# Nozioni fondamentali di LaTeX

A differenza dei comuni elaboratori di testi come Word di Microsoft o Writer di LibreOffice, di solito LaTeX non è WYSIWYG (What You See Is What You Get, ‘ciò che vedi è ciò che ottieni’). Con LaTeX, si prende del testo non formattato (plain text) e lo si arricchisce con istruzioni di marcatura. Questo linguaggio di marcatura informa LaTeX sul significato logico di determinati elementi del testo, analogamente a quanto fa HTML.

Prendiamo, per esempio, l’elemento <h2>, che in un documento HTML segnala l’inizio di una nuova sezione. Anche LaTeX ha un comando per fare la stessa cosa: il comando \section.

## Il flusso di lavoro di LaTeX

Dal momento che i file di LaTeX (di solito li riconosci dall’estensione .tex) non costituiscono il documento vero e proprio, ma sono piuttosto istruzioni che informano su che cosa e come ciascuna parte del documento dovrebbe essere, di solito non sono questi a essere inviati ad altre persone. Dopo aver scritto il tuo codice sorgente, invece, lancerai su di esso LaTeX (normalmente adoperando un programma chiamato pdflatex) per generare un file PDF (il documento composto o finale). È questo PDF che poi condividerai con gli altri utenti.

Questo processo viene descritto in modi diversi. Poiché adoperare LaTeX è un po’ come programmare, spesso si dice compilare il documento, sebbene comporre sia un termine più preciso.

## Esecuzioni multiple di LaTeX

Con i file semplici, per ottenere il PDF finale il file sorgente va composto una volta soltanto. Ma appena incominci ad aggiungere cose più complicate, come riferimenti incrociati, citazioni, immagini e indici, potresti dover eseguire LaTeX più di una volta. Te lo diremo quando sarà il caso.

LaTeX o pdfLaTeX o…

Nella [prossima lezione](https://www.learnlatex.org/it/lesson-02) vedremo che LaTeX non è un solo programma. Per semplificare le cose, per creare i tuoi PDF ci concentreremo su un particolare programma LaTeX, pdfLaTeX. Più avanti nel corso vedremo alcuni altri programmi e perché in particolari circostanze potrebbero fare al caso tuo.

# Struttura di un documento LaTeX

Questa lezione spiega la struttura di base di un documento LaTeX, come trasformarlo in un file PDF e i principali caratteri speciali adoperati per controllare LaTeX.

Il tuo primo documento LaTeX sarà molto semplice: l’idea è mostrarti come appare un documento e come scriverlo con successo. È anche la tua prima occasione per vedere [come lavorare con gli esempi](https://www.learnlatex.org/it/help) di learnlatex.org.

Se stai lavorando con un’installazione locale di TeX, crea un nuovo file con il tuo editor e registralo come primo.tex; poi, copia e incolla nel file il codice qui sotto oppure scrivicelo tu direttamente.

Se stai lavorando con un sistema on-line, per provare l’esempio puoi semplicemente premere i bottoni TeXLive.net o Overleaf sotto al codice!

Ti suggeriamo di provare i servizi on-line anche se hai configurato LaTeX localmente: è una buona occasione per vedere come funzionano le due cose.

%%%%%%%%%%%%%% PRIMO %%%%%%%%%%%%%%

\documentclass{article}

\usepackage[T1]{fontenc}

\begin{document}

Ciao, mondo!

Questo è il mio primo documento.

\end{document}

Salva le modifiche e trasformalo in un documento PDF. Se stai lavorando con un’installazione locale di TeX, l’esatto pulsante da premere per comporre il codice dipende dall’editor che hai scelto. Al termine della composizione, dovresti ottenere un file PDF che contiene il testo qui sopra più un numero di pagina: LaTeX lo aggiunge automaticamente.

Visualizza il risultato primo.pdf con il tuo visualizzatore di PDF preferito. Non male, complimenti!

## Cosa fare con gli errori

Gli errori capitano a tutti. Innanzitutto, verifica che ogni riga nel tuo file corrisponda esattamente al testo nell’esempio qui sopra. A volte, modifiche apparentemente esigue al sorgente
producono cambiamenti sensibili nel risultato, compreso l’arresto della composizione. Se vedi che tutto è bloccato, prova a cancellare il documento e a copiare nuovamente il codice dall’esempio.

Se la composizione si arresta mostrando un messaggio che termina con con un punto di domanda, puoi uscirne dando x e Invio.

I messaggi di errore di LaTeX cercano di essere utili, ma non sono dello stesso genere dei messaggi dei comuni elaboratori di testo. In alcuni editor è anche difficile vedere il testo ‘completo’ di un errore, che può nascondere i dettagli chiave per risolvere il problema.

LaTeX crea sempre un registro di ciò che sta facendo: è un file di testo con estensione .log. Potrai sempre vedere i messaggi di errore completi lì e, se hai un problema, gli esperti di LaTeX spesso ti chiederanno una copia di questo file.

Che cos’hai imparato finora

Il primo documento ti ha mostrato le basi del lavoro con LaTeX. I documenti LaTeX sono un misto di testo e comandi. I comandi incominciano con una barra rovescia (\) e a volte hanno un argomento obbligatorio tra parentesi graffe ({...}) o, talvolta, argomenti facoltativi tra parentesi quadre ([...]). Infine, dopo aver lanciato LaTeX sul codice sorgente, otterrai il risultato composto in forma di file PDF .

Ogni documento LaTeX deve contenere la stringa \begin{document} (che indica l’inizio del documento) e una stringa corrispondente \end{document} (che ne indica la fine). Tra questi due comandi si trova il corpo del documento, cioè tutti i contenuti che LaTeX comporrà. Nell’esempio qui sopra, il corpo del documento è formato da due capoversi (in LaTeX, per separare i capoversi basta lasciare una o più righe vuote tra l’uno e l’altro). La parte prima di \begin{document} si chiama preambolo del documento e contiene il codice per impostare l’aspetto del documento. Il comando \usepackage, descritto in una [lezione successiva](https://www.learnlatex.org/it/lesson-06), è adoperato nella maggior parte degli esempi di questo sito per impostare la codifica del font.

La sintassi di LaTeX prevede altre coppie del tipo \begin{...} e \end{...}: vengono chiamate ambienti. Queste coppie di comandi vanno abbinate con grande cura, in modo che per ogni \begin{x} ci sia un \end{x}. Se li annidi (cioè se inserisci un ambiente all’interno di un altro ambiente), allora devi verificare che ogni \end{y} ... \end{x} corrisponda a \begin{x} ... \begin{y}: in altre parole, le dichiarazioni di apertura e chiusura degli ambienti debbono corrispondersi nel medesimo ordine.

LaTeX permette di aggiungere al codice dei commenti, cioè appunti, promemoria e altro testo utile, per esempio, per descrivere un frammento di codice particolarmente complesso, o a qualunque altro scopo. I commenti si scrivono facendoli precedere dal segno di percento %. Vediamoli all’opera per descrivere la struttura dell’esempio seguente:

%%%%%%%%%%%%%% SECONDO %%%%%%%%%%%%%%

\documentclass[a4paper,12pt]{article} % Dichiarazione di classe con alcune opzioni

\usepackage[T1]{fontenc}

% Commento nel preambolo

\begin{document}

% Commento nel corpo del documento

Ecco un semplice

documento\footnote{con una nota nel piè di pagina.}

Questo è un nuovo capoverso.

\end{document}

L’esempio è composto di due capoversi, come si vede dalla riga vuota tra le due frasi. Osserva inoltre (nel PDF composto) che più spazi consecutivi vengono considerati un solo spazio.

Talvolta potrebbe servirti uno spazio ‘indivisibile’ che non si perda tra una riga e l’altra: in LaTeX si ottiene ‘legando’ insieme due parole con ~. La cosa tornerà particolarmente utile quando più avanti nel corso impareremo come si creano i riferimenti incrociati.

## Caratteri speciali

Probabilmente hai notato che i tre caratteri \, { e } hanno per LaTeX un significato speciale. La barra rovescia \ incomincia un comando, cioè un’istruzione impartita a LaTeX. Le parentesi graffe { e } vengono adoperate per racchiudere gli argomenti obbligatori, cioè informazioni richieste obbligatoriamente dai comandi.

Ci sono altri caratteri con un significato speciale: abbiamo appena visto che ~ indica uno spazio ‘indivisibile’, per esempio. Quasi tutti questi caratteri sono molto inconsueti in un testo normale, motivo per cui sono stati scelti per veicolare significati particolari. Se hai bisogno di scrivere uno di questi caratteri come testo normale, puoi scoprire come fare nella [pagina di approfondimento di questa lezione](https://www.learnlatex.org/it/more-03).

## Esercizio

Sperimenta l’editor e il sistema di composizione on-line; premi il pulsante per comporre l’esempio, poi modificalo direttamente sulla pagina e componilo nuovamente.

Aggiungi del testo al tuo primo documento, componilo e osserva i cambiamenti nel PDF. Scrivi ulteriori capoversi e aggiungi spazi variabili. Esplora le funzionalità del tuo editor; clicca su un punto del sorgente e trova il modo di andare alla riga corrispondente del PDF. Inserisci alcuni spazi indivisibili e osserva come influenzano l’interruzione delle righe.

# Struttura logica

Questa lezione mostra alcuni comandi di formattazione di base e li confronta con la formattazione semantica ottenuta con i comandi di sezionamento e gli ambienti per gli elenchi.

LaTeX permette sia di concentrarsi sulla struttura logica del documento, sia di impostarne direttamente l’aspetto. Il più delle volte è meglio mantenere una certa separazione tra contenuto e aspetto, privilegiando una marcatura che descrive la struttura del documento, perché rende più facile riutilizzarne o modificarne l’aspetto al bisogno.

## Struttura e presentazione visuale

Incominciamo con un esempio che confronta due modi per evidenziare un testo: il primo adopera uno dei comandi di marcatura logica più comuni in LaTeX, \emph; il secondo adopera il comando \textit{...}, che stampa il proprio argomento in corsivo. Nei documenti stampati, infatti, il corsivo serve a evidenziare qualcosa.

%%%%%%%%%%%%%% TERZO %%%%%%%%%%%%%%

\documentclass{article}

\usepackage[T1]{fontenc}

\begin{document}

Testo con \emph{enfasi e contenuto \emph{annidato}}.

Testo in \textit{corsivo e contenuto \textit{annidato}}.

\end{document}

Probabilmente avrai già indovinato da te che il comando \textit mette il proprio argomento in corsivo, ma il punto è che lo fa sempre, perciò non funziona se il testo da evidenziare è annidato in un altro testo. Al contrario, \emph tiene conto del contesto ed evidenzia in tondo un testo annidato in un altro testo in corsivo Tieni presente che il corsivo non è l’unico mezzo per evidenziare del testo: per esempio, nelle videopresentazioni è meglio adoperare il colore. Grazie alla marcatura logica, non dovrai preoccuparti di questi dettagli nel corpo del documento.

Parleremo della [formattazione manuale](https://www.learnlatex.org/it/lesson-11) più avanti, ma per ora aggiungiamo ai comandi che già conosci \textbf: mette il proprio argomento in nero.

## Comandi di sezionamento

Probabilmente hai adoperato almeno una volta un elaboratore di testi, con il quale la maggior parte degli utenti per incominciare una sezione scrive il titolo, ne ingrandisce corpo e spessore mettendolo in nero e lo fa seguire da una riga vuota. In LaTeX, la marcatura logica è uno strumento molto più facile di questa formattazione manuale: basta dare il comando \section, che modifica il corpo del font e inserisce lo spazio verticale opportuno prima e dopo il titolo, garantendo un risultato uniforme in tutto il documento.

%%%%%%%%%%%%%% QUARTO %%%%%%%%%%%%%%

\documentclass{article}

\usepackage[T1]{fontenc}

\begin{document}

Ciao, mondo!

Questo è il mio primo documento.

\section{Titolo del primo paragrafo}

Testo del primo paragrafo.

Secondo capoverso.

\subsection{Sottoparagrafo del primo paragrafo}

Testo del sottoparagrafo.

\section{Secondo paragrafo}

Testo del secondo paragrafo.

\end{document}

La configurazione predefinita della classe article prevede paragrafi e sottoparagrafi numerati e con il titolo in nero. [Nella prossima lezione](https://www.learnlatex.org/it/lesson-05) ti mostreremo come cambiare un po’ questo aspetto.

LaTeX è in grado di suddividere i documenti a diversi livelli:

* \chapter (questo comando è ammesso solo con \documentclass{book} o \documentclass{report})
* \section (paragrafo)
* \subsection (sottoparagrafo)
* \subsubsection (sotto-sottoparagrafo)

Ma possiamo scendere a profondità ancora maggiori: il successivo livello verso il basso è \paragraph, anche se la grande maggioranza delle volte non serve essere così dettagliati. (E sì: \paragraph è un comando di sezionamento, e non un modo per incominciare un nuovo capoverso!)

A questo punto, potresti chiederti come si fa a scrivere il titolo di un documento. Esistono alcuni comandi speciali per farlo, ma non tutti i documenti li adoperano, perciò li descriviamo [nella lezione di approfondimento](https://www.learnlatex.org/it/more-04).

# Liste

Un’altra situazione in cui la marcatura logica si rivela vincente è nella composizione delle liste. LaTeX conosce due tipi di liste di base:

%%%%%%%%%%%%%% QUINTO %%%%%%%%%%%%%%

\documentclass{article}

\usepackage[T1]{fontenc}

\begin{document}

Lista ordinata (o numerata)

\begin{enumerate}

 \item Una voce

 \item Un'altra voce

 \item Però! Tre voci

\end{enumerate}

Lista non ordinata (o ‘puntata’)

\begin{itemize}

 \item Una voce

 \item Un'altra voce

 \item Però! Tre voci

\end{itemize}

\end{document}

Osserva che ogni voce dell’elenco incomincia con \item e che il marcatore viene aggiunto automaticamente da LaTeX, a seconda del tipo di lista.

## Esercizi

Sperimenta differenti livelli di sezionamento. Dichiara \documentclass{report} invece di \documentclass{article} e adopera comandi \chapter. Che aspetto hanno le sezioni nel documento composto? Prova \paragraph e (addirittura) \subparagraph per osservarli all’opera: nota che per impostazione predefinita producono titoli non numerati.

Componi alcune liste e annidane una dentro l’altra. Come cambia il formato dei numeri o dei marcatori? Tieni presente che LaTeX permette al massimo quattro livelli di annidamento. Dunque, se ti accorgi di doverne usare di più, è segno che le liste non sono lo strumento migliore per visualizzare i tuoi dati e forse è meglio strutturare diversamente il testo.

# Classi di documento

Questa lezione spiega che cos’è una classe di documento e in che modo può influire sull’aspetto generale di un documento. Inoltre, elenca le principali classi disponibili in una distribuzione di TeX.

Forse hai già notato che tutti i documenti LaTeX creati fino a qui incominciano con la riga \documentclass, e che \documentclass{article} è stata di gran lunga la scelta prevalente. (Nella [lezione precente](https://www.learnlatex.org/it/lesson-04), tuttavia, abbiamo avuto bisogno di nominare \documentclass{report} per introdurre il comando \chapter.) Questa riga, chiamata in gergo dichiarazione di classe, è richiesta in tutti i documenti LaTeX, ed è (quasi) sempre il primo comando che si dovrebbe dare nel preambolo.

Che cosa ‘fa’ una classe di documento

La classe di documento imposta la presentazione generale del documento. Per esempio, si occupa di quanto segue:

* dell’aspetto: margini, font, spaziature, eccetera
* di determinare se ci possono essere i capitoli oppure no
* di determinare se il titolo deve stare in una pagina a sé oppure no

Più in generale, le classi di documento possono definire nuovi comandi rispetto a quelli previsti da LaTeX standard: la cosa è particolarmente vera per le classi molto specialistiche, come quelle destinate alla creazione di videopresentazioni o lettere.

Nella riga della dichiarazione di classe puoi impostare anche le opzioni globali, cioè parametri che si applicano all’intero documento e si scrivono tra parentesi quadre: \documentclass[<opzioni>]{<classe>}. Questa sintassi, con le informazioni facoltative tra parentesi quadre, è comune a numerosi comandi di LaTeX.

## Le classi standard

LaTeX è corredato di una serie di classi standard, il cui aspetto, abbastanza simile nel complesso, si differenzia per gli elementi seguenti:

* article
per brevi documenti senza capitoli
* report
per documenti più lunghi con capitoli, pensati per la stampa su un solo lato del foglio
* book
per documenti più lunghi con capitoli, pensati per la stampa in fronte e retro, con materiale iniziale e finale (come una prefazione e un indice, per esempio)
* letter
per la corrispondenza, senza suddivisione in sezioni
* slides
per videopresentazioni (ma vedi sotto)

Le classi article, report e book mettono a disposizione comandi molto simili, come quelli che abbiamo già visto. Quando si scrive una letter, invece, i comandi sono un po’ diversi:

%%%%%%%%%%%%%% SESTO %%%%%%%%%%%%%%

\documentclass{letter}

\usepackage[T1]{fontenc}

\begin{document}

\begin{letter}{Un destinatario\\Una strada\\Una città}

\opening{Caro signore, cara signora}

Qui ci va il testo della lettera.

\closing{Cordialmente,}

\end{letter}

\end{document}

Osserva come le righe dell’indirizzo sono separate con \\; i modi per interrompere una riga verranno descritti [un po’ più avanti](https://www.learnlatex.org/it/lesson-11). Osserva anche come letter richieda comandi speciali e un nuovo ambiente (letter) per ogni lettera.

Le classi standard article, report e book prendono comunemente due opzioni: 10pt, 11pt e 12pt, per cambiare il corpo del font; e twocolumn, per comporre il documento su due colonne.

## Classi più ricche di funzioni

Le classi principali sono molto stabili, il che le rende anche piuttosto conservative sia nell’aspetto, sia nella varietà di comandi disponibili. Nel tempo, sono state scritte classi più potenti, che permettono di modificare l’aspetto del documento senza dover fare le cose a mano (argomento di cui parleremo [un po’ più avanti](https://www.learnlatex.org/it/lesson-11)).

L’American Mathematical Society fornisce le classi amsart e amsbook, varianti delle classi standard con un aspetto più tradizionale, simile a quello adoperato nelle riviste di matematica.

Le due classi ‘estese’ più grandi e popolari sono la suite KOMA-Script e la classe memoir. KOMA-Script è un insieme di classi ‘parallele’ alle classi standard: scrartcl, scrreprt, scrbook e scrlttr2, mentre la classe memoir, unica, è più simile a un’estensione di book.

Queste classi estese possono essere personalizzate in molti modi, che esploreremo un po’ in un esercizio. Se ti stai chiedendo come facciamo a conoscerli, lo scoprirai da te [in una lezione successiva](https://www.learnlatex.org/it/lezione-16), ma nulla ti vieta di darci un’occhiata anche adesso!

## Videopresentazioni

La classe slides è stata sviluppata verso la metà degli anni ‘80 del Novecento per produrre lucidi da proiezione, e quindi non offre alcuna funzionalità per creare videopresentazioni interattive basate su file PDF. Esistono però classi più recenti che fanno esattamente questo: siccome sono in qualche modo ‘specialistiche’ rispetto ai documenti LaTeX di carattere più generale, le abbiamo [descritte nella lezione di approfondimento](https://www.learnlatex.org/it/more-05).

## Esercizi

Fa’ delle prove con le classi standard, quelle della suite KOMA e memoir, e osserva come cambia l’aspetto del documento.

%%%%%%%%%%%%%% SETTIMO %%%%%%%%%%%%%%

\documentclass{article} % Cambia la classe qui

\usepackage[T1]{fontenc}

\begin{document}

\section{Introduzione}

Questo è un documento d'esempio con del testo

di prova\footnote{e una nota}. Questo capoverso

è piuttosto lungo, dal momento che vogliamo

mostrare l'effetto della composizione su due

colonne.

\end{document}

Aggiungi l’opzione di classe twocolumn e osserva come cambia l’aspetto del documento.

Nell’esempio precedente cambia il comando \section in \chapter e scopri che effetto hanno le opzioni di classe seguenti quando adoperi la classe scrreprt:

* chapterprefix
* headings=small
* headings=big
* numbers=enddot

# Estendere LaTeX

Questa lezione mostra come estendere le funzionalità di LaTeX per soddisfare esigenze compositive particolari e come modificarne ulteriormente il risultato predefinito con i diversi pacchetti e le definizioni di comandi personali.

Dopo aver dichiarato una classe, nel preambolo del documento puoi modificare le funzionalità standard di LaTeX aggiungendo uno o più pacchetti. I pacchetti possono:

* cambiare il modo in cui certe parti di LaTeX funzionano
* modificare l’aspetto del documento
* aggiungere nuovi comandi a LaTeX

## Modificare il funzionamento di LaTeX

L’utente ha ben poche possibilità di personalizzare il ‘kernel’ (o nucleo) di LaTeX, ma grazie ai pacchetti aggiuntivi, previsti dalla naturale modularità del programma, si possono risolvere alcuni problemi comuni. Il primo problema in cui ti imbatterai è la selezione della lingua del documento, alla quale dovrai adattare il comportamento di LaTeX in quanto a sillabazione, punteggiatura, stile delle citazioni, uso delle virgolette, localizzazione, eccetera. Lingue diverse hanno regole diverse, perciò è importante dire a LaTeX quale (o quali) adoperare. Questo aspetto è gestito automaticamente dal pacchetto babel.

%%%%%%%%%%%%%% OTTAVO %%%%%%%%%%%%%%

\documentclass{article}

\usepackage[T1]{fontenc}

%\usepackage[italian]{babel}

\usepackage[width = 6cm]{geometry} % Per ottenere cesure a fine riga

\begin{document}

Questo è un bel po' di testo riempitivo che dimostrerà come LaTeX sillaba

il materiale e che sarà in grado di darci almeno un punto di sillabazione.

Questo è un bel po' di testo riempitivo che dimostrerà come LaTeX sillaba

il materiale e che sarà in grado di darci almeno un punto di sillabazione.

\end{document}

Prova a decommentare la (chiaramente ingannevole) riga che carica babel con l’opzione per l’italiano e osserva il risultato. (Nota che le regole di sillabazione predefinite sono quelle dell’inglese americano.)

A seconda della lingua impostata, il pacchetto babel fa molto più che sillabare semplicemente il testo: se ti servono, ti diamo [alcuni dettagli aggiuntivi](https://www.learnlatex.org/it/more-06).

## Modificare l’aspetto del documento

Spesso è utile sapere come modificare alcuni aspetti del documento indipendentemente dalla classe dichiarata, come i margini della pagina, per incominciare da quelli più ovvi. Nell’esempio qui sopra abbiamo caricato il pacchetto geometry, ma è meglio fare un esempio dedicato espressamente alla modifica dei margini (attenzione: il PDF composto sarà di più pagine).

%%%%%%%%%%%%%% NONO %%%%%%%%%%%%%%

\documentclass{book}

\usepackage[T1]{fontenc}

\usepackage[margin=1in]{geometry}

\begin{document}

Ehi, mondo!

Questo è il mio primo documento.

% ++++++++++++++++

\chapter{Capitolo Uno}

Introduzione al primo capitolo.

% ================

\section{Titolo del primo paragrafo}

Testo del primo paragrafo.

Secondo capoverso.

%-----------------

\subsection{Sottoparagrafo del primo paragrafo}

Testo del sottoparagrafo.

% ================

\section{Secondo paragrafo}

Testo del secondo paragrafo.

\end{document}

Ora commenta la riga che carica geometry e osserva come cambia il documento composto.

## Aggiungere nuove funzionalità

Uno dei punti di forza di LaTeX è la possibilità di scegliere tra migliaia di pacchetti, tra i quali quelli per la scrittura della matematica, per i collegamenti ipertestuali, per funzionalità sofisticate come l’uso dei colori, eccetera. Nelle lezioni successive vedremo all’opera alcuni tra i pacchetti più comuni.

Definire comandi personali

Lavorando con LaTeX, ti potrà succedere di aver bisogno di un comando specifico per il tuo documento, o perché quella funzionalità che ti serve non è contemplata dai pacchetti disponibili, o, più semplicemente, per inserire più facilmente nel testo un’espressione che ricorre con una certa frequenza.

L’esempio seguente mostra come definire un comando che stampa il proprio argomento in uno stile specifico (il nero corsivo, in questo caso).

%%%%%%%%%%%%%% DECIMO %%%%%%%%%%%%%%

\documentclass{article}

\usepackage[T1]{fontenc}

\newcommand\kw[1]{\textbf{\itshape #1}} % definizione del comando \kw

\begin{document}

Qualcosa a proposito di \kw{mele} e \kw{arance}.

\end{document}

Nella definizione, [1] indica il numero degli argomenti del comando (uno, in questo caso) e #1 il primo (e unico) argomento che gli viene passato (mele o arance, in questo esempio). Un comando può prendere fino a nove argomenti, ma di solito è meglio definire comandi a un solo argomento, o talvolta senza del tutto addirittura.

La definizione dei comandi non riduce solo la quantità di caratteri da digitare, ma cosa più utile, aiuta a separare il contenuto dalla forma. Se per qualche motivo a un certo punto si decide di volere le parole chiave in uno stile diverso, invece di modificare tutte le loro occorrenze nel documento, basterà modificarne la definizione nel preambolo. Nell’esempio seguente carichiamo il pacchetto xcolor per aggiungere i colori e adoperiamo il blu anziché il nero.

%%%%%%%%%%%%%% UNDICESIMO %%%%%%%%%%%%%%

\documentclass{article}

\usepackage[T1]{fontenc}

\usepackage{xcolor}

\newcommand\kw[1]{\textcolor{blue}{\itshape #1}}

\begin{document}

Qualcosa a proposito di \kw{mele} e \kw{arance}.

\end{document}

Attenzione, però: definire troppi comandi o definirne con molti argomenti può rendere il file sorgente difficile da leggere, perché la sintassi non è familiare. La possibilità di definire comandi ad hoc per un documento, quindi, dovrebbe essere sfruttata con parsimonia.

## Esercizi

Prova a scrivere del testo in altre lingue europee e osserva come babel influisce sulla sillabazione: puoi trovare del testo a questo scopo su Internet e indovinare le opzioni corrette da passare al pacchetto.

Prova a modificare i margini impostati nell’esempio con geometry. Puoi impostare separatamente i singoli margini top, bottom, left e right dichiarando le opzioni in un elenco e separandole con la virgola.

Prova a caricare il pacchetto lipsum e aggiungi al tuo documento il comando \lipsum. Riesci a indovinare perché questo pacchetto è utile per scrivere esempi?

Prova a modificare la definizione di \kw per ottenere uno stile diverso.

# Inserimento e collocazione sulla pagina di immagini esterne

Questa lezione mostra come inserire immagini esterne nel documento, come cambiarne l’aspetto e come metterle sulla pagina o farle ‘galleggiare’ automaticamente.

Per importare in LaTeX file grafici esterni, devi caricare il pacchetto graphicx, che aggiunge ai comandi di LaTeX il comando \includegraphics, all’opera nell’esempio seguente:

%%%%%%%%%%%%%% DODICESIMO %%%%%%%%%%%%%%

\documentclass{article}

\usepackage[T1]{fontenc}

\usepackage{graphicx}

\begin{document}

Questa figura

\begin{center}

 \includegraphics[height=2cm]{example-image}

\end{center}

è un PDF importato.

\end{document}

In un documento LaTeX puoi inserire file EPS, PNG, JPG, e PDF. Se hai a disposizione più di una versione di uno stesso file grafico, puoi caricarlo specificandone l’estensione: esempio-imagine.png, per esempio. (Se non indichi alcuna estensione, graphicx tenterà di indovinarla.)

Avrai notato che qui abbiamo adoperato un nuovo ambiente, center, per centrare orizzontalmente l’immagine sulla pagina. Diremo qualcosa di più su spaziatura e posizionamento delle immagini [un po’ più avanti](https://www.learnlatex.org/it/lesson-11).

## Modificare l’aspetto di un file grafico

Il comando \includegraphics prevede molte opzioni per controllare la grandezza e la forma delle immagini e per rifilarle. Alcune di esse sono molto usate, per cui vale la pena conoscerle.

La cosa più ovvia da impostare è la width (larghezza) o la height (altezza) di un’immagine, che spesso vengono indicate relativamente alla \textwidth (o alla \linewidth) e alla \textheight (altezza della gabbia del testo). La differenza tra \textwidth e \linewidth è sottile, tanto che spesso il risultato è lo stesso. \textwidth è la larghezza della gabbia del testo sulla pagina fisica, mentre \linewidth è la larghezza corrente, che potrebbe essere localmente diversa dalla prima (tale differenza è più evidente se si è dichiarata l’opzione di classe twocolumn). LaTeX ridimensionerà automaticamente l’immagine rispettandone le proporzioni.

%%%%%%%%%%%%%% TREDICESIMO %%%%%%%%%%%%%%

\documentclass{article}

\usepackage[T1]{fontenc}

\usepackage{graphicx}

\begin{document}

\begin{center}

 \includegraphics[height = 0.5\textheight]{example-image}

\end{center}

Un po' di testo

\begin{center}

 \includegraphics[width = 0.5\textwidth]{example-image}

\end{center}

\end{document}

Puoi anche scalare le immagini con l’opzione scale o ruotarle sulla pagina di un certo angolo con angle. Le altre cose di cui potresti avere bisogno sono il ritaglio (opzione clip) e il rifilo (opzione trim) di un’immagine.

%%%%%%%%%%%%%% QUATTORDICESIMO %%%%%%%%%%%%%%

\documentclass{article}

\usepackage[T1]{fontenc}

\usepackage{graphicx}

\begin{document}

\begin{center}

 \includegraphics[clip, trim = 0 0 50 50]{example-image}

\end{center}

\end{document}

## Far ‘galleggiare‘ le immagini

Tradizionalmente, nella composizione dei documenti e specialmente di quelli tecnici e scientifici, le immagini e le tabelle possono finire più in là nel documento o essere raggruppate in pagine a loro dedicate per riempire meglio lo spazio disponibile e non lasciare grandi vuoti sulle pagine. Immagini e tabelle di questo tipo prendono il nome di oggetti galleggianti (float, in inglese).

%%%%%%%%%%%%%% QUINDICESIMO %%%%%%%%%%%%%%

\documentclass{article}

\usepackage[T1]{fontenc}

\usepackage{graphicx}

\usepackage{lipsum} % genera testo fittizio come riempitivo

\begin{document}

\lipsum[1-4] % vogliamo solo qualche capoverso di riempimento

Prova di collocamento.

\begin{figure}[ht]

 \centering

 \includegraphics[width=0.5\textwidth]{example-image-a.png}

 \caption{Un'immagine d'esempio}

\end{figure}

\lipsum[6-10] % ancora qualche capoverso di riempimento

\end{document}

Qui LaTeX sposta l’immagine e la didascalia da sotto Prova di collocamento all’inizio della seconda pagina, perché nella parte inferiore della prima non c’è abbastanza spazio. L’opzione ht condiziona la posizione in cui LaTeX può mettere l’immagine: con lo specificatore h gli chiediamo di metterla esattamente nel punto in cui sta nel sorgente (se possibile); con lo specificatore t gli chiediamo di metterla nella parte superiore di una pagina. Puoi indicare fino a quattro specificatori di posizione:

* h qui (*here*) se possibile
* t all’inizio (*top*) di una pagina
* b alla fine (*bottom*) di una pagina
* p in una pagina di soli float

[Più avanti](https://www.learnlatex.org/it/lesson-09), vedremo come creare riferimenti incrociati che puntino ai float.

Probabilmente avrai notato che qui abbiamo centrato l’immagine con \centering e non con l’ambiente center. Per centrare orizzontalmente il contenuto di un float, dovresti adoperare \centering, per evitare che center aggiunga dell’ulteriore spazio verticale a quello già inserito dal float.

## Esercizi

Prova a inserire nel documento un’immagine creata da te, sostituendola a quelle ‘standard’ adoperate negli esempi.

Esplora che cosa puoi fare con le chiavi height, width, angle e scale.

Adopera la chiave width per impostare la dimensione di un file grafico relativamente a \textwidth prima e a \linewidth poi. Osserva come si comportano con o senza l’opzione twocolumn.

Adopera lipsum per comporre un esempio ragionevolmente lungo, poi prova a inserire dei float con differenti specificatori di posizione. Come interagiscono tra loro?

# Tabelle

Questa lezione mostra come comporre le tabelle con LaTeX, come impostare l’allineamento delle celle e come unirle, come aggiungere filetti.

In LaTeX, le tabelle si costruiscono nell’ambiente tabular. In questa lezione assumiamo anche il caricamento del pacchetto array, che aggiunge ulteriori funzionalità alle tabelle standard, e che non è integrato nel kernel di LaTeX solo per ragioni storiche. Scrivi dunque la riga seguente nel tuo preambolo e sarai pronto per incominciare:

\usepackage{array}

Per comporre una tabella nell’ambiente tabular, LaTeX ha bisogno di sapere due cose: quante colonne servono e come debbono essere allineate. Questa dichiarazione va fatta in un argomento obbligatorio – spesso chiamato preambolo della tabella – di tabular, nel quale le colonne vengono specificate con nomi di una sola lettera, chiamati preamble-token (token di preambolo). La tabella seguente mostra i tipi di colonna standard disponibili:

|  |  |
| --- | --- |
| Tipo | Descrizione |
| l | colonna allineata a sinistra |
| c | colonna centrata |
| r | colonna allineata a destra |
| p{larghezza} | colonna di larghezza fissa; il testo va a capo automaticamente ed è giustificato |
| m{larghezza} | come p, ma messo in alto rispetto al resto della riga |
| b{larghezza} | come p, ma messo in basso |
| w{allineamento}{larghezza} | produce una colonna di larghezza fissa e allineamento l, c o r; se troppo largo, il contenuto della colonna finisce nel margine. |
| W{allineamento}{larghezza} | come w, ma se il contenuto è troppo largo, LaTeX emette un avviso di overfull box. |

Oltre a quelli appena visti, sono disponibili alcuni altri preamble-token che non definiscono una colonna, ma potrebbero essere ugualmente utili:

|  |  |
| --- | --- |
| Tipo | Descrizione |
| \*{numero}{stringa} | ripete la stringa per numero volte nel preambolo; utile per dichiarare più colonne identich. |
| >{dichiarazione} | mette la dichiarazione prima del contenuto di ogni cella nella colonna successiva (utile, per esempio, per impostare un font differente per la colonna in questione) |
| <{dichiarazione} | mette la dichiarazione dopo il contenuto di ogni cella nella colonna precedente |
| | | aggiunge un filetto verticale |
| @{dichiarazione} | sostituisce lo spazio tra due colonne con la dichiarazione |
| !{dichiarazione} | aggiunge la dichiarazione al centro dello spazio tra due colonne |

Le due tabelle qui sopra mostrano tutti i tipi di colonna disponibili in LaTeX standard e con il pacchetto array. Ulteriori tipi di colonna definiti dai diversi pacchetti vengono presentati nella [pagina di approfondimento](https://www.learnlatex.org/it/more-08) di questa lezione.

Le colonne l, c e r avranno la larghezza naturale della cella più larga. Vanno dichiarate tutte le colonne di cui hai bisogno: se ti servono tre colonne centrate, per esempio, nel preambolo della tabella scriverai ccc (o c c c, dato che gli spazi vengono ignorati).

Nel corpo della tabella, le colonne si separano con il carattere & e si termina una riga con \\.

Ora sai tutto ciò che ti serve per comporre la tua prima tabella. Nota che nel codice seguente & e \\ sono incolonnati. LaTeX non lo richiede, ma è una buona pratica per rendere il sorgente più leggibile.

%%%%%%%%%%%%%% SEDICESIMO %%%%%%%%%%%%%%

\documentclass{article}

\usepackage[T1]{fontenc}

\usepackage{array}

\begin{document}

\begin{tabular}{lll}

 Animale & Cibo & Taglia \\

 cane & carne & media \\

 cavallo & fieno & grande \\

 rana & mosche & piccola \\

\end{tabular}

\end{document}

Se una colonna contiene molto testo, il risultato con i semplici l, c e r non sarà buono. Osserva che cosa succede nell’esempio seguente:

%%%%%%%%%%%%%% DICIASETTESIMO %%%%%%%%%%%%%%

\documentclass{article}

\usepackage[T1]{fontenc}

\usepackage{array}

\begin{document}

\begin{tabular}{cl}

 Animale & Descrizione \\

 cane & Il cane è un membro del genere Canis, che fa parte dei canidi

 simili ai lupi, ed è il carnivoro terrestre più largamente

 diffuso. \\

 gatto & Il gatto è una specie domestica di piccolo mammifero carnivoro.

 È l'unica specie domestica della famiglia Felidae ed è spesso

 indicato come il gatto domestico per distinguerlo dai membri

 selvatici della famiglia. \\

\end{tabular}

\end{document}

Il problema è che il materiale di una colonna l viene stampato su una sola riga e con la sua larghezza naturale, anche se la pagina non è sufficientemente grande. Per evitarlo, al suo posto puoi specificare la colonna p{larghezza}, che sistema il proprio contenuto come un capoverso della larghezza da te indicata e allineato verticalmente in alto (questa impostazione predefinita andrà bene nella grandissima maggioranza delle volte). Confronta il risultato dell’esempio precedente con questo:

%%%%%%%%%%%%%% DICIOTTESIMO %%%%%%%%%%%%%%

\documentclass{article}

\usepackage[T1]{fontenc}

\usepackage{array}

\begin{document}

\begin{tabular}{cp{9cm}}

 Animale & Descrizione \\

 cane & Il cane è un membro del genere Canis, che fa parte dei canidi

 simili ai lupi, ed è il carnivoro terrestre più largamente

 diffuso. \\

 gatto & Il gatto è una specie domestica di piccolo mammifero carnivoro.

 È l'unica specie domestica della famiglia Felidae ed è spesso

 indicato come il gatto domestico per distinguerlo dai membri

 selvatici della famiglia. \\

\end{tabular}

\end{document}

Se la tua tabella contiene molte colonne dello stesso tipo, dichiararle una alla volta nel preambolo può essere farraginoso. È molto più veloce scrivere \*{<numero>}{<specificatore>}, che ripete lo specificatore per numero volte. In altre parole, \*{6}{c} equivale a scrivere cccccc. Per mostrarti come funziona la cosa, ecco di nuovo la prima tabella di questa lezione con la sintassi appena descritta:

%%%%%%%%%%%%%% DICIANNOVESIMO %%%%%%%%%%%%%%

\documentclass{article}

\usepackage[T1]{fontenc}

\usepackage{array}

\begin{document}

\begin{tabular}{\*{3}{l}}

 Animale & Cibo & Taglia \\

 cane & carne & media \\

 cavallo & fieno & grande \\

 rana & mosche & piccola \\

\end{tabular}

\end{document}

## Aggiungere i filetti

Prima di affrontare l’argomento delle linee (o filetti), ti diamo un consiglio: i filetti nelle tabelle andrebbero adoperati con grande parsimonia, evitando il più possibile quelli verticali, perché producono un risultato tipograficamente scadente. Per ottenere tabelle di aspetto professionale, inoltre, è meglio evitare i filetti standard di LaTeX e adoperare al loro posto quelli messi a disposizione dal pacchetto booktabs, motivo per cui lo descriviamo qui come prima cosa. Se vuoi saperne di più sui filetti standard di LaTeX, trovi qualche altra informazione nella [pagina di approfondimento](https://www.learnlatex.org/it/more-08).

booktabs definisce quattro tipi diversi di filetto, ciascuno dei quali si ottiene con un comando. Questi comandi vanno dati come prima cosa all’inizio di una riga di tabella o debbono seguire immediatamente un altro filetto. I tre comandi più adoperati sono \toprule, \midrule e \bottomrule. Il loro nome indica chiaramente dove vanno messi nel codice:

%%%%%%%%%%%%%% VENTESIMO %%%%%%%%%%%%%%

\documentclass{article}

\usepackage[T1]{fontenc}

\usepackage{array}

\usepackage{booktabs}

\begin{document}

\begin{tabular}{lll}

 \toprule % filetto iniziale

 Animale & Cibo & Taglia \\

 \midrule % filetto intermedio

 cane & carne & media \\

 cavallo & fieno & grande \\

 rana & mosche & piccola \\

 \bottomrule % filetto finale

\end{tabular}

\end{document}

Il quarto filetto definito da booktabs è \cmidrule. Può essere adoperato per disegnare una linea che non copre l’intera larghezza della tabella, ma solo un determinato intervallo di colonne, che si indica come un intervallo numerico: {\_\_`-`\_\_`}`. Dovrai indicare un intervallo anche se vuoi che il filetto sia largo una colonna soltanto (in questo caso, i due numeri saranno uguali).

%%%%%%%%%%%%%% VENTUNESIMO %%%%%%%%%%%%%%

\documentclass{article}

\usepackage[T1]{fontenc}

\usepackage{array}

\usepackage{booktabs}

\begin{document}

\begin{tabular}{lll}

 \toprule

 Animale & Cibo & Taglia \\

 \midrule

 cane & carne & media \\

 \cmidrule{1-2} % filetto che copre la prima e la seconda colonna

 cavallo & fieno & grande \\

 \cmidrule{1-1} % filetto che copre la prima colonna

 \cmidrule{3-3} % filetto che copre la terza colonna

 rana & mosche & piccola \\

 \bottomrule

\end{tabular}

\end{document}

\cmidrule presenta un’altra caratteristica utile: puoi accorciarlo a entrambe le estremità con un argomento facoltativo tra parentesi tonde:

%%%%%%%%%%%%%% VENTIDUESIMO %%%%%%%%%%%%%%

\documentclass{article}

\usepackage[T1]{fontenc}

\usepackage{array}

\usepackage{booktabs}

\begin{document}

\begin{tabular}{lll}

 \toprule

 Animale & Cibo & Taglia \\

 \midrule

 cane & carne & media \\

 \cmidrule{1-2}

 cavallo & fieno & grande \\

 \cmidrule(r){1-1}

 \cmidrule(rl){2-2}

 \cmidrule(l){3-3}

 rana & mosche & piccola \\

 \bottomrule

\end{tabular}

\end{document}

Osservando il codice, hai certamente indovinato che le lettere r e l indicano che il filetto va accorciato alla sua estremità destra (right) e sinistra (left) rispettivamente.

A volte un filetto orizzontale può essere una separazione eccessiva tra due righe, ma ti accorgi che la tabella sarebbe più chiara distinguendole comunque in qualche modo. In casi come questo, puoi inserire un piccolo spazio verticale con il comando \addlinespace.

%%%%%%%%%%%%%% VENTITREESIMO %%%%%%%%%%%%%%

\documentclass{article}

\usepackage[T1]{fontenc}

\usepackage{array}

\usepackage{booktabs}

\begin{document}

\begin{tabular}{cp{9cm}}

 \toprule

 Animale & Descrizione \\

 cane & Il cane è un membro del genere Canis, che fa parte dei canidi

 simili ai lupi, ed è il carnivoro terrestre più largamente

 diffuso. \\

 \addlinespace

 gatto & Il gatto è una specie domestica di piccolo mammifero carnivoro.

 È l'unica specie domestica della famiglia Felidae ed è spesso

 indicato come il gatto domestico per distinguerlo dai membri

 selvatici della famiglia. \\

 \bottomrule

\end{tabular}

\end{document}

## Unire le celle

Per unire celle orizzontali c’è il comando \multicolumn, che in una cella va dato come prima cosa. \multicolumn prende tre argomenti:

1. il numero di celle da unire
2. l’allineamento della cella risultante
3. il contenuto della cella risultante

Il secondo argomento può contenere qualsiasi cosa sia ammessa nel preambolo di tabular, ma, fa’ attenzione, puoi metterci un solo tipo di colonna.

%%%%%%%%%%%%%% VENTIQUATTRESIMO %%%%%%%%%%%%%%

\documentclass{article}

\usepackage[T1]{fontenc}

\usepackage{array}

\usepackage{booktabs}

\begin{document}

\begin{tabular}{lll}

 \toprule

 Animale & Cibo & Taglia \\

 \midrule

 cane & carne & media \\

 cavallo & fieno & grande \\

 rana & mosche & piccola \\

 furlong & \multicolumn{2}{c}{sconosciuto} \\

 \bottomrule

\end{tabular}

\end{document}

# Riferimenti incrociati

Questa lezione mostra come creare un riferimento agli elementi numerati di un documento, come immagini, tabelle e sezioni.

Quando si scrive un documento, non importa quanto lungo sia, spesso bisogna fare riferimento ad alcuni suoi elementi numerati come le immagini, le tabelle o le formule matematiche. Per fortuna, LaTeX è in grado di aggiungere automaticamente i numeri corretti: dobbiamo solo imparare qualche semplice comando.

Il meccanismo di \label e \ref

Per fare in modo che LaTeX ricordi un punto preciso nel documento, prima devi ‘etichettarlo’ con una stringa di caratteri univoca e poi potrai riferirti a esso da altri punti del testo.

%%%%%%%%%%%%%% VENTICINQUESIMO %%%%%%%%%%%%%%

\documentclass{article}

\usepackage[T1]{fontenc}

\begin{document}

Ciao, mondo!

Questo è il mio primo documento.

\section{Titolo del primo paragrafo}

Testo del primo paragrafo.

\subsection{Sottoparagrafo del primo paragrafo}

\label{subsec:etichetta-uno}

Testo del primo sottoparagrafo.

\begin{equation}

 e^{i\pi}+1 = 0

\label{eq:etichetta-due}

\end{equation}

Nel sottoparagrafo~\ref{subsec:etichetta-uno}

c'è l'equazione~\ref{eq:etichetta-due}.

\end{document}

L’esempio contiene due comandi \label{...}, uno dopo il sottoparagrafo e l’altro dentro all’ambiente matematico equation, che serve a scrivere formule numerate. Questo comando ‘attacca’ le etichette ai due oggetti. Nell’ultima frase, infine, ci sono i riferimenti
veri e propri, che si ottengono mettendo le etichette degli oggetti nell’argomento del comando \ref{...}.

Quando viene lanciato, LaTeX memorizza le informazioni relative alle etichette in un file ausiliario (con estensione .aux, lo troverai nella cartella di lavoro). Di \label{subsec:etichetta-uno} LaTeX sa che è l’etichetta di un sottoparagrafo, perciò registra nel file ausiliario il numero di questa sezione. Di \label{eq:etichetta-due} LaTeX sa che è l’etichetta di una formula matematica numerata, perché l’ultimo elemento che produce un numero è l’ambiente equation, perciò registra nel file ausiliario il numero della formula. Quando si richiede un riferimento incrociato con \ref, il programma stampa i numeri giusti prendendoli dal file .aux.

Per la verità, LaTeX ignora le stringhe subsec: e eq:, perché non le conosce: sono solo una convenzione per aiutarti a ricordare a che cosa l’etichetta si riferisce. Sei libero di adoperare qualunque altra convenzione. LaTeX tiene solo una traccia di quanto ha elaborato più di recente.

Potrà capitarti di vedere nel PDF composto dei riferimenti incrociati in forma di due punti interrogativi in nero: ??. Ciò accade perché la prima composizione del documento si limita a scrivere le informazioni delle etichette sul file ausiliario senza registrarle. Basta lanciare LaTeX una seconda volta e tutto andrà a posto. (A ogni modo, la cosa non costituisce un problema, perché quando scrivi un documento eseguirai LaTeX parecchie volte comunque.)

Osserva il carattere di tilde (~) prima del comando \ref. Prendi la buona abitudine di mettercelo sempre: inserisce uno ‘spazio indivisibile’ che impedisce a LaTeX di spezzare su due righe un’espressione come paragrafo 3.1, per esempio.

## Dove va messo \label?

Il comando \label si riferisce sempre all’elemento numerato immediatamente precedente: un paragrafo, una formula, un float, eccetera. Ciò significa che \label va dato sempre dopo l’elemento a cui vuoi riferirti. Con i float, in particolare, va dato dopo (o, meglio, dentro) il comando \caption, ma dentro l’ambiente galleggiante.

# Matematica

Questa lezione presenta il modo matematico di LaTeX e insegna come si scrivono le formule in linea e in display, le funzionalità aggiuntive fornite dal pacchetto amsmath e come si selezionano i font nella scrittura della matematica.

La scrittura della matematica, da quella più semplice alle formule più complesse, è uno dei punti di forza di LaTeX. È possibile scrivere formule matematiche con la marcatura logica nel cosiddetto ‘modo matematico’.

Modo matematico

Nel modo matematico, LaTeX ignora gli spazi e applica (quasi sempre) la spaziatura corretta tra i caratteri.

Esistono due forme del modo matematico:

* in linea
* in display (cioè, ‘in mostra’ sulla pagina)

%%%%%%%%%%%%%% VENTISEIESIMO %%%%%%%%%%%%%%

\documentclass{article}

\usepackage[T1]{fontenc}

\begin{document}

Una frase con della matematica in linea: $y = mx + c$.

Un'altra frase con della matematica in linea: $5^{2}=3^{2}+4^{2}$.

Un secondo capoverso contenente della matematica in display.

\[

 y = mx + c

\]

Osserva in che modo il capoverso continua dopo il display.

\end{document}

## Modo matematico in linea e notazione matematica

Come puoi vedere nell’esempio qui sopra, il modo matematico in linea è delimitato da una coppia di dollari $...$. Al suo posto si può anche adoperare la notazione equivalente \( ... \). Le espressioni semplici si scrivono nel sorgente senza bisogno di marcature particolari. Osserva come nel documento finale i vari elementi delle formule sono ben distanziati tra loro e le lettere vengono messe automaticamente in corsivo.

Nella matematica in linea, la dimensione verticale delle formule è limitata in modo da non disturbare, per quanto possibile, l’avanzamento di riga del capoverso in cui compaiono.

Nota bene che tutta la matematica dovrebbe essere inserita nel sorgente come tale (tra dollari, cioè) anche se si tratta di un solo carattere: dunque, è bene scrivere ... $2$ ... e non ... 2 .... In caso contrario, dovendo scrivere un numero negativo in modo matematico per ottenere il corretto segno meno, per esempio, la scrittura ... $-2$ ... potrebbe adoperare caratteri matematici diversi da quelli usati per le altre cifre presenti nel testo (a seconda della classe del documento). Invece, fa’ molta attenzione a quando copi simboli matematici da una fonte terza: i simboli di valuta come $ o i nomi di file nei quali è presente \_ produrranno altrettanti errori: dovrai correggerli scrivendo rispettivamente \$ e \\_.

Si possono aggiungere facilmente esponenti e pedici marcandoli con ^ e \_ rispettivamente.

%%%%%%%%%%%%%% VENTISETTESIMO %%%%%%%%%%%%%%

\documentclass{article}

\usepackage[T1]{fontenc}

\begin{document}

Un esponente $a^{b}$ e un pedice $a\_{b}$.

\end{document}

(Qua e là nella rete troverai facilmente esempi in cui esponenti e pedici semplici sono scritti senza parentesi graffe, ma questa non è la sintassi ufficiale e potrebbe non funzionare: per non sbagliare, mettile sempre.)

Esistono numerosissimi comandi per scrivere la matematica specialistica. Il significato di alcuni di essi si indovina abbastanza facilmente: per esempio, \sin per il seno, \log per il logaritmo o \theta per la corrispondente lettera greca.

%%%%%%%%%%%%%% VENTOTTESIMO %%%%%%%%%%%%%%

\documentclass{article}

\usepackage[T1]{fontenc}

\begin{document}

Un po' di matematica: $y = 2 \sin \theta^{2}$.

\end{document}

## Matematica in display

Gli identici comandi adoperati per la matematica in linea possono essere adoperati per mostrarla in display. Per impostazione predefinita, un display matematico viene composto al centro della pagina ed è pensato per equazioni più grandi che sono ‘parte di un capoverso’. Tieni presente che siccome gli ambienti matematici in display non possono contenere la fine di un capoverso, non sono ammesse righe vuote al loro interno.

Il capoverso dovrebbe incominciare sempre prima del display, matematico, quindi non lasciare una riga vuota prima. Se hai bisogno di più righe di matematica, non adoperare display matematici consecutivi (la cosa produce una spaziatura incoerente), ma uno degli ambienti matematici ‘multiriga’ come align del pacchetto amsmath, descritto più avanti.

I display matematici sono particolarmente utili per formule contenenti simboli voluminosi come gli integrali, come si vede nell’esempio seguente:

%%%%%%%%%%%%%% VENTINOVESIMO %%%%%%%%%%%%%%

\documentclass{article}

\usepackage[T1]{fontenc}

\begin{document}

Un capoverso che riguarda una formula voluminosa:

\[

\int\_{-\infty}^{+\infty} e^{-x^2} \, dx = \sqrt{\pi}

\]

\end{document}

Spesso nei documenti si richiedono formule numerate, cosa che si può fare automaticamente con l’ambiente equation. Riproviamo di nuovo con il solito esempio:

%%%%%%%%%%%%%% TRENTESIMO %%%%%%%%%%%%%%

\documentclass{article}

\usepackage[T1]{fontenc}

\begin{document}

Un capoverso che riguarda una formula voluminosa:

\begin{equation}

\int\_{-\infty}^{+\infty} e^{-x^2} \, dx

\end{equation}

\end{document}

# Il pacchetto amsmath

La notazione matematica è molto ricca, il che significa che gli strumenti integrati nel nucleo di LaTeX non possono coprire tutte le necessità. Il pacchetto amsmath estende il supporto di base per soddisfare numerose esigenze. La [documentazione di amsmath](http://texdoc.org/pkg/amsmath) contiene molti più esempi di quanti non possiamo mostrarne in questa lezione.

%%%%%%%%%%%%%% TRENTUNESIMO %%%%%%%%%%%%%%

\documentclass{article}

\usepackage[T1]{fontenc}

\usepackage{amsmath}

\begin{document}

Risolvi la seguente ricorrenza per $n,k\geq 0$:

\begin{align\*}

 Q\_{n,0} &= 1 \quad Q\_{0,k} = [k=0]; \\

 Q\_{n,k} &= Q\_{n-1,k}+Q\_{n-1,k-1}+\binom{n}{k}, \quad\text{for $n$, $k>0$.}

\end{align\*}

\end{document}

L’ambiente align\* allinea le equazioni alle ‘e commerciali’ (&) proprio come in una tabella. Osserva come abbiamo adoperato \quad per inserire un po’ di spazio e \text per inserire del testo normale in modo matematico. Qui c’è anche un altro comando matematico: \binom, per un binomio.

La maggior parte degli ambienti matematici numera le formule per impostazione predefinita. Le loro varianti con asterisco (che si ottengono scrivendo un \* subito dopo il nome dell’ambiente, come abbiamo fatto qui con align\*), invece, disabilitano la numerazione.

Il pacchetto amsmath definisce anche molti altri ambienti utili, come quelli per le matrici.

%%%%%%%%%%%%%% TRENTADUESIMO %%%%%%%%%%%%%%

\documentclass{article}

\usepackage[T1]{fontenc}

\usepackage{amsmath}

\begin{document}

Matrici di AMS.

\[

\begin{matrix}

 a & b & c \\

 d & e & f

\end{matrix}

\quad

\begin{pmatrix}

 a & b & c \\

 d & e & f

\end{pmatrix}

\quad

\begin{bmatrix}

 a & b & c \\

 d & e & f

\end{bmatrix}

\]

\end{document}

Font in modo matematico

A differenza di quanto accade nel testo normale, nel modo matematico i cambiamenti dello stile del font veicolano in genere un significato molto specifico, il che richiede la grande maggioranza delle volte di scriverli esplicitamente. A questo punto, hai bisogno di conoscere una serie di comandi:

* \mathrm: romano (tondo)
* \mathit: corsivo, spazieggiato come testo normale
* \mathbf: nero
* \mathsf: senza grazie
* \mathtt: a spaziatura fissa (tipo macchina da scrivere)

Tutti questi comandi prendono come argomento lettere dell’alfabeto latino. Per scrivere una matrice, dunque, possiamo fare così:

%%%%%%%%%%%%%% TRENTATREESIMO %%%%%%%%%%%%%%

\documentclass{article}

\usepackage[T1]{fontenc}

\begin{document}

La matrice $\mathbf{M}$.

\end{document}

Nota che per impostazione predefinita il corsivo matematico spazieggia le lettere, perché implicitamente le considera come un prodotto di variabili. Per scrivere una parola in corsivo, invece, devi adoperare \mathit.

I comandi per i font matematici del tipo \math.. caricano font specifici per la matematica. A volte, tuttavia, è necessario incorporare nella formula del testo che appartiene alla struttura esterna della frase e richiede perciò il carattere del testo corrente. Per farlo, puoi adoperare il comando \text{...} (richiede amsmath) o stili di carattere specifici come \textrm{..}.

%%%%%%%%%%%%%% TRENTAQUATTRESIMO %%%%%%%%%%%%%%

\documentclass{article}

\usepackage[T1]{fontenc}

\usepackage{amsmath}

\begin{document}

$\text{testo corrente } corrente \neq \mathit{corrente} \neq \mathrm{corrente}$

\textit{$\text{testo corrente } corrente \neq \mathit{corrente} \neq \mathrm{corrente}$}

\end{document}