Authors and contributors
Fabio Proietti

Feedback
Please direct any comments or suggestions about this document to
fabio.proietti AT istruzione DOT it

Publication date and version

last modified 2012-11-04

bibliografia

http://it.wikipedia.org/wiki/Ascii

http://en.wikipedia.org/wiki/Western_Latin_character_sets_%28computing%29

Le estensioni dei file

In quale modo si vedono i nomi dei file sul computer di casa?
Nella seguente figura ci sono due possibili modalità di visualizzazione dei file: quale differenza c'è tra questi due modi di vedere i file?
Nella parte destra della figura il nome del file è più lungo che nella parte sinistra, perché è presente anche l'estensione del file.

mario **mario.jpg**

Illustration 1: Due modalità di visualizzazione dei file

L'estensione di un file è la sigla che compare in fondo al nome, dopo il punto, e aiuta a stabilire il tipo di dati contenuto nel file. Purtroppo, se il sistema operativo nasconde queste ultime tre lettere, per l'utente può risultare difficile comprendere il tipo di dati che esso contiene.
A titolo di esempio, altri tipi di comuni estensioni sono:

- `file.txt` file di testo non formattato
- `file.jpg` file di immagini "compresse" nelle dimensioni con perdita di qualità
- `file.png` file di immagini senza perdita di qualità
- `file.zip` file che può contenere altri file (compressi)
- `file.doc` file di testo formattato secondo Microsoft Word fino al 2007
- `file.docx` file di testo formattato secondo Microsoft Word a partire dal 2007
- `file.odt` file di testo formattato secondo LibreOffice writer
- `file.xls` file di Microsoft Excel
- `file.ods` file di LibreOffice Calc
- `file.pdf` file di testo formattato, in genere non modificabile

Attività: come si possono visualizzare le estensioni dei file a scuola e a casa propria?
Purtroppo il sistema operativo Microsoft di solito nasconde le estensioni e per visualizzarle si deve seguire una delle procedure descritte di seguito:

Microsoft Windows XP	Aprire Risorse del computer, aprire il menu Strumenti, Opzioni, e la scheda Visualizzazione: togliere il segno sulla casella "nascondi le estensioni dei tipi di file conosciuti".
Microsoft Windows Vista	Aprire dal menu di avvio, aprire il Pannello di controllo, aprire Aspetto e personalizzazione, Opzioni cartella, scheda Visualizza, Pulsante Impostazioni avanzate: togliere il segno di spunta su "Nascondi le estensioni dei tipi di file conosciuti".

La domanda nasce spontanea: perché questi sistemi operativi nascondono le estensioni?
Per fortuna ci sono altri sistemi operativi, più "onesti", come MAC OS X e GNU/Linux, che non nascondono nulla all'utente. Tutti i file con questi tipi di estensioni sono detti genericamente "file" o anche "file di dati" perché contengono dati di un certo tipo, organizzati in una determinata struttura logica.

Attività: provare a creare un file di testo vuoto. Si vede l'estensione .txt? Provare a rinominarlo. Rinominandolo è necessario conservare la stessa estensione

Di solito al momento del salvataggio l'utente può scegliere il formato in cui salvare, cioè il modo in cui i dati sono salvati nel file. Quando si salvano i propri dati bisogna scegliere l'estensione che non costringa nessuno, in futuro, ad acquistare il programma necessario a visualizzarli.

I nomi dei file

Oggi è possibile dare ad un file un nome lungo diverse decine di caratteri, usando ogni tipo, o quasi, di simboli, sia maiuscoli che minuscoli, ma per evitare di avere problemi quando si scambia un file con altre persone, che possono avere un sistema diverso dal nostro, si adottano le seguenti rigorose regole:

- non usare le maiuscole nei nomi dei file, se non è necessario
- non usare lo spazio nei nomi dei file, ma, al suo posto, usare un trattino
- usare nomi abbastanza brevi, magari di una sola parola

attività: controllare quali nomi vengono usati per le immagini wikimedia: ci sono spazi?

I programmi

Oltre ai file di dati esiste un altro tipo di file, i **file eseguibili**, o applicazioni, o ancora, **programmi**, che non contengono dati, ma istruzioni e comandi che il computer esegue quando gli viene richiesto dall'utente. Che relazione esiste tra i file di dati e questi "programmi"? Un programma è usato per "vedere" o "modificare" il contenuto di un file di dati. Ad esempio, per vedere il contenuto di un file.txt si usa un editor di testo (notepad, blocco note o gedit).
Si avvia l'editor, si clicca sul menu File, si clicca su Apri e si sfogliano le cartelle finché non si trova il file desiderato e lo si apre.
In alternativa, questa sequenza di passaggi può svolta automaticamente dal sistema operativo, facendo solo un doppio click sull'icona del file.txt.
Quando tentiamo di aprire uno di questi file, eseguendo un doppio click con il mouse, il sistema controlla l'estensione del file di dati e avvia il programma adatto a mostrare il contenuto di quel tipo di file.

attività: provare a cambiare l'estensione del file da ".txt" in ".abc": cosa altro cambia?

Cosa succede se si fa doppio click?

Provare a ripristinare la corretta estensione.

Modificando manualmente l'estensione di un file, si modifica solo il suo aspetto, non il contenuto.

Attività: provare a cambiare l'estensione di un'immagine (.bmp) in .txt e ad aprirla.

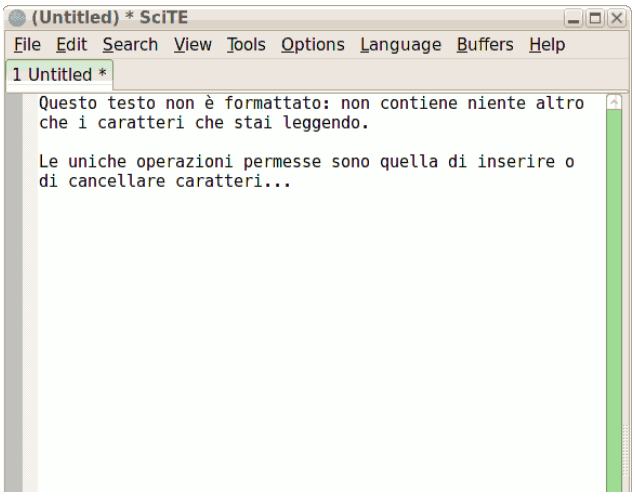
Nella memoria del computer anche le immagini sono costituite da numeri.

I formati digitali dei dati

La parola digitale deriva dall'inglese "digit" che significa "cifra numerica". Il primo esempio di comunicazione “digitale” è stato forse il codice Morse, risalente al 1840, che usava il punto, la linea e le pause per rappresentare le lettere e le parole. Oggi i computer rappresentano le lettere attraverso i numeri.
I dati vengono sempre salvati in modo numerico, ma possono essere disposti, raggruppati e organizzati in molti modi diversi a seconda delle necessità. Ogni organizzazione diversa è un diverso **formato**. Di solito ad ogni formato corrisponde una propria **estensione**, ma l'estensione fa parte del nome del file e il nome non garantisce sempre una corretta descrizione del contenuto del file (qualcuno potrebbe averlo rinominato).
Anche se oggi esistono file di diversi tipi (immagini, suoni, video, ecc.) è particolarmente importante provare ad approfondire il formato usato per i file di tipo **testuale**, sia perché sono semplici da studiare, ma anche perché sono molto usati.
I testi possono essere di due tipi: *formattati* oppure *non formattati*.

Il testo non formattato

Dopo aver scritto un documento al computer è possibile scegliere il formato in cui salvarlo, ad esempio, si potrebbe scegliere tra l'estensione .txt e l'estensione .doc.
La nostra scelta determinerà il modo in cui i dati verranno organizzati all'interno della memoria del computer. Per effettuare una scelta consapevole bisogna conoscere il significato delle **estensioni** dei file.
Ad esempio, è poco utile cambiare il colore del testo e poi salvarlo usando l'estensione **.txt**, un'estensione usata per il testo "non formattato" (plain text).
Un programma che permette di creare facilmente testo non formattato è il blocco note, o notepad.

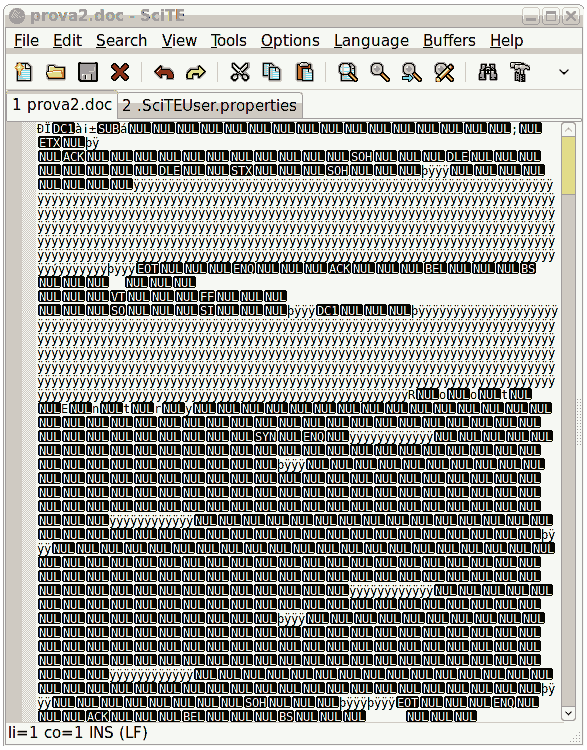


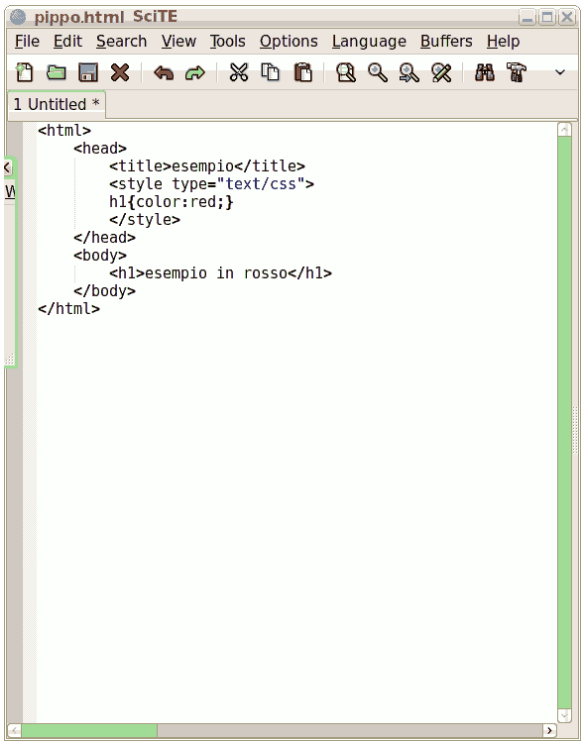
Il testo formattato

Quando al testo viene applicato un particolare "stile grafico", si parla di "testo formattato" (rich text). Se vogliamo colorare o usare il grassetto sul testo dobbiamo usare un programma di scrittura che consente di formattare il testo. Nel prossimo esempio si analizzeranno due diversi file, che hanno lo stesso contenuto: una semplice parola, colorata di **rosso**. Il primo, però, sarà con estensione **.doc**, mentre l'altro con estensione **.htm**.
Entrambe le estensioni consentono di ottenere un testo formattato.

Per aprire questi file useremo un semplice editor di testo (come il blocco note) Aprendo il file .doc vedremo una lunga lista di strani caratteri, come quella in basso a sinistra. È abbastanza difficile pensare che sia necessario tutti questi caratteri per poter memorizzare una sola parola colorata di rosso... Infatti, la stessa parola in rosso, in un file.htm, appare memorizzata in modo molto più semplice (a destra).

I caratteri strani sono in realtà Byte (numeri) che non dovrebbero essere visualizzati come caratteri in un editor, ma che ci permettono comunque di avere un'idea di quale sia il contenuto del file.





In informatica la parola **formattare** ha diversi significati che poterbbero essere confusi:

- formattare **un testo** vuol dire dare una forma estetica al testo
- formattare **un'unità di memoria** di massa vuol dire preparare un'area di memoria e organizzarla in modo che vi possano essere memorizzati i dati.

Il formato dei dati

Le domande che ci possiamo porre sono:

- Quale è il migliore formato per il testo formattato, ad esempio, tra .doc e .htm?
- Perché il formato .doc è così complicato rispetto al .htm, se entrambi riescono a produce la stessa cosa?

La semplicità del formato html consente facilmente di capire il suo contenuto anche usando un semplice editor come il notepad, e consente:

1. di **condividere** più facilmente i nostri file con gli altri;
2. di mantenere la **portabilità** dei dati, anche tra sistemi operativi molto diversi tra loro;
3. di **risparmiare** il costro della licenza di un programma che potrebbe essere indispensabile per leggere e/o modificare i file;
4. di poter **conservare e riutilizzare** per molto tempo importanti dati.

Infatti, se usiamo il formato .doc, nessuno ci garantisce che i computer del futuro saranno in grado di aprire un formato così "strano" come quello proposto da Microsoft Word. Questo dipende dalle scelte di Microsoft.
Da questo punto di vista non sembra una scelta saggia usare il formato .doc.
Chi ci guadagna ad usare questo formato? Chi lo usa quali vantaggi ha?

Il formato .doc è un formato "segreto" o meglio "proprietario", mentre il formato .htm è un formato "aperto". Chi usa un formato proprietario diventa **dipendente** dal proprietario di quel formato di dati. Un formato aperto invece è facile da comprendere anche senza l'aiuto di nessuno.
Siamo ancora abbastanza tranquilli, anche se il Ministero della Difesa italiano utilizza un formato di dati proprietario (cioè, segreto) di un'azienda americana?

Attività: Navigando su wikipedia.org è possibile leggere anche il codice sorgente delle pagine (aprendo un articolo e cliccando su modifica). In che formato vengono salvati i dati di wikipedia per consentire una collaborazione più facile tra gli utenti? Si tratta di testo formattato oppure di testo non formattato?

Il formato aperto

In opposizione al formato proprietario, come quello .doc, si potrebbe usare un formato aperto, come quello .htm, ma in pratica è molto più utilizzato il .doc.

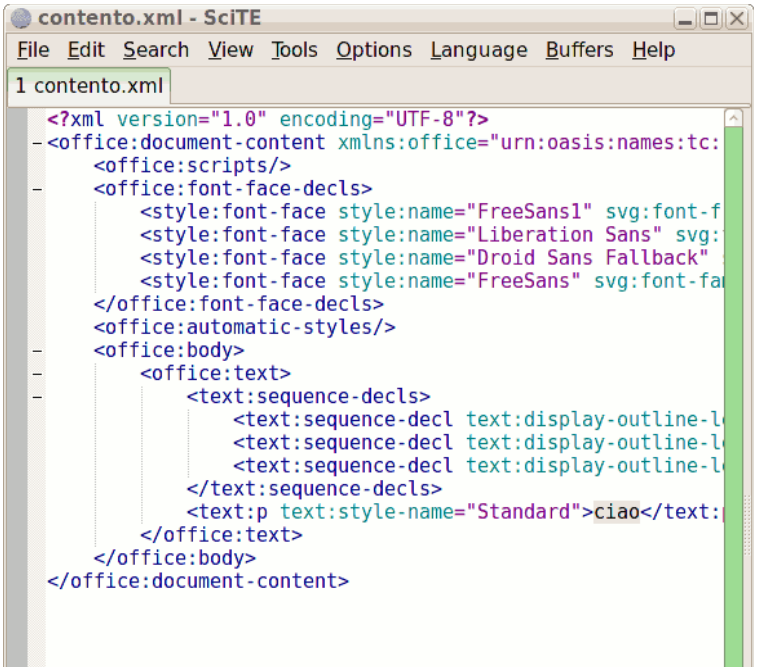
- Il formato dei file .doc consente di inserire anche le immagini.
- Il formato .htm consente di vedere le immagini ma esse vengono memorizzate all'esterno del file e questo non è comodo.

Da diversi anni, per il testo formattato, è nato un nuovo standard (aperto) chiamato **.odt**. Questo è uno standard talmente importante che è stato riconosciuto a livello internazionale dall'Organizzazione Internazionale degli Standard (detta ISO).
Lo usano tutti i programmi di videoscrittura, soprattutto LibreOffice e Apache OpenOffice.
Anche se non è ancora molto diffuso, il .odt potrebbe sostituire facilmente il formato proprietario .doc e superare tutti i problemi dei formati proprietari.

Per capire come funziona questo formato, si può aprire con un editor di testo, come fatto con il .doc.
Prima bisogna rinominare il .odt in .zip, poiché il .odt è un archivio compresso.
Decomprimendo il contenuto si può trovare un file chiamato **content.xml**

Per leggere più facilmente il testo si consiglia di riformattarlo usando questo comando:
xmlindent -o output.xml < content.xml

Aprendo il contenuto del file conten.xml si scoprirà che è molto simile al formato .htm e quindi di facile comprensione.

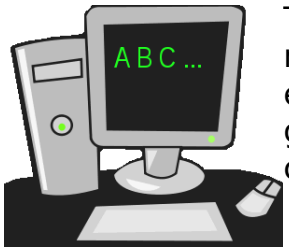


I numeri nel computer

All'interno del computer i numeri interi (che possono misurare fino 4 Byte) possono essere memorizzati usando diversi metodi che si differenziano tra loro per l'ordine con cui si dispongono le cifre (i Byte): ad esempio, si può memorizzare e leggere le cifre del numero da sinistra verso destra o viceversa, cioè il numero 23 può essere rappresentato in memoria come 0023 oppure 2300, utilizzando, rispettivamente, le rappresentazioni **big-endian** o **little-endian**. Questa scelta dipende dall'architettura elettronica della CPU di un computer. Ad esempio le CPU **Intel** adottano la disposizione **big-endian** mentre le CPU **Motorola** quella **little-endian**.

Se un file viene condiviso tra due computer che usano una diversa rappresentazione dei numeri interi, possono verificarsi dei problemi nell'interpretazione di questi numeri. Per questa ragione, si può aggiungere nei file un codice BOM (Byte Order Mark) che specifica il tipo di disposizione delle cifre (big-endian o little-endian), altrimenti sarebbe impossibile comprendere la disposizione dei numeri. Al momento del salvataggio di un file contenente solo numeri interi, dovrebbe essere possibile scegliere la convenzione da utilizzare.

I caratteri nel computer



Tutto ciò che si trova nella memoria del computer è rappresentato come un numero, anche la musica, le immagini e il testo. In un file di testo, per trasformare le lettere e tutti gli altri simboli in numeri, si utilizza una tabella di conversione.

simbolo	numero
A	$(65)_{10}$
B	$(66)_{10}$
C	$(67)_{10}$

Su ogni riga della tabella c'è un simbolo e il numero corrispondente. I numeri sono astrazioni, ma sulla carta possono essere rappresentati in diversi modi equivalenti tra loro: base 2, base 10 o base 16. Il suo valore (e il carattere a cui è associato) è unico.

Nel testo "formattato" il computer può rappresentare graficamente il carattere “A” usando diversi fonti tipografiche (in inglese, font), come Arial, Serif, ecc., ma questo solo se si lavora usando un testo formattato.

Se si usa il blocco note, il testo non è formattato e la rappresentazione grafica non ha nessuna importanza, quello che viene memorizzato è solo il valore numerico del simbolo “A” e questo non dipende dal font. Chi leggerà quel valore numerico, per poterlo interpretare correttamente, dovrà usare la stessa codifica, ma questo non è garantito perché in diversi computer sono in uso diverse codifiche.

La codifica dei caratteri

In passato, per memorizzare i simboli della tastiera, era adottata la tabella di codifica ASCII che conteneva solo 127 elementi (elementi dell'alfabeto inglese e altri simboli), per i quali era sufficiente usare un singolo Byte per ogni carattere. Grazie a questo non si poneva il problema della memorizzazione di numeri con più cifre (big-endian o little-endian).

Nelle seguenti tabelle sono elencati prima i **caratteri non stampabili**, come quello corrispondente al tasto "invio" (return) o "esc", e poi quelli stampabili, come i simboli delle lettere e dei numeri.

Alcuni dei caratteri non stampabili sono di semplice comprensione, come DELETE, che cancella un carattere in avanti, mentre altri sono pensati per essere usati nelle comunicazioni tra dispositivi elettronici, come FORM FEED, che invia alla stampante un comando per passare ad un nuovo foglio.

Questi caratteri vengono chiamati anche caratteri di controllo e si possono ottenere premendo il tasto CTRL insieme ad una lettera della tastiera. Il tasto CTRL viene indicato con l'accento circonflesso ^ (caret).

Nel terminale dei comandi di Linux è possibile usare alcuni di questi caratteri di controllo: ^D effettua il logout dell'utente, ^A sposta il cursore ad inizio riga,...

Dec	Abbrev.	CS	CEC	Descrizione
0	NUL	^@	\0	Null character
1	SOH	^A		Start of Header
2	STX	^B		Start of Text
3	ETX	^C		End of Text
4	EOT	^D		End of Transmission
5	ENQ	^E		Enquiry
6	ACK	^F		Acknowledgment
7	BEL	^G	\a	Bell
8	BS	^H	\b	Backspace
9	HT	^I	\t	Horizontal Tab
10	LF	^J	\n	Line feed
11	VT	^K	\v	Vertical Tab
12	FF	^L	\f	Form feed
13	CR	^M	\r	Carriage return
14	SO	^N		Shift Out
15	SI	^O		Shift In
16	DLE	^P		Data Link Escape
17	DC1	^Q		Device Control 1 (oft. XON)
18	DC2	^R		Device Control 2
19	DC3	^S		Device Control 3 (oft.
20	DC4	^T		Device Control 4
21	NAK	^U		Negative Acknowledgement
22	SYN	^V		Synchronous Idle
23	ETB	^W		End of Trans. Block
24	CAN	^X		Cancel
25	EM	^Y		End of Medium
26	SUB	[^Z		Substitute
27	ESC	^[		Escape
28	FS	^\		File Separator
29	GS	^]		Group separator
30	RS	^^		Record Separator
31	US	^_		Unit Separator
127	DEL	^?		Delete

Dec	Glifo
32	Spazio
33	!
34	"
35	#
36	$
37	%
38	&
39	'
40	(
41	)
42	*
43	+
44	,
45	-
46	.
47	/
48	0
49	1
50	2
51	3
52	4
53	5
54	6
55	7
56	8
57	9
58	:
59	;
60	<
61	=
62	>
63	?
64	@
65	A
66	B

Dec	Glifo
68	D
69	E
70	F
71	G
72	H
73	I
74	J
75	K
76	L
77	M
78	N
79	O
80	P
81	Q
82	R
83	S
84	T
85	U
86	V
87	W
88	X
89	Y
90	Z
91	[
92	\
93	]
94	^
95	_
96	`
97	a
98	b
99	c
100	d
101	e
102	f

Dec	Glifo
105	i
106	j
107	k
108	l
109	m
110	n
111	o
112	p
113	q
114	r
115	s
116	t
117	u
118	v
119	w
120	x
121	y
122	z
123	{
124	|
125	}
126	~
127	DEL

Oggi, per poter adottare anche lettere di alfabeti diversi da quello inglese, come quello italiano, francese, greco o cirillico, sono stati sviluppati nuove codifiche dei caratteri, come, ad esempio: ISO 8859, Windows-1252 e UTF-8.

Vediamo le differenze tra questi tre tipi:

- Lo standard internazionale **ISO 8859** si divide in 16 gruppi di alfabeti. Ad esempio, l'alfabeto dell'europa occidentale si trova nella ISO 8859-1 (detta anche latin1) e nella ISO 8859-15 (che contiene anche il simbolo dell'euro).
- Microsoft, per i suoi sistemi operativi, per il gruppo di lingue dell'europa dell'ovest, ha creato la codifica chiamata **Windows-1252** che è molto simile alla ISO 8859-1, ma rispetto alla quale possiede alcune piccole differenze. Questo causa, anche oggi, l'errata convinzione che Windows-1252 e ISO 8859-1 siano la stessa cosa, ma il simbolo euro e alcune virgolette non corrispondono. Oltre a questo problema, la codifica Windows-1252 viene chiamata nei programmi del sistema operativo Microsoft con il termine **ANSI** (American National Standards Institute), ma in verità, questa codifica non è mai stata uno standard ANSI.
- **Universal Character Set** (UCS) è un insieme di caratteri, che contiene circa 100000 simboli, definiti da un nome e un numero intero. Tale numero può essere usato per memorizzare un carattere nella memoria del computer. La più diffusa codifica usata dai computer per codificare nelle memorie dei calcolatori i caratteri dell'insieme UCS è l'**UTF-8**, ma essendo una codifica che utilizza fino a 4 Byte, si può distingure tra UTF-8 big-endian, little-endian, con o senza BOM.

Gli ultimi sistemi operativi Linux adottano la codifica UTF-8.

Gli ultimi sistemi operativi Microsoft adottano la codifica Windows-1252.

Gli ultimi sistemi operativi Apple adottano la codifica macintosh (Mac OS Roman).

La maggior parte di queste codifiche è compatibile con la codifica ASCII, nel senso che chi le adotta riesce a leggere correttamente anche i vecchi file di testo, creati in ASCII. Grazie a questa compatibilità verso il passato, tutte le codifiche si possono sovrapporre per i primi 127 elementi e per lo stesso motivo la memoria occupata da un "vecchio" carattere è ancora un Byte.

Quando nei file di testo si usano ulteriori caratteri, invece, diventa difficile condividerli tra i tre precedenti sistemi operativi. Come è possibile, allora, che questi tre sistemi navighino sulle pagine internet senza troppe difficoltà? La codifica usata dalle pagine web è quasi sempre specificata al loro interno e il browser deve solo adottare la codifica richiesta.

Ricordiamo, comunque, che sul web è raccomandato l'uso della codifica UTF-8, la stessa usata anche dal sistema operativo Linux.

attività: Aprire l'editor del testo (notepad di Windows) e digitare 123 sul tastierino numerico tenendo premuto il tasto ALT: si possono generare anche simboli che non sono presenti sulla nostra tastiera. Chi usa il sistema operativo GNU Linux, può usare un altro editor di testo come Gedit e, tenendo premuto CRTL + SHIFT, digitare 'u', seguita da un numero esadecimale (da 2 a 8 cifre) (come U+20AC).

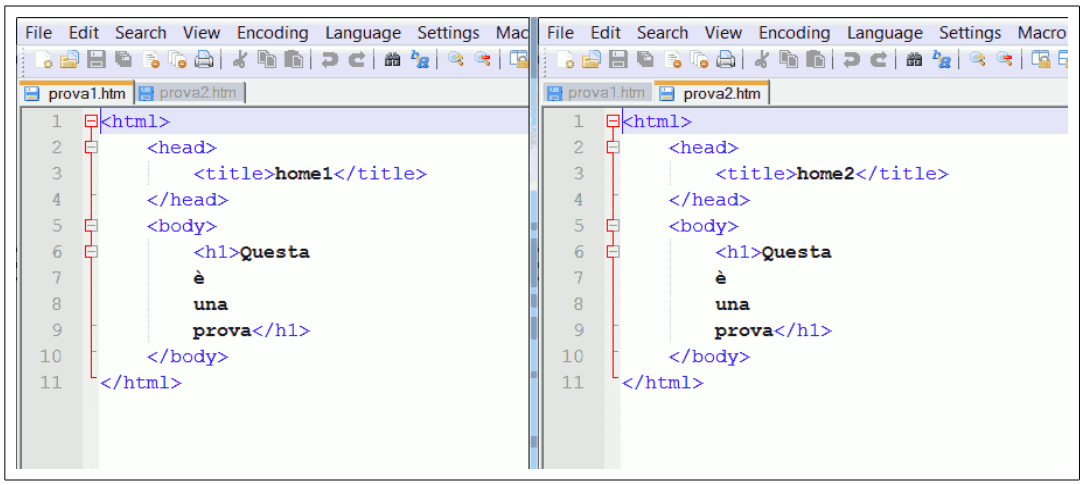
Esistono programmi per generare immagini in ASCII Art

Quando si usa un file di testo formattato, non vi è più il problema della codifica,

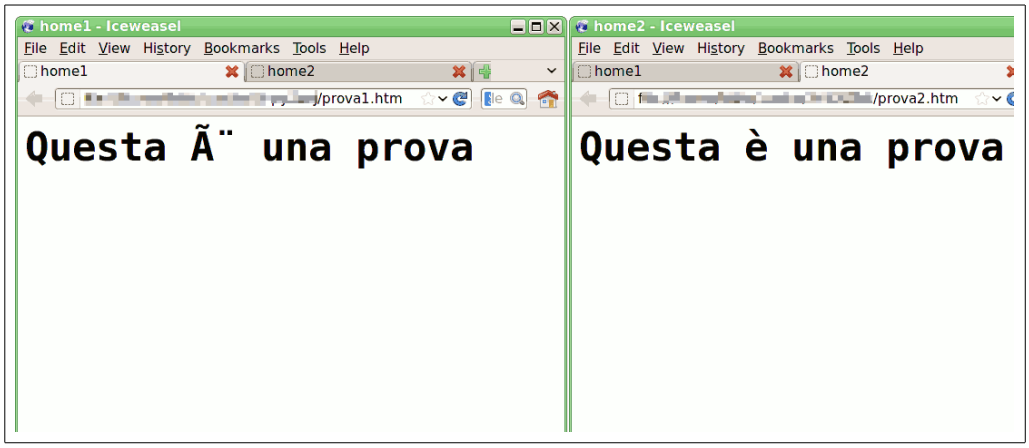
perché di solito questa viene specificata all'interno dello stesso file.

Esempio di cattiva codifica dei caratteri nelle pagine web

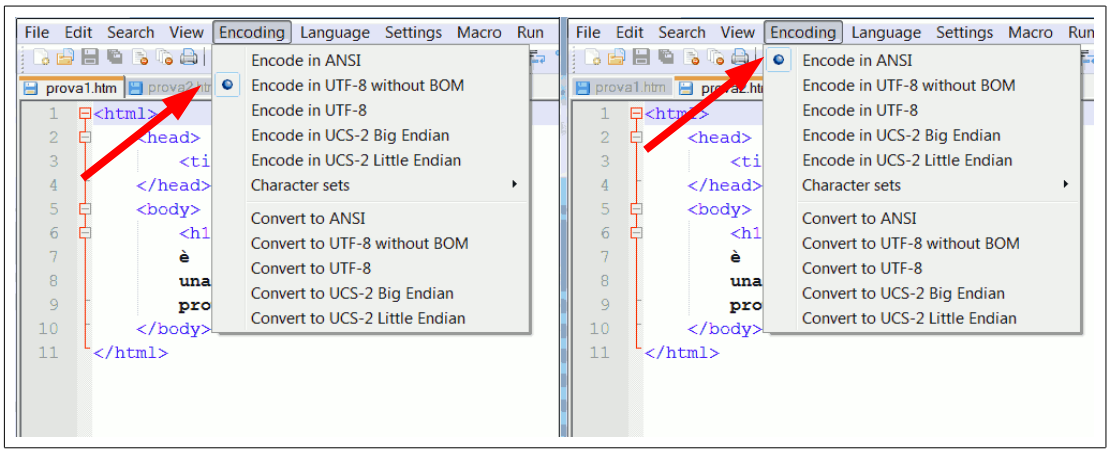
Durante un'esercitazione sono stati aperti (usando Notepad++) due file (prova1.htm e prova2.htm) apparentemente identici nel codice: http://www.illuminamente.org/dokuwiki/lib/exe/fetch.php?media=appunti3s:test.zip



il broser però mostrava queste due pagine con una piccola differenza



Che dipende dalla differente codifica dei caratteri usata durante la realizzazione della pagina e durante il salvataggio



Per evitare questo tipo di problemi nella codifica dei caratteri, si deve aggiungere **sempre** una riga in ogni pagina, per dire al browser la codifica che

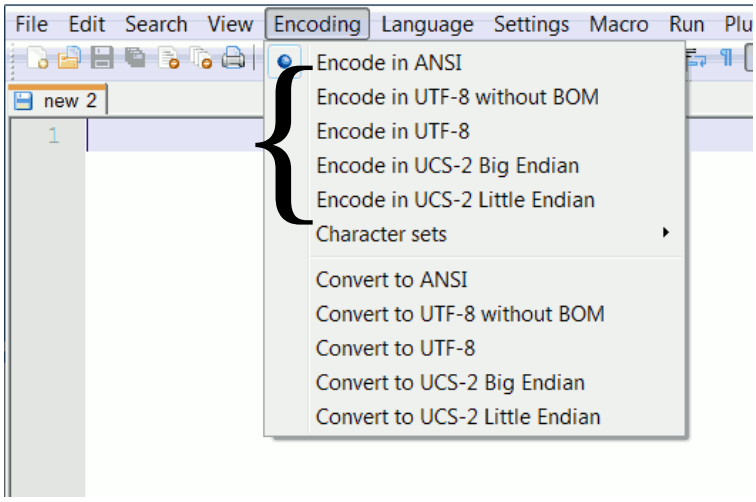
deve usare, usando il tag <meta>.
Ad esempio, in prova1.htm si deve aggiungere (usando HTML 4)

```
<head>
<meta http-equiv="content-type" content="text/html; charset=utf-8" />
</head>
```

mentre, in prova2.htm si deve aggiungere

```
<head>
<meta http-equiv="content-type" content="text/html; charset=Windows-1252"
/>
</head>
```

Questi esempi hanno dimostrato che **prima** di iniziare a lavorare con un file di testo non formattato è importante stabilire quale codifica dei caratteri sarà adottata. La codifica è il meccanismo che trasforma i caratteri in numeri (al salvataggio).



Nel caso delle pagine web è sempre consigliata la codifica UTF-8. Ovviamente, la codifica deve essere dichiarata sempre anche nel tag <meta>.
Come si usa <meta> in HTML 5?...