

# LATEX

## Что такое LaTeX?

LaTeX — это собирательное название для системы подготовки (верстки) документов. Она включает набор инструментов, которые из текстовых файлов, записанных с использованием специального языка разметки формирует готовые к печати документы (как правило в формате PDF). Собственно TeX — это низкоуровневый язык разметки и программирования который лежит в основе этой системы.

## Почему LaTeX?

Создание LaTeX документа это программирование: Вы создаёте текстовый файл в LaTeX-разметке, макросы LaTeX обрабатывают его и производят конечный документ.

Такой подход отличается от используемого в WYSIWYG (What You See Is What You Get — что видишь, то и получишь) программах, таких, как Openoffice.org Writer или Microsoft Word.

В LaTeX:

- Во время редактирования документа Вы не можете (обычно) увидеть его окончательный вариант.

- Вам, как правило, необходимо знать нужные команды разметки LaTeX.

- Иногда сложно получить требуемый вид документа

С другой стороны, у LaTeX есть и достоинства:

- Оформление текста отделено от содержания. Вы полностью сосредотачиваетесь на структуре и содержании документа и забываете о том, как будет выглядеть печатный вариант.

- Стиль, шрифты, оформление таблицы и рисунков т. д. согласованы во всём документе.

- Одно и то же оформление можно использовать для любого числа документов.

- Легко набирать математические формулы.

- Легко создаются алфавитные указатели, сноски, ссылки и библиографические списки.

- Большие документы можно разбивать на несколько файлов и работать с ними отдельно, в том числе с использованием систем управления версиями.

- Вы не обязаны вручную настраивать шрифты, размер текста, высоту строк или читаемость текста — за всё это LaTeX отвечает сам.

- Вам придётся правильно структурировать Ваш документ.

- Файлы с исходными текстами можно просмотреть и изменить в любом текстовом редакторе.

- Так как исходный документ содержит просто текст, с помощью программных средств на любом языке программирования можно создать таблицы, рисунки, формулы и т. д.

Подход LaTeX к созданию документа можно назвать WYSIWYM (What You See Is What You Mean — что видишь, то и думаешь): во время набора текста Вы не видите окончательный вариант документа, только логическую структуру этого документа. Об оформлении вместо Вас позаботится LaTeX.

## WYSIWYG

WYSIWYG является аббревиатурой от англ. What You See Is What You Get, «что видишь, то и получишь») — свойство прикладных программ или веб-интерфейсов, в которых содержание отображается в процессе редактирования и выглядит максимально близко похожим на конечную продукцию, которая может быть печатным документом, веб-страницей или презентацией. В настоящее время для подобных программ также широко используется понятие «визуальный редактор».

## История

До появления технологии WYSIWYG для создания сложноформатированных документов использовались программы, применяющие язык разметки. В этих программах для форматирования документа необходимо было указывать специальные коды (теги), невидимые в конечном результате работы. Теги определяли стиль текста (жирный, наклонный и т. п.), изменения шрифта, расположение текста и иллюстраций и т. д.

Первой программой, использующей WYSIWYG считается текстовый редактор Bravo. Bravo был разработан в Xerox PARC для компьютеров Alto. Программа была

разработана Батлером Лемпсоном (Butler Lampson), Чарльзом Симони (Charles Simonyi) и др. в 1974 году. Bravo не был выпущен на рынок, однако программное обеспечение компьютеров Xerox Star вероятно является прямым потомком этого редактора.

Параллельно с Bravo, но независимо от Xerox PARC, Hewlett Packard разработала и выпустила в конце 1978 года первую коммерческую программу, использующую WYSIWYG — приложение для создания диапозитивов или то, что сегодня называют презентационной графикой. Первый релиз программы, названной BRUNO, работал на мини-компьютерах HP 1000. BRUNO был портирован на HP 3000 и переиздан как «HP Draw».

В 1970-х — начале 1980-х популярным домашним компьютерам не хватало графических возможностей, необходимых для отображения WYSIWYG документов. Такие приложения, как правило, использовались достаточно редко, в основном на мощных рабочих станциях, которые были слишком дорогими для широкого распространения. Однако к середине 1980-х всё стало меняться. Улучшение технологии позволило производить дешёвые графические дисплеи, и программы с WYSIWYG начали появляться на более дешёвых и популярных компьютерах, в том числе LisaWrite для Apple Lisa, выпущенный в 1983 году, и MacWrite для Apple Macintosh, выпущенный в 1984 году.

## TeX

TeX — система компьютерной вёрстки, разработанная американским профессором информатики Дональдом Кнутом в целях создания компьютерной типографии. В неё входят средства для секционирования документов, для работы с перекрёстными ссылками. В частности, благодаря этим возможностям, TeX популярен в академических кругах, особенно среди математиков и физиков.

Название произносится как «тех» (от греч. τέχνη — «искусство», «мастерство»). В написании буква E опущена ниже T и X. В самой программе название форматируется как TEX.

Тип MIME для TeX — application/x-tex.

TeX является свободным ПО.

## История

Первый том книги «Искусство программирования» Д. Кнута был опубликован в 1969 году и печатался методом монотипии, технологии XIX века, которая давала на выходе издание в «хорошем классическом стиле», что нравилось Кнуту. Когда в 1976 году публиковалось второе издание второго тома, всю книгу пришлось набирать вновь, поскольку монотипия почти повсеместно была замещена фотографической техникой, и оригинальные шрифты больше не использовались. Однако 30 марта 1977 года, когда Кнут получил новые оттиски, он увидел, что они выглядят ужасно. Примерно в это же время Кнут впервые увидел результат работы высококачественной цифровой типографической системы и заинтересовался возможностями цифровой типографии. Не оправдавшие ожиданий оттиски дали ему дополнительный толчок к тому, чтобы, разработав свою типографическую систему, решить проблему раз и навсегда. 13 мая 1977 года он написал заметку самому себе, описывающую базовые возможности TeX'a.

Он планировал завершить систему во время своего творческого отпуска 1978 года, но финальная версия языка появилась на свет лишь в 1979 году. Летом 1978 года, когда Кнут писал первую версию TeX'a, в Стэнфорде находился Гай Стил. Осенью того же года он вернулся в MIT и переписал систему ввода-вывода TeX под операционную систему ITS. Первая версия TeX'a была написана на языке программирования SAIL и работала на PDP-10 под операционной системой WAITTS. Для следующих версий Кнут изобрёл концепцию «грамотного программирования» (англ. literate programming), способ получения совместимого исходного кода и документации к нему (конечно, в виде текста на TeX'e) из одного и того же оригинального файла. Этот язык был назван WEB и производил программы на Паскале.

Новая, переписанная с нуля версия TeX'a, была издана в 1982 году и названа TeX82. Помимо других изменений первоначальный алгоритм переносов был заменён новым, написанным Франком Ляном (Frank Liang). Чтобы обеспечить воспроизводимость результата на различном оборудовании, вместо арифметики с плавающей запятой TeX82 использовал арифметику с фиксированной запятой. Помимо этого под нажимом Гая Стила в TeX82 появился настоящий язык программирования.

В 1989 году Дональд Кнут выпустил новые версии систем TeX и METAFONT. Вопреки своему желанию сохранить программу неизменной, Кнут осознал, что 128 различных символов недостаточно, чтобы обеспечить ввод текста на разных языках. Таким образом, главным

изменением в версии 3.0 была возможность работать с 8-битными входными данными, которые позволяли использовать 256 различных символов.

С версии 3.0 TeX использует оригинальную систему нумерации версий: каждое обновление добавляет дополнительную десятичную цифру в конце номера версии так, что она асимптотически приближается к  $\pi$ . Это отражает тот факт, что текущая версия TeX'a — 3.1415926 — очень стабильна, и возможны лишь мелкие обновления. Последнее обновление было в марте 2008 года. На версии 3.0 дизайн системы был заморожен, поэтому добавление новой функциональности не планируется, и все новые версии будут содержать только исправления ошибок.

## Математические формулы в TeX

Первое и главное - пользуйтесь тем, что предоставляет вам интегрированная среда. Обзор интегрированных сред для LaTeX можно прочитать по этой ссылке. В её вкладках должны быть таблицы греческих символов, основных математических операторов и прочего. Самое главное, что символы в таком виде гораздо проще найти и вставить, не перелистывая талмуд со специальными символами.

Заучивать команды для математических символов в LaTeX не нужно - вы и так большинство команд запомните за их красивые и лаконичные названия. Едва ли вы испытаете серьёзные затруднения с названиями греческих букв  $\alpha$ ,  $\gamma$  или  $\delta$ . Основные математические символы тоже должны быть в вашей интегрированной среде, такие как столь любимые физиками приближённые равенства  $\approx$  или интегралы с суммами.

## Быстрый старт - набор простых формул

Формулы можно вставлять в строке или торжественно на отдельной строке, по центру и с номером. Малозначительные формулы типа  $f(x) = a \cdot x + b$  вставляются, как правило, в строчку, а что-то серьёзное, вроде разложения в ряд Фурье:

- `\begin{equation}\label{eq:fourierrow}`
- `f(x) = \frac{A_0}{2} + \sum \limits_{n=1}^{\infty} A_n \cos`
- `\left( \frac{2 n \pi x}{\nu} - \alpha_n \right)`
- `\end{equation}`

оформляется с помощью окружения `\begin{equation}\label{ссылка} ... \end{equation}`.

Набранная формула выглядит так:

$$f(x) = \frac{A_0}{2} + \sum_{n=1}^{\infty} A_n \cos \left( \frac{2n\pi x}{\nu} - \alpha_n \right)$$

При этом напротив неё будет помещён номер, сгенерированный автоматически. Для того, чтобы сослаться на эту формулу, в LaTeX тексте будем ставить `(\ref{ссылка})` чтобы на неё сослаться. Имена ссылкам стоит давать на латинице, во избежание проблем. Если формула приводится для пояснений и сослаться на неё не надо, следует поставить после equation звёздочку, то есть `equation*` и после этого номер для данной формулы генерироваться не будет. Смысл использования LaTeX - создание логичного, структурированного документа, а не каши из кривого оформления и потока сознания.

Ссылки должны быть осмысленными и безошибочно говорить автору текста, какая формула имеется в виду. Не стоит потворствовать лени и глупости, проявляющихся в ссылках типа `\label{uravnenieб}` - следует спросить себя, что означает эта формула и дать ей осмысленное имя.

## Вставка формул в текст

Чтобы вставить формулу внутрь текста, используем окружение `$ $`, внутри которого помещаем формулу. Окружение `$ $` переводит LaTeX в математический режим и будет отрисовывать формулы. Например: `\alpha_0` даст нам греческую букву АЛЬФА с индексом 0.

Подчёркиванием в LaTeX делается нижний индекс, и использовать подчёркивание в тексте нельзя (для подчёркивания в тексте есть пакет расширений `ulem`). Если хотите набирать длинные пассажи в подчёркивании - ставьте фигурные скобки `$x_{i,j}`.

Кроме того, LaTeX категорически против двойного нижнего индекса, и команда `$x_j_k` приведёт к ошибке. Но фигурные скобки позволяют нам надурить LaTeX и сделать

двойной индекс так:  $x_{j_k}$ . Если же вам нужен верхний индекс в формуле, используем символ  $^$  так:  $x^2$ . LaTeX возведёт в степень только первый символ после крышки, и если в степень нужно возвести сразу много символов, экранируем их фигурными скобками (они не отображаются в тексте):  $x^{2x+1}$ .

Если же вы хотите набрать в формуле фигурные скобки, следует заэкранировать их вот так:  $\{$  и они будут отрисованы в формуле, например  $x \{j \}$ .

## Как LaTeX расширяет TeX

Сначала был TeX. Программисты со стажем знают профессора Гарвардского университета Дональда Кнута (Knuth, Donald) как автора многотомной монографии «Искусство программирования для ЭВМ». Ровно 25 лет назад, в 1978 году, он опубликовал первую версию системы обработки печатных документов, известную ныне как TeX. Многие специалисты безоговорочно относят её к одному из выдающихся достижений XX столетия, приравнивая к созданию печатного станка Гутенбергом (Gutenberg, Johann). TeX предвосхитил идеи, получившие признание на рубеже третьего тысячелетия. Система команд TeX по сути была первым языком разметки гипертекстов, к которым принадлежит широко известный ныне HTML (Hyper Text Markup Language) — язык разметки документов для интернета. Исполняемая программа `tex`, выполняющая преобразование размеченного текста в документ, пригодный для высококачественной печати, была чуть ли не первой из программ, которые сейчас принято называть парсерами (`parser`).

TeX общепризнанно считается наиболее качественной системой подготовки печатных публикаций. Как сказано в словаре компьютерных терминов, TeX определяет стандарт, к которому пытаются приблизиться другие настольные издательские системы.

## Затем пришел LATEX

Следующий шаг сделал Лесли Лампорт (Lamport, Leslie). В начале восьмидесятых годов XX века он разработал систему подготовки печатных документов LATEX, основанную на форматировании средствами TeX'a. LATEX позволил пользователю сконцентрировать свои усилия на содержании и структуре текста, не заботясь о деталях его оформления. Как и профессор Кнут, Л. Лампорт опередил своё время. Идея отделения содержания от формы, реализованная в системе LATEX, нашла своё продолжение в XML — расширяемом языке разметки (eXtensible Markup Language), появившемся в конце девяностых годов XX века. Простая замена стиля документа в системе LATEX, как и замена стиля XSL (eXtensible Style Language), «надеваемого» на разметку XML, способна одинаково радикально изменить внешний вид документа.

LATEX реализован в виде формата, то есть надстройки над компактной системой базовых команд, встроенных в исполняемую программу `tex`. Надстройка, созданная самим Кнутом, называется «формат Plain TeX», или просто TeX. Формат TeX входит составной частью в формат LATEX.

LATEX содержит удобные средства генерации алфавитного указателя, списков литературы, рисунков и таблиц, развитые средства импортирования графики, обеспечивает автоматическую нумерацию формул, ссылок и других подобных объектов в сочетании с эффективным механизмом перекрёстного цитирования. Подлинного совершенства TeX и LATEX достигли в форматировании математических формул. Ни одна другая издательская система не сумела достичь тех же вершин в этой области издательского ремесла. Поэтому LATEX особенно популярен в научных кругах. За два десятилетия после изобретений Д. Кнута и Л. Лампорта появились прекрасные текстовые процессоры, но TeX и LATEX сохраняют ранее завоёванные позиции. Причина очевидна: уникальное качество подготовки печатной продукции, помноженное на полную совместимость версий TeX'a и LATEX'a для различных компьютеров.

В конце восьмидесятых годов TeX и LATEX достигли России. Был разработан алгоритм автоматического переноса русских слов. Кириллические шрифты разрабатывались в разных местах: в Вашингтонском университете, в Институте высоких энергий в Протвино, в издательстве «Мир».

## Основные средства разметки LaTeX.

Поскольку все документы LaTeX являются чистым текстом, оформление в них описывается командами разметки, которые при сборке разворачиваются в примитивы TeX (те самые размеры шрифтов, начертание и т.д. которые зачастую выставляются в Word вручную). Для написания работ в готовом шаблоне требует знания лишь базовых, очень простых команд

разметки. Во многих редакторах они вынесены прямо на панели или в меню, поэтому запутаться в них достаточно сложно.

#### 1) Команды структуры:

Команды структуры описывают иерархическую структуру документа и формируют содержание.

`\part => \chapter => \section => \subsection => \subsubsection => \paragraph => \subparagraph`

Все данные команды имеют возможность указания короткого варианта названия для содержания в квадратных скобках.

Также, если к этим командам дописать звёздочку, то раздел не будет пронумерован и отображён в содержании.

Нужно добавить приложения с другой нумерацией? Используйте команду `\appendix` и все разделы после неё сменят нумерацию (по умолчанию с цифр на буквы). Многие редакторы умеют отслеживать структуру документа по этим командам, например на картинке ниже на левой показана структура документа, а по центру список для быстрой вставки.

Пример использования:

```
\chapter[Большое и отчаяние]{Как студент заключение экспертной комиссии подписывать ходил}
```

#### 2) Текстовые команды:

Команды указывающие на размер и начертание текста. Присутствуют, но напрямую в тексте практически не используются.

#### 3) Команды форматирования

Казалось бы, LaTeX запрещает ручное управление форматированием. Но есть несколько особенностей/команд которые нужно знать для успешного использования.

#### 4) Перечисления

В LaTeX есть отлично работающие перечисления. Они представляют собой конструкции вида:

```
\begin{description}
\item [Что описываем] описание
\item [Что еще описываем] описание
\end{description}
```

Какие бывают списки:

нумерованный --- `enumerate`

с точками --- `itemize`

предмет-описание --- `description`

#### 5) Маркеры и ссылки

LaTeX имеет лучшую систему ссылок в индустрии, и не просто лучшую, а ещё и простейшую! Работать с ней исключительно просто:

а) добавляем маркер (на что ссылаться будем) вплотную к нужному объекту --- `\label{eq:superteorema}`.

б) ссылаемся на маркер. Автор использует пакет для умных ссылок `cleveref`, позволяющий надёжно ссылаться на все типы объектов двумя командами `\cref{eq:superteorema}` и `\Cref{eq:superteorema}`, где первый вариант для ссылки внутри предложения, а второй в начале предложения. Для ссылки на несколько объектов нужно просто перечислить их или указать диапазон через дефис.

## Создание презентаций в LaTeX

Все необходимые средства для создания презентаций в расширенной поставке LaTeX есть. Это означает, что нужно устанавливать некоторые отдельные пакеты, но при правильной настройке программы они устанавливаются автоматически.

- **Первый фрейм - титульный лист**

Создаем первый фрейм - обычно это титульный лист с рядом параметров (название, авторы, место работы, дата). Данные пишем до `\begin{document}`

- **Второй и третий фрейм - автоматическое построение содержания и блоки**

- **Четвертый фрейм - добавляем цвета**

Подключаем пакет для работы с цветом (в самом начале документа, после определения его стиля и других пакетов).