkniętą pomiędzy nawiasami nazwą mogą zawierać jeszcze dodatkowe informacje – tak zwane **atrybuty**:

```
<znacznik atrybut1="wartość1" atrybut2="wartość2" > ... </znacznik>
```

np.

```
<FONT FACE="Arial" COLOR="Red" SIZE="10">.....</FONT>
```

Atrybuty oddzielone są od nazwy i pozostałych atrybutów znakami spacji, zaś wartość atrybutu zapisywana jest zaraz za nazwą atrybutu, jest oddzielona od niej znakiem równości i ujęta w znaki cudzysłowu:

```
atrybut1="wartość1" np. FACE="Arial"
```

Tak jak w przypadku znaczników nazwy atrybutów możemy zapisywać zarówno dużymi jak i małymi literami. Jeśli chodzi o wartości atrybutów wielkość liter może mieć znaczenie.

Wewnątrz elementu HTML – a więc pomiędzy znacznikiem otwierającym i zamykającym dany element – znajdować się może tekst treści tego elementu lub dowolna liczba innych elementów języka HTML.

**<znacznik\_1>**

Tekst treści elementu 1

**<znacznik\_2>**

Tekst treści elementu 2

**</znacznik\_2>**

**<znacznik\_3>**

Tekst treści elementu 3

**</znacznik\_3>**

**</znacznik\_1>**

Bezwzględnie należy pamiętać o tym, aby elementy nie przeplatały się wzajemnie. Oznacza to, aby elementy były zamykane w kolejności odwrotnej do kolejności ich otwierania:

**<z1>**

**<z2>**

....

**</z2>**

**</z1>**

**a nie np:**

**<z1>**

**<z2>**

....

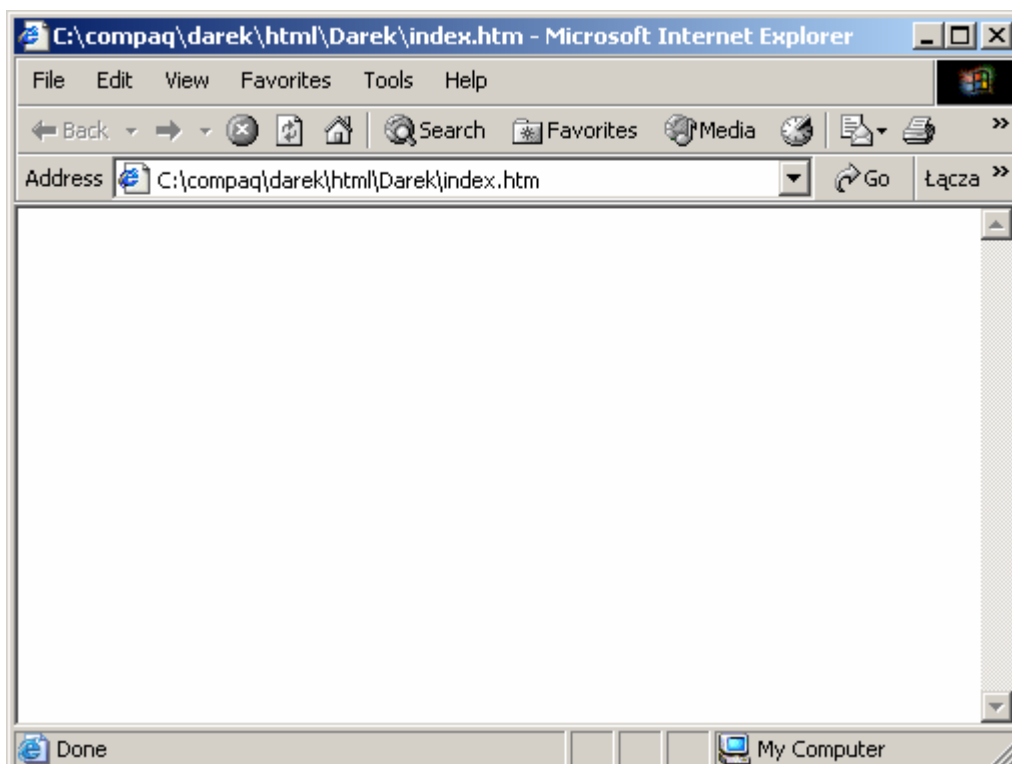
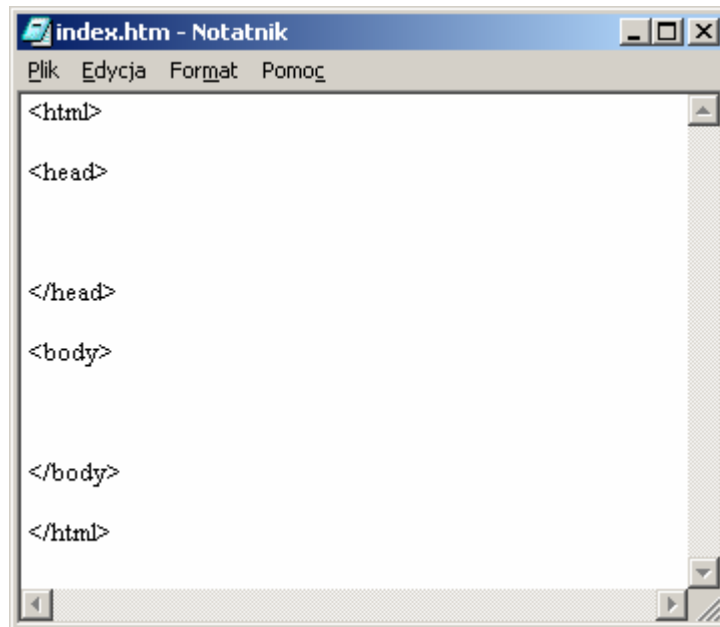
**</z1>**

**</z2>**

Język HTML posiada kilkadziesiąt znaczników, część z nich poznamy w trakcie dalszej części zajęć.

## 7.4 Budujemy pierwszą stronę WWW.

Uruchamiamy nasze narzędzie do tworzenia strony HTML – w naszym wypadku będzie to program Notatnik.



Każda strona WWW powinna mieć tytuł określający skrótowo jej zawartość lub przynależność do większego serwisu internetowego. Tytuł strony wyświetlany jest zawsze na pasku tytułowym przeglądarki WWW.

Zaleca się nie przekraczanie 40 znaków. W kodzie HTML odpowiedzialny jest za to znaczniki <title> umieszczany w obszarze nagłówka strony.

Kod nagłówka powinien zostać zmodyfikowany do postaci:

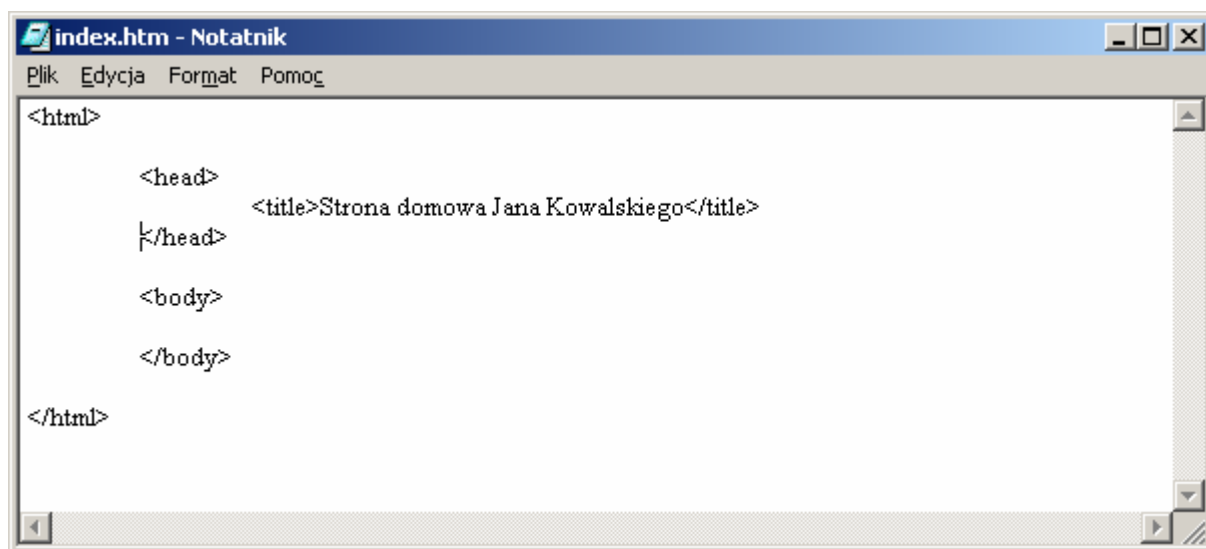
```
<head>
```

```
    <title>Strona domowa Jana Kowalskiego</title>
```

```
</head>
```

**Ważna uwaga:** Nie należy używać w tytułach samych WIELKICH LITER, gdyż wiele wyszukiwarek zignoruje stronę. Nie wpisujemy także w ramach znacznika <title> adresów internetowych, gdyż skutek będzie taki sam.

Warto także pamiętać, że wyszukiwarki nadają temu znacznikowi wysoki priorytet, a więc warto w nim podać istotne informacje, np. nie "Moja strona domowa", lecz raczej "Strona domowa Jana Kowalskiego".

A screenshot of a Notepad window titled "index.htm - Notatnik". The window contains the following HTML code:

```
<html>

    <head>
        <title>Strona domowa Jana Kowalskiego</title>
    </head>

    <body>

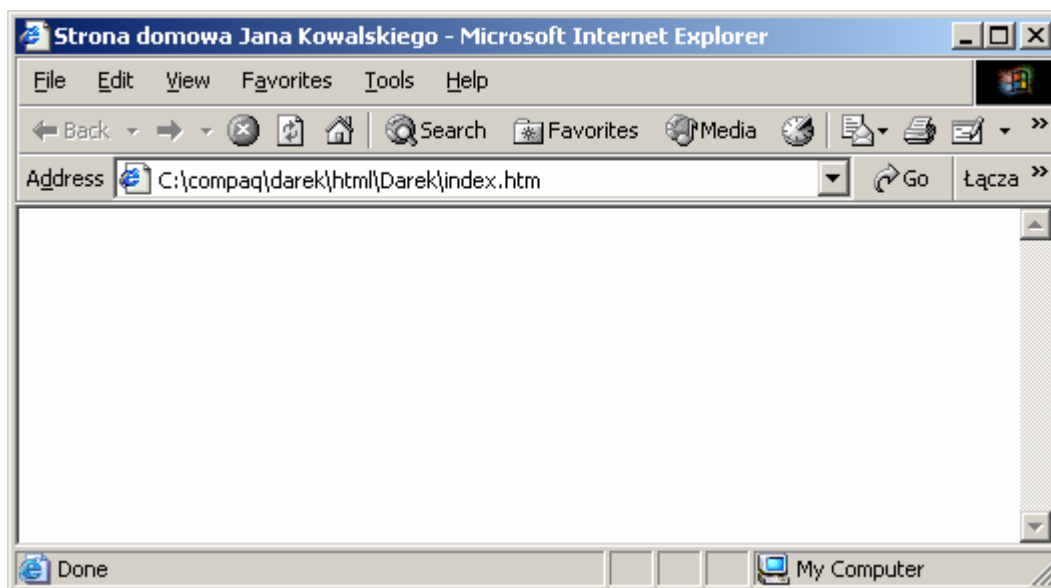
    </body>

</html>
```

The code is indented to show the hierarchy of the HTML elements. The window has a menu bar with "Plik", "Edycja", "Format", and "Pomoc".

Na powyższym rysunku można zaobserwować odstępy przed znacznikami <head>, <title>. Odstępy te są całkowicie zbędne, ale pozwalają wizualnie podkreślić hierarchię znaczników. Odstępy te, nazywane przez programistów wcięciami, będą się pojawiać we wszystkich kolejnych przykładach. Jest to zabieg czysto kosmetyczny ułatwiający wyszukiwanie elementów w kodzie.

Zapisujemy zbudowaną stronę na dysku komputera i uruchamiamy plik w przeglądarce internetowej. A oto efekt pracy:



### ***Deklarowanie sposobu kodowania polskich znaków diakrytycznych***

Często na stronach WWW można zauważyć, że nie zawsze wyświetlane są prawidłowo polskie znaki diakrytyczne. Zamiast nich znajdują się „dziwne symbole”. Dzieje się tak dlatego, że nie ma w kodzie strony informacji o sposobie zakodowania polskich znaków, a przeglądarka nie jest w stanie automatycznie wykryć i ustawić odpowiedniego kodowania polskich znaków.

Kod nagłówka pliku HTML modyfikujemy o linię:

**<meta http-equiv="Content-Type" content="text/html; charset=windows-1250" />**



W zależności od stosowanego oprogramowania do tworzenia kodu HTML stosuje się różne standardy kodowania polskich liter. Najczęściej używanymi są:

✓ windows-1250

✓ iso-8859-2

W znaczniku `<meta>` został użyty drugi sposób zamykania znaczników. Jest to znak „/” umieszczony tuż przed nawiasem kończącym znacznik `...windows-1250” />`. Z tą sztuczką można się spotkać bardzo często.

Niestety nie do wszystkich znaczników można ją stosować

### ***Wprowadzanie tekstu.***

Podstawowym elementem tekstowym na stronie WWW jest akapit reprezentowany przez znacznik

**`<P> tekst </P>`**

Rozmieszczenie tekstu pomiędzy znacznikami akapitu, nie ma znaczenia. Przeglądarka i tak ułoży go w jeden ciąg znaków (oczywiście zachowując odstępy pomiędzy wyrazami) wypełniający całą szerokość okna.

Do tworzonego wcześniej przykładu dodajemy w sekcji BODY (czyli pomiędzy znacznikami `<body>` i `</body>`) akapit tekstu:

`<p>`

Akapit tekstowy. Obojętnie jak go wprowadzimy i tak zostanie wyświetlony

tak aby wypełnił całą szerokość okna

`</p>`

```
index - Notatnik
Plik Edycja Format Widok Pomoc
<html>
  <head>
    <title>Strona domowa Jana Kowalskiego</title>
    <meta http-equiv="Content-Type" content="text/html; charset=windows-1250" />
  </head>
  <body>
    <p>
      Akapit tekstowy. obojętnie jak go wprowadzimy i tak
      zostanie wyświetlony
      tak aby wypełnił całą szerokość okna
    </p>
    zżąśęćńół
  </body>
</html>
```

A oto efekt:



Gdy zastosujemy wspomniane wcześniej kodowanie [ISO-8859-2](#) wygląda to tak:

```
index - Notatnik
Plik Edycja Format Widok Pomoc
<html>
  <head>
    <title>Strona domowa Jana Kowalskiego</title>
    <meta http-equiv="Content-Type" content="text/html; charset=|iso-8859-2" />
  </head>
  <body>
    <p>
      Akapit tekstowy. obojętnie jak go wprowadzimy i tak
      zostanie wyświetlony
      tak aby wypełnił całą szerokość okna
    </p>
    zżąśęćńół
  </body>
</html>
```

A tak wygląda po wczytaniu przez przeglądarkę:

