

# Noțiuni elementare

Acest tutorial are rolul de a vă familiariza cu structura de bază a [LaTeX](#)-ului.

Înainte de a începe, asigurați-vă că aveți LaTeX instalat pe calculator (vezi [Instalare](#) pentru instrucțiuni legate de instalare și programele necesare). Vom începe prin a crea un fișier sursă LaTeX, iar apoi vom trece prin etapele necesare pentru a produce un format al rezultatului de calitate, de genul postscript (PS) sau PDF.

## Codul sursă LaTeX

LaTeX folosește un limbaj de markup pentru a descrie structura și prezentarea documentului. LaTeX transformă textul sursă, combinat cu markup, într-un document de calitate ridicată. Pentru a face o comparație, paginile web funcționează într-o manieră similară: textul scris sub formă de HTML este folosit pentru a descrie documentul, însă browserul este acela care îl prezintă cu toate trăsăturile generate de markup - culori, fonturi, dimensiuni diferite, etc.

Inputul pentru LaTeX este un fișier text ASCII simplu. Puteți să creați un astfel de fișier cu orice editor de texte. Acesta conține textul documentului, ca și comenzile ce indică LaTeX-ului cum să formateze textul.

Dacă sunteți nerăbdători să vedeți cum merge, un exemplu minimal arată cam în felul următor (comenzile vor fi explicate mai târziu):

```
\documentclass{article}

\begin{document}
Salut, lume!
\end{document}
```

## Spații

"Spațiile albe" (*whitespace*), precum spațiu sau tab, sunt tratate uniform ca "spații" în LaTeX. Mai multe spații albe consecutive sunt tratate ca un singur "spațiu". Spațiul alb la începutul unei linii este în general ignorat, iar un singur sfârșit de linie este tratat ca "spațiu alb". O linie ce nu conține nimic sau doar spații albe între două linii de text definește sfârșitul unui paragraf.

Mai multe linii goale sunt tratate la fel ca și una singură. Textul de mai jos constituie un exemplu. În partea stângă este textul din fișierul de intrare, iar în partea dreaptă se află rezultatul formatat.

Nu contează dacă scrii  
unul  
sau mai multe  
spații  
după un cuvânt.

O linie goală marchează  
începutul unui nou  
paragraf.

Nu contează dacă scrii unul sau  
mai multe spații după un cuvânt.

O linie goală marchează începutul  
unui nou paragraf.

## Caractere speciale

Simbolurile următoare sunt caractere rezervate care fie au o semnificație specială în LaTeX, fie nu sunt disponibile pentru toate fonturile. Dacă le scrieți așa cum sunt în text, în mod normal nu se vor tipări, ci vor determina programul LaTeX să facă lucruri pe care n-ați avut intenția să le faceți.

# \$ % ^ & \_ { } ~ \

După cum veți vedea, aceste caractere pot fi folosite în documente așa cum sunt scrise dacă adăugați un prefix backslash:

`\# \ $ \% \textasciicircum{} \& \_ \{ \} \~{} \textbackslash{}`

Celelalte simboluri și multe altele pot fi tipărite cu comenzi speciale în formule matematice sau ca accente.

Caracterul backslash `\nu` poate fi scris prin adăugarea unui alt backslash în fața lui (`\\`); această secvență este folosită pentru a marca sfârșitul de linie. Pentru a introduce un backslash în modul matematic (*math mode*), puteți folosi în schimb `\backslash`.

Comanda `\~` produce un caracter tilda care este plasat pe litera următoare. Spre exemplu, `\~n` produce ñ. Pentru a produce doar caracterul `~`, folosiți `\~{}`, care pune un `~` peste o căsuță goală.

În mod asemănător, comanda `\^` pune o căciuliță peste caracterul următor, spre exemplu `\^{o}` produce ô. Dacă trebuie să afișați în text simbolul `^`, va trebui să folosiți comanda `\textasciicircum`.

Dacă vreți să inserați text ce ar putea conține anumite simboluri particulare (ca în cazul URIs), puteți folosi comanda `\verb`, care va fi discutată mai târziu, în secțiunea [Formatare](#) (inserează textul așa cum este scris).

## Comenzi LaTeX

Comenzile LaTeX sunt case sensitive, având unul din următoarele două formate:

- Încep cu un backslash `\`, iar apoi au un nume ce conține numai litere. Denumirile comenzilor se termină cu un spațiu, un număr sau cu orice alt caracter ce nu este literă.
- Constau într-un backslash `\` și exact un caracter ce nu este literă.

Unele comenzi solicită un argument, care trebuie specificat între acolade `{ }` după numele comenzii. Unele comenzi suportă parametri opționali, care se adaugă după denumirea comenzii în paranteze drepte `[ ]`. Sintaxa generală este:

```
\numecmd[optiune1,optiune2,...]{argument1}{argument2}...
```

## Medii LaTeX

*Environments* (mediile) în LaTeX au un rol foarte asemănător comenzilor, însă au de obicei efect asupra unei părți mai mari a documentului. Sintaxa acestora este:

```
\begin{numemediu}  
text ce va fi afectat  
\end{numemediu}
```

Între `\begin` și `\end` puteți plasa oricâte alte comenzi sau medii imbricate doriți. În general, mediile pot accepta la rândul lor argumente, însă această trăsătură nu prea este folosită, astfel că o vom discuta numai în părțile mai avansate ale documentului.

Orice lucru în LaTeX poate fi exprimat sub formă de comenzi și medii.

## Comentarii

Când LaTeX întâlnește un caracter % în timp ce procesează un fișier de intrare, ignoră restul liniei curente, sfârșitul de linie, și toate spațiile albe de la începutul liniei următoare.

Comentariile pot fi folosite pentru a scrie note în fișierul de intrare, care nu vor face parte din versiunea tipărită.

```
Acesta este un  
exemplu: % stupid  
% Mai bine:  
instructiv <----  
    Supercal%  
        ifragilist%  
icexpialidocious
```

```
Acesta este un exemplu:  
Supercalifragilisticexpialidocious
```

Caracterul % poate fi de asemenea folosit pentru a împărți liniile lungi de intrare, în care nu se permit spații albe sau caractere de sfârșit de linie.

## Structura fișierului de intrare

Când LaTeX procesează un fișier de intrare, se așteaptă ca acesta să aibă o anumită structură. Astfel, orice fișier de intrare trebuie să înceapă cu comanda

```
\documentclass{...}
```

Aceasta specifică ce tip de document intenționați să scrieți. După aceasta, puteți include comenzi ce determină stilul întregului document, sau puteți încărca pachete care să adauge noi trăsături sistemului LaTeX. Pentru a încărca un astfel de pachet, puteți folosi comanda

```
\usepackage{...}
```

Când ați terminat cu aceste setări ale documentului, începeți să scrieți textul efectiv al documentului cu comanda

```
\begin{document}
```

Acum scrieți textul împreună cu unele comenzi LaTeX utile. La sfârșitul documentului, adăugați comanda

```
\end{document}
```

care spune sistemului LaTeX să termine de analizat documentul. Orice urmează acestei comenzi va fi ignorat de LaTeX. Zona dintre `\documentclass` și `\begin{document}` se numește preambul (*preamble*).

## O sesiune tipică din linia de comandă

LaTeX nu are el însuși o interfață grafică (*GUI - graphical user interface*), de vreme ce este doar un program care prelucrează fișierele de intrare și produce un fișier DVI sau PDF. Unele sisteme LaTeX au o interfață grafică, unde puteți compila un fișier LaTeX printr-un simplu clic pe un buton corespunzător. În alte sisteme, s-ar putea să fie nevoie să scrieți câteva comenzi la o consolă, așadar iată cum puteți determina sistemul LaTeX să vă compileze fișierele de intrare dacă aveți doar o interfață text. Vă rugăm să notați că această descriere presupune că aveți deja instalat un sistem LaTeX pe calculator.

1. Editați/Creați fișierul de intrare (sursă) LaTeX. Acest fișier trebuie să fie text scris numai cu caractere ASCII. În Unix, toate editoarele de texte vor crea numai așa ceva. În Windows, ar fi bine să salvați fișierul în format ASCII sau *Plain Text*. Când alegeți un nume pentru fișier, aveți grijă să aibă extensia `.tex`.
2. Rulați fișierul de intrare în sistemul LaTeX. Dacă ați realizat compilarea cu succes, veți obține un fișier de ieșire cu extensia `.dvi`. S-ar putea să trebuiască să rulați comanda LaTeX de compilare de mai multe ori pentru a include cuprinsul (tabla de materii) și toate referințele interne. Când fișierul sursă are o eroare, sistemul LaTeX va descrie această eroare și va opri procesarea fișierului.

Tipăriți CTRL-D ca să reveniți la linia de comandă.

```
latex foo.tex
```

Acum puteți vizualiza fișierul DVI. Într-un sistem Unix cu X11, puteți scrie la linia de comandă `xdvi foo.dvi`, în Windows puteți folosi un program numit *yap* (*yet another previewer* - încă un program de previzualizare).

(*Evince* și *Okular*, programele standard de vizualizare a documentelor pentru numeroase distribuții Linux, permit afișarea fișierelor DVI.)

Puteți urma o procedură similară cu `pdflatex` pentru a produce un document PDF din sursa inițială `.tex`. La fel ca mai înainte, scrieți la consolă comanda:

```
pdflatex foo.tex
```

Acum puteți vizualiza fișierul PDF, `foo.pdf`.

## Primul nostru document

Acum suntem în măsură să creăm primul nostru document. Vom scrie numai minimul absolut necesar pentru a produce un rezultat; binecunoscuta abordare **Salut, lume!** (*Hello, world!*) este foarte potrivită aici.

- Deschideți editorul favorit de texte. Dacă folosiți [Vim](#) sau [Emacs](#), și acestea colorează textul corespunzător sintaxei LaTeX, lucru care vă va ajuta să scrieți fișierele.
- Copiați textul următor în editorul de texte. Aceasta este sursa LaTeX:

```
% salut.tex - Primul nostru exemplu LaTeX!  
\documentclass{article}  
\begin{document}  
Salut, lume!  
\end{document}
```

- Salvați fișierul cu numele `hello.tex`.

## Ce înseamnă toate astea?

% salut.tex - Primul  
nostru exemplu LaTeX!

Prima linie este un *comentariu* deoarece începe cu simbolul procent (%); când LaTeX vede asta, pur și simplu ignoră restul liniei. Comentariile sunt utile pentru a explica anumite părți ale fișierului sursă. Spre exemplu, puteți pune informații despre autor și dată, sau despre orice altceva.

\documentclass{article}

Această linie este o comandă ce spune sistemului LaTeX să folosească clasa de documente

	<code>article</code> (articol). O clasă de documente definește formatarea, care în acest caz este un format generic de articol. Înfățișarea întregului document se poate schimba ușor, punând altă clasă în loc de articol.
<code>\begin{document}</code>	Această linie marchează începutul mediului numit <code>document</code> ; anunță sistemul LaTeX că urmează să înceapă conținutul documentului. Orice este scris înainte de această comandă este cunoscut în general sub numele de preambul ( <i>preamble</i> ).
Salut, lume!	Aceasta este de fapt singura linie ce are un conținut real - textul pe care vrem să-l afișăm pe pagină.
<code>\end{document}</code>	Mediul <code>document</code> se termină aici. Această comandă transmite sistemului LaTeX că sursa documentului este completă, orice lucru scris în fișier după aceasta fiind ignorat.

După cum am spus mai devreme, fiecare comandă LaTeX începe cu un backslash (`\`). Acesta este modul în care sistemul LaTeX este anunțat ca oricând vede un backslash, să se aștepte la o comandă. Comentariile nu sunt considerate comenzi, de vreme ce tot ceea ce transmit sistemului LaTeX este să ignore linia curentă. Comentariile nu afectează niciodată documentul rezultat.

## Generarea documentului

În mod sigur nu va fi cel mai grozav document pe care l-ați văzut vreodată, însă vrem să-l vedem oricum. Presupun că sunteți la linia de comandă, deja în directorul în care se găsește `salut.tex`.

1. Dați comanda: `latex salut` (extensia `.tex` nu este necesară, deși puteți s-o includeți dacă vreți).
2. Se vor afișa diverse informații despre fișierul sursă LaTeX și despre progresul compilării. Dacă totul a mers bine, ultimele două linii afișate la consolă vor fi:

```
Output written on salut.dvi (1 page, 228 bytes).
Transcript written on salut.log.
```

Asta înseamnă că fișierul sursă a fost procesat și documentul rezultat este *hello.dvi*, care are o pagină și ocupă 228 octeți. În felul acesta, ați creat fișierul DVI, însă cu același fișier sursă puteți crea un document PDF. Pașii de urmat sunt exact aceiași ca mai devreme, însă trebuie să înlocuiți comanda `latex` cu `pdflatex`:

1. Dați comanda: `pdflatex salut` (ca mai înainte, extensia `.tex` este opțională).
2. Se vor afișa diverse informații despre fișierul sursă LaTeX și despre progresul compilării. Dacă totul a mers bine, ultimele două linii afișate la consolă vor fi:

```
Output written on salut.pdf (1 page, 12618 bytes).  
Transcript written on salut.log.
```

Observați că documentul PDF este mult mai mare decât fișierul DVI, chiar dacă conține exact aceleași informații. Principalele diferențe între formatele DVI și PDF sunt următoarele:

- **DVI** are nevoie de mai puțin spațiu pe disc și se creează mai rapid. Nu include fonturile din document, astfel că dacă vreți ca documentul să fie vizualizat corespunzător pe alt calculator, trebuie să aveți toate fonturile necesare instalate. Nu suportă nici un fel de interactivitate cum sunt hyperlinkurile sau imaginile animate. Programele de vizualizare pentru DVI nu sunt foarte întâlnite, astfel încât puteți să aveți în vedere utilizarea unui atare program pentru a previzualiza documentul pe măsură ce-l scrieți.
- **PDF** necesită mai mult spațiu pe disc și se generează mai lent, însă include toate fonturile necesare din document, astfel că nu veți avea nicio problemă de portabilitate. Oferă suport pentru hyperlinkuri interne și externe. Suportă de asemenea facilități tipografice avansate: [hanging punctuation](#), extinderea fonturilor și ajustarea marginilor (*margin kerning*), aducând mai multă flexibilitate sistemului TeX și un aspect mai plăcut. În prezent este standardul *de facto* pentru documentele tipărite sau publicate, așa că-l puteți avea în vedere pentru versiunea finală a documentului.

Până acum, ați văzut că puteți crea atât documente DVI, cât și PDF din aceeași sursă. Asta e adevărat, însă devine ceva mai complicat dacă vreți să introduceți imagini sau linkuri. Aceste lucruri vor fi explicate în detaliu în capitolele următoare, însă pe moment să zicem că putem compila în ambele formate, DVI și PDF, fără probleme.



Notăți, în acest exemplu, că datorită simplității fișierului, este suficient să rulați comanda de compilare LaTeX o dată. Cu toate acestea, dacă începeți să creați documente complexe, inclusiv bibliografii și referințe, etc, comanda LaTeX trebuie să fie executată de mai multe ori pentru a rezolva referințele. Însă aceste lucruri vor fi discutate pe parcurs, pe măsură ce se ivesc.