

Internet

Internet è la rete di computer più estesa e popolata del mondo (rete delle reti). Nasce agli inizi degli anni '70 del secolo scorso, nel periodo della “guerra fredda”, su iniziativa del Dipartimento della Difesa degli Stati Uniti. Fu finanziato un progetto per individuare un mezzo in grado di funzionare anche se interrotto in più punti, in modo da consentire alle autorità governative di continuare a comunicare anche a seguito di un attacco nucleare. I principi fondamentali sui quali si è basato tale progetto erano: la rete doveva essere considerata inaffidabile, i nodi di comunicazione dovevano essere gerarchicamente uguali ed in grado di originare, trasmettere e ricevere i messaggi, i messaggi stessi dovevano essere suddivisi in porzioni più piccole di informazioni (pacchetti), ciascun pacchetto separatamente indirizzato poteva seguire un diverso percorso.

Nel 1969 furono connessi tra loro 4 nodi, e la rete così formata venne denominata ARPANet. Successivamente, nel 1972, il numero dei nodi connessi era salito a 37 e, in quel periodo, nacque il primo servizio di posta elettronica.

Agli inizi degli anni '90 Internet comincia ad essere conosciuta dal grande pubblico, diventando un fenomeno di massa.

Servizi/applicazioni Internet

Di seguito sono sinteticamente esposti i servizi principali e le applicazioni più importanti disponibili sulla rete Internet.

Posta elettronica

Consente lo scambio di messaggi in formato elettronico. Con la posta elettronica si superano i problemi di distanza e di fuso orario, è possibile inviare simultaneamente messaggi a più destinatari, l'effettiva consegna del messaggio avviene in qualche secondo. Un indirizzo di posta elettronica, che equivale ad un indirizzo postale nella rete, è composto da due parti separate dal simbolo @ (es. nomecognome@provider.it). La parte a sinistra del simbolo è la casella postale, la parte a destra rappresenta il *dominio* di appartenenza. Per gestire la posta elettronica sul computer dell'utente esistono numerosi programmi. La posta elettronica non è un mezzo di comunicazione sicuro, nel senso che non garantisce la riservatezza e l'inalterabilità dei messaggi scambiati. Scambiare messaggi con sicurezza e riservatezza richiede l'utilizzo di ulteriori mezzi e accorgimenti tecnici.

Newsgroups

Si tratta di bacheche composte da articoli creati ed inviati da diversi utenti, in genere suddivise per argomento. Lo scopo è quello di condividere informazioni ed esperienza fra diverse persone.

FTP (File Transfer Protocol)

Questo servizio consente di trasferire file di qualsiasi tipo da un sistema ad un altro. Il trasferimento verso (o la copia da) un server FTP richiede la connessione al server mediante un identificativo e una password (account) validi. Una volta eseguita la connessione si trasferiscono o si copiano i file verso/dal server utilizzando gli opportuni comandi FTP.

Telnet

Questo servizio consente di connettersi ad un altro computer e di utilizzarne le risorse come se si lavorasse direttamente su tale computer.

WWW (World Wide Web)

Ideato al CERN di Ginevra nel 1983, consente di costruire documenti ipertestuali e multimediali, con il supporto del linguaggio HTML (Hyper Text Markup Language).

Il protocollo¹ di accesso ai servizi WEB è chiamato HTTP (Hyper Text Transport Protocol), che rappresenta il linguaggio con il quale i web client e i web server comunicano tra loro.

¹ Un protocollo è un insieme di regole per la comunicazione fra computer. Può essere paragonato ad un linguaggio con le proprie regole grammaticali che deve essere conosciuto dai sistemi per potersi scambiare informazioni.

Introduzione al World Wide Web

Concetti base

Le informazioni presenti sul Web² sono contenute in *siti web*. Gli utenti usano i *web client* o *browser* (programmi come Explorer, Navigator, Opera, ecc.) per accedere ai siti ed alle informazioni pubbliche in essi ospitate. I browser rappresentano quindi i programmi di interfaccia che consentono agli utenti di esplorare il Web, e sono corredati da una serie di pulsanti (avanti, indietro, home page, stampa, ecc.) che permettono, tra l'altro, di muoversi tra le diverse pagine informative e di stamparle.

Le informazioni di un sito sono ospitate in documenti (*risorse*) generati tramite il linguaggio HTML, “capito” dai browser che provvedono alla loro visualizzazione. Gli autori dei documenti sono pertanto i responsabili dei contenuti pubblicati sul Web, mentre i browser si occupano di visualizzare tali contenuti. Va tenuto presente che i dispositivi di visualizzazione delle risorse Web non sono esclusivamente personal computer ma possono essere costituiti anche da altri tipi di apparati come computer palmari, telefoni cellulari, ecc.

L'accesso alle informazioni contenute su un sito web è gestito da un *web server*. Si tratta di un software (programma) che invia le pagine di risposta alle richieste che gli utenti formulano tramite il browser.

L'insieme delle regole (*protocollo*) che governa lo scambio di informazioni tra web server e web client è l'http (HyperText Transfer Protocol).

Nella figura successiva sono rappresentate le diverse componenti sinora trattate.

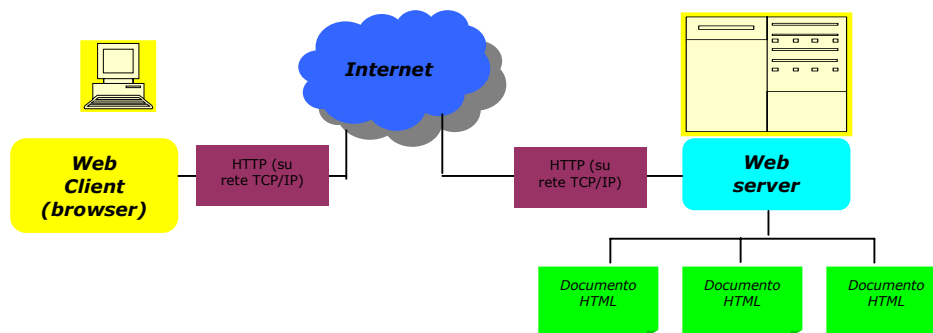


Fig. 1: Web client, Web server, documenti HTML

La visualizzazione di una pagina web sul computer dell'utente avviene secondo i semplici passi qui di seguito riepilogati:

- il browser richiede una documento HTML;
- il web server individua il documento;
- il web server invia il documento al browser che ne ha fatto richiesta;
- il browser visualizza il documento.

² Web e World Wide Web sono qui usati come sinonimi.

Lo spostamento fra le pagine dei documenti HTML e tra pagine diverse (appartenenti allo stesso sito o a siti diversi) avviene cliccando sugli appositi *link* (collegamenti ipertestuali) predisposti nei documenti.

I documenti e le parti di documento raggiungibili sono identificati da un indirizzo URL (Uniform Resource Locator) del tipo <http://www.nomesito.it/documento.htm>.

Nell'esempio l'URL identifica il metodo (protocollo) di connessione (http), il sito da contattare (nomesito), e lo specifico documento da visualizzare (documento.htm).

Ogni risorsa presente sul Web ha un indirizzo URL univoco. L'URL può contenere anche valori inseriti dall'utente in una *form* (maschera video).

Cliccando su un link l'URL corrispondente è inviato al server del sito web indicato e successivamente il web server individua il documento e lo invia, tramite internet, al web client.

In un sito web basato su file (*file based*) ogni risorsa è costituita da un file a sé stante. Ne consegue una grande difficoltà di gestione dovuta al gran numero di file da gestire e alle loro versioni, ma soprattutto al corretto mantenimento dei collegamenti (link) tra i documenti. Inoltre, nella realtà, molte informazioni che in genere risiedono su database sono di tipo dinamico, essendo soggette a variazioni. In questi casi oltre all'aggiornamento delle informazioni nel database occorre gestire anche i corrispondenti aggiornamenti nei file pubblicati sul sito. Queste ragioni hanno determinato lo sviluppo di architetture e tecniche rivolte alla pubblicazione diretta di database, o di sue parti, dando luogo a siti a contenuto dinamico.

Pagine statiche e dinamiche

Un documento HTML memorizzato in un file rappresenta una pagina web statica. Il suo contenuto ed il suo aspetto, definiti dal progettista/sviluppatore, rimangono invariati fino a quando il file stesso non è aggiornato. Una pagina dinamica è invece una pagina generata nel momento stesso in cui è acceduta. Un esempio di pagina generata dinamicamente può essere costituito da una pagina relativa ad un elenco di prodotti che mostra sia le loro informazioni generali e le loro immagini (parte statica) sia i prezzi e le quantità disponibili, ospitate nel database (parte dinamica). La parte statica è costituita da informazioni più stabili nel tempo mentre la parte dinamica è formata dalle componenti che variano con maggiore frequenza, come appunto nel caso delle quantità disponibili e del prezzo dei relativi prodotti.

Di seguito si riportano alcune caratteristiche delle pagine dinamiche:

- possono fornire informazioni costantemente aggiornate;
- possono restituire informazioni specifiche in base ai dati forniti dall'utente tramite una maschera video;
- possono essere personalizzate in relazione alle preferenze espresse dall'utente (aree tematiche di interesse, evidenziazione di "novità", livello di esperienza utente, ecc.).

Le pagine dinamiche, in particolare, possono contenere:³

- testo HTML statico, sempre uguale ad ogni visualizzazione della pagina;
- istruzioni di programma, ad esempio per eseguire le interrogazioni sul database ospitato sul server;
- dati dinamici provenienti dal database, ad ogni visualizzazione della pagina.

³ E' opportuno osservare, anche alla luce delle caratteristiche appena descritte, che la presenza di immagini in movimento o di file video e/o audio in una pagina HTML non sono sufficienti per classificare la pagina stessa come dinamica. Una pagina statica, anche se corredata con immagini animate, è pur sempre una pagina statica.

Le componenti statiche della pagina possono essere ospitate in documenti HTML detti anche *template* che contengono il testo HTML statico, che non varia al caricamento della pagina, le istruzioni del programma che generano i dati dinamici e le istruzioni di formattazione ovvero la rappresentazione “tipografica” delle informazioni.

L’elaborazione delle pagine HTML dinamiche avviene secondo i passi riportati di seguito:

- il browser richiede la pagina web dinamica;
- il web server individua la pagina e la passa all’application server;
- l’application server analizza la pagina individuando le istruzioni e, in base a tali istruzioni, la completa;
- l’application server trasmette la pagina elaborata e completata al web server;
- il web server invia la pagina completata al browser che ne ha fatto richiesta;
- il browser visualizza la pagina.

Si osservi che l’application server rimuove le istruzioni di elaborazione dalla pagina e ciò che il browser riceve è, comunque, una pagina statica in quanto l’elaborazione ha popolato la pagina stessa con le informazioni appropriate. La pagina che arriva al browser contiene quindi solo codice HTML.

Database e web

Le istruzioni del programma sono interpretate dall’application server e non sono visibili dal client in quanto vengono sostituite delle informazioni generate durante la creazione della pagina.

Per ricercare le informazioni aggiornate presenti nel database e per far sì che il Web Server possa elaborarle è richiesta la presenza di un’application server (o gateway).

L’application server è un software che, unitamente al web server, elabora pagine web che contengono le istruzioni da eseguire sul server (ad es. per selezionare determinate informazioni). L’application server deve essere provvisto di un’interfaccia verso il web server e di un’interfaccia verso il database (DBMS). L’interfaccia web server più comune è denominata CGI (Common Gateway Interface). L’interfaccia DBMS è nativa di ogni DBMS, e sono disponibili anche interfacce di tipo ODBC (Open Database Connectivity).

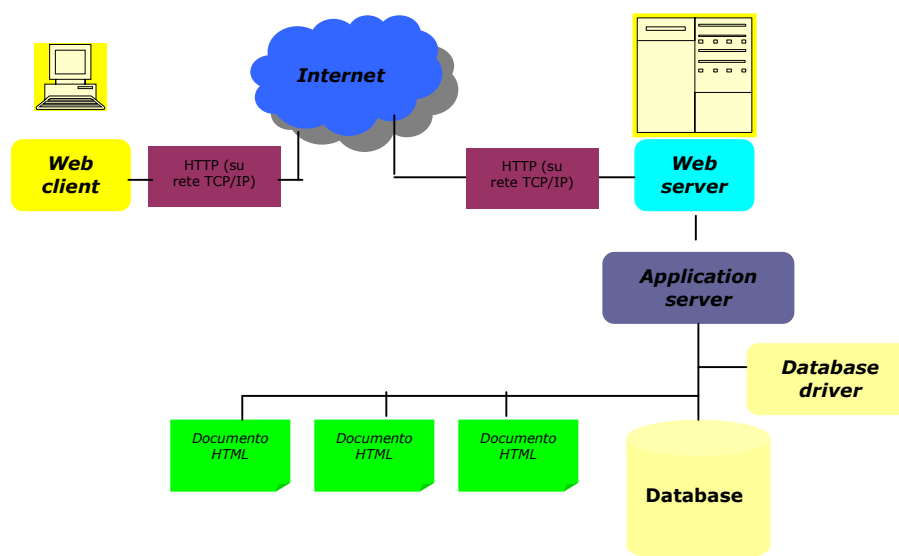


Fig. 2: Web client, Web server, Application server, documenti HTML e database

CGI è un'interfaccia web server standard usata per “connettere” non solo DBMS ma qualsiasi applicazione esterna con il web sever. Quando il web server riceve un URL corrispondente ad una risorsa CGI esegue un programma CGI (come uno script Perl o un altro tipo di application server) che si connette al DBMS, esegue l'elaborazione e restituisce i risultati ottenuti al web server affinché siano inviati al web client. L'application server comunica con il database attraverso un *database driver* che agisce da interprete tra database e application server.

L'accesso ad un database avviene secondo i passaggi indicati di seguito:

- il browser richiede la pagina web dinamica;
- il web server individua la pagina e la passa all'application server,
- l'application server analizza la pagina individuando le istruzioni di interrogazione (query) al database;
- l'application server invia la query al database driver;
- il database driver esegue la query sul database;
- il database driver invia i dati ottenuti tramite la query (recordset) all'application server;
- l'application server inserisce i dati nella pagina e la trasmette al web server;
- il web server invia la pagina completata al browser che ne ha fatto richiesta;
- il browser visualizza la pagina.

HTML - Hypertext Markup Language

HTML, acronimo di *HyperText Markup Language*, è il linguaggio standard per descrivere e definire il contenuto e l'aspetto delle pagine (risorse) sul World Wide Web. Può essere definito anche come un metodo di descrizione del formato delle pagine.

HTML non è un linguaggio di *programmazione* né un linguaggio di *descrizione dei dati* (come invece è XML), ma è un linguaggio di *markup* (marcatura o contrassegnazione). Ciò significa che tramite HTML si contrassegnano specifiche parti di testo, seguendo la sintassi prevista dal linguaggio, attribuendo così a tali parti un determinato significato. La parola hypertext dell'acronimo si riferisce alla caratteristica del linguaggio di poter creare in determinati punti del testo collegamenti (hyperlink) ad altre risorse (altre parti dello stesso testo, altre pagine, altri siti, ecc.). In genere si intende per documento o pagina HTML il file contenente il testo e le relative istruzioni in linguaggio HTML mentre con pagina Web si fa riferimento al risultato visualizzato sullo schermo. La pagina Web è quindi la rappresentazione del documento HTML.

Le pagine HTML sono i documenti di base del World Wide Web. Possono contenere testi, immagini, suoni, filmati, collegamenti (hyperlink ad altre pagine o fonti informative) che danno luogo agli *ipertesti*, nei quali si possono includere più fonti di informazione in un unico schema documentale. Le pagine Web possono anche essere usate per l'immissione dei dati da parte dell'utente, come ad esempio per la compilazione di maschere video (*form*) di risposta ad un sondaggio o per l'esecuzione di transazioni commerciali. Le pagine HTML contengono comandi incorporati che forniscono informazioni sulla struttura, l'aspetto e il contenuto della pagina stessa, denominati *tag*. I browser utilizzano le informazioni fornite dai tag per determinare come visualizzare la pagina e anche per eseguire le azioni richieste dall'utente, come nel caso dell'attivazione dei collegamenti ipertestuali o dell'invio di un modulo di risposta mediante un click su un apposito pulsante. In sintesi, ciò è analogo a quanto avviene con gli elaboratori di testo nei quali, tramite gli opportuni comandi, si applica al testo la *formattazione*, come il tipo e la dimensione con cui rappresentare i caratteri, l'allineamento del testo, la spaziatura tra le righe, ed altro ancora, determinando in tal modo l'applicazione di uno *stile*. Nei programmi di elaborazione di testi, però, la formattazione avviene tramite istruzioni di marcatura proprie dei differenti programmi (si dice che si tratta di istruzioni di tipo *proprietario*). Di conseguenza si viene a determinare una stretta dipendenza tra programmi e documenti, rendendo necessaria una conversione di formato per lavorare sugli stessi documenti con differenti programmi applicativi. HTML, invece, assicura l'indipendenza della rappresentazione dei dati da tecnologie proprietarie. L'apprendimento di HTML è inoltre facilitato perché l'insieme dei tag ed il loro significato è predefinito e valido per tutte le pagine.⁴ La formattazione del testo, in HTML, si realizza mediante l'utilizzo dei *tag* (nome di un formato incluso tra parentesi angolari).

Esempio

`<b>` è un tag che indica di formattare il testo in grassetto.⁵

Di seguito si riportano due righe di codice HTML che si riferiscono, rispettivamente, a una stringa di testo in carattere normale e ad un'altra in carattere grassetto.

```
stringa in carattere normale
<b>stringa in carattere grassetto</b>
```

⁴ Ciò costituisce, però, uno dei limiti del linguaggio.

⁵ “b” sta per bold (grassetto).

Visualizzando questo codice mediante un browser si ottiene la schermata di Fig. 3.

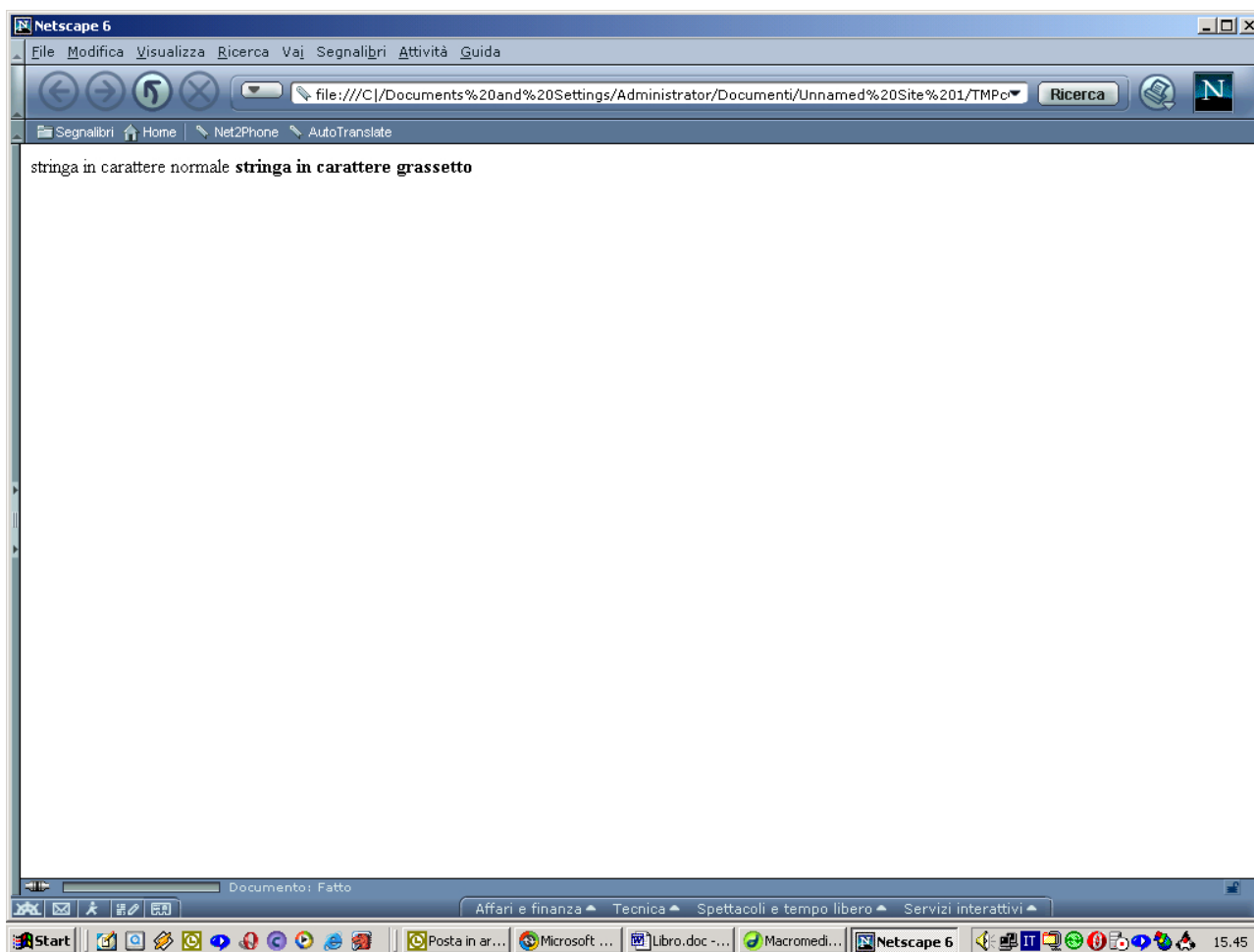


Fig. 3: Stringhe in carattere normale e in carattere grassetto

In questa semplicissima pagina Web si nota che la prima stringa di caratteri è scritta in modalità normale mentre la seconda, che è la “zona” influenzata dall’effetto del tag `<b>`, è invece mostrata in modalità grassetto. Si osservi, intanto, che mentre il codice è scritto su due differenti righe di testo la rappresentazione a video del codice avviene su un’unica riga. Ciò è del tutto normale ed è una delle caratteristiche del linguaggio HTML che verranno successivamente espone.

I tag HTML non sono sensibili alle differenze tra caratteri maiuscoli e minuscoli per cui, ad esempio, scrivere `<b>` è del tutto equivalente allo scrivere `<B>`.⁶ Va comunque tenuto presente che il World Wide Web Consortium (W3C, www.w3c.org), il consorzio internazionale composto da oltre 500 enti ed aziende, e preposto alla regolamentazione delle tecnologie Web, nelle proprie raccomandazioni HTML 4 *suggerisce* l’uso di tag scritti in caratteri minuscoli mentre XHTML (la successiva generazione HTML) *richiede* necessariamente che i tag siano in caratteri minuscoli. E’ buona norma, quindi, abituarsi il prima possibile a scrivere i tag con caratteri minuscoli.

Gli standard HTML sono creati dal W3C, e sono stati aggiornati più volte. Tuttavia le diverse versioni del linguaggio sono state implementate in modo differente dai diversi browser. Inoltre alcuni produttori hanno introdotto caratteristiche valide solo per i browser di propria produzione (caratteristiche *proprietarie*).

⁶ Si dice che i tag sono “case-insensitive” mentre il caso opposto è invece detto “case-sensitive”.

Come abbiamo appena visto nell'esempio precedente, la formattazione del testo è eseguita dal browser solo attraverso la lettura e l'interpretazione dei tag presenti nel documento HTML. Per questo motivo gli spazi o le righe in bianco eventualmente contenute in un testo HTML non vengono rappresentate nel modo in cui appaiono nel codice HTML. Sia le linee lasciate in bianco sia gli spazi nel documento HTML vengono rappresentati e visualizzati come un unico spazio. Per visualizzare il testo con una determinata formattazione, quindi, non vanno mai utilizzati spazi e righe in bianco ma si deve ricorrere all'utilizzo di particolari tag che consentono la visualizzazione del testo esattamente come è rappresentato nel codice. La caratteristica appena discussa consente di scrivere il codice HTML aggiungendo liberamente spazi in bianco e indentazioni nel codice, rendendolo più facilmente leggibile senza alterarne il significato. A ciò va comunque aggiunto che, per diversi motivi, non è possibile sapere in anticipo come verrà esattamente visualizzato un testo in un browser. I principali motivi di questa apparente anomalia derivano dalla differente grandezza degli schermi usati dagli utenti, dalla diversa dimensione che l'utente può impostare per la dimensione dei caratteri e dal fatto che, di norma, l'utente può ridimensionare liberamente la pagina Web che sta visualizzando trascinando con il mouse i bordi della finestra.

Per quanto riguarda i browser, che come sappiamo rappresentano l'interfaccia tramite la quale gli utenti accedono al Web, possono essere suddivisi in due principali categorie: browser grafici e browser testuali. I browser grafici sono in grado, in particolare, di visualizzare immagini e di riprodurre suoni e rappresentano quindi strumenti di tipo multimediale. I browser testuali, invece, non riproducono immagini e suoni, e per questo motivo sono generalmente più veloci di quelli grafici. In proposito si rammenta che è opportuno che tutti i documenti HTML siano verificati dagli autori o dal team di collaudo sia in modalità grafica sia in modalità testo. Oltre alle due categorie menzionate si stanno diffondendo recentemente anche browser per non vedenti, dotati della capacità di interpretare le istruzioni HTML e di convertirle in impulsi vocali e/o in codice Braille.

Scrittura del codice HTML

Il codice HTML può essere scritto usando semplici programmi di editazione di testo come il blocco note di Windows o Wordpad. Esistono, comunque, strumenti di creazione di codice HTML estremamente sofisticati, come Dreamweaver di Macromedia e FrontPage di Microsoft, in grado, tra l'altro, di generare automaticamente il testo HTML partendo dal disegno a video della pagina che l'autore ha eseguito. In questo modo il processo di creazione delle pagine risulta drasticamente abbreviato. Tuttavia nel momento in cui si inizia ad apprendere il linguaggio HTML l'uso di tali sofisticati strumenti, per diversi motivi, non è consigliabile. Tra questi motivi rientra la semplicità degli esempi didattici normalmente usati all'inizio del percorso di apprendimento e che non richiede il ricorso a tali strumenti. Inoltre è opportuno focalizzare la propria attenzione sulle caratteristiche e sul funzionamento dei diversi comandi previsti dal linguaggio, anche apportando deliberatamente variazioni al codice per verificarne l'effetto all'atto della visualizzazione. Inoltre gli strumenti automatici aggiungono solitamente righe e parti di codice supplementari rispetto a quanto strettamente necessario per il corretto funzionamento degli esempi, rendendoli, come si dice, più "pesanti" e aumentando la difficoltà di lettura del codice da parte dell'autore. Cercare di apprendere i principi della scrittura dei documenti HTML partendo direttamente dagli strumenti di generazione automatica può essere dunque paragonabile all'utilizzo di traduttori automatici per una lingua straniera: non conoscendo ancora la nuova lingua non si è in grado di comprendere la versione tradotta e, di conseguenza, non è possibile riconoscere e correggere gli eventuali errori di traduzione commessi dallo strumento automatico. Infine, oltre ai possibili errori che gli strumenti automatici potrebbero generare, potrebbero rendersi maggiormente necessarie modifiche e personalizzazioni alle pagine costruite automaticamente, soprattutto se si opera con siti dinamici e non con "semplici"

pagine statiche. Tra i semplici editor di testo, come il blocco note di Windows, e gli strumenti automatici per la generazione e la gestione di siti web, come Dreamweaver di Macromedia e Frontpage di Microsoft, si collocano altri strumenti in grado di leggere il codice e di verificarne la sintassi. Un esempio è dato dal programma di utilità Tidy, disponibile gratuitamente sul sito del W3C. Questo programma, in particolare, individua e inserisce eventuali tag finali mancanti correggendo anche l'eventuale errata nidificazione dei tag di apertura e di chiusura, aggiunge gli apici mancanti attorno ai valori degli attributi, esegue l'indentazione del codice, ecc.⁷ Per quanto riguarda in particolare l'indentazione si ricorda nuovamente che tutto il codice funzionerebbe correttamente anche se scritto senza interporre spazi o righe in bianco, ma che in questo modo la lettura e le operazioni di correzione risulterebbero inutilmente complicate.

Un'altra categoria di strumenti per la scrittura automatica di codice HTML è data dai cosiddetti *convertitori*. Si tratta di funzionalità software, incluse nei programmi di videoscrittura, che consentono il salvataggio di file di testo in formato HTML, in genere mediante l'opzione "Salva come HTML". Anche i convertitori, però, aggiungono solitamente tag non necessari e generano un codice molto più complesso rispetto a quello minimo necessario per ottenere l'effetto desiderato.

Un eccellente modo di iniziare a scrivere codice HTML consiste nell'esaminare ciò che è già stato scritto. HTML consente l'esame del codice di qualsiasi pagina visualizzata nel browser e questo costituisce una fonte preziosa per l'apprendimento, che si aggiunge alle risorse presenti su Internet e destinate specificatamente alla didattica. Infine, per quanto concerne gli errori più comuni che si possono commettere nella scrittura del codice HTML, si può provare a citarne qualcuno. Possono essere, a titolo esemplificativo, il dimenticare di includere il tag finale di qualche elemento, il non rispettare l'ordine di chiusura dei tag, il non racchiudere i valori degli attributi tra virgolette. In genere tali errori sono commessi da autori alle prime armi, e la loro frequenza diminuisce drasticamente all'aumentare del numero di documenti creati.

W3C

Il World Wide Web Consortium (W3C) è un consorzio internazionale, fondato nel 1994, composto da esperti ed aziende del settore informatico, articolato in gruppi di lavoro, che ha, tra l'altro, la finalità di definire gli standard HTML.

I membri del W3C sviluppano le specifiche HTML che, al momento della loro presentazione, vengono definite bozze (*working draft*). Se tali specifiche vengono accettate entrano a far parte delle raccomandazioni proposte (*proposed recommendations*). Successivamente, una volta raggiunta l'intesa da parte dei membri su una raccomandazione proposta, questa, essendo promossa a standard, diviene una raccomandazione (*recommendation*). A questo punto i produttori di browser dovrebbero cominciare ad inserire i nuovi standard nelle nuove versioni dei browser. Il lavoro del W3C continua con il periodico aggiornamento degli standard per far fronte a possibili errori che possono essere stati commessi.

Uno degli aspetti dei lavori del W3C per l'aggiornamento delle specifiche HTML è costituito dall'eliminazione di vecchi tag. Ciò avviene in due diversi momenti: con il primo i vecchi tag sono definiti deprecati (*deprecated*), con il secondo sono poi classificati come *obsoleti*. La differenza tra le due tipologie è data dal fatto che i tag deprecati continuano ad essere riconosciuti dai browser mentre i tag obsoleti non sono più supportati dalle nuove versioni dei browser. Naturalmente se l'utente non cambia la versione del browser che sta già utilizzando l'eventuale attribuzione ai tag delle definizioni di deprecato e obsoleto non avrà alcun effetto sulla visualizzazione dei documenti HTML.

⁷ Le funzionalità e le modalità di utilizzo del programma sono disponibili nell'apposita sezione del sito www.w3c.org.

Tag obsoleti e deprecati

Per quanto concerne i tag previsti dalla specifica HTML 4.01 occorre ricordare la presenza di elenchi di tag *obsoleti* e *deprecati*. Con il termine obsoleti si indicano tag non più supportati dalle più recenti versioni dei browser, che di conseguenza potrebbero essere interpretati non correttamente. Con il termine deprecati si indicano tag non più aggiornati e sostituiti da altri tag per ottenere effetti simili. Tuttavia i tag deprecati dovrebbero essere interpretati correttamente dai browser. Sia i tag obsoleti sia i tag deprecati non dovrebbero essere usati anche se possono essere ancora presenti in pagine web scritte per utenti che utilizzano versioni non aggiornate dei browser.

In merito ai tag deprecati va detto che lo scopo originario di HTML non era quello di contenere informazioni destinate alla formattazione dei documenti, ma di contenere solo informazioni sulla struttura dei documenti. Il tag `<p>`, ad esempio, si riferisce ad un paragrafo e pertanto alla struttura, e non dovrebbe includere informazioni sul come questo paragrafo deve essere visualizzato. Con il passare del tempo il progressivo proliferare dei tag di formattazione ha contribuito a rendere più complesso il codice HTML e la manutenzione delle pagine. La specifica HTML 4.0 separa il modo con cui il documento è presentato dalla sua struttura. A titolo di esempio un altro tag deprecato è `<center>` che deve essere sostituito da `<div align="center">`. Come già accennato, l'uso di tag e/o attributi deprecati non dovrebbe comportare errori nella visualizzazione delle pagine, ma rende di fatto tali documenti non conformi allo standard corrente.

Elementi e tag

Gli elementi e i tag costituiscono concetti fondamentali di HTML. Il codice HTML si basa su coppie di *tag* di apertura e di chiusura che definiscono ogni elemento di una pagina Web, come ad esempio un paragrafo di testo, una tabella o un'immagine. Il linguaggio HTML contrassegna quindi i componenti di una pagina Web con un insieme di *elementi* predefiniti. Gli elementi sono costituiti da intestazioni, paragrafi, elenchi, tabelle, immagini e collegamenti. Ogni elemento è costituito da un *tag* iniziale formato da una parte di testo preceduto da una parentesi angolare sinistra (`<`) e seguita da una parentesi angolare destra (`>`). Il tag finale (in qualche caso non è obbligatorio) è simile al tag iniziale con la differenza che include una barra rovesciata (`/`). Il testo compreso tra il tag iniziale e quello finale è il *contenuto*. L'elemento può pertanto essere definito come la sequenza di tag di apertura, contenuto e tag di chiusura. La figura 2 riassume i componenti sinora esaminati.

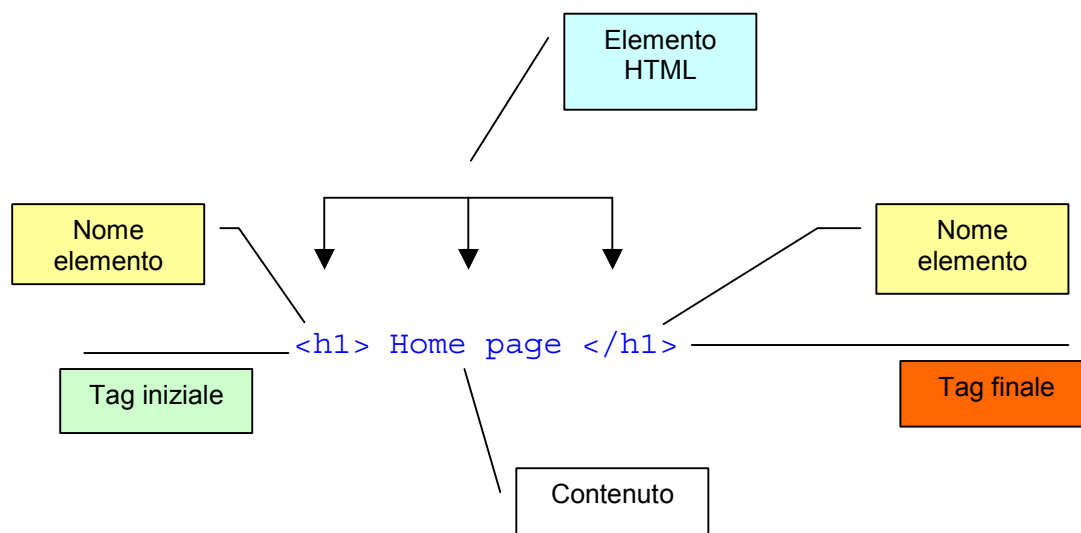
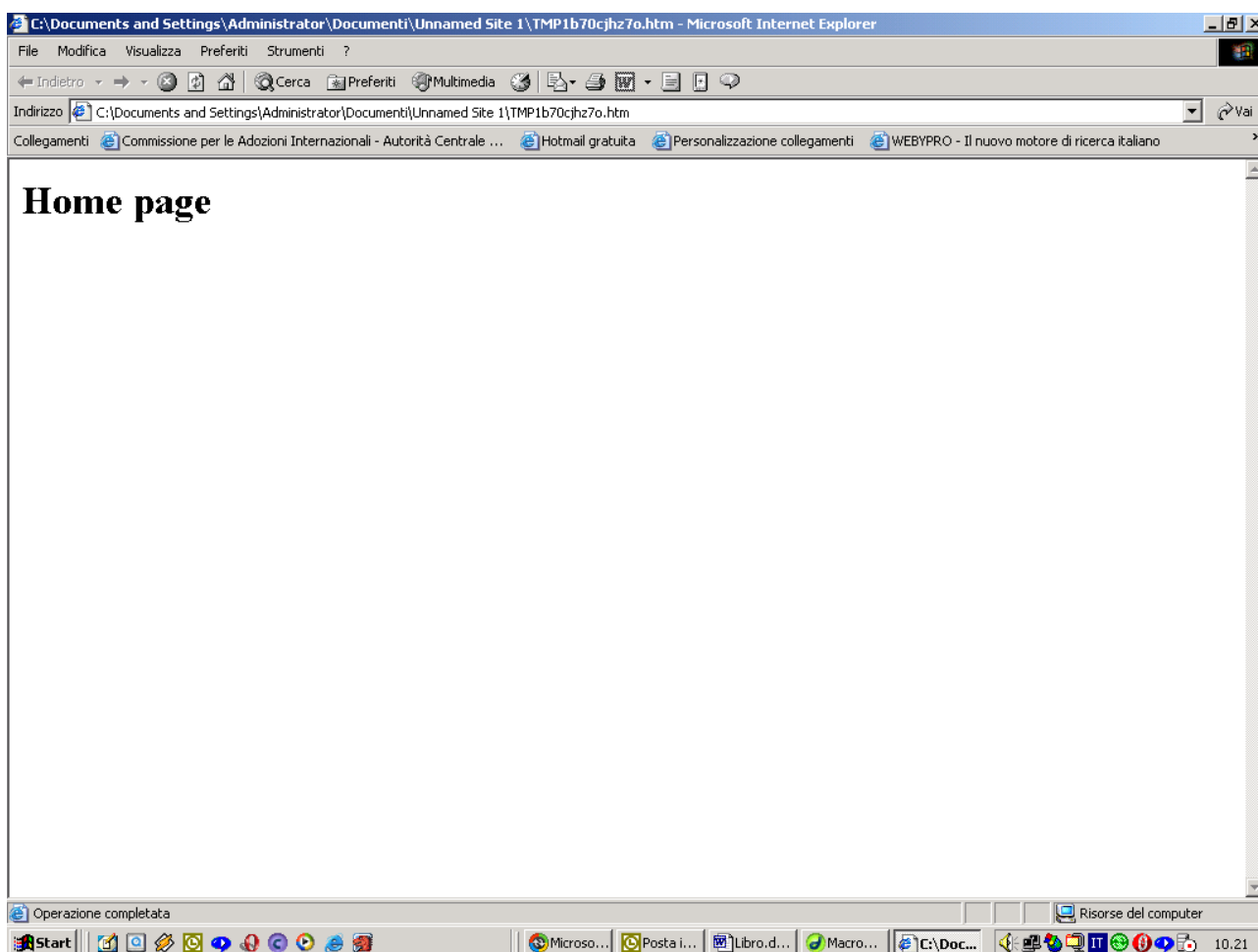


Fig. 4: Elementi e tag

Nella figura 4 è rappresentata una semplice riga di codice HTML. La pagina visualizzata tramite il browser presenterà la scritta “Home page” (contenuto dell’elemento) con la formattazione prevista per il tag *h1* (elemento). Quando il browser legge questa riga di istruzione il tag iniziale “attiva” l’elemento al quale si riferisce (nell’esempio: *h1*, ovvero intestazione di primo livello) e il tag finale “disattiva” l’elemento. In altri termini il contenuto dell’elemento, costituito dal testo compreso tra il tag iniziale e il tag finale, è reso sulla pagina Web con la modalità che il browser prevede per il tag stesso. Nella figura 5 è mostrato l’effetto che questa semplice istruzione produce quando è visualizzata attraverso un browser.

**Fig. 5: Un'intestazione di tipo h1**

Si noti che non tutti gli elementi prevedono un tag di chiusura e che, in tal caso, sono definiti elementi vuoti.⁸ In proposito è bene precisare che il linguaggio HTML prevede due categorie di tag: tag *vuoti* e tag *contenitori*. I tag vuoti hanno lo scopo di formattare la pagina, non hanno testo al proprio interno e, come già affermato, non prevedono il tag di chiusura, mentre possono essere corredati da attributi.⁹ Un esempio di tag vuoto è dato dal tag `<img>` (immagine) che, richiamando

⁸ Quando è presente la coppia di tag di apertura e chiusura i tag stessi sono anche definiti *bilaterali*, mentre quando è presente il solo tag di apertura il tag è anche detto *monolaterale*.

⁹ Gli attributi sono discussi successivamente.

un file di tipo image, fa apparire l'immagine stessa nella pagina Web, nella posizione e con le modalità indicate dal tag. I tag contenitori sono invece costituiti da un tag iniziale, con la presenza di eventuali attributi, da un testo e da un tag finale che ha lo scopo di segnalare il punto in cui deve terminare l'effetto previsto dal tag. Un esempio di tag finale è dato da `</html>` che rappresenta il tag con il quale deve terminare ogni documento HTML.

Si osservi che la combinazione di più tag deve essere scritta annidandoli e non sovrapponendoli ovvero “chiudendo” prima quelli più interni e via via quelli più esterni.

```
<h1><body></h1></body>
```



Fig. 6: Esempio di tag sovrapposti (errato)

```
<h1><body></body></h1>
```



Fig. 7: Esempio di tag annidati (corretto)

Dalle figure 6 e 7 si vede che i tag devono essere chiusi in ordine inverso a quello con il quale sono stati aperti.

Sebbene i browser siano generalmente “tolleranti” nei confronti di errori nella codifica del linguaggio HTML e che pertanto siano in grado di rappresentare correttamente anche pagine contenenti combinazioni di tag non rispondenti alla regola appena indicata è buona norma scrivere le aperture e le chiusure dei tag in base alla regola indicata. Infatti i futuri sviluppi delle specifiche del linguaggio prevedono che la codifica avvenga secondo questa regola.

Nella tabella successiva si riportano alcuni elementi e la parte di pagina che ognuno di essi contrassegna.

<i>Elemento HTML</i>	<i>Parte di pagina contrassegnata</i>
html	intera pagina
head	informazioni relative all'intestazione
title	titolo (appare nella finestra del browser)
body	corpo principale del testo
h1	intestazione di primo livello
h2	intestazione di secondo livello
h3	intestazione di terzo livello
p	paragrafo
img	immagine

a	ancoraggio per il collegamento ad altra pagina o parte di pagina
b	blocco di testo in grassetto
i	blocco di testo in corsivo
ol	elenco ordinato (ordered list)
li	singolo elemento di un elenco
table	tabella

In seguito questi ed altri elementi verranno discussi ed usati in diversi esempi.

Il browser è in grado di formattare e visualizzare gli elementi, secondo le caratteristiche previste per ciascuno di essi.

Elementi e attributi

Abbiamo visto che gli elementi identificano parti specifiche di un testo HTML ed assumono un determinato significato. Gli elementi HTML possono essere (e in genere sono) accompagnati da attributi, che rappresentano un modo per assegnare proprietà aggiuntive ed aggiungere informazioni supplementari all'elemento cui si riferiscono. In termini non rigorosi si può affermare che un tag dice al browser di *fare* qualcosa, un attributo del tag dice al browser *come* farlo. Ad esempio il tag `<body>` può essere accompagnato da un attributo *bgcolor* (colore di background – sfondo) che comunica al browser il colore di sfondo con il quale la pagina deve essere visualizzata. Gli attributi si presentano come una coppia costituita dal nome dell'attributo e dal suo valore, uniti da un segno di uguale, e vengono aggiunti al tag iniziale di un elemento HTML. E' possibile, per uno stesso tag, definire più di un attributo ed in questo caso gli attributi sono separati l'uno dall'altro con uno spazio. L'ordine con il quale si specificano i diversi attributi non è rilevante. Ci si potrebbe chiedere, a questo punto, quali sono gli attributi che è possibile specificare per ciascun elemento. La risposta è che ogni elemento possiede una lista di attributi ammissibili e di relativi valori, definiti nelle specifiche HTML. I valori degli attributi possono essere racchiusi indifferentemente tra apici o tra doppi apici. Se, ad esempio, il valore dell'attributo contiene apici al suo interno per delimitare tale valore devono essere usati i doppi apici, e viceversa.

I nomi degli attributi sono *case-insensitive* ma spesso i loro valori sono invece *case-sensitive*.

Nella figura 8 è illustrato un esempio di sintassi per l'utilizzo di un attributo.

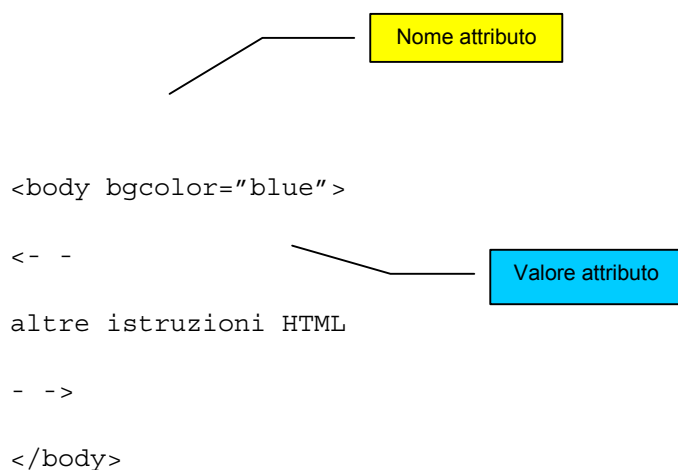


Fig. 8: La valorizzazione dell'attributo bgcolor

Nell'esempio appena mostrato è stato aggiunto al tag iniziale `body` l'attributo *bgcolor* seguito dal valore *blue* che è uno dei possibili valori previsti per l'attributo stesso. In questo modo il colore di sfondo della sezione `body` della pagina visualizzata avrà il colore blu, che è stato indicato quale valore dell'attributo.

Default dei caratteri

La grandezza di default dei caratteri è pari a 3. Ogni browser ha un set di caratteri di default, colore e grandezza. I font indicati nelle pagine HTML non sono trasferiti dal server verso il computer dell'utente (tranne, in casi particolari, nel caso in cui si faccia uso dei fogli di stile) e pertanto sono visualizzati soltanto se già installati sul computer dell'utente.¹⁰

Se i font specificati dall'autore di una pagina non sono installati sul computer verranno usati quelli previsti dal browser come caratteri di default. Solitamente nelle pagine HTML si specificano più tipi di font, come illustrato nell'esempio successivo.

Esempio

```
<font face="Arial, Helvetica, Lucida Sans">Stringa di esempio</font>
```

¹⁰ Nei sistemi Windows i caratteri sono, di norma, installati nella cartella `c:\windows\fonts`.

In questo caso il browser, per rappresentare la stringa di caratteri “Stringa di esempio”, cercherà di usare i caratteri *nell’ordine* con cui sono indicati nel tag font. Il browser tenta quindi, in un primo momento, di visualizzare la stringa con il carattere Arial. Se questo carattere non è disponibile sul computer dell’utente il browser tenta di usare il carattere Helvetica, e così via. Per evitare la sostituzione dei caratteri indicati nella pagina con quelli di default, è preferibile che gli autori specifichino caratteri sufficientemente diffusi. Una possibile alternativa, utile nel caso si voglia necessariamente visualizzare caratteri particolari, è quella di non usare il tag font o i fogli di stile ma di utilizzare immagini appositamente costruite. Si tenga presente che il tag è deprecato nelle più recenti versioni di HTML. Le proprietà di visualizzazione e di formattazione dovrebbero pertanto essere definite attraverso i fogli di stile (CSS), che saranno esaminati nei capitoli successivi.

Struttura di un documento HTML

Tutti i documenti HTML hanno la stessa struttura di base, come riportato di seguito:

- intestazione: contiene informazioni di controllo usate dal browser e dal server;
- body;
- contenuto;
- tag che controllano la formattazione del contenuto.

Esempio ¹¹

```
<html>
  <head>
    <title>Primo documento HTML </title>
  </head>
  <body>
    <h1> Esempio intestazione di primo livello</h1>
    <h2> Esempio intestazione di secondo livello</h2>
    <p>Esempio di un primo paragrafo</p>
    <p>Esempio di un secondo paragrafo</p>
  </body>
</html>
```

Visualizzando il codice attraverso un browser si ottiene il risultato presentato nella figura 9.

¹¹ Il codice HTML degli esempi proposti può essere scritto in un semplice elaboratore di testi, come ad es. il blocco note di Windows, e visualizzato successivamente mediante un browser. E’ fortemente consigliato apportare modifiche al codice esemplificativo per verificarne poi l’effetto, sempre tramite il browser. Si osservi che le variazioni apportate al codice sorgente saranno visualizzate dal browser se il file è caricato nuovamente, ad es., utilizzando il tasto di aggiornamento/reload della pagina presente sul browser stesso.

Le immagini riportate nel testo sono ottenute con i browser Internet Explorer, Netscape, Opera e Mozilla

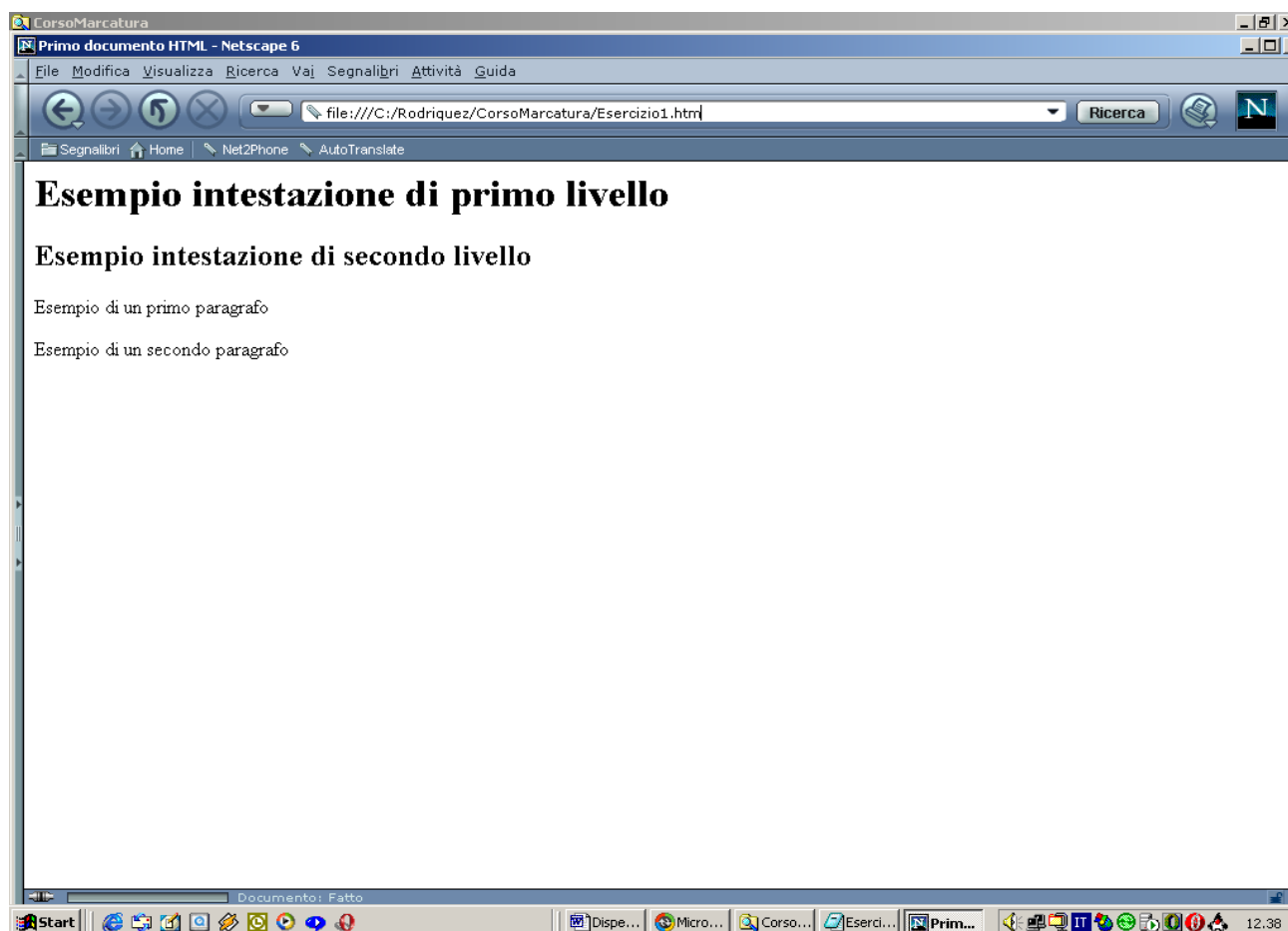


Fig. 9: Esempi di intestazioni e paragrafi

Esaminando in dettaglio il codice esemplificativo appena illustrato osserviamo che:

- i tag iniziale `<html>` e finale `</html>` comunicano al browser, rispettivamente, l'inizio e la fine del documento HTML;
- la porzione di testo compresa tra il tag iniziale `<head>` ed il tag finale `</head>` costituisce la sezione di intestazione, e non viene visualizzata nella finestra del browser;
- il contenuto del tag `title` appare sulla riga del titolo della finestra del browser;
- la porzione di testo compresa tra il tag iniziale `<body>` ed il tag finale `</body>` costituisce la sezione che viene visualizzata nella finestra del browser;
- il contenuto dei tag `h1` e `h2`, che rappresentano righe di intestazione appare, rispettivamente, con enfasi maggiore e minore;
- il contenuto dei tag `p` appare uguale in entrambe le righe e rappresenta i paragrafi nei quali è articolata la pagina.

I file HTML possono essere salvati sui supporti di memorizzazione (dischi) sia con estensione ¹² `.htm` sia con estensione `.html`.

L'estensione `.htm` è più diffusa e probabilmente tale maggiore diffusione è dovuta al fatto che i sistemi operativi del recente passato consentivano di denominare i file attribuendo all'estensione del nome la lunghezza massima di tre caratteri.¹³

¹² L'estensione di un file è la parte del nome del file che segue il punto.

¹³ In tal caso, di norma, il nome del file non può superare la lunghezza di 8 caratteri per cui la struttura dei nomi è detta appartenente al tipo "8.3".

La sezione head

La sezione `head` di una pagina HTML, detta intestazione del documento, è sempre compresa tra il tag `<html>` e il tag `<body>`. A causa della posizione “obbligata” nella quale questo tag deve trovarsi molti browser sono in grado di stabilire automaticamente la sua posizione e per questo motivo, spesso, sono presenti sul Web documenti senza l’esplicita presenza di questo tag. In base alle specifiche HTML nella sezione `head` possono essere inclusi solo determinati tag, come `TITLE` e `META`. Tutto ciò che fa parte di questa sezione¹⁴ non è visualizzato dal browser, ad eccezione di quanto previsto dal tag `<title>`. In realtà, se nella sezione `head` vengono introdotti tag che generano una visualizzazione, come ad es. `h1` oppure `p`, molti browser visualizzano comunque tali elementi. In ogni caso le informazioni previste in questa sezione rivestono notevole importanza perché sono utilizzate anche dai motori di ricerca per indicizzare le pagine pubblicate e trovare i documenti disponibili sul Web quando gli utenti eseguono le ricerche.

Commenti

Come negli altri linguaggi anche per l’HTML è prevista la possibilità di inserire commenti nel codice sorgente. I commenti sono informazioni inserite a fini di documentazione e quindi non incidono sul funzionamento delle pagine. L’autore di pagine HTML dovrebbe sempre inserire nelle pagine commenti per documentare opportunamente il codice che scrive. I commenti vengono ignorati dal browser e sono visibili solo accedendo al codice sorgente della pagina. Proprio perché chiunque, esaminando il codice sorgente, è in grado di leggere i commenti inseriti non andrebbe scritto nulla che non possa essere pubblicamente letto. I commenti sono utili per fornire chiarimenti o informazioni aggiuntive finalizzate alla manutenzione del codice HTML e sono particolarmente importanti nel caso di pagine complesse e/o se il lavoro è svolto da più sviluppatori. In quest’ultimo caso servono a comunicare efficacemente a tutti i componenti del gruppo di sviluppo le ragioni ed i procedimenti logici che hanno portato a compiere determinate scelte in fase di codifica.

I commenti possono essere usati anche in fase di verifica e correzione degli errori (*debugging*), eliminando temporaneamente parti del codice senza necessità di cancellare effettivamente ed eventualmente riscrivere il codice eliminato.

Di seguito si riportano alcuni esempi di commenti che può essere utile inserire nelle proprie pagine.

- nome dell’applicazione;
- descrizione dello scopo dell’applicazione;
- autore della pagina;
- data di prima creazione;
- data ultimo aggiornamento;
- numero versione della pagina;
- inizio e/o fine di parti di pagina;
- fonte delle immagini originali pubblicate nella pagina.

I commenti possono essere scritti in qualunque punto del documento HTML e possono essere costituiti da una o più righe. Sono racchiusi tra i tag “`<!--`” e “`-->`”.

¹⁴ La presenza della sezione `head` è facoltativa secondo le specifiche HTML 3.2 ma torna ad essere un prerequisito nella versione delle specifiche 4.01.

Esempio:

istruzioni HTML

```
<!-- commento  
    articolato  
    su tre righe -->
```

altre istruzioni HTML

Si noti che il punto esclamativo va inserito prima della parentesi angolare di apertura ma non prima della parentesi angolare di chiusura.

Titolo del documento

Il titolo della pagina è un elemento particolarmente importante. Definisce e descrive in maniera sintetica il contenuto della pagina stessa (ciò che l'utente si aspetta di trovare) e pertanto dovrebbe essere al tempo stesso esplicativo e sufficientemente breve. Il titolo è definito mediante il tag `<title> ... </title>` che fa apparire la stringa di caratteri contenuta in tale elemento sulla riga di intestazione della finestra del browser. Il contenuto di questo tag, inoltre, dà il nome al collegamento nel momento in cui si aggiunge la pagina nei siti "preferiti" del browser. Alcuni motori di ricerca indicizzano il contenuto di questo tag. Queste due ultime caratteristiche del tag `<title>` devono essere opportunamente temperate. Se da un lato, infatti, è utile inserire numerose parole per una accurata descrizione del sito e per una conseguente idonea indicizzazione da parte dei motori di ricerca, dall'altro lato una descrizione troppo lunga impedisce un'agevole e rapida aggiunta del sito agli indirizzi preferiti dell'utente. Inoltre un titolo troppo lungo, all'atto della visualizzazione, potrebbe essere troncato. Un'altra caratteristica del titolo è che si tratta del primo elemento che appare quando il browser inizia a caricare una pagina Web. Ciò significa che, in presenza di una connessione Internet lenta, la visualizzazione del titolo della pagina, di fatto, comunica all'utente che la pagina richiesta è stata trovata e che è in corso il caricamento. Per ciascun documento HTML è previsto un solo titolo.

Elemento h

Tramite l'elemento `h`¹⁵ è possibile creare nelle pagine intestazioni di diverse dimensioni e quindi di diverso impatto visivo. Queste intestazioni, o titoli, non vanno confuse con il titolo della pagina HTML generato mediante il tag `title`. Si tratta infatti di componenti che svolgono funzioni simili a quelle svolte dai titoli di una qualsiasi pubblicazione, nella quale separano parti di testo e riassumono gli argomenti trattati nelle diverse parti, e nelle quali ad argomenti più importanti corrispondono, di norma, titoli più grandi. In HTML esistono 6 livelli di intestazione che appaiono con diversa formattazione. Il primo livello (`h1`) è quello che appare con caratteri più grandi rispetto a quelli degli altri livelli, ed è quindi contraddistinto da una maggiore enfasi. Usando, ad esempio, il tag `<h3>` comunichiamo al browser una specifica formattazione che può essere costituita da una combinazione di caratteri e colori da applicare, analogamente a quanto avviene con gli elaboratori di testo. Si osservi che nel visualizzare la pagina, HTML aggiunge una riga bianca prima e dopo ciascuna tag di intestazione.

Di seguito si riporta un esempio di sintassi con l'uso dell'attributo *align*, che regola l'allineamento del testo.

```
<h1 [align="left"|"center"|"right"]> .... </h1>
<h2 [align="left"|"center"|"right"]> .... </h2>
<h3 [align="left"|"center"|"right"]> .... </h3>
```

Per vedere l'effetto prodotto a video dai tag di intestazione si veda la figura 9.

Paragrafi

I paragrafi di una pagina sono definiti ricorrendo al tag `<p>`. I tag di paragrafo sono utili per assicurarsi che il browser visualizzi il testo con la formattazione desiderata in quanto, come sappiamo, gli spazi in bianco presenti nei testi non vengono considerati. Il tag paragrafo dispone di un tag di apertura e di un tag di chiusura, anche se il tag di chiusura in genere è omesso. E' diffusa l'abitudine di applicare spazi in bianco tra blocchi di testo ricorrendo a una successione di tag `<p>`, senza il corrispondente tag di chiusura. Questa pratica, però, non risulta particolarmente valida in quanto, a seconda dei browser usati, è anche possibile che all'atto della visualizzazione venga considerato solo il primo dei tag `p`, con la conseguenza che non verrà visualizzato tutto lo spazio desiderato. Come per i tag di intestazione (`h`) anche in questo caso nel visualizzare la pagina l'HTML aggiunge una riga bianca prima e dopo ciascun paragrafo.

Di seguito si riporta un esempio di sintassi con l'uso dell'attributo *align* (allineamento del testo).

```
<p [align="left"|"center"|"right"]> .... </p>
```

Linee orizzontali

La leggibilità di un testo può essere migliorata suddividendolo in sezioni tramite apposite linee orizzontali, tracciate sul dispositivo di rappresentazione della pagina. In HTML le linee orizzontali

¹⁵ h sta per heading (intestazione).

si ottengono utilizzando il tag `<hr>`.¹⁶ Per questo elemento è previsto il solo tag di apertura in quanto non deve contenere nulla al proprio interno. Le linee tracciate con questo tag non occupano necessariamente tutta la lunghezza della pagina che è invece definita tramite un apposito attributo previsto per il tag.

Di seguito è mostrato un esempio di sintassi con l'utilizzo di alcuni attributi.

```
<hr [align="left"|"center"|"right"] [size="n"] [noshade] [width="nn%"]>
```

L'attributo *align*, così come per altri tag, regola l'allineamento orizzontale dell'elemento. L'attributo *size* imposta l'altezza della riga in pixel (il valore di partenza è 1) mentre l'attributo *width* determina la larghezza della riga. Width può essere impostato come valore fisso in pixel (es. `size="50"`) oppure come valore percentuale (es. `size="50%"`). Nel caso del valore fisso la lunghezza della riga corrisponde al valore impostato mentre nel caso del valore espresso in percentuale l'estensione della riga all'interno della finestra del browser corrisponde al valore percentuale assegnato, indipendentemente dalle effettive dimensioni della finestra. In altri termini il valore in percentuale dell'estensione della riga viene mantenuto anche nel caso in cui la finestra del browser è modificata manualmente dall'utente. Impostando un valore percentuale pari a 100 la linea si estende per tutta la larghezza della finestra del browser. Per default la linea viene tracciata sul dispositivo di emissione con un effetto tridimensionale, particolarmente evidente quando la linea ha un notevole spessore (a partire da 5 o 6 pixel). Usando l'attributo *noshade* la riga è riprodotta senza l'effetto predefinito di ombreggiatura. Questo attributo non richiede la presenza di un segno di uguale e, per disattivare l'impostazione predefinita, è richiesta solo la sua presenza. Gli attributi di questo tipo, che attivano o disattivano le impostazioni predefinite semplicemente con la loro presenza, sono definiti commutatori (*toggle*). Si tenga presente, infine, che gli attributi possono essere elencati in qualsiasi ordine.

Esempio

```
<html>
  <head>
    <title>Esempi di righe orizzontali</title>
  </head>
  <body>
    <hr>
    <hr align="center" size="5" width="50">
    <hr align="center" size="5" width="50%">
    <hr align="center" size="10" width="50%">
    <hr align="center" size="10" width="50%" noshade>
  </body>
</html>
```

Il codice produce a video il risultato illustrato nella figura 10.

¹⁶ hr sta per horizontal rule (righello).

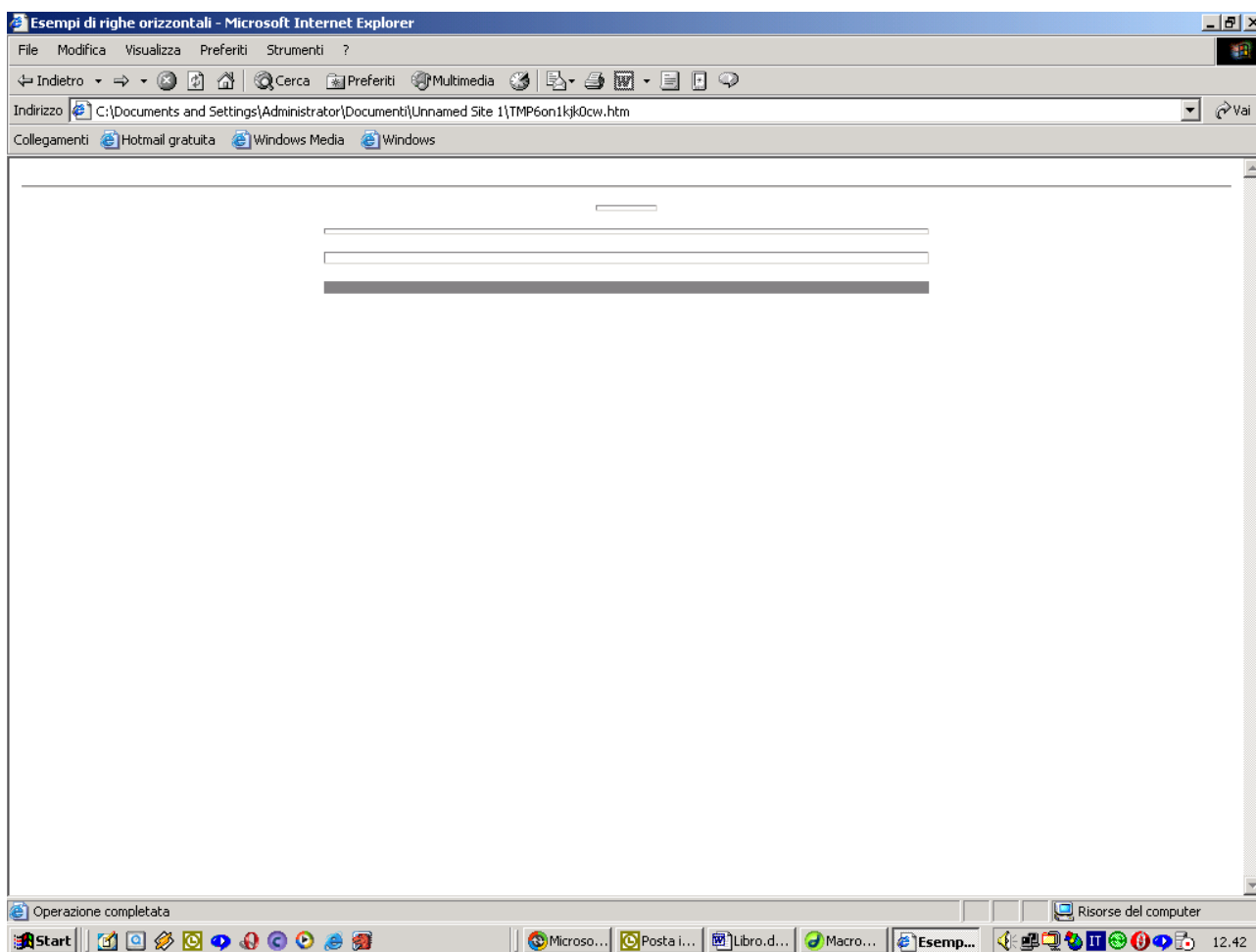


Fig. 10: Esempi di linee orizzontali ottenute con il tag hr

Linee di interruzione

Talvolta si desidera iniziare una nuova riga di testo senza creare un nuovo paragrafo con il contestuale inserimento di una riga in bianco. In HTML per far cominciare il testo su una nuova riga si usa il tag `<br />`. La digitazione del tasto “Invio” nel codice sorgente non produce alcun effetto in quanto i punti in cui le righe si interrompono sono determinati dalla larghezza della finestra del browser, che di norma l’utente può ridimensionare a suo piacimento.

Con il tag `br` il browser viene forzato a visualizzare una linea di interruzione. Questo tag non prevede un tag di chiusura (si tratta quindi di un elemento vuoto). E’ inoltre possibile utilizzare successioni di tag `br` che in diversi browser vengono resi con blocchi di spazio in bianco più o meno estesi a seconda del numero di tag `br` inseriti.

Esempio

```
<html>
  <head>
    <title>Tag br</title>
  </head>
  <body>
    <p>Linea di testo nella quale qui
      è stato premuto il tasto Invio durante la digitazione</p>

    <p>Linea di testo nella quale qui<br />è stato inserito il tag
    br</p>

    <p>Si può osservare che la prima linea di testo non si interrompe
    mentre la seconda sì</p>
  </body>
</html>
```

Il codice produce a video il risultato illustrato nella figura 11.

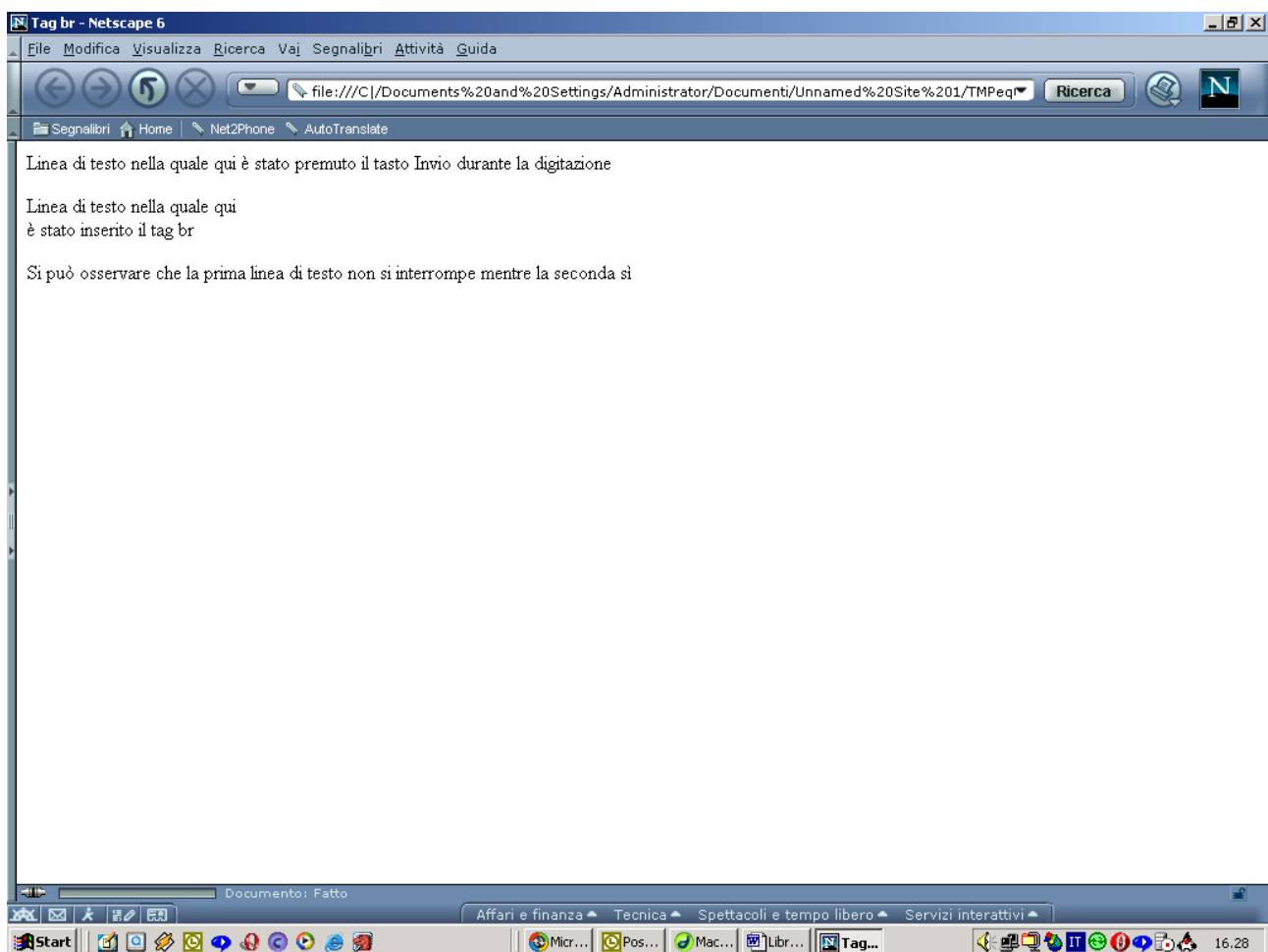


Fig. 11: Interruzioni di linea ottenute con il tag br

Un attributo del tag `br` che è opportuno esaminare più in dettaglio è *clear*, usato per gestire la disposizione del testo attorno alle immagini. In questo caso l'interruzione di linea che si desidera introdurre può comportare la prosecuzione del testo su una nuova riga sia a fianco dell'immagine sia al di sotto della stessa. Con l'attributo *clear* si determina tale comportamento.

L'esempio riportato di seguito dovrebbe chiarire meglio il contesto in cui l'attributo agisce

```
<html>
  <head>
    <title>Attributo clear del tag br</title>
  </head>
  <body>
    
    <p>Questo testo è adiacente all'immagine presente nella pagina e
    tutte le parole in esso presenti continuano ad essere adiacenti,
    tranne nel caso di ridimensionamento della finestra che potrebbe
    far spostare le parole al di sotto dell'immagine</p>

    <p>In questo testo è usato l'attributo clear a partire da qui <br
    clear="left" /> ed il testo continua al di sotto dell'immagine a
    partire dal margine sinistro della prima riga disponibile</p>
  </body>
</html>
```

Il codice produce a video il risultato illustrato nella figura 12.

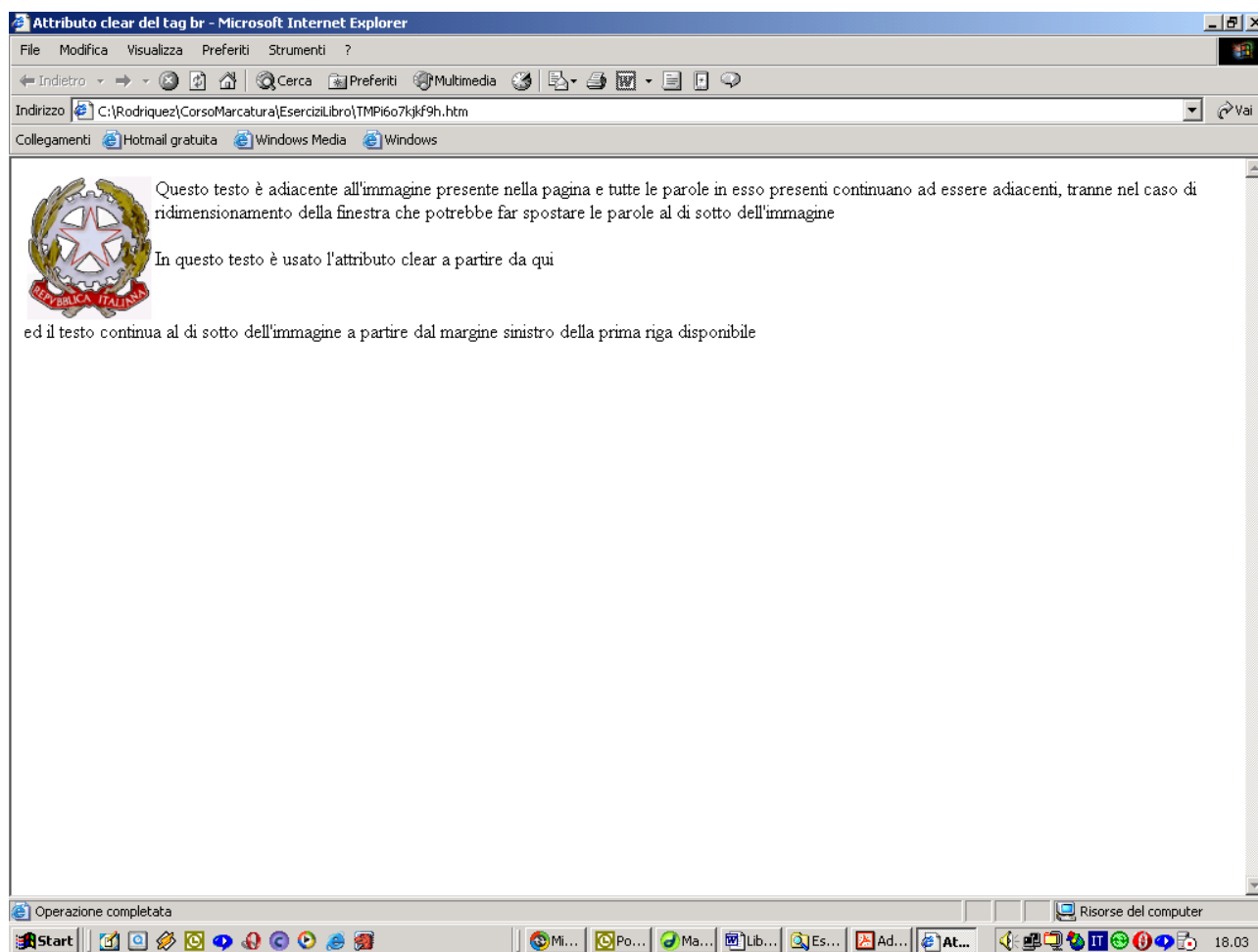


Fig. 12: Effetto dell'attributo clear in presenza di un'immagine

I valori ammissibili dell'attributo clear sono *left*, *right*, *all* e *none*. L'effetto del valore *left* è stato appena illustrato. *Right* fa apparire il testo sul margine destro della prima riga disponibile mentre *all* richiede che sia il margine sinistro sia il margine destro siano disponibili per far continuare il testo.

Infine con *none*, che è anche il valore di default, il testo continua a scorrere nella riga successiva.

Si propone ora il codice seguente che riepiloga alcuni dei tag sinora esaminati

```

<html>
  <head>
    <title>Università ...</title>
  </head>

  <body>
    <h1>Università ...</h1>
    <h2> Informazioni...</h2>
    <p>Questo sito fornisce informazioni sui servizi offerti a docenti,
    studenti e terze parti interessate.</p>
    <hr>
    <h3>Corsi</h3>
    <p align="center"> Elenco corsi</p>
    <hr width="50%">
  </body>
</html>

```

Il codice genera il risultato illustrato nella figura 13.



Fig. 13: Sezioni head e body con altri tag

Nell'esempio si noti, in particolare, il tag *hr* usato sia senza alcun valore sia con un valore espresso in percentuale. Nel primo caso la riga occupa sempre l'intera larghezza della finestra del browser mentre nel secondo caso la riga occupa sempre il 50% della finestra, anche nel caso in cui questa venga ridimensionata manualmente dall'utente.

Formattazione dei caratteri

La formattazione dei caratteri, unitamente ad altre caratteristiche, assume notevole importanza nel determinare il livello di efficacia ottenibile attraverso la comunicazione della parola scritta. Un semplice esempio è offerto dalla decisione in merito alle dimensioni da attribuire ai caratteri poiché a minori dimensioni corrisponde, a parità di spazio disponibile sulla pagina, un maggior numero di informazioni rappresentabili, al quale si accompagna, di contro, una possibile maggiore difficoltà di lettura. HTML è corredato di opportuni tag volti a definire e modificare nei modi più diversi le modalità di visualizzazione del testo. Ad esempio, oltre alla possibilità di scegliere il tipo font da utilizzare per la visualizzazione del testo, l'autore della pagina HTML può impostarne le dimensioni ed il colore così come può far apparire determinate stringhe o parti di testo in grassetto, corsivo, sottolineato, ecc. Questi tag possono essere usati in combinazione fra loro, per cui una stringa può ad esempio essere visualizzata contestualmente in corsivo ed in grassetto. Deve comunque essere tenuto presente che parte di questi tag è stata deprecata e che la specifica HTML suggerisce di usare al loro posto i fogli di stile, trattati successivamente, per ottenere gli stessi effetti.

Il tag per applicare un determinato tipo di carattere alla pagina è *font*. Prevede diversi attributi tra i quali qui si menzionano:

- *Size*, per indicare la dimensione dei caratteri; la sintassi è `size="N"` in cui N può variare da 1 (minima dimensione) a 7 (massima dimensione). Si osservi che nel caso siano indicati valori minori o maggiori, rispettivamente, di quelli minimi e massimi consentiti l'effettiva dimensione sarà ricondotta al ed espressa con il valore minimo o massimo consentito. Il valore di default dell'attributo `size` è 3. È anche possibile usare espressioni del tipo `size="+2"` o `size="-2"` che genera la rappresentazione dei caratteri, rispettivamente, aumentandola o diminuendola di due corpi rispetto alla precedente dimensione.
- *Face*, per specificare il nome del carattere o del gruppo di caratteri. In genere si elencano più nomi di caratteri ed il browser userà il primo dell'elenco, se installato sul computer dell'utente, passando poi a usare gli altri nello stesso ordine con il quale i caratteri sono elencati, nel caso in cui i caratteri non siano installati. Il processo continua sino a quando non si individua un carattere compreso nell'elenco e presente sul computer. In caso negativo il browser utilizzerà il proprio carattere di default.¹⁷ Per questo motivo risulta preferibile utilizzare nelle proprie pagine font abbastanza comuni, disponibili sui computer degli utenti, in modo che le pagine siano visualizzate con le caratteristiche di formattazione previste dall'autore. La sintassi è `face="NomeCarattereA, NomeCarattereB, NomeCarattereC"`
- *Color*, per indicare il colore con il quale il carattere deve essere rappresentato. Il colore può essere indicato attraverso il suo nome o tramite il suo corrispondente valore esadecimale.

Esempio

```
<html>
  <head>
    <title>Elemento font</title>
  </head>
  <body>
```

¹⁷ Per verificare l'elenco dei font installati sul proprio computer, in ambiente Windows, è possibile selezionare l'icona "Tipi di carattere" accessibile con la sequenza Avvio/Start, Impostazioni, Pannello di controllo.

```

<p>
<font size="6" face="Verdana, Arial, Helvetica, sans-serif"
color="blue">L</font>a prima lettera di questo paragrafo è più
grande degli altri caratteri ed appare in colore blu.
</p>
</body>
</html>

```

Il codice produce il risultato illustrato nella figura 14.

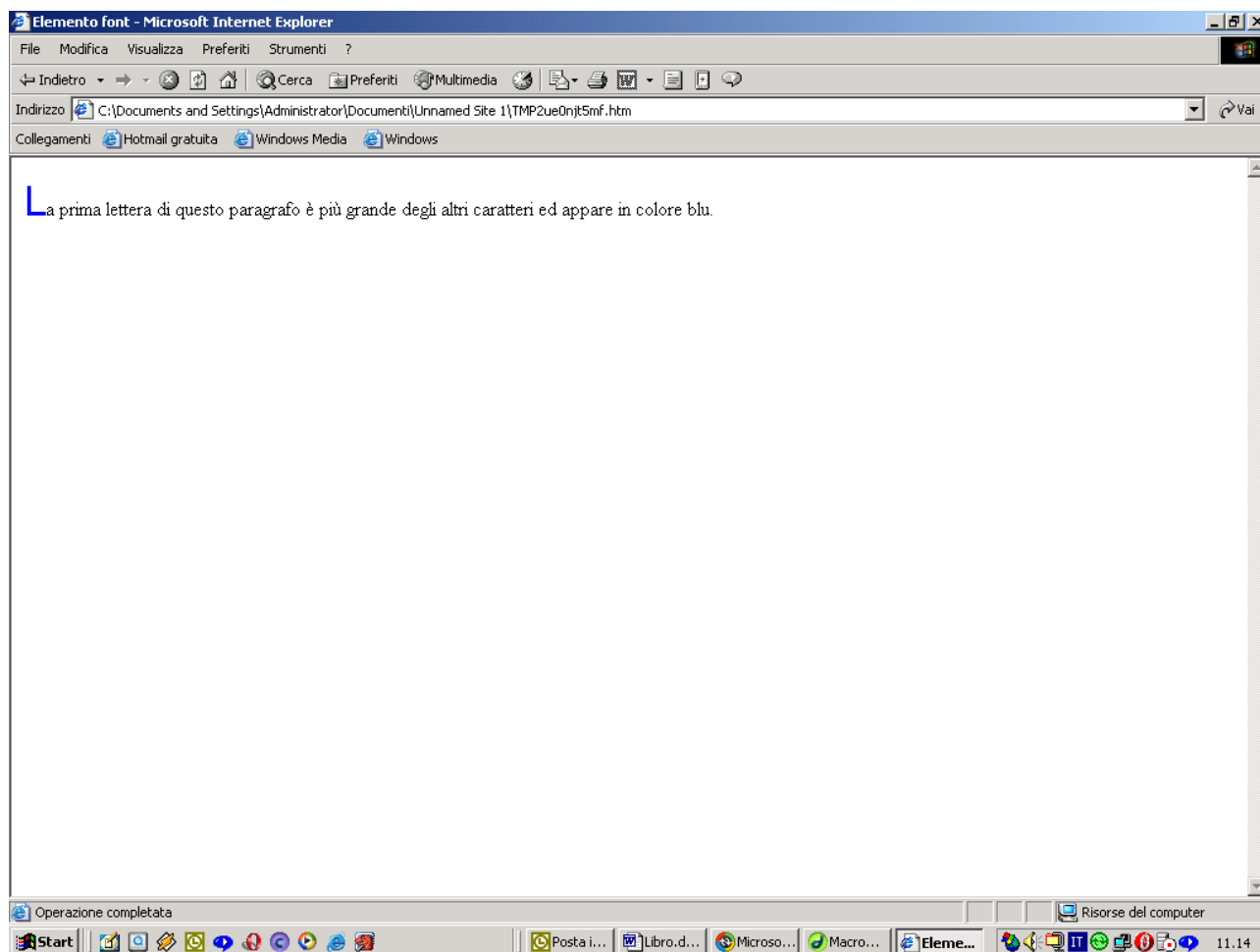


Fig. 14: Esempio di utilizzo del tag font

Anziché lavorare con singole porzioni di testo è possibile attribuire ad un intero documento HTML una dimensione predefinita dei caratteri tramite l'elemento *basefont*, i cui principali attributi sono gli stessi illustrati per il tag *font*. Il tag *basefont* non dispone di un tag finale poiché le informazioni occorrenti per le funzioni che deve svolgere sono già tutte comprese all'interno del tag iniziale. E' possibile utilizzare utilmente, nello stesso documento, sia il tag *font* sia il tag *basefont*, impostando tramite *font* le dimensioni dei caratteri in maniera *relativa* rispetto a quanto specificato da *basefont*. Il seguente codice esemplificativo dovrebbe risultare utile per chiarire il meccanismo.

```

<html>
  <head>
    <title>Elementi basefont e font</title>
  </head>

```

```

<body>
  <basefont size="4" face="Arial, Helvetica, sans-serif">
  <p>Testo basato sul tag basefont con attributo size="4"
</p>
  <p>
    <font size="-1"> Testo ridotto di una misura rispetto a basefont
(3)</font>
  </p>
  <p>
    <font size="+1"> Testo aumentato di una misura rispetto a basefont
(5)</font>
  </p>
  <p>
    <font size="+2"> Testo aumentato di due misure rispetto a basefont
(6)</font>
  </p>
  <p>
    <font size="+3"> Testo aumentato di tre misure rispetto a basefont
(7)</font>
  </p>
  <p>
    <font size="+5"> Testo aumentato di cinque misure rispetto a basefont (9)
e ricondotto al massimo (7)</font>
  </p>
</body>
</html>

```

Il codice produce il risultato illustrato nella figura 15.

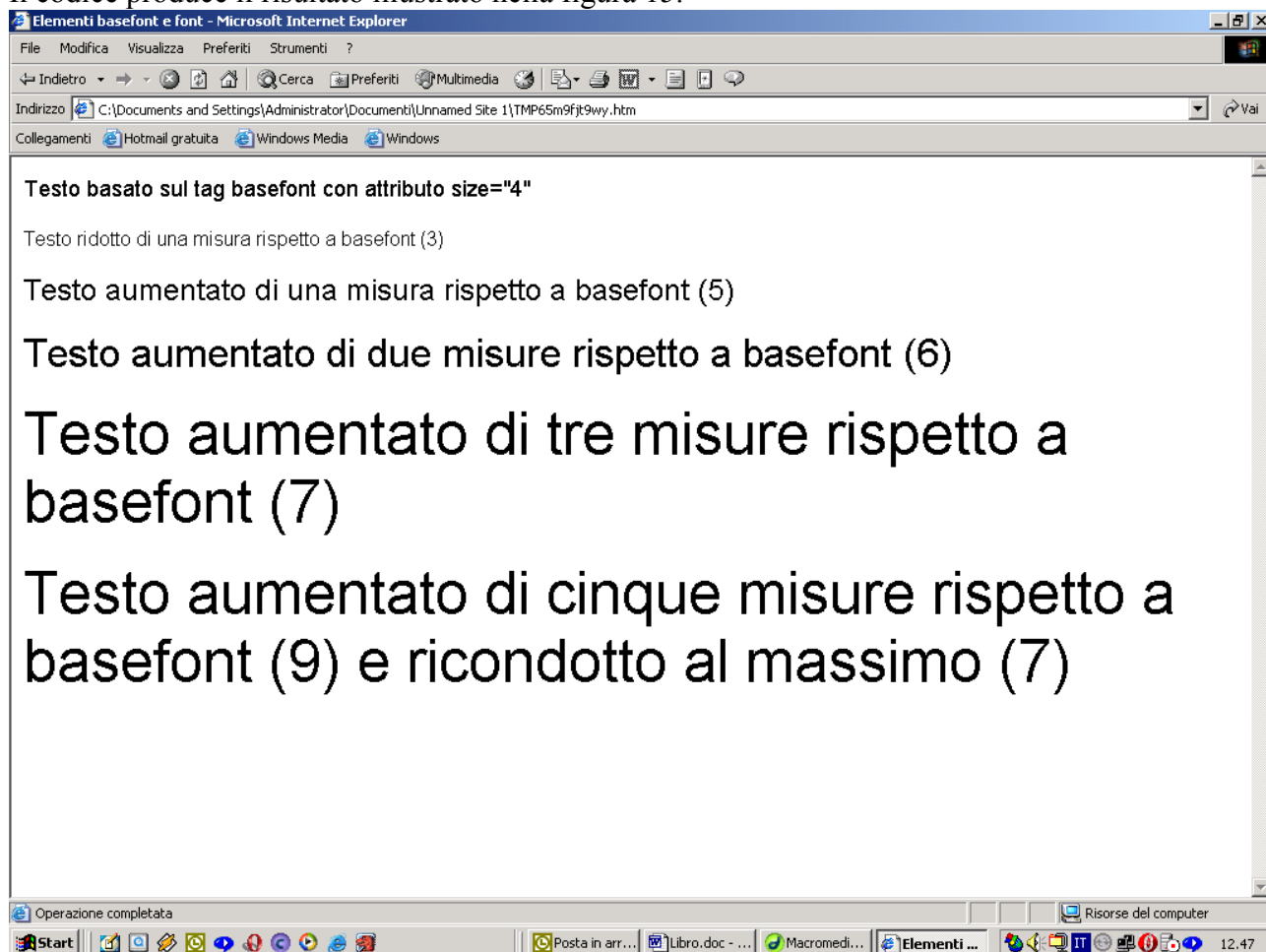


Fig. 15: Ridimensionamento del testo con i tag basefont e font

Si osservi che anche nel caso in cui le dimensioni dei caratteri siano impostate in modalità relativa non possono comunque ricadere al di fuori dei limiti minimi (1) e massimi (7) previsti. Per i valori assegnati all'attributo *face* si veda quanto esposto al paragrafo “Famiglie di caratteri”.

Famiglie di caratteri

Abbiamo già discusso dell'opportunità, all'interno di una stessa pagina HTML, di indicare più nomi di caratteri che saranno usati dal browser nello stesso ordine in cui sono elencati, sino a trovare un carattere disponibile sul computer dell'utente. Un'ulteriore possibilità di far visualizzare le pagine nel modo più aderente possibile rispetto a quello progettato è data dall'utilizzo di nomi di *famiglie generiche di caratteri*. Tali famiglie generiche sono cinque, come elencato di seguito.

- *serif*: è resa sempre con caratteri con grazie;
- *sans-serif*: è resa sempre con caratteri senza grazie;
- *cursive*: è resa sempre con caratteri in corsivo;
- *fantasy*: è resa sempre con caratteri “fantasiosi”;
- *monospace*: è resa sempre con caratteri monospaziati, nei quali ad ogni carattere è riservata una medesima larghezza.

In quanto ultima alternativa normalmente le cosiddette famiglie di caratteri sono posizionate alla fine della lista dei nomi dei caratteri nelle istruzioni HTML. In questo modo se ad esempio si vuole rappresentare un testo o una parte di esso con caratteri con grazie, oltre agli specifici nomi di caratteri aventi questa caratteristica, si terminerà la lista con il nome della famiglia di caratteri *serif*. L'effettivo carattere usato dipende dal sistema operativo e dal browser con il quale il documento è visualizzato ma, con questa tecnica, si è certi che la rappresentazione avverrà con uno dei caratteri appartenenti alla famiglia indicata.

Colori

I colori, contribuendo a guidare l'utente attraverso la struttura di un sito e le diverse sezioni di cui è costituito, rappresentano un altro fondamentale elemento della progettazione delle pagine Web. Di seguito si propongono i principali termini usati per la descrizione delle proprietà dei colori.

Tinta – E' il nome di un colore dello spettro (giallo, rosso, blu, ecc.);

Saturazione – E' l'intensità di un colore;

Valore – E' la luminosità o lo scuro relativo di un colore;

Ombra – E' la gradazione di colore ottenuta tramite l'aggiunta di nero;

Colori primari – sono i tre colori principali di un modello cromatico;

Colori secondari – sono i colori prodotti attraverso la combinazione di due colori primari.

La rappresentazione dei colori negli schermi televisivi e nei monitor dei computer è ottenuta tramite il modello cromatico *RGB* (Red, Green, Blue) che è un modello *additivo*, vale a dire che i colori si ottengono componendo le onde luminose, sommando i componenti principali dei colori per ottenere tutte le possibili tonalità. In questo modello i colori primari sono rosso, verde e blu mentre i colori secondari sono ciano, magenta e giallo. Nelle pagine HTML i colori possono essere indicati attraverso il loro nome o mediante un valore esadecimale a sei cifre. I nomi dei colori sono più

semplici da utilizzare e devono essere scelti da un'insieme di nomi contenuti nelle specifiche HTML o supportati dai diversi browser.¹⁸ Delle sei cifre del valore esadecimale le prime due cifre si riferiscono al colore rosso, le seconde due al colore verde e le ultime due al colore blu.

L'assenza di colore è indicata con il valore 00 mentre la presenza del colore alla sua massima intensità è espressa con il valore FF. I colori si ottengono quindi combinando diversamente i valori di ciascuna coppia indicanti l'intensità dei tre colori primari da usare per l'ottenimento del colore finale. Se tutti i valori sono espressi con la massima intensità possibile si ottiene il colore bianco mentre l'assenza di tutti e tre i valori determina la riproduzione del colore nero. La sintassi di HTML prevede che le tre coppie di valori siano precedute dal simbolo “#”. Tale simbolo (detto *octothorp*) ha lo scopo di comunicare al browser che il valore che segue è il numero del colore espresso in esadecimale e non il nome del colore stesso. In HTML i colori sono espressi utilizzando combinazioni dei colori primari rosso, verde e blu, in base al modello RGB. Ognuno dei tre colori primari incide in base ad un valore compreso tra 0 e 255, per complessivi 256 valori. Il numero massimo di colori ottenibili è dunque pari a 256^3 (16,7 milioni di colori circa). La scelta dei colori con i quali rappresentare le pagine HTML dovrebbe sempre avvenire nell'ambito dei possibili colori previsti dalla *tavolozza dei colori* per il Web, il cui sviluppo risale al 1994. In tale periodo, nel quale il Web cominciava a diffondersi, i monitor dei computer usavano una codifica dei colori ad 8 bit e potevano pertanto visualizzare sino a 256 colori. Questi 256 colori sono quelli considerati sicuri per l'utilizzo sul Web.¹⁹ La sicurezza è qui intesa come garanzia che i colori scelti per la riproduzione su una pagina HTML siano correttamente rappresentati. Se le pagine sono progettate per visualizzare colori non *Web-safe* e se il monitor dell'utente non è in grado di visualizzarli il browser potrebbe cercare di sostituirli, alterando quindi l'aspetto delle pagine e conducendo a risultati cromatici non del tutto prevedibili. In taluni casi il testo delle pagine potrebbe anche risultare illeggibile. Numerosi editor grafici HTML consentono di selezionare i colori direttamente da una cosiddetta tavolozza che appare sullo schermo durante l'editazione della pagina, inserendo automaticamente nel codice sorgente i valori esadecimali associati alle scelte cromatiche effettuate dall'autore. In Internet sono disponibili diversi strumenti di ausilio all'utilizzo dei colori, come elenchi dei colori appartenenti alla tavolozza di colori Web-safe, utilità per eseguire la conversione dei valori dei colori da decimale in esadecimale, ed altro ancora.

Infine, ma l'argomento non è meno importante, deve essere considerato che ai colori sono in genere associati significati diversi, sia positivi sia negativi, e che possono generare reazioni emotive di diversa intensità nei visitatori. A ciò deve essere aggiunto il diverso significato che uno stesso colore può avere in relazione alle differenti etnie e culture. Non dimentichiamo, infatti, che i siti Web sono “vetrine” aperte 24 ore su 24 e 7 giorni su 7, accessibili da qualsiasi postazione connessa ad Internet, da qualsiasi parte del mondo.

Di seguito è illustrato un esempio di codice che imposta il colore di sfondo della pagina in blu ed il colore del testo in bianco. Per modificare il colore di default dello sfondo si usa l'attributo *bgcolor* mentre per modificare il colore di default di un testo si utilizza l'attributo *text*, entrambi utilizzabili all'interno della sezione *body*.

```
<html>
  <head>
    <title>Esempio di colori caratteri e sfondo</title>
  </head>
  <body bgcolor="#0000FF" text="#FFFFFF">
    Questa riga di testo è in bianco su sfondo blu
  </body>
</html>
```

Il codice produce il risultato illustrato nella figura 16.

¹⁸ Le specifiche HTML 4.01 prevedono 16 nomi di colore, ma quelli supportati dai browser sono molti di più.

¹⁹ Da parte di alcuni specialisti si ritiene che solo 212 colori tra i 256 siano sicuri.

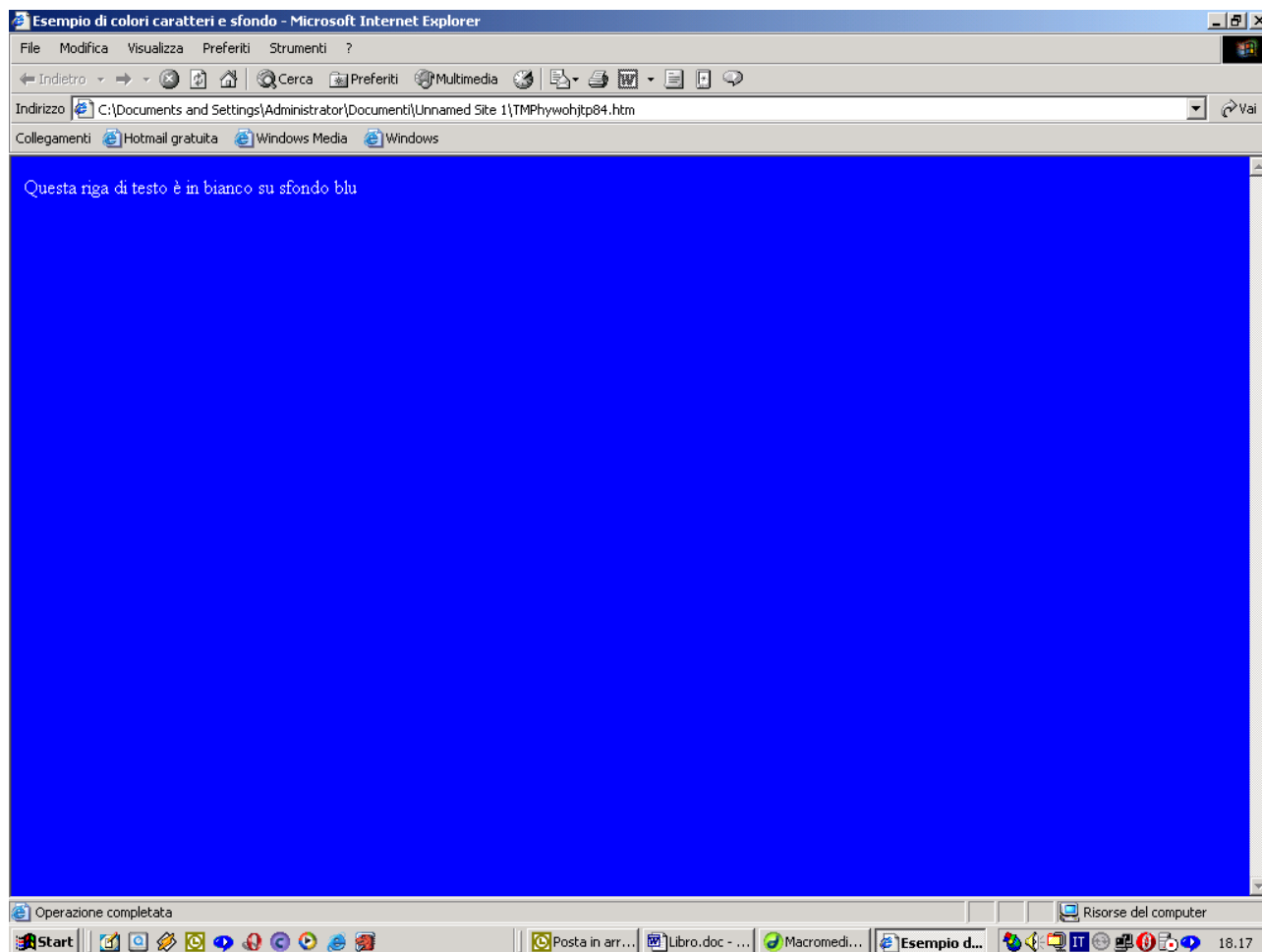


Fig. 16: Variazione dei colori nella sezione body con alcuni suoi attributi

Si può ottenere lo stesso effetto indicando i colori attraverso il loro nome anziché i valori esadecimali, come nella riga seguente:

```
<body bgcolor="blue" text="white">
```

Tag di stile fisici e logici

Oltre agli interventi di formattazione sui caratteri, un testo può essere modificato da numerosi altri tag che possono essere suddivisi concettualmente in tag *fisici* e tag *logici*.

Per comprendere questi due concetti può essere opportuno riferirsi ad un esempio. Si supponga di voler evidenziare una parte di testo utilizzando, ad esempio, il carattere corsivo. Ciò può essere ottenuto ricorrendo al tag `<i>` oppure al tag `<em>`. Il primo, che è un tag fisico, rende in corsivo la parte di testo contrassegnata, indipendentemente dal browser e dalla piattaforma di rappresentazione utilizzata. Il secondo, che rientra invece nella categoria dei tag logici, evidenzia la parte di testo contrassegnata, *generalmente*, in corsivo. Nel secondo caso, in altri termini, si ottiene comunque l'effetto di evidenziazione ma non si è certi del modo in cui ciò avviene. A questo punto possiamo affermare che un tag fisico ha lo scopo di contrassegnare le parti del documento per definirne la struttura eseguendo contestualmente l'opportuna formattazione mentre un tag logico definisce la struttura prescindendo dallo specifico aspetto. Volendo fornire un altro esempio, il tag `<b>`

visualizza in grassetto il testo incluso nell'elemento ed ha pertanto lo scopo di formattare opportunamente il testo. Si tratta perciò di un tag fisico. Il tag `<address>` definisce il testo dell'elemento come indirizzo, prescindendo dalla sua formattazione (che viene generalmente resa in corsivo). E' pertanto un tag logico.

Il concetto sottostante l'utilizzo dei tag logici sta nell'ottimizzazione della rappresentazione delle pagine in rapporto alla piattaforma usata per tale rappresentazione. Ritorniamo a considerare il tag `em` prima esemplificato. Sappiamo che evidenzia il testo contrassegnato. Il punto è che le pagine possono essere accedute e rappresentate tramite diversi strumenti, tra i quali rientrano ad es. i sistemi vocali che leggono le pagine trasformando le informazioni digitali in messaggi vocali. Nel caso del tag `em` la parte di testo evidenziata verrà, in genere, visualizzata in corsivo se la pagina è acceduta tramite browser e verrà, in genere, riprodotta con un volume maggiore rispetto alla restata parte del testo, nel caso venga acceduta tramite un sistema vocale. L'evidenziazione ottenuta tramite il tag `<i>` sarebbe quindi valida solo nel primo caso ma nel secondo non avrebbe avuto effetto. Non esiste un modo per pronunciare un corsivo! Il tag logico viene pertanto reso a seconda di come è interpretato dalla piattaforma di rappresentazione effettivamente utilizzata. In conclusione i tag logici dovrebbero essere usati al posto di quelli fisici nel caso in cui si ritiene che le pagine siano accedute attraverso dispositivi di diversa tipologia.

Nell'esempio successivo si riportano alcuni tag, fisici e logici, per la formattazione dei caratteri.

```
<html>
<head>
<title>Formattazione caratteri</title>
</head>

<body>
<h1>Esempi di formattazione caratteri</h1>

<p>Testo normale</p>
<p><b>Testo in grassetto</b></p>
<p><i>Testo in corsivo</i></p>
<p>Testo in<sup>apice</sup></p>
<p>Testo in<sub>pedice</sub></p>
<p><tt>Testo monospaziato</tt></p>
<p>Testo<s>sbarrato</s></p>
<p>Testo con formattazione<pre>
      p
      r
      o
      p
      r
      i
      a</pre></p>

</body>
</html>
```

Il codice produce a video quanto rappresentato nella figura 17.

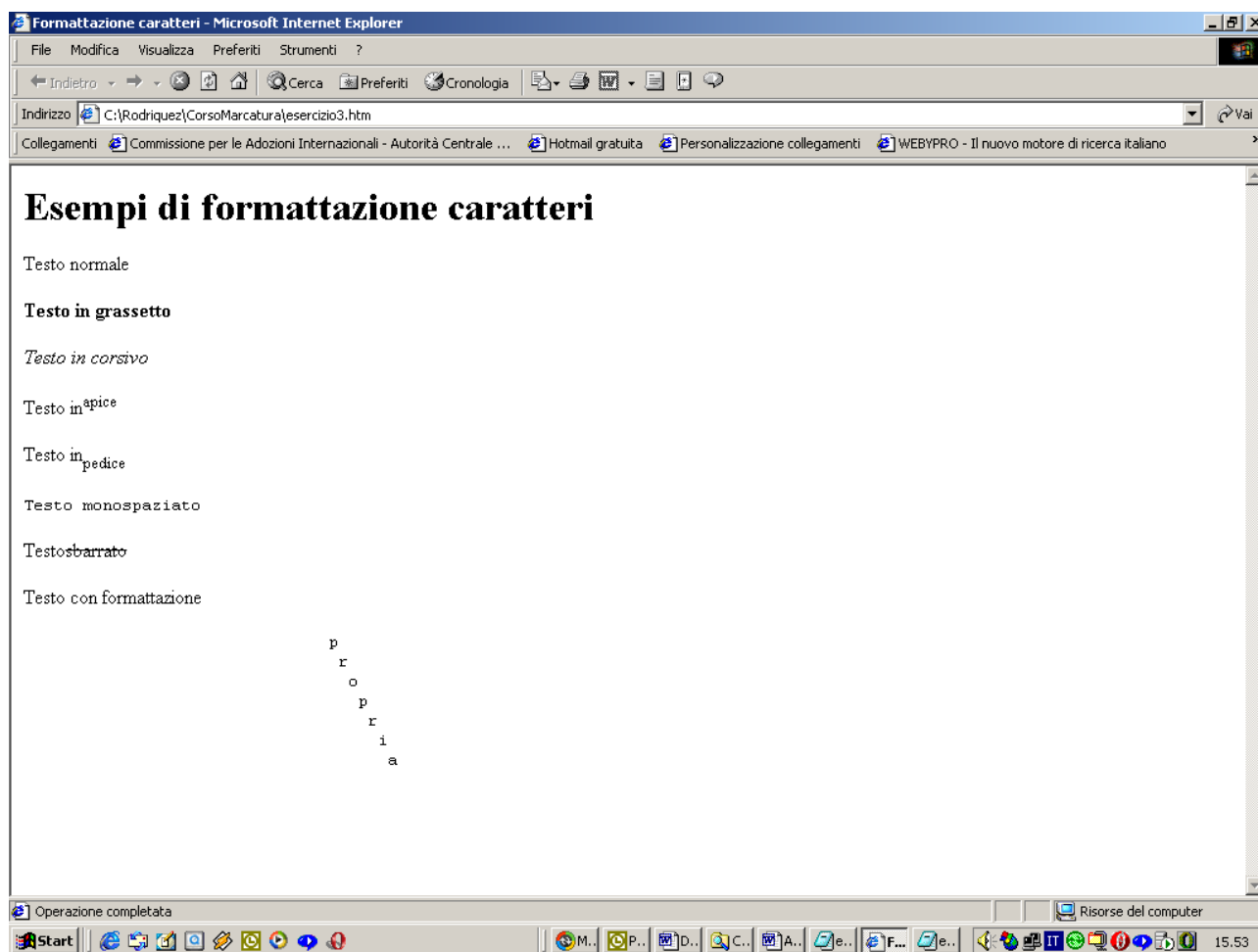


Fig. 17: Esempi di tag per la formattazione dei caratteri

Dall'immagine risultante a video si può osservare che:

- il tag (logico) <sup> visualizza il testo in apice;
- il tag (logico) <sub> visualizza il testo in pedice;
- il tag (fisico) <s> trasforma il testo in un testo sbarrato, tracciando su di esso una linea orizzontale;
- il tag (fisico) <tt> attribuisce al testo un carattere in cui ad ogni lettera o spazio è riservata una medesima dimensione (carattere monospaziato);
- il tag <pre> visualizza il testo con la sua formattazione predefinita, nella quale rimangono invariati gli spazi e le interruzioni di riga.

Collegamenti ipertestuali

Le pagine HTML possono diventare “parte attiva” nel Web aggiungendo ad esse i collegamenti ipertestuali. Si tratta, in termini estremamente semplificati, di “zone” della pagina opportunamente evidenziate cliccando sulle quali avviene il collegamento ad altre risorse. I link possono puntare a risorse di altri siti, dello stesso sito, presenti in altre parti della stessa pagina. Le risorse possono essere costituite da pagine HTML, immagini, altri oggetti multimediali, servizi di invio mail, ecc.

I link consentono la trasformazione del testo da un ambiente a due dimensioni ad un ambiente multidimensionale. In un testo tradizionale i collegamenti possono essere creati definendo un indice degli argomenti, un glossario, una bibliografia o inserendo delle note nel testo. E’ evidente che in questo tipo di collegamenti manca l’aspetto dinamico che è invece presente nel Web. Con i link ipertestuali ciascun visitatore è in grado di creare il proprio autonomo percorso all’interno dell’enorme patrimonio informativo disponibile sulla rete, assimilabile ad una vastissima biblioteca virtuale. Al posto di una “semplice” bibliografia potrebbero essere direttamente inseriti i collegamenti alle fonti citate, se disponibili sulla rete. Mediante la tecnica degli hyperlink risulta quindi possibile inserire tutte le fonti informative e gli approfondimenti che l’autore ritiene opportuno senza che tali inserimenti interrompano in alcun modo la narrazione. Il lettore è libero di seguire il flusso principale o di volgere la propria attenzione verso le possibili fonti di informazione aggiuntive previste dall’autore. L’utilizzo delle note in un testo tradizionale, in particolare, pur costituendo punti di chiarimento e approfondimento inseriti dall’autore ove ritenuto opportuno, non possono non avere natura di informazioni di tipo secondario, proprio perché non facenti parte del testo. I collegamenti ipertestuali, rappresentando uno strumento aggiuntivo e flessibile di navigazione, devono essere costruiti ponendo particolare attenzione al fatto che il contesto di riferimento sia sufficientemente adeguato. L’utente deve, in altri termini, avere un’idea chiara di dove porta il collegamento. Seguire un link significa abbandonare la pagina o la parte di pagina corrente e questa azione dovrebbe essere accompagnata dalla consapevolezza di poter ottenere altrove ciò di cui si ha bisogno. L’utilizzo dei collegamenti ipertestuali può inoltre essere dovuto esclusivamente a motivazioni tecniche, con lo scopo di ridurre i tempi di scaricamento (*download*) delle pagine HTML. Possono essere usati, ad esempio, per collegarsi a più documenti invece di avere un unico, voluminoso documento, oppure per mostrare immagini miniaturizzate sulle quali il visitatore può cliccare per ottenere le immagini di dimensioni normali, una volta scelta quella di proprio interesse. Tali tecniche sono da preferire rispetto al mettere a disposizione dei visitatori un’unica voluminosa risorsa, scomoda da consultare e che richiede tempi lunghi per lo scaricamento. I visitatori, e ciò vale soprattutto per i siti commerciali, non amano attendere troppo a lungo le risorse Web di cui hanno bisogno e potrebbero essere pronti a rivolgersi alla concorrenza.

Storyboard

La sequenza delle possibili videate nelle quali si articola un sito, e che include oltre al contenuto anche lo schema dei link tra le pagine, è detta *storyboard*. Usualmente uno storyboard ha inizio nel momento in cui l’autore del sito inizia a disegnare su carta lo schema del sito apportando via via le opportune modifiche. Questo schema è poi sottoposto a periodiche revisioni, anche con la partecipazione dell’eventuale committente, sino a diventare definitivo. Il diagramma così ottenuto è poi utilizzato nel corso dell’effettiva realizzazione del sito ed è particolarmente utile nel caso in cui il lavoro di sviluppo venga svolto contemporaneamente da più persone.

Gli schemi dei link possono essere classificati in tre categorie: *sequenziali*, *gerarchici* e *personalizzati*. Lo schema di tipo sequenziale, adatto anche a testi tra loro correlati, ed anche estesi,

può essere realizzato, a sua volta, in due diverse modalità: in una stessa pagina oppure su più pagine. Nel primo caso la parte iniziale della pagina contiene link a parti specifiche della stessa pagina, che generalmente corrispondono ai diversi argomenti. Nel secondo caso, relativo a più pagine, i link attivano collegamenti a pagine specifiche, come ad esempio nel caso di un libro i cui link aprono pagine relative a diversi capitoli. Ogni pagina contiene quindi un link alla pagina successiva, da cui deriva il nome di schema sequenziale. Lo schema gerarchico, anch'esso sviluppato su più pagine, è invece adatto per testi che non devono necessariamente essere letti in ordine sequenziale, come nel caso di manuali tecnici per i quali deve essere data la possibilità di consultare direttamente la parte di proprio interesse. La pagina principale di questo schema contiene quindi tutti i link verso tutti i possibili argomenti in cui il testo è stato suddiviso. Infine, lo schema personalizzato può essere definito come un ibrido tra quello sequenziale e quello gerarchico, presentando una serie di link che associano un punto ad un altro del sito, senza seguire del tutto uno dei due schemi. La scelta su quale schema di link usare dipende, oltre che dal tipo di contenuto del sito, anche dallo stile che si intende dare al sito stesso.

Creazione dei link

I link sono composti da una sorgente (origine) e da un obiettivo (destinazione). La sorgente è l'elemento che funge da collegamento, l'obiettivo è ciò che viene visualizzato attivando il link. La sorgente può essere data da una singola parola, un insieme di parole, un'immagine, un pulsante. Il tag usato per tutti i tipi di link è: `<a>`²⁰

Link ad altra pagina

Di seguito si riporta la sintassi prevista per realizzare un collegamento ipertestuale ad una pagina diversa da quella corrente.

```
<a href="Pagina seguente.htm">Pagina successiva</a>
```

Il testo compreso tra i tag `<a>` (nell'esempio: pagina successiva) è quello che viene visualizzato e sul quale si clicca con il mouse per eseguire il collegamento. Di norma è di colore blu ed è sottolineato (il colore è predefinito ma l'utente può modificarne l'impostazione tramite il browser). Il valore dell'attributo `"href"`²¹ (nell'esempio: Pagina seguente.htm) è la pagina HTML con la quale si intende eseguire il collegamento. Affinché l'esempio funzioni correttamente occorre inserire nella pagina di origine il link illustrato e creare il file di destinazione con estensione `".htm"` nella stessa cartella in cui è stata salvata la pagina di origine.

Link ad altro sito

Di seguito si riporta la sintassi prevista per realizzare un collegamento ipertestuale verso una pagina appartenente ad un altro sito. L'esempio esegue il collegamento alla home page del sito della Presidenza della Repubblica.

```
<a href="http://www.quirinale.it">Presidenza della Repubblica</a>
```

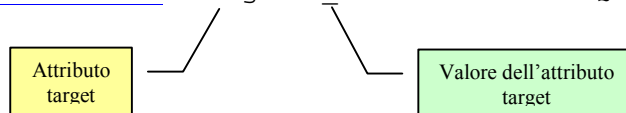
²⁰ A sta per anchor (ancora).

²¹ Href sta per Hypertext Reference.

L'elemento `<a>` dispone di un attributo denominato *target*, tramite il quale è possibile specificare dove dovrà essere aperta la risorsa collegata. Ad esempio, per aprire un link in una nuova finestra del browser il valore dell'attributo *target* dovrà essere uguale a `"_blank"`.

Si riporta, di seguito, un esempio della sintassi utilizzata per realizzare un link verso il sito della Presidenza della Repubblica, che deve essere aperto in una nuova finestra del browser:

```
<a href="http://www.quirinale.it" target="_blank">Sito del Quirinale</a>
```



Un altro modo per aprire una finestra indipendente del browser è:

```
<a href="http://www.quirinale.it" target="_new">Sito del Quirinale</a>
```

Si osservi che le due soluzioni indicate non consentono di stabilire le dimensioni e le opzioni della nuova finestra. Per fare ciò è possibile utilizzare il linguaggio JavaScript.

Così come per le immagini, anche per i link è possibile aggiungere del testo descrittivo che appare quando il puntatore è posizionato sul link sotto forma di un piccolo suggerimento (*tip*), come riportato nell'esempio successivo.

```
<a href="http://www.quirinale.it" TITLE="Visita il sito della Presidenza della Repubblica">Presidenza della Repubblica</a>
```

Posizionando il puntatore sul link si ottiene quanto riprodotto nella figura 18.

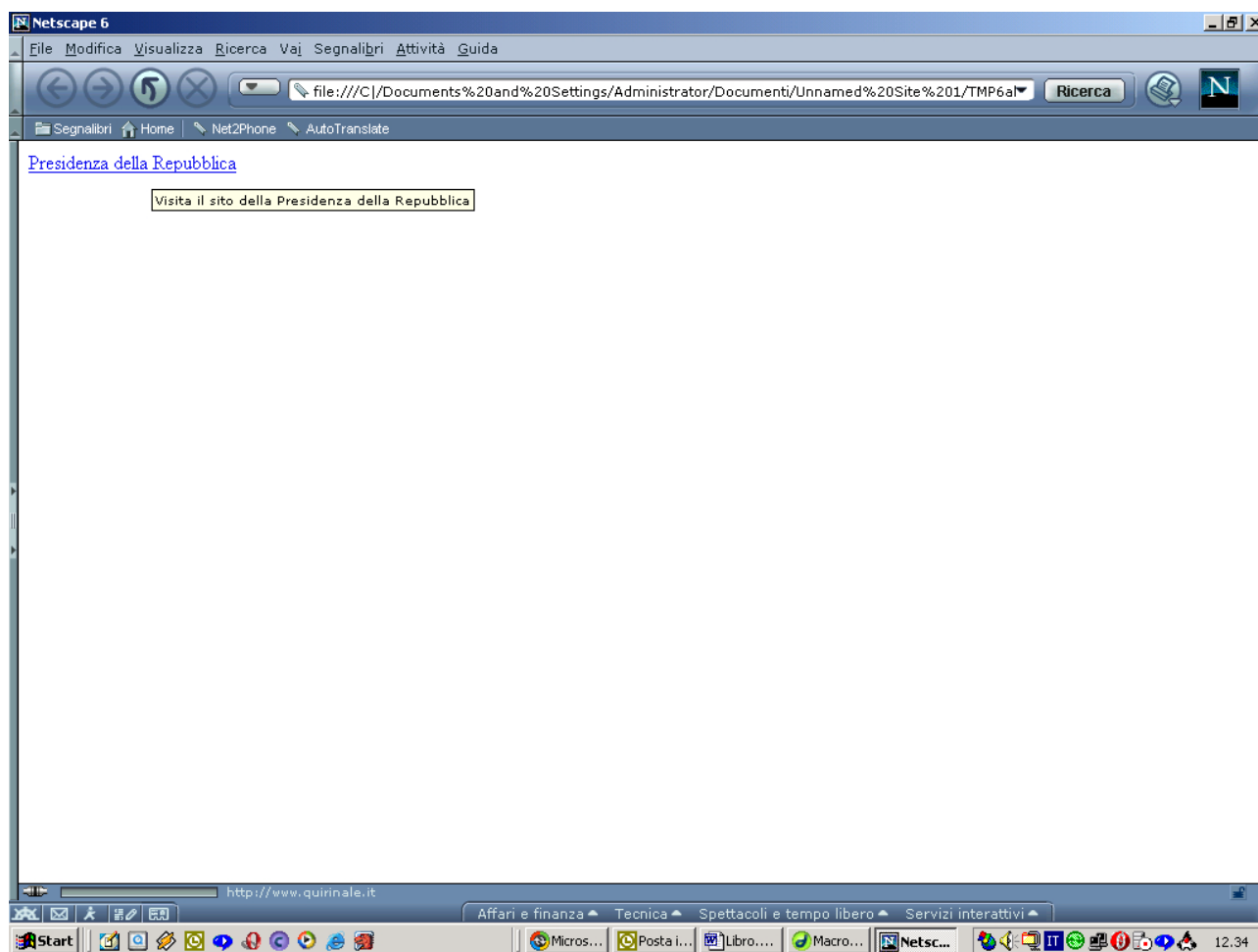


Fig. 18: Esempio di link con testo descrittivo aggiunto tramite l'attributo title

Di seguito è proposta una pagina HTML che esegue quanto appena discusso

```
<html>
  <head>
    <title>Link ad altra pagina</title>
  </head>
  <body>
    <h1>Esempio di link ad altra pagina</h1>
    <p>Di seguito è mostrato un esempio di collegamento ad un'altra pagina.
    <p>
    In questo caso l'altra pagina si trova su su un altro <i>sito</i>.
    <p>
    Per attivare il link fare click sulla parola seguente:
    <a href ="http://www.governo.it">Governo italiano</a>.
    <p>
    <hr>
  </body>
</html>
```

Il codice produce il risultato illustrato in figura 19.

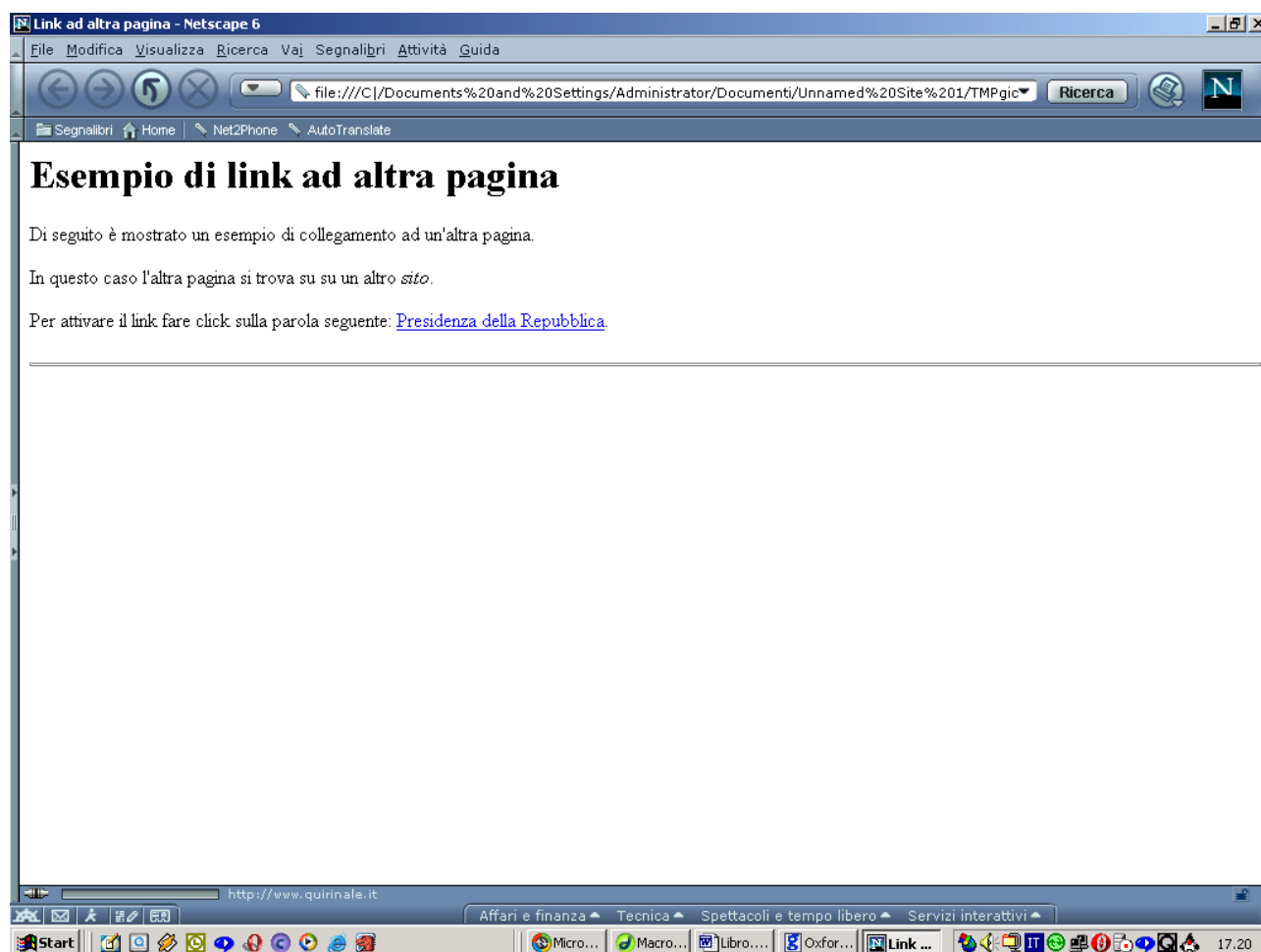


Fig. 19: Esempio di intestazione, paragrafi e link ad un sito

Oltre ai link testuali è possibile utilizzare un'immagine come link. Ciò avviene sostituendo la denominazione del link (nell'esempio "Presidenza della Repubblica") con il tag `<img>`, come di seguito riportato.

```
Sito della Presidenza della Repubblica<a href="http://www.quirinale.it/"></a>
```

Con l'istruzione mostrata, l'immagine di tipo gif contenuta nella pagina HTML diviene un link la cui attivazione tramite il mouse genera l'apertura del sito indicato.

Nel tag `img` dell'esempio sono stati usati gli attributi *src* che indica il nome del file che rappresenta l'immagine da visualizzare, *width* e *height* che rappresentano, rispettivamente, la larghezza e l'altezza con le quali l'immagine è visualizzata nella finestra del browser ed infine l'attributo *border* che, se impostato a zero, elimina il bordo dall'immagine che rappresenta il link sullo schermo.

Link a sezioni specifiche

I link a parti specifiche (link interni alle pagine) sono usati in genere per consentire un veloce spostamento all'interno della stessa pagina. Un tipico utilizzo di questa tipologia di link è dato da un indice di argomenti collocato nella parte superiore di una pagina. Cliccando su una delle lettere dell'alfabeto che costituiscono l'indice l'utente è indirizzato alla parte di pagina interessata. Ciò avviene in quanto per ciascuna lettera dell'alfabeto è stato creato il corrispondente link alla relativa parte di pagina. L'utilizzo del link a sezioni specifiche richiede la definizione di un *punto di ancoraggio* con il tag `<a>` cui il link si riferisce.

Il funzionamento di questa tipologia di link richiede la definizione di una destinazione, attraverso un nome, e di un collegamento a tale destinazione, come successivamente illustrato.

Destinazione: `<a name="nomeparte">< /a>`²²

Con l'attributo *name* di questo tag si definisce il nome da usare per individuare il punto di ancoraggio (destinazione).

Link indirizzo: `<a href="#nomeparte">Vai a destinazione</a>`

Con questa istruzione si crea il link che punta alla destinazione. Il carattere `"#"` deve precedere il link di destinazione ed indica il riferimento ad un punto di ancoraggio presente nella stessa pagina. Se la parte specifica da raggiungere con il link si trova su un altro documento l'indirizzo web del documento deve precedere il carattere `"#"`. E' evidente che mentre ogni singolo autore di pagine HTML ha tecnicamente la possibilità di poter collegare le proprie pagine a qualsiasi pagina accessibile dal Web i link a sezioni specifiche di pagina possono essere realizzate solo dall'autore della pagina stessa in quanto è necessario definire sulla pagina il punto di destinazione ed è quindi indispensabile avere accesso al relativo file HTML.

Infine, si sottolinea che se il browser non trova il punto di ancoraggio specificato nell'indirizzo non si verifica alcun errore, ed il browser si posiziona all'inizio della pagina.

²² Tra il tag `"a"` di apertura e chiusura non occorre inserire alcun testo, in quanto semplice punto di ancoraggio.

Link al client di posta elettronica

Una particolare tipologia di link è data dalla possibilità di creare sulla pagina HTML un collegamento al programma di posta elettronica dell'utente. Attivando questo link, se sul computer dell'utente è installato un programma di posta elettronica, viene aperto tale programma con il testo del messaggio vuoto e con l'indirizzo già compilato, in base a quanto contenuto nell'istruzione di link. Si compone utilizzando la parola chiave *mailto* come di seguito mostrato.

Per ulteriori informazioni inviate una
`href="mailto:informazioni@indirizzosocieta.com">mail</A>`

Nella figura 20 è illustrato il risultato che si ottiene con il codice esemplificativo appena visto.

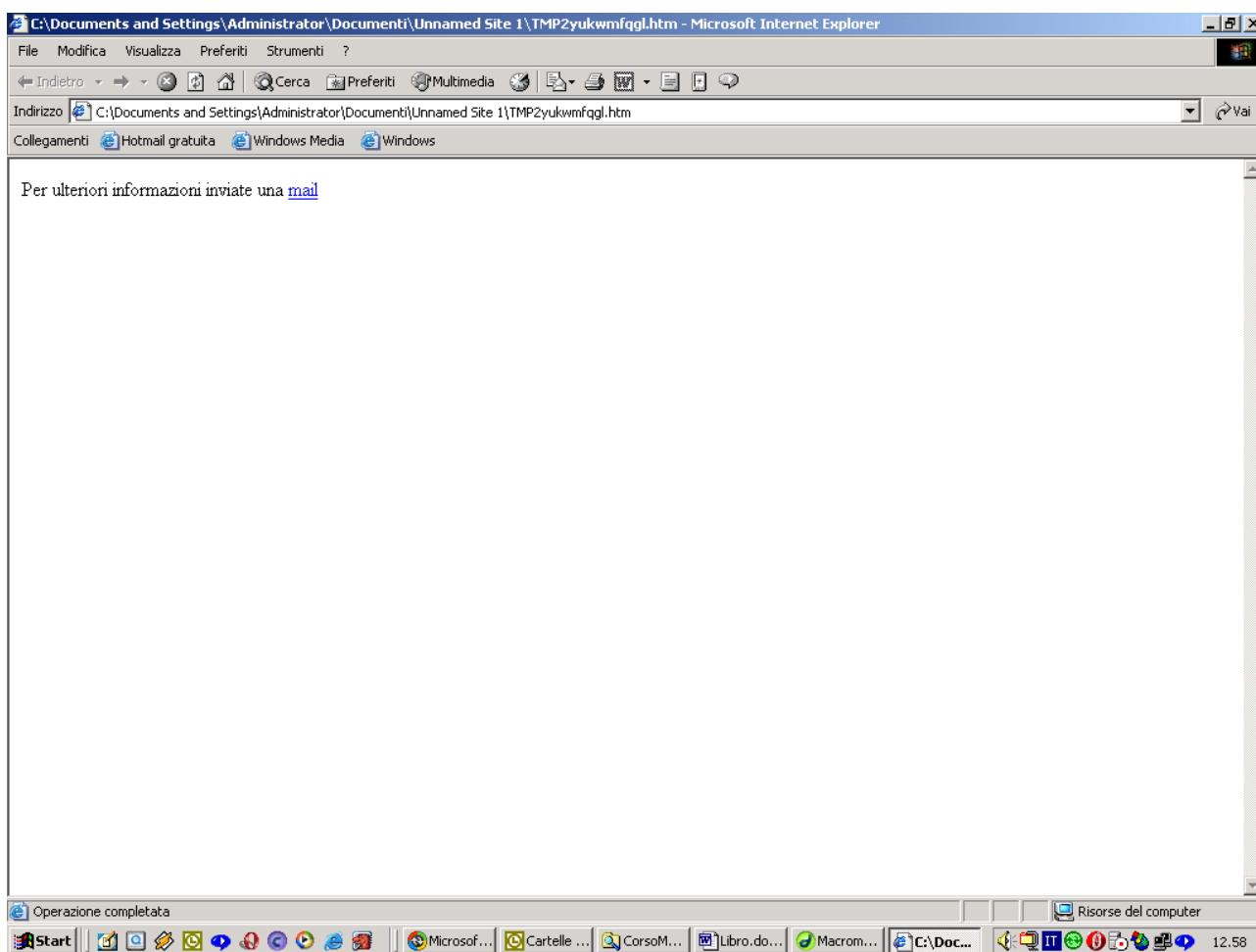


Fig. 20: Il collegamento ipertestuale ottenuto tramite la parola chiave mailto.