

Elenchi

Con HTML è possibile generare tre tipi di elenco:

- a punti (non ordinato)
- numerati
- di definizioni

Gli elenchi possono contenere al loro interno altri elenchi.

Tag utilizzati:

`<ul>...</ul>` (unordered list – elenco a punti)

Questo tag definisce il contenuto di un elenco puntato. E' disponibile un attributo *type* per specificare il tipo di punto da visualizzare, che può assumere i seguenti valori:

- *square* per un piccolo quadrato
- *circle* per un punto vuoto circolare
- *disc* (simbolo predefinito) per un punto pieno circolare

`<li>...</li>`

Questi tag contengono le voci degli elenchi puntati e numerati

Si noti che all'interno di ciascuna voce è possibile inserire interruzioni di riga, paragrafi, link, altri elenchi, ecc.

`<ol>...</ol>` (ordered list – elenco ordinato)

Questi tag definiscono il contenuto di un elenco numerato. Esiste un attributo *type* per specificare lo schema di numerazione da usare (es. da quale numero iniziare). I valori dell'attributo *type* sono:

I per i numeri romani maiuscoli
i per i numeri romani minuscoli
a per lettere minuscole
A per lettere maiuscole

Si noti che il *type* predefinito visualizza numeri arabi.

L'attributo *start* stabilisce il numero o la lettera dalla quale l'elenco deve avere inizio
Ad es. *start=5* farà iniziare l'elenco dal numero 5 (arabo o romano) o dalla lettera E (quinta lettera dell'alfabeto). L'attributo *start* accetta solo valori numerici.

`<dl>...</dl>` (definition list)

Questo tag indica un elenco di definizioni, ovvero una lista di termini e delle spiegazioni relative a ciascun termine.

`<dt>...</dt>` (definition terms)

Questo tag contiene l'elenco delle voci da definire (nell'esempio successivo: Prima voce elenco definizioni)

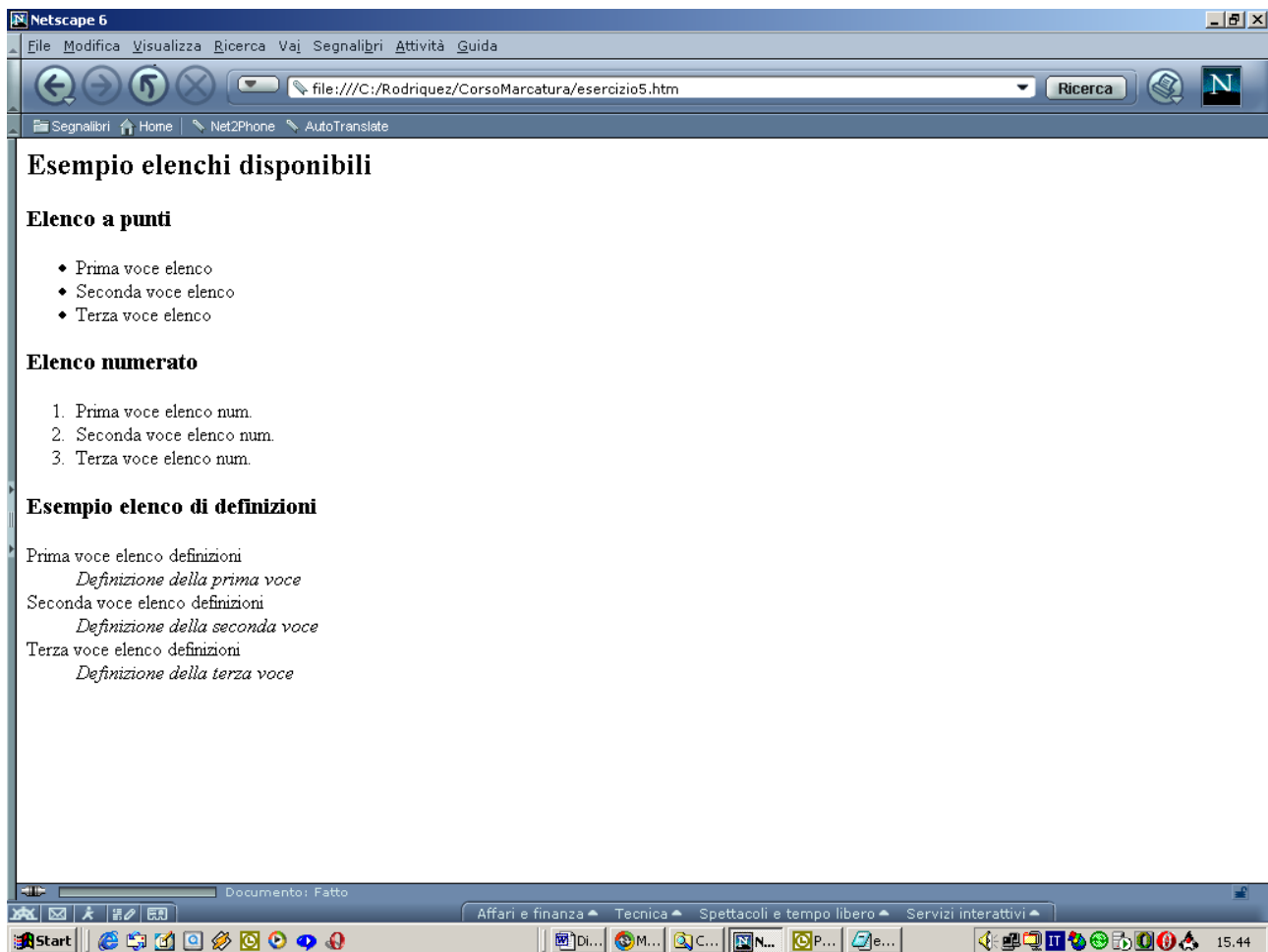
`<dd>...</dd>` (definitions)

Questo tag contiene le definizioni delle voci contenute nell'elenco

Esercizio

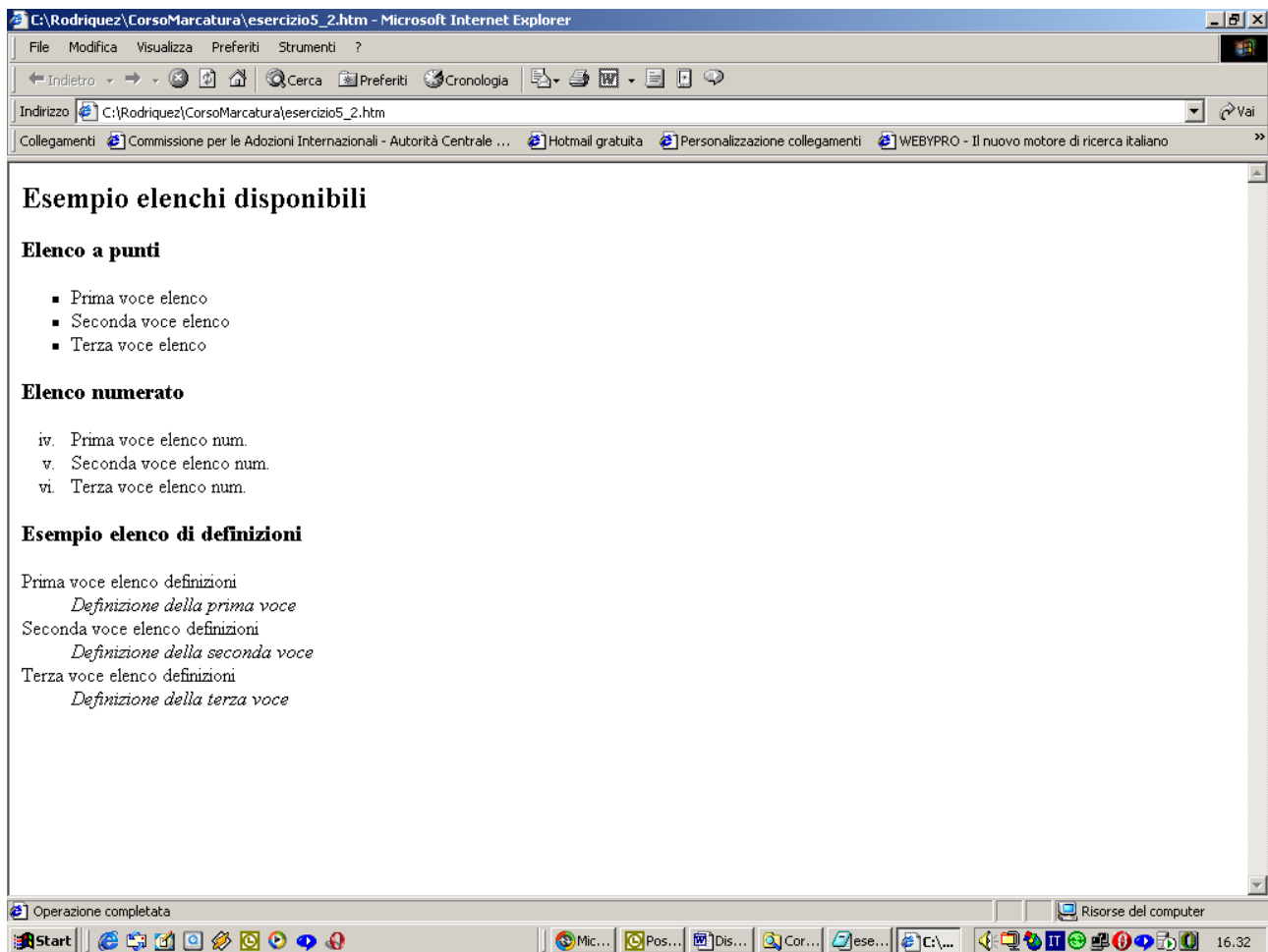
```
<html>
<body>
<h2>Esempio elenchi disponibili</h2>
<h3>Elenco a punti</h3>
<ul>
<li>Prima voce elenco</li>
<li>Seconda voce elenco</li>
<li>Terza voce elenco</li>
</ul>
<h3>Elenco numerato</h3>
<ol>
<li>Prima voce elenco num.</li>
<li>Seconda voce elenco num.</li>
<li>Terza voce elenco num.</li>
</ol>
<h3>Esempio elenco di definizioni</h3>
<dl>
<dt>Prima voce elenco definizioni</dt>
<dd><i>Definizione della prima voce</i></dd>
<dt>Seconda voce elenco definizioni</dt>
<dd><i>Definizione della seconda voce</i></dd>
<dt>Terza voce elenco definizioni</dt>
<dd><i>Definizione della terza voce</i></dd>
</dl>
</body>
</html>
```

Il codice genera il seguente risultato:



Esercizio con variazione di attributi

```
<html>
<body>
<h2>Esempio elenchi disponibili</h2>
<h3>Elenco a punti</h3>
<ul type="square">
<li>Prima voce elenco</li>
<li>Seconda voce elenco</li>
<li>Terza voce elenco</li>
</ul>
<h3>Elenco numerato</h3>
<ol type="i" start="4">
<li>Prima voce elenco num.</li>
<li>Seconda voce elenco num.</li>
<li>Terza voce elenco num.</li>
</ol>
<h3>Esempio elenco di definizioni</h3>
<dl>
<dt>Prima voce elenco definizioni</dt>
<dd><i>Definizione della prima voce</i></dd>
<dt>Seconda voce elenco definizioni</dt>
<dd><i>Definizione della seconda voce</i></dd>
<dt>Terza voce elenco definizioni</dt>
<dd><i>Definizione della terza voce</i></dd>
</dl>
</body>
</html>
```



Tabelle

Una tabella può essere definita una griglia (insieme di celle), formata da righe e colonne, che contiene informazioni. Ciascuna cella può contenere testo, righe, immagini, elenchi, form, e anche altre tabelle (tabelle innestate). Le tabelle sono estremamente utili per disporre i dati sulla pagina in maniera ordinata e sistematica. A tal fine sono spesso usate per mostrare informazioni disposte in formato tabellare senza visualizzare la corrispondente griglia bordata. Un esempio può essere dato dal formato simile alle colonne giornalistiche, nel quale il testo appare incolonnato e privo di bordi. Una tabella può essere usata anche per contenere parti di un'immagine completa per ridurre il tempo di download.

Sintassi:

```
<table [align="center"|"left"|"right"] [border [= "n"]]  
[cellpadding="n"] [width="nn%"] [cellspacing="n"] > ...  
</table>
```

Gli elementi *table* contengono la tabella. Tutto ciò che è compreso tra questi tag fa parte della tabella.

Si noti che gli attributi di questo tag si riferiscono all'intera tabella e non alle singole celle.

L'attributo *border* esprime l'ampiezza del bordo in pixel. Se non è specificato la tabella risulterà priva di bordo.

Relativamente alla spaziatura delle tabelle esistono gli attributi *cellpadding* e *cellspacing*.

L'attributo *cellpadding*, espresso in pixel, può essere usato per specificare lo spazio tra il contenuto della cella e il bordo interno (a valori numerici crescenti dell'attributo aumenta lo spazio tra celle e bordi).

L'attributo *cellspacing*, espresso in pixel, specifica invece lo spazio vuoto tra le celle.

L'attributo *width* determina lo spazio sullo schermo destinato alla tabella. Il valore dell'attributo può essere espresso in pixel o come un valore percentuale dello spazio disponibile tra i margini sinistro e destro. Usando un valore percentuale lo spazio è variato proporzionalmente nel caso in cui l'utente modifichi manualmente la dimensione della finestra del browser.

```
<tr [align="left"|"center"|"right"]  
[valign="top"|"center"|"bottom"] > ... </tr>
```

L'elemento *tr* (table row) contiene ogni riga della tabella.

```
<td >contenuto1, contenuto2,..., contenutoN</td>
```

L'elemento *td* (table data) indica il contenuto di ciascuna cella

Nel caso in cui il contenuto di determinate celle sia vuoto è probabile che il browser non visualizzi correttamente la cella, producendo sull'intera tabella una cattiva rappresentazione visiva. Per evitare ciò è preferibile, nelle celle vuote, usare sempre il carattere speciale ` `, illustrato precedentemente.

Le intestazioni delle tabelle sono specificate con il tag `<th>`.

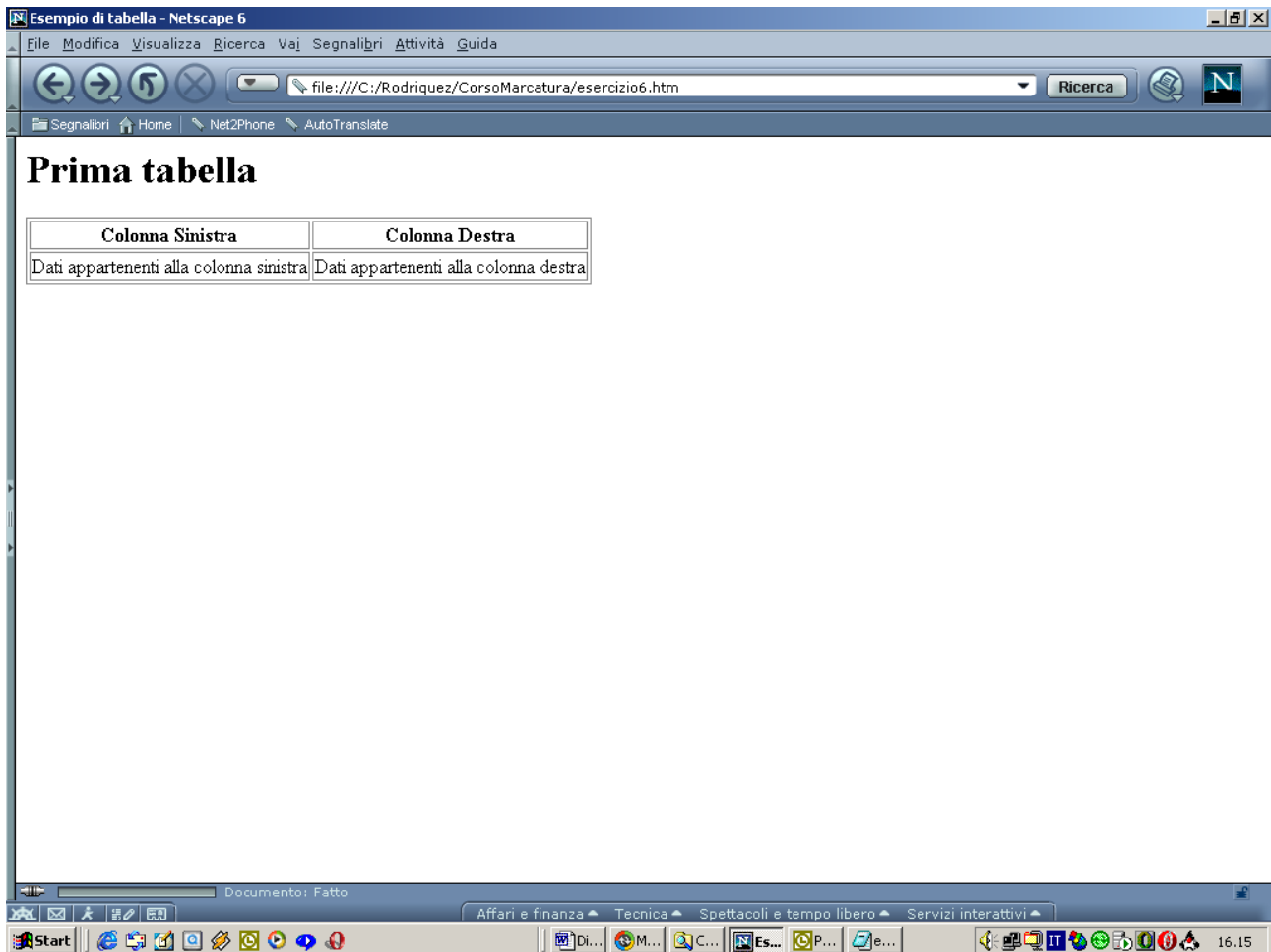
L'impostazione di default del browser centra l'elemento *th* (intestazioni) e allinea a sinistra gli elementi *td* (celle). L'allineamento di default può essere modificato con i valori `“left”`, `“center”`, `“right”`.

Per centrare il contenuto in verticale è disponibile l'attributo *valign* con i possibili valori `“top”`, `“middle”`, `“bottom”`, utilizzabile sia per le righe complete sia per singole celle. Il browser per default usa il valore `“middle”` per l'intestazione e il valore `“top”` per il contenuto di ogni singola cella.

Esercizio:

```
<html>
<head>
<title>Esempio di tabella</title>
</head>
<body>
<h1>Prima tabella</h1>
<table border="1">
<tr>
<th>Colonna Sinistra</th>
<th>Colonna Destra</th>
</tr>
<tr>
<td>Dati appartenenti alla colonna sinistra</td>
<td>Dati appartenenti alla colonna destra</td>
</tr>
</table>
</body>
</html>
```

Il codice genera il seguente risultato:



Sintassi

```
<th [align="left"|"center"|"right"]  
[valign="top"|"center"|"bottom"]  
[nowrap] [colspan="n"] [rowspan="n"] > ... </th>
```

L'elemento *th* è usato per le celle che contengono l'intestazione di tabella.

Il tag `<br>` può essere usato per forzare l'"a capo" del testo nelle celle.

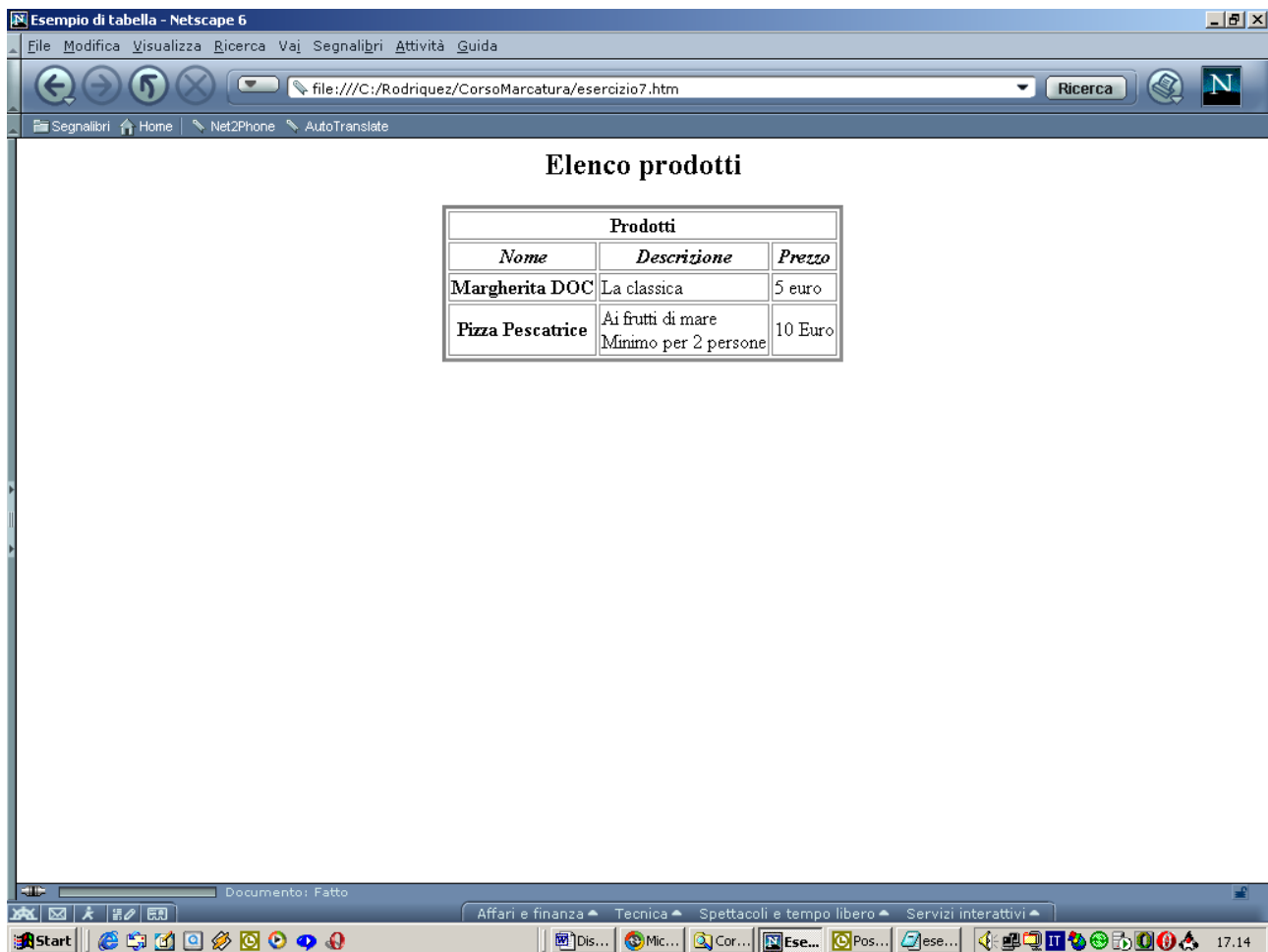
Gli attributi *colspan* e *rowspan* consentono di espandere lo spazio destinato ad una cella, rispettivamente, in orizzontale e in verticale

```
<td [align="left"|"center"|"right"]  
[valign="top"|"center"|"bottom"]  
[nowrap] [colspan="n"] [rowspan="n"] > ... </td>
```

Il tag *td* si riferisce al contenuto delle celle

Esercizio

```
<html>
<head>
<title>Esempio di tabella</title>
</head>
<body>
<h2 align="center">Elenco prodotti</h2>
<table border="3" align="center">
<tr>
<th colspan="3" align="center"> Prodotti </th>
</tr>
<tr>
<th><i>Nome</i></th>
<th><i>Descrizione</i></th>
<th><i>Prezzo</i></th>
</tr>
<tr>
<th>Margherita DOC</th>
<td>La classica</td>
<td>5 euro</td>
</tr>
<tr>
<th> Pizza Pescatrice</th>
<td>Ai frutti di mare<br> Minimo per 2 persone<br></td>
<td>10 Euro</td>
</tr>
</table>
</body>
</html>
```



Sintassi

`<caption> DescrizioneTabella</caption>`

Questo elemento opzionale è utile per fornire una breve descrizione del contenuto della tabella. Deve seguire il tag `table` e deve precedere l'elemento `tr` della prima riga. Può essere posizionato prima o dopo la tabella.

`<thead> ... </thead>`

`<tfoot> ... </tfoot>`

`<tbody> ... </tbody>`

`<colgroup [span="n"] [width="n"]> ... </colgroup>`

Questo tag è usato per raggruppare logicamente le colonne all'interno di una tabella. Il valore `width` verrà applicato a tutte le colonne del gruppo prive di un proprio valore di ampiezza.

`<col [span="n"] [width="n"]> ... </col>`

Questo tag è usato per impostare i valori delle singole colonne.

Esercizio

```
<html>
<head>
<!--Esempio con altri elementi di tabella -->

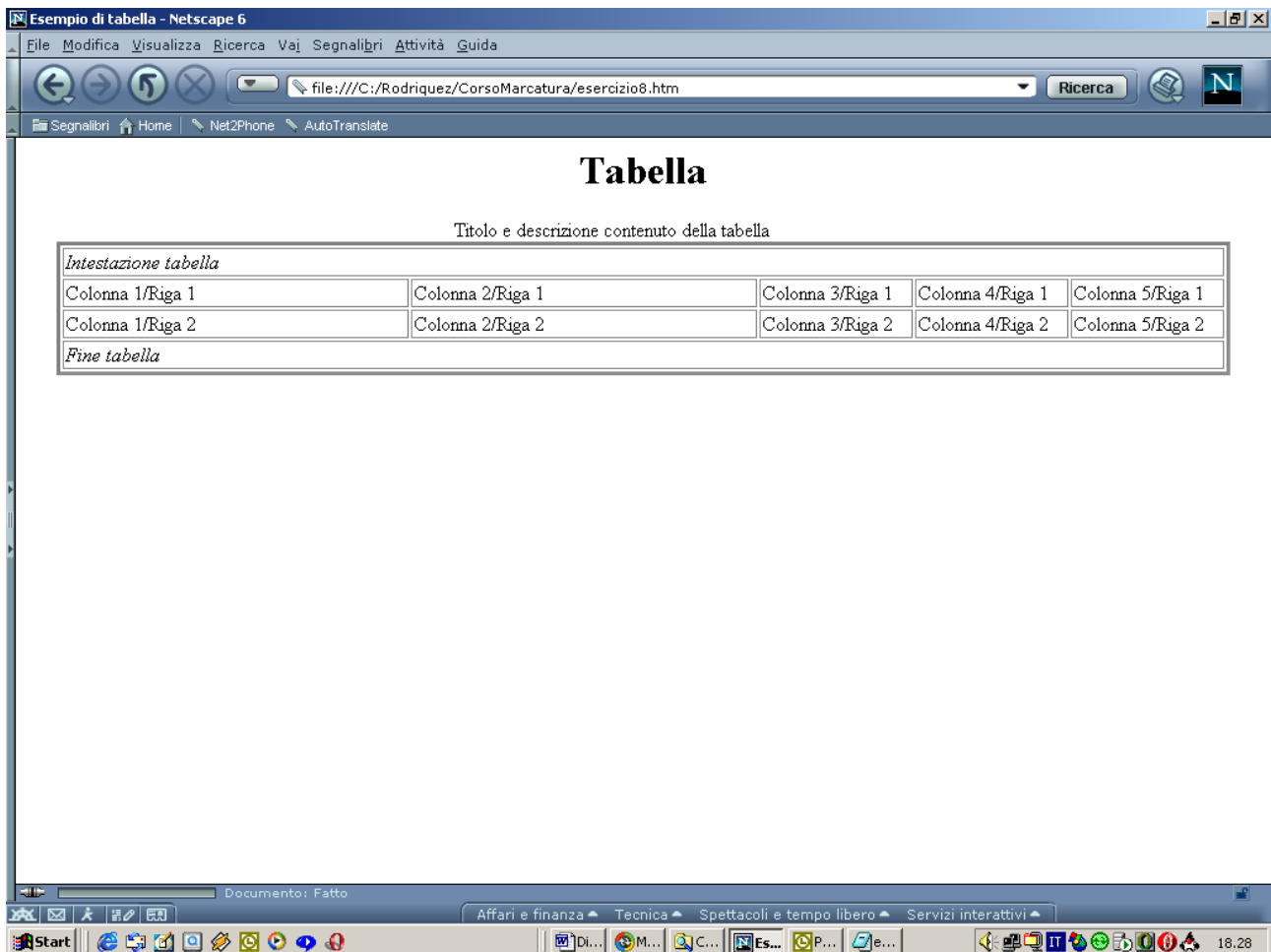
<title>Esempio di tabella</title>
</head>
<body>
<h1 align="center">Tabella</h1>

<table align="center" width="95%"border="3">
<caption>Titolo e descrizione contenuto della tabella</caption>

<colgroup width="30%" span="2"></colgroup>
<!-- L'ampiezza delle prime 2 colonne è pari al 30% -->

<colgroup span="3"></colgroup>
<thead>
<tr><td colspan="5"><i>Intestazione tabella</i></td></tr>
</thead>
<tbody>
<tr>
<td>Colonna 1/Riga 1</td>
<td>Colonna 2/Riga 1</td>
<td>Colonna 3/Riga 1</td>
<td>Colonna 4/Riga 1</td>
<td>Colonna 5/Riga 1</td>
</tr>
<tr>
<td>Colonna 1/Riga 2</td>
<td>Colonna 2/Riga 2</td>
<td>Colonna 3/Riga 2</td>
<td>Colonna 4/Riga 2</td>
<td>Colonna 5/Riga 2</td>
</tr>
</tbody>
<tfoot>
<tr><td colspan="5"><i>Fine tabella</i></td></tr>
</tfoot>
</table>
</body>
</html>
```

Il codice produce il seguente risultato:



Si noti la modalità di indicazione dell'ampiezza delle prime 2 colonne.

Gli attributi di dimensione non sono strettamente necessari in quanto, in mancanza di specifiche istruzioni, le dimensioni della tabella saranno tali da poter contenere i dati in essa presenti.

Per rendere invisibile la griglia della tabella si possono impostare gli attributi `border="0"` e `cellspacing="0"`.

Un attributo utile per dispositivi corredati con riproduttori acustici delle pagine è *scope* che prevede come valori *row* e *col*. Row fornisce l'intestazione per la riga cui appartiene la cella, col fornisce l'intestazione per la colonna cui appartiene la cella.

Esempio

```
<tr align="center">
  <th scope="col">Anno 2001</th>
  <th scope="col">Anno 2002</th>
</tr>
```

Spazi

```
H1 { margin-top: 7em; margin-bottom: 5em; }
```

Queste proprietà permettono di controllare direttamente gli spazi da visualizzare prima e dopo un blocco di testo (nell'es.: spaziatura prima dell'intestazione di primo livello).

L'unità di misura *em* consente di mantenere inalterato l'aspetto della pagina web al variare della dimensione del carattere eventualmente impostata dall'utente (l'unità di misura espressa in punti non è resa allo stesso modo da browser diversi).

Si noti che il margine inferiore di un'intestazione non viene aggiunto al margine superiore di un paragrafo posizionato al di sotto dell'intestazione. Quindi il valore applicato per la spaziatura sarà dato dal valore maggiore tra il margine inferiore e superiore dei blocchi di testo coinvolti.

Se non indicato diversamente gli spazi in bianco contenuti nel testo non sono riprodotti e i caratteri sono visualizzati di seguito uno dietro l'altro.

Per andare a capo (interruzione di riga) si utilizza il tag `<br>` mentre il tag `<p>`, oltre ad andare a capo, aggiunge un'interlinea.

Il browser adatta automaticamente il testo entro i margini della finestra. Per evitare l'adattamento tra due parole, nel caso le stringhe di caratteri debbano essere riprodotti fedelmente (ad es. marchi di fabbrica), può essere usato il tag:

```
&nbsp;
```

Caratteri speciali

Dal momento che in HTML alcuni caratteri rivestono un particolare significato (es. "<"), la visualizzazione di tali caratteri richiede l'utilizzo di *entità o caratteri speciali*.

Il carattere "&" indica l'inizio di un carattere speciale e il carattere ";" ne indica la fine. Al centro è contenuto il nome dell'entità oppure il carattere "#" ed il numero di un'entità. Ad esempio la visualizzazione del carattere "<" è effettuata scrivendo `<` oppure `<`. La differenza tra i due modi illustrati è che il primo, pur essendo più facile da usare in quanto l'uso di nomi e abbreviazioni è più agevole rispetto all'utilizzo dei numeri, non è supportato dai browser meno recenti mentre il secondo è supportato dai browser in maniera pressoché completa.

Di seguito è riportato un elenco dei principali caratteri speciali espressi con il nome dell'entità:

```
&nbsp; (non-breaking space; spazio)
< (minore)
> (maggiore)
& (e)
" (virgolette)
' (trattino)
© Copyright
® Trademark
```

Testo preformattato

Per evitare lo spostamento del testo al variare della dimensione della finestra del browser può essere usato il tag:

```
<pre> ... </pre>
```

Immagini

Il tag per inserire un'immagine è: `<img>`. Si tratta di un tag vuoto (empty) ovvero non richiede tag di chiusura e contiene solo attributi.

Sintassi:

```
1</sup> contiene l'URL del file immagine che si vuole visualizzare. La maggior parte dei browser riconosce le immagini in formato JPEG , GIF e PNG. Questi tre formati supportano il “progressive rendering” (l'immagine scaricata viene rifinita progressivamente).

Gli attributi *height* e *width*, espressi in pixel, impostano lo spazio sulla pagina da destinare all'immagine. Non sono obbligatori ma se valorizzati contribuiscono alla riduzione dei tempi di scaricamento dell'immagine.<sup>2</sup> Inoltre il testo di commento dell'immagine (attributo *alt*) sarà subito disponibile.

L'attributo *alt*<sup>3</sup> contiene del testo alternativo utilizzato per descrivere l'immagine. Il browser visualizza la stringa di caratteri quando il mouse è posizionato sull'immagine o se l'immagine non è caricata dal browser (ad es. perché l'utente ha inibito sul browser la visualizzazione delle immagini).

Si noti che l'uso dell'attributo *alt* è particolarmente importante per consentire ai programmi, in genere utilizzati da persone disabili, di riprodurre acusticamente il testo delle pagine web o, comunque, nel caso se l'utente ha a disposizione solo un browser di tipo testuale. Per questi motivi ogni immagine dovrebbe sempre includere un attributo *alt* coerentemente valorizzato.

---

<sup>1</sup> Src sta per source.

<sup>2</sup> I documenti HTML arrivano sul browser “a pezzi” e sono posizionati quando arrivano anche le sezioni circostanti.

Anche a causa della diversa risoluzione degli schermi non si conosce a priori l'area disponibile sul browser.

<sup>3</sup> Alt sta per alternate (text).

Va inoltre ricordato che i tempi occorrenti per lo scaricamento delle immagini sul browser suggeriscono di ricorrere alle immagini con moderazione.

Per quanto riguarda tutti gli aspetti connessi all'accessibilità il riferimento è costituito dalle "Linee guida per l'accessibilità ai contenuti del Web" del W3C (WAI Guidelines). WAI è l'acronimo di Web Accessibility Iniziative, il cui scopo è garantire a tutti parità di accesso all'informazione, incluse le categorie di persone svantaggiate.

Il testo che segue un'immagine è visualizzato, per default, sotto il bordo inferiore dell'immagine, mentre per mostrare il testo a fianco dell'immagine è possibile fare uso di una tabella.

L'attributo *usemap* è usato per definire una mappa immagine suddivisa in aree cliccabili, ad es. nel caso di cartine geografiche cliccabili su specifiche regioni.

```
<body background="URL" > ... </body>
```

Questo tag imposta l'immagine da usare come sfondo della pagina, che a partire dall'angolo superiore sinistro della finestra è ripetuta automaticamente fino a riempire la pagina. Tale processo è definito *tiling*.

Sono inoltre disponibili due ulteriori attributi: *hspace* e *vspace* che determinano, rispettivamente, lo spazio orizzontale a destra e sinistra dell'immagine e lo spazio verticale, sopra e sotto l'immagine.

Trattando le immagini di sfondo non deve essere dimenticato che queste interferiscono sempre con il testo che si trova in primo piano. Ogni pagina Web che include immagini di sfondo deve essere pertanto attentamente verificata per impedire che la lettura del testo risulti in qualche misura compromessa.

Esercizio

```
<html>

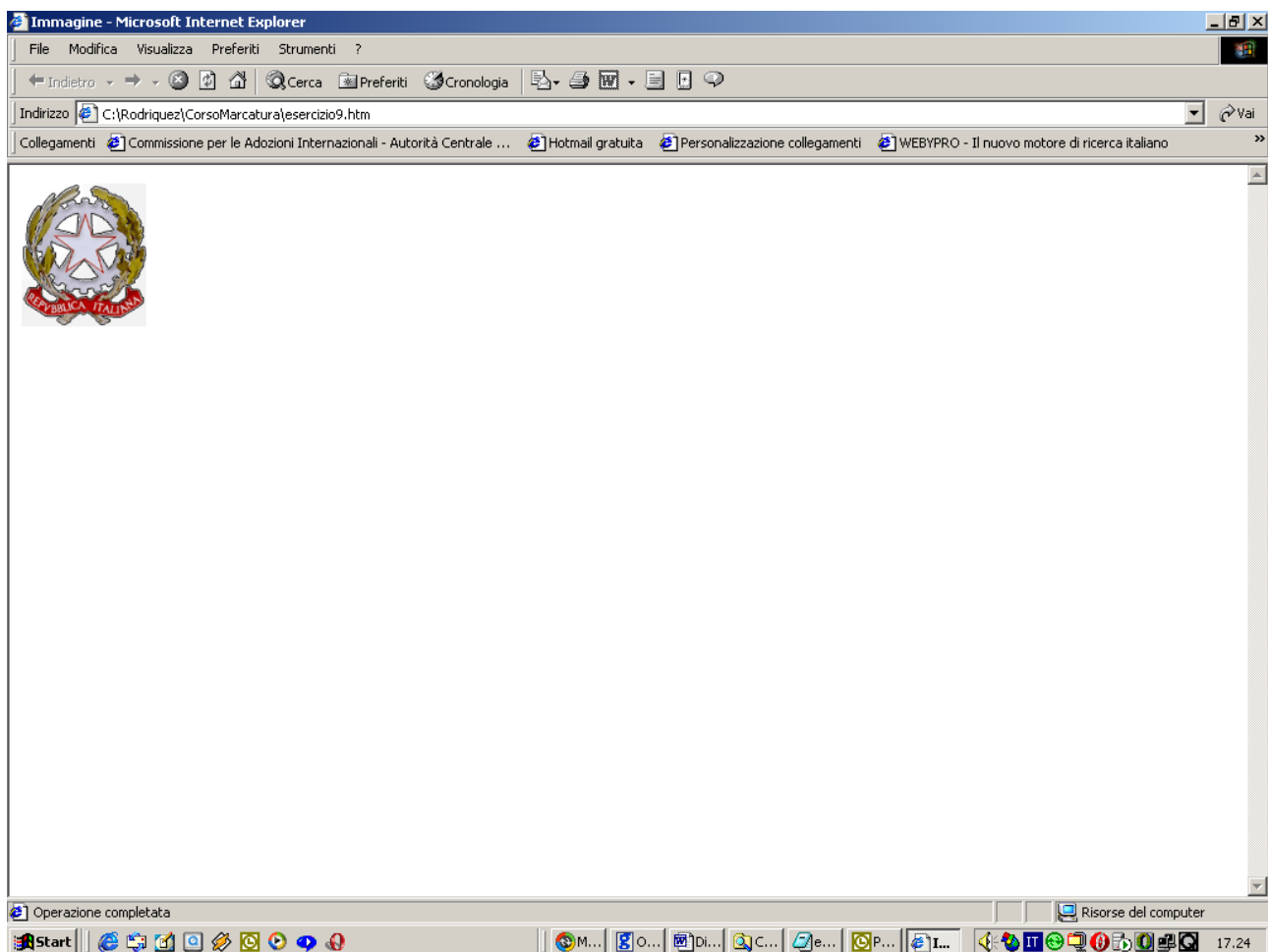
<head>
<!--Esempio immagine -->
<title>Immagine</title>
</head>

<body bgcolor="white">

</body>

</html>
```

Il codice genera il seguente risultato:



Può essere sottolineata la differenza tra gli attributi *bgcolor* e *background* del tag `<body>`. L'attributo *bgcolor* imposta il colore di sfondo mentre l'attributo *background* imposta un'immagine di sfondo.

L'attributo `background` è valorizzato tramite l'URL (relativo o assoluto) dell'immagine da visualizzare. Si tenga presente che nel caso in cui l'immagine è di dimensioni minori rispetto alla finestra del browser sarà automaticamente ripetuta sino a riempire l'intera finestra.

Si noti che l'URL deve indicare, oltre al nome dell'immagine, anche dove l'immagine stessa si trova. Nel caso indicato, non essendo presente il percorso (*path*) dell'immagine, il browser la cerca nella stessa cartella in cui si trova la pagina HTML.

Nella tabella successiva si presentano alcune semplici situazioni e il relativo percorso da indicare:

Immagine e pagina HTML nella stessa cartella	Src="stellone.gif"
Immagine in una cartella (cartella "images") di livello inferiore a quella della pagina HTML che la contiene	Src="images/stellone.gif"
Immagine in una cartella (cartella "images") di livello superiore a quella della pagina HTML che la contiene	Src="../stellone.gif"
Immagine in una cartella (cartella "images") di due livelli superiore a quella della pagina HTML che la contiene	Src="../../stellone.gif"

I riferimenti illustrati sono detti *relativi*, ovvero includono il percorso in modo parziale, in rapporto alla cartella in cui si trova la pagina HTML che contiene l'immagine.

Esiste anche la possibilità di utilizzare riferimenti *assoluti*, ovvero che includono completamente il percorso in cui si trova l'immagine. I riferimenti assoluti, tuttavia, dovrebbero essere usati solo quando si lavora *in locale* (sul proprio PC), mentre per le pagine da pubblicare su un server internet è preferibile usare i riferimenti relativi. Il percorso assoluto andrà usato necessariamente quando l'immagine risiede su un diverso server.

## Immagini e testo

```
<p> testo che appare
sul lato destro dell'immagine</p>
```

```
<p> testo che appare
sul lato destro dell'immagine</p>
```

Per far continuare il testo al di sotto dell'immagine:

```
<p> testo che appare
sul lato destro dell'immagine.<br clear="all"
Questa linea comincia al di sotto dell'immagine</p>
```

## Form

Attraverso le form (maschere video, moduli) l'utente può inviare informazioni al server che ospita il sito. Ciò richiede la possibilità di eseguire sul server script CGI, con le modalità previste dal fornitore del servizio internet (ISP - Internet Service Provider).

Le form aggiungono interattività alle pagine HTML, consentendo all'utente di interagire con il sito per richiedere informazioni, inviare risposte e, nei casi più complessi, eseguire transazioni commerciali. Quest'ultimo tipo di transazioni richiede particolari accorgimenti per garantirne la sicurezza.

Le form, definite tramite il tag `<form>`, contengono propri elementi (campi di testo, bottoni di tipo radio, menu, ecc.) attraverso i quali l'utente immette le informazioni.

Nelle form il tag più usato è il tag `<input>`, la cui tipologia è indicata dall'attributo *type*.

### Sintassi

```
<input type="text"|"password"|"checkbox"|"radio"|"password"|"submit" |
"reset"|"button"|"image"|"hidden"
name="NomeDaDareAlInput" [value="StringaDiDefault"] [checked] [size="n"]
[maxlength="n"] [src="URL"]
[align="top"|"bottom"|"middle"|"left"|"right"]>
```

La tipologia *text* è quella più diffusa per immettere dati in una form. Questa tipologia accetta una singola riga di testo il cui numero di caratteri può essere definito tramite l'attributo *maxlength*. Nella maggior parte dei browser la lunghezza di default del campo di tipo text è pari a 20 caratteri. L'inserimento di due o più righe è ottenibile ricorrendo al tag *textarea*. Se l'utente immette un numero di caratteri superiore allo spazio disponibile sullo schermo per la *textarea* appare una barra di scorrimento verticale per consentire lo scorrimento del testo. Il numero di caratteri che è consentito immettere in una *textarea* può essere controllato e limitato con l'uso di un opportuno programma di script.

Il riconoscimento dell'input richiede un nome, dato dal parametro *name*, con il quale il sistema è in grado di gestire l'input per le successive elaborazioni. Il valore dell'input immesso dall'utente (contenuto del campo) verrà quindi associato al nome (parametro *name*).

In una stessa form coesistono tipicamente più input, di tipo uguale o diverso.

L'attributo facoltativo *value*, se definito, costituisce il valore di default che appare nell'input al caricamento della form, e può essere modificato dall'utente. Ove possibile è preferibile proporre sempre valori di default in quanto riducono i tempi di immissione dei dati, suggeriscono il contesto dei possibili valori e, in genere, fanno apparire la form maggiormente comprensibile.

La tipologia *password* è simile al tipo text, con la differenza che non appaiono sullo schermo i caratteri digitati ma un asterisco per ciascun carattere immesso.

Si noti che i dati immessi tramite il tipo password non vengono criptati e sono pertanto inviati al server in chiaro.

Il tipo *checkbox* (caselle di scelta) mostra sullo schermo una casella che è possibile valorizzare con un click del mouse. Ciascuna casella potrà avere solo i valori “on” oppure “off”. Se la casella è contrassegnata il valore inviato al server sarà “Name=on”. In genere le checkbox sono utilizzate per consentire all’utente di scegliere tra una o più opzioni nell’ambito di un limitato numero di proposte prestabilite. Ad esempio per conoscere quali sono gli hobby dell’utente (le risposte potrebbero essere più di una tra le proposte presentate) useremo il tipo checkbox. Per consentire un’unica scelta fra tutte le proposte si usa il tipo “radio”.

Il tipo *radio* (pulsanti) raggruppa più checkbox con un unico nome (e con valori diversi). Viene visualizzato con la modalità dei pulsanti di selezione e consente un’unica scelta. Ad esempio per conoscere la fascia di età dell’utente (la risposta ricadrà solo in una delle proposte presentate) useremo il tipo radio. Si noti, quindi, che la scelta non avviene solo tra due possibilità (ad es. maschio/femmina) ma è la risposta che è unica. L’attributo *checked* può essere usato per preselezionare una scelta specifica nell’ambito delle possibili proposte.

Il valore scelto dall’utente è trasmesso al server con la coppia univoca nome + valore.

Si noti che nel tipo checkbox l’attributo name è diverso e l’attributo value è lo stesso (è possibile contrassegnare contemporaneamente più caselle) mentre nel tipo radio (pulsanti) l’attributo name è lo stesso e l’attributo value cambia (è possibile selezionare solo uno tra i pulsanti previsti).

Il tipo *submit* invia i dati al server. Il testo che appare sul bottone (solitamente “Invia”) può essere comunque modificato mediante l’attributo value.

Il tipo *reset* elimina dallo schermo i valori digitati dall’utente, ripristinando contemporaneamente gli eventuali valori di default previsti dalla pagina.

Il tipo *image* fa apparire sul bottone un’immagine.

Il tipo *hidden* (costituito da una coppia nome/valore) è inviato al server ma non è visualizzato sullo schermo. Può essere usato per l’invio di informazioni necessarie alle successive elaborazioni del server e che non interessano l’utente (ad es. invio di informazioni sulla versione della pagina HTML usata).

```
<form action="URL" method="post"|"get"> ... </form>
```

Il tag *form* è il tag di definizione della form. L’attributo *action* indica la locazione e il nome di uno script CGI per eseguire l’elaborazione, ovvero il nome di un file al quale inviare il contenuto della form. Lo script indicato nell’attributo action eseguirà le elaborazioni previste con le informazioni ricevute.

L’attributo *method* può assumere i valori *get* e *post*: il metodo *get* contatta il server e le informazioni sono incluse in coda all’URL, inviando i dati in un’unica trasmissione; con il metodo *post* viene prima contattato il server, le informazioni faranno parte del corpo del messaggio e la trasmissione dei dati avviene separatamente.

## Esercizio

```
<html>
<head>
<title>Società di Servizi - Sondaggio</title>
</head>
<body>
<form action="mailto:casella@indirizzosocietà.com"
method="post">
<p align="left"> Nominativo:
<p align="left"> <input type="text"
name="nominativo" maxlength="30" size="50">
<p align="left">Interessi:

<input type="checkbox" name="Viaggi" checked>Viaggi

<input type="checkbox" name="Foto" >Foto

<input type="checkbox" name="Musica" >Musica

<input type="checkbox" name="Altro" checked>Altro

<p align="left">Genere:

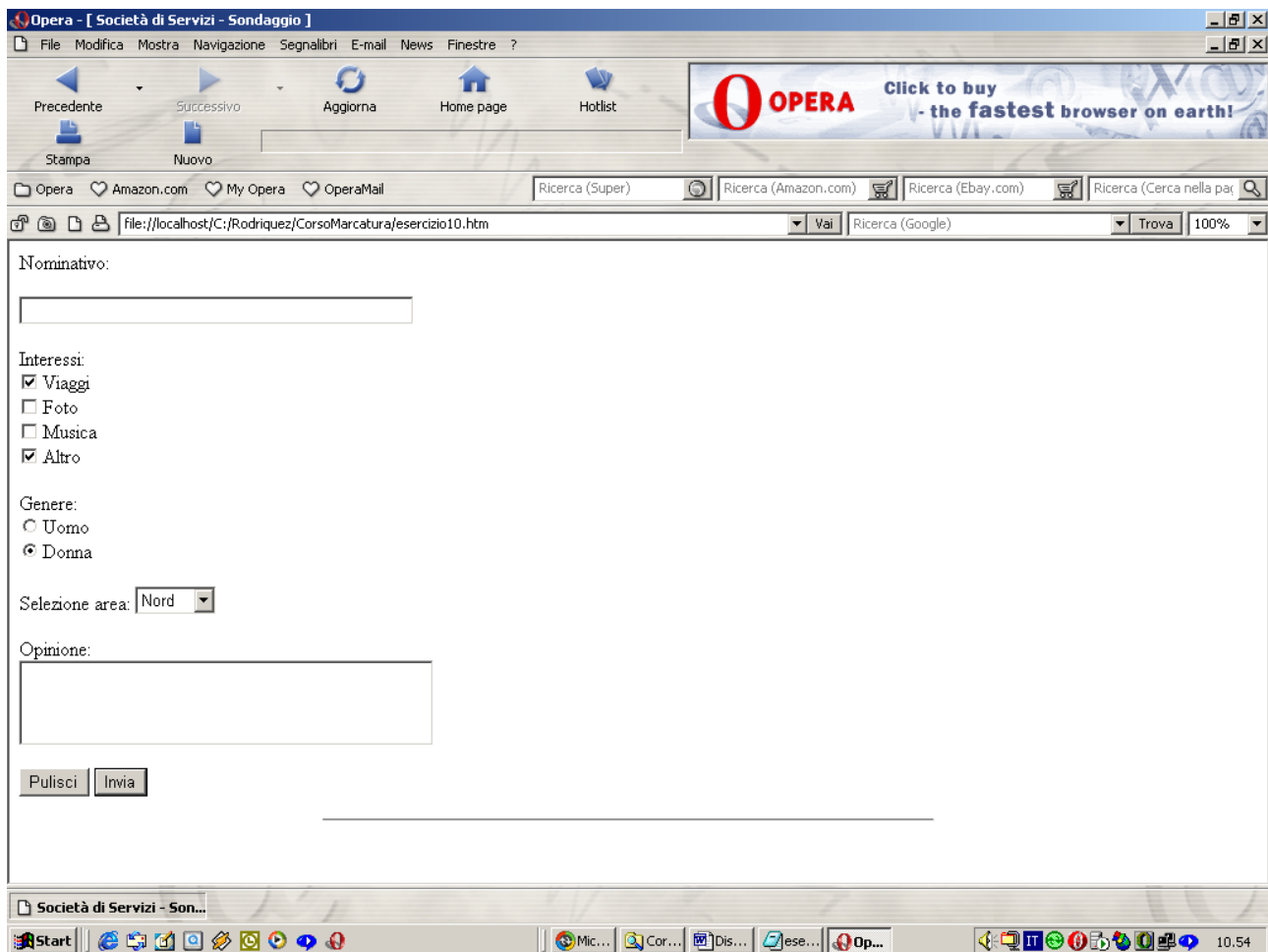
<input type="radio" name="genere" value="uomo">Uomo

<input type="radio" name="genere" value="donna" checked>Donna

<p align="left">Selezione area:
<select name="area" size="1">
<option value="Nord" selected> Nord
<option value="Centro"> Centro
<option value="Sud"> Sud
</select>
<p>Opinione:

<textarea name="opinione" rows="4" cols="40">
</textarea>
<p align="left"><input type="reset" name="reset" value="Pulisci">
<input type="submit" name="risposta" value="Invia">
</form>
<hr width="50%">
</body>
</html>
```

Il codice genera il seguente risultato:



I valori che appaiono selezionati sono quelli dichiarati nel codice come valori predefiniti. Si noti che è possibile inserire valori di default non solo tramite l'attributo checked previsto per i relativi controlli ma anche per i campi di tipo testo e per la textarea. Variando tali valori e/o inserendo caratteri nei campi di tipo testo e cliccando sul bottone "Pulisci", appaiono nuovamente sulla form i valori predefiniti.