

## Introdução ao LaTeX: os primeiros passos

Em 1978, [Donald Knuth](#) desenvolveu um sistema de compilação textual chamado TeX (lê-se *téc*). Mais tarde, nos anos 80, [Leslie Lamport](#) transformou o sistema TeX em uma linguagem *markup* chamada [LaTeX](#) (lê-se, geralmente, *lêi-téc*).

Basicamente, trata-se de uma “linguagem” em que você trabalha com primitivos, funções e comandos, que então são transformados (**compilados**) em um arquivo **pdf**. Ou seja, você tem um “código” que é transformado em um artigo, por exemplo, falo de artigo porque geralmente é o tipo de texto que mais produzimos no mundo acadêmico. No futuro, falarei de *pôster* e *slides* em LaTeX.

Em Word, você escreve e visualiza o seu texto na mesma janela — de forma intuitiva. Quando você quer sublinhar uma palavra, basta usar o mouse ou um atalho no teclado. Em LaTeX, você lida com **dois ambientes**: o código-fonte e o “produto final”. Ou seja, conteúdo e forma são duas coisas **separadas** em LaTeX. Assim, todo resultado visto na **forma** vem de um comando específico adicionado ao **conteúdo**. Isso pode parecer trabalhoso, mas o resultado final é infinitamente superior a editores de texto comuns.

O Word é excelente para documentos simples. Contudo, quando você precisa escrever artigos longos, utilizar caracteres especiais,

fazer referências entre figuras e texto, entre outras coisas, o sistema da Microsoft deixa a desejar. Existem diversas maneiras de aprimorar o uso do aplicativo, mas certas coisas simplesmente não ficam boas em Word. Por essa razão, o mundo científico/acadêmico utiliza LaTeX como *standard*.

Neste artigo, falarei do (bem) básico, para que em artigos futuros seja mais fácil acompanhar os temas específicos. Vamos lá!

## 1. Instale o LaTeX ([clique aqui para baixar](#))

O pacote LaTeX é gratuito, e virá com alguns arquivos. O arquivo que nos interessa agora se chama **TeXShop**, o editor *default* do pacote. Esse editor fará as duas coisas de que precisamos: servirá para produzirmos o **conteúdo** e, também, para gerar o arquivo final em pdf — a **forma**. Você não precisa necessariamente utilizar o TeXShop: há diversas opções (pagas e gratuitas) online. Pessoalmente, acho que há opções melhores, mas esse assunto será abordado em um artigo futuro.

## 2. Abra o TeXShop

Você sentirá um “vazio”: não há botões de formatação, fonte, parágrafo ou qualquer outro. Em vez disso, há apenas uma tela vazia, com um ou dois botões no topo. Na versão para Mac, à esquerda, você verá o botão **Typeset**. Esse será o único botão de que você precisará, acredite. Você produzirá um **conteúdo** e, quando estiver pronto, compilará esse conteúdo (Typeset) em um arquivo pdf (“**forma**”). “*Compile*” significa “*transformar código em*

um arquivo pdf”.

### 3. Produzindo um documento em LaTeX

**Salve** o arquivo em branco com algum nome qualquer: não é possível compilar algo que não está salvo. O formato utilizado pelo LaTeX é **\*.tex**.

A primeira coisa que você precisa fazer é especificar que tipo de documento deseja. Neste caso, vamos escrever um artigo. Para isso, a primeira linha do documento será:

```
1 \documentclass{article}
```

Perceba que há uma barra invertida (\) antes de “documentclass”.



A BARRA INVERTIDA É ESSENCIAL – TODA FUNÇÃO E TODO COMANDO EM LATEX É INICIADO COM UMA BARRA INVERTIDA (\).



Imagine que “documentclass” é uma *função*, e que “article” é o seu principal *argumento*, e precisa estar entre chaves {} — não esqueça de *sempre* fechar as chaves. Em seguida, adicionamos **título** e **data**:

```
1 \documentclass{article}
2 \title{Meu primeiro artigo em \LaTeX}
3 \date{\today}
```

O comando “\LaTeX” gerará o logotipo da linguagem LaTeX — por isso a barra antecede a palavra e o “l”, “t” e “x” estão em maiúscula. Para a data, utilizei um comando que insere automaticamente a data atual, mas você poderia simplesmente inserir a data que quisesse, e no formato que quisesse.

As três linhas escritas acima estarão no início do documento, no **preâmbulo**. O preâmbulo é o local em que também carregamos pacotes específicos, que servem para diversos fins. Pense desta forma: o documento oferece o básico, o essencial.

Contudo, você talvez queira alguns “adicionais”, como, por exemplo, inserir um cabeçalho com três componentes, ou transcrições fonéticas, ou um símbolo especial no seu texto; como a língua padrão do sistema é o inglês, utilizar português também requer um pacote.

Para cada “adicional” que você queira utilizar, é preciso carregar um pacote. É algo bastante simples, e falarei disso em outro artigo. Aqui, faremos um documento **básico**, e apenas falarei do pacote para língua portuguesa.

O próximo passo é iniciar *de fato* o documento. Até agora, dissemos que tipo de documento queremos, demos um título e uma data ao arquivo, mas ainda não começamos a escrever, certo? Para iniciar o documento, basta utilizar o comando “\begin”.

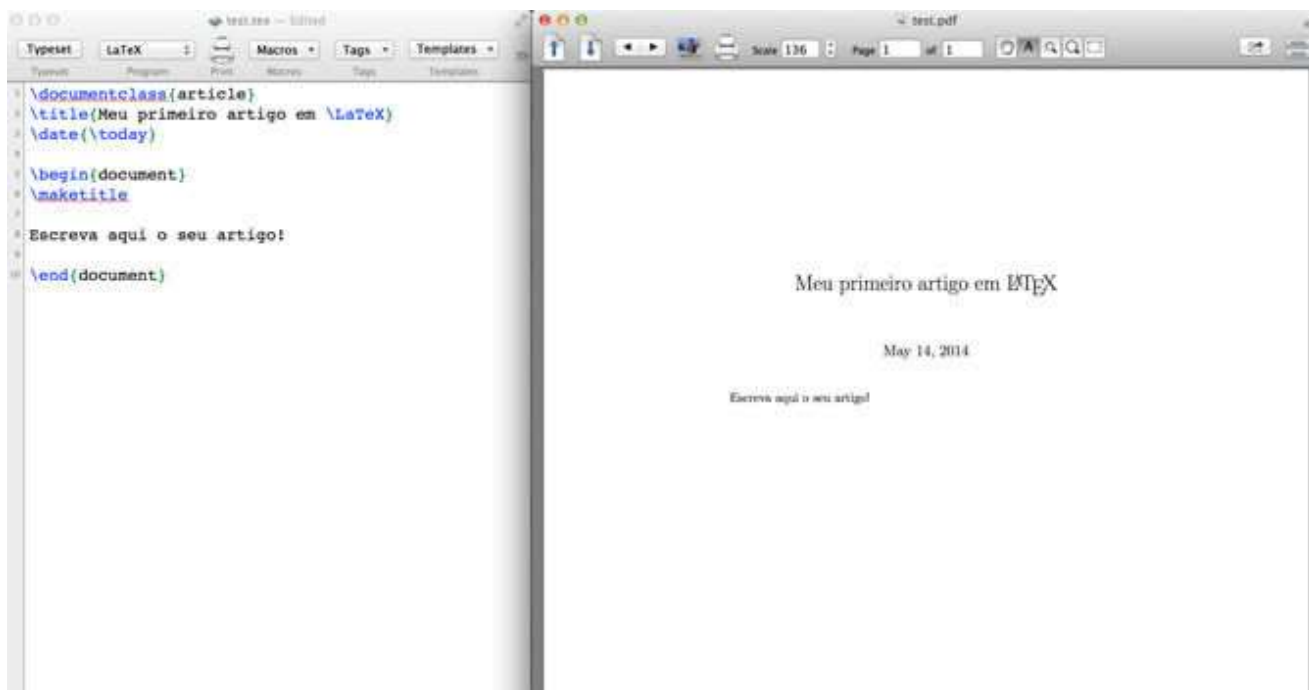
Da mesma forma, criaremos o **fim** do documento, com o comando

`“\end”`. Em ambos os casos, o argumento é `“document”`, já que é o documento que estamos iniciando e finalizando. O primeiro componente de nosso documento é o título, que já especificamos no **preâmbulo**.

Para fazer com que o título esteja **dentro** do documento de fato, utilizaremos o comando `“\maketitle”`. Pronto: o seu artigo estará entre `“\begin”` e `“\end”`, logo após `“\maketitle”`. Veja abaixo:

```
1 \documentclass{article}
2 \title{Meu primeiro artigo em \LaTeX}
3 \date{\today}
4 \begin{document}
5 \maketitle
6
7 Escreva aqui o seu artigo!
8 \end{document}
```

**Neste momento**, você pode salvar seu documento e compilá-lo, clicando em **Typeset** ou utilizando o atalho **Cmd + T** (Mac). Ao compilar o seu documento **tex**, você perceberá que alguns arquivos serão gerados. Não se preocupe: você quer apenas o **pdf**. Esses arquivos são gerados a cada vez que você compila um **tex**. Ou seja: desde que você guarde o arquivo **tex**, os outros podem ser excluídos sem problemas. Na figura abaixo, à esquerda, temos o código-fonte; à direita, o arquivo compilado.



Documento básico em LaTeX compilado

Você deve ter percebido que a data está em inglês, a língua padrão do sistema.

Se você deseja escrever artigos em português, precisará inserir uma linha a mais no seu preâmbulo: trata-se do pacote **babel**, que basicamente “traduz” o seu documento para a língua que você deseja. Para carregar um pacote, utiliza-se o comando “`\usepackage`”.

O argumento principal desse comando será o nome do pacote que desejamos carregar — neste caso, “babel”. Contudo, aqui também incluiremos um argumento *secundário* para selecionar a **língua** que desejamos utilizar (trata-se de um argumento para “babel”). A sintaxe completa será: `\usepackage[língua]{babel}`.

Perceba que o argumento entre **[colchetes]** está relacionado ao

pacote “babel”, e **não** ao comando “\usepackage”. Veja abaixo o código completo — que gerará o mesmo documento visto na figura acima, exceto pela data, que agora estará em português — a alteração de língua apenas afetará comandos que gerem texto automático, é claro.

```
1 \documentclass{article}
2 \usepackage[portuguese]{babel}
3 \title{Meu primeiro artigo em \LaTeX}
4 \date{\today}
5 \begin{document}
6 \maketitle
7
8 Escreva aqui o seu artigo!
9 \end{document}
```

Por fim, você pode fazer comentários no conteúdo que não aparecerão na forma — eis uma das vantagens de separar forma de conteúdo. Fazer comentários é algo extremamente comum em linguagens de programação: em Python e R, usa-se #; em **LaTeX**, usa-se %.



QUANDO VOCÊ QUISER COMENTAR ALGO (UMA  
NOTA MENTAL, POREXEMPLO), BASTA INICIAR A LINHA  
COM %.



Tudo que você escrever após % não será compilado, mas será visível no código-fonte. Isso é útil por diversas razões, já que você certamente faz diversos comentários no que escreve.

Além disso, é didaticamente interessante anotar o que você faz em

cada parte do seu código. O % não precisa necessariamente iniciar uma linha: ele pode vir depois de uma frase que será compilada. No código abaixo, o que está vermelho não será compilado.

```
1 \documentclass{article}
2 \usepackage[portuguese]{babel}
3 \title{Meu primeiro artigo em \LaTeX}
4 \date{\today}
5 \begin{document}
6 \maketitle
7
8 Escreva aqui o seu artigo! % e faça quantos comentários quiser
9 % comentários não são compilados e portanto não aparecem no pdf
10
11 \end{document}
```

— Fim —

## ALGUMAS NOTAS GERAIS SOBRE O LaTeX:

✓ **Espaços em LaTeX:** um dos problemas mais conhecidos do Word é a má administração de **espaçamentos** do aplicativo. Você verá que em LaTeX os espaçamentos são mais inteligentes (e automatizados). Por exemplo, se você inserir, sem querer, **dois** espaços entre duas palavras, apenas um espaço será compilado. Ou seja: “**esta       frase**” é compilada como “**esta frase**”. Veja uma comparação visual entre LaTeX, Word e Adobe [aqui](#). Basicamente, em LaTeX você tem:

- (1) menos hifenizações;
- (2) menos espaçamentos acima da média entre palavras;
- (3) um menor desvio padrão de espaçamentos entre palavras.



Ou seja: o seu texto fica mais uniforme e regular.

✓ **Pastas e organização::** é interessante que você crie seu **tex** em uma pasta só para ele. Assim, arquivos da compilação serão gerados **nessa** pasta, o que evita uma desorganização de arquivos. Além disso, ter uma pasta específica para o seu artigo é essencial para inserir figuras, por exemplo — trata-se de um hábito importante na organização de seus artigos, e será **essencial** com LaTeX. Não é algo **necessário**, mas certamente é recomendável. Em breve você verá por quê.

✓ Você pode criar um documento **tex** em qualquer editor de texto simples — inclusive em **Bloco de Notas**. Basta salvar o arquivo *como* **tex**. Mas então por que baixar o pacote LaTeX? Porque sem ele você não **compila** arquivos **tex**. De qualquer forma, é interessante pensar que você pode criar um código-fonte em qualquer editor de textos, não? Ou seja: tudo o que fizemos até aqui pode ser feito no **Bloco de Notas**, *exceto* o arquivo pdf compilado, é claro.

Note que não especifiquei fonte, tamanho, espaçamento, parágrafo, tamanho de folha etc. Essas variáveis foram automaticamente definidas pelo sistema. No próximo artigo, veremos como alterar cada um delas, para que o documento fique exatamente da forma que desejamos.

✓ A transição entre Word e LaTeX pode ser um pouco complexa, e

certamente exigirá muita paciência e dedicação. Contudo, o resultado será excelente, e seus artigos certamente ficarão mais bem estruturados. Além disso, quando falarmos de referências bibliográficas e caracteres especiais, você entenderá por que o LaTeX é tão importante — e prático.

No início, você talvez se pergunte por que é preciso tantos detalhes para produzir coisas tão simples. Com o tempo, contudo, você perceberá que certas coisas só são bem feitas assim. Depois que você passar o estágio inicial e dominar a linguagem, tudo fluirá com mais naturalidade — e você certamente não se arrependerá!