

4. Оформление документов

Л^AT_EX в Вышке

22 апреля 2022 г.

Для создания любого документа полезно скопировать преамбулу текущего файла.

1 Некоторые типографские символы

Кавычки-елочки пишутся «так», а лапки „так“. Если в тексте нужно использовать кавычки внутри кавычек, то «„стоит писать так“». Если внешняя и внутренняя кавычки соседствуют, то их следует разделить пробелом.

В тексте для многоточия следует использовать команду...`\textellipsis`

В издательских системах, основанных на T_EX, различают длинное тире «—», короткое тире «-», дефис «-» и знак минуса «-». Чтобы получить на печати дефис, короткое тире или длинное тире, надо в исходном тексте набрать один, два или три знака - соответственно. В русских текстах часто используют длинное тире в качестве тире как такового, а короткое тире — в сочетаниях типа «я вернусь через 2-3 часа» (обратите внимание на отсутствие пробелов вокруг тире в исходном тексте). Длинное тире в русском тексте обычно окружают (следуя традиции) пробелами.

2 Промежутки

pt (пункт)	≈ 0.35 mm
pc (пика)	= 12 pt
mm	миллиметр
cm (сантиметр)	= 10 mm
in (дюйм)	25,4 mm

Таблица 1: Единицы длины в L^AT_EX

Бывают случаи, когда промежутки между символами в формулах, выбранные T_EX'ом автоматически, выглядят неудачно. В этом случае в формулу можно включить команды, задающие промежутки в явном виде. Вот основные из них:

`\quad` Пробел в 1em: | |
`\qquad` Пробел в 2em: | | |
`\,` «Тонкий пробел», или тонкая шпация: ||
`\:` «Средний пробел»: ||
`\;` «Толстый пробел»: ||
`\!` «Отрицательный тонкий пробел» | |

Команда `\!` из этой таблицы уменьшает промежуток на столько же, на сколько команда `\!` его увеличивает.

Иногда необходимо обеспечить, чтобы два соседних слова не попали на разные строки. В этом случае между ними надо вставить «символ неразрывного пробела» `\~`. Такая необходимость возникает, например, в сочетаниях типа «на с. 5»: нельзя отрывать номер страницы от сокращения «с.».

Согласно отечественным полиграфическим правилам строка не должна начинаться с тире; однобуквенное слово, начинающее предложение, не должно стоять последним в строке. Полный список русских однобуквенных слов (который может быть полезен, если вы автоматически расставляете символы ~ после них с помощью текстового редактора) содержится в магическом слове «АВИКОСУЯ».

Как команда «backslash с пробелом» \ , так и символ неразрывного пробела ~ генерируют пробел, но делать пробелы вручную с помощью набора чего-нибудь вроде ~~~ или \ \ \ неразумно, поскольку эти пробелы, как правило, могут растягиваться или сжиматься ради выравнивания строк, и вы не сможете проконтролировать реальный размер пустого пространства, полученного таким способом. Если необходимо задать промежуток с указанием конкретной длины, можно воспользоваться командой \hspace{длина}. Команда для вертикального промежутка \hspace{длина} действует аналогично (обычно для дополнительного отступа между абзацами)

Для дополнительного расстояния между строк можно использовать (если переход к новой строке сделан вручную) уже известную команду перехода на новую строку \\[5pt]

3 Шрифты

Выделим лишь основные моменты.

Текст можно набирать **полужирным**, *курсивом*, в стиле пишущей машинки, или вовсе подчеркнутым с помощью команд с аргументом. Если желаете с помощью команд без аргументов (что делают чаще), то можно поступить (**вот так**, *или так вот*).

Размер менять можно так: мелкий, большой и очень большой.

4 Верстка абзацев (тут что-то пошло не так)

4.1 Борьба с переносами

Не смотря на автоматические переносы в словах, может потребоваться указать L^AT_EX’у в явном виде места в каком-либо слове, где делать переносы можно. Делается с помощью команды \- прямо посреди слова: **тво\-рог**.

Если хотите запретить перенос в каком-либо слове, можно использовать \mbox{слово-без-переносов}. При этом аргумент воспринимается как одна буква. Способ может вызвать одну из распространенных проблем — переполненные строки (те, что за поля выползают).

Для борьбы с такими строками либо расставляйте ручками переносы в последнем слове на строке, либо напишите в самом конце *абзаца*, где находится переполненная строка, пустую строку в группе с командой \slorru:

«Если хотите запретить перенос в каком-либо слове, можно использовать \mbox{слово-без-переносов}. При этом аргумент воспринимается как одна буква. Способ может вызвать одну из распространенных проблем — переполненные строки (те, что за поля выползают).»

4.2 Цитаты, выравнивание, перечни, сноски

Если вам нужно включить в текст цитату, пример, предупреждение и т. п., то удобно воспользоваться окружением **quote**. Это окружение набирает текст, отодвинутый от краев. Пример:

Каждый сознательный гражданин должен понимать, что

весьма небезопасно содержать экзотических животных в городской квартире.

Поэтому подумайте, прежде чем покупать на рынке крокодила.

Для выравнивания текста используются окружения **center** (для центрирования), а также **flushleft** и **flushright** (для выравнивания по левому и правому краю соответственно).

Внутри каждого из этих окружений можно в принципе набирать и самый обычный текст, стандартным образом разбитый на абзацы с помощью пустых строк, но при этом каждая строка

получающегося «абзаца» будет центрирована (для окружения `center`) или выровнена по левому/правому краю (для окружений `flushleft` и `flushright` соответственно).

левый
марш

наше дело
правое

а вот мы позиционируемся
в центристской части
политического спектра

Перечни (списки) можно делать с помощью окружения `itemize`.

«Дни недели»:

- будние:
 - понедельник
 - вторник
 - среда
 - четверг
 - пятница
- выходные:
 - суббота
 - воскресенье

Текст с использованием сносок выглядит так ¹.

4.3 Мелочи

Дабы в новом абзаце подавить первый отступ, пишем в начале абзаца команду `\noindent`; дабы буквально воспроизвести набор символов (не заботясь о том, есть там команды L^AT_EX'а или нет) используют команду `\verb"набор-символов"`. Полезно для набора программного кода.

Для перехода на новую страницу пишем команду `\newpage`.

¹ тут содержание сноски. Если хотите поменять номер на другой, то пропишите его в необязательном аргументе перед обязательным

5 Оформление текста в целом

5.1 Классы, пакеты и классовые опции

Команда `\documentclass`, с которой начинается любой ЛАТ_ЭX'овский файл, имеет один обязательный аргумент — название основного класса — и один необязательный, размещающийся перед обязательным, — список, через запятую, «классовых опций». После `\documentclass` может идти одна или несколько команд `\usepackage`; аргумент этой команды — это список, через запятую, стилевых пакетов, подключаемых к нашему документу.

Список стилевых опций пакета задается в необязательном аргументе команды (перед обязательным) `\usepackage` (через запятую, если опций несколько).

Например, если включить в преамбулу строку `\usepackage{amsmath}` то вам откроются дополнительные возможности набора математических формул; если же написать `\usepackage[noamsfonts]{amsmath}`, то у вас будут все эти возможности, кроме использования готического и ажурного шрифтов.

Стандартные классы, предоставляемые ЛАТ_ЭX'ом: **article**, **report**, **book**, **proc** и **letter**.

Класс **article** удобно применять для статей, класс **report** — для более крупных статей, разбитых на главы, или небольших книг, класс **book** — для книг.

Самые часто употребляемые классовые опции — это **11pt** и **12pt**. Они означают, что основной текст документа будет набран шрифтом кегля 11 или 12 соответственно. Если этих опций не указывать, то будет шрифт кегля 10. Можно указать классовую опцию, задающую формат используемой бумаги, после чего ЛАТ_ЭXсам подберет поля и отступы.

Опция **twoside** задает печать с разными полями на нечетных и четных страницах (как в книгах), а опция **oneside** — печать с одинаковыми полями на всех страницах. В классе **book** по умолчанию установлена опция **twoside**, в классах **article** и **report** — **oneside**.

Опция **draft** пригодна для любого класса. Если она включена, то каждая выбивающаяся на поля строка помечается на полях «марашкой». Это удобно при подготовке корректур (английское слово **draft** как раз и означает «набросок»).

5.2 Стиль оформления страниц

Для задания стиля оформления страницы в ЛАТ_ЭX'e предусмотрена команда `\pagestyle`. Эта команда имеет один обязательный аргумент — слово, обозначающее этот стиль. При использовании стандартными классами документов это слово должно быть одним из следующих:

empty	нет ни колонтитулов, ни номеров страниц;
plain	номера страниц ставятся внизу в середине строки, колонтитулов нет;
headings	присутствуют колонтитулы (включающие в себя и номера страниц).

Наряду с командой `\pagestyle`, задающей стиль оформления всех страниц, есть и команда `\thispagestyle`, задающая стиль оформления одной отдельно взятой страницы. Она принимает такой же аргумент, как и `\pagestyle`, но указываемое этим аргументом оформление относится только к той странице, на которую попал текст, окружающий эту команду. Заранее предугадать, на какую страницу попадет данный фрагмент текста, обычно невозможно. Поэтому, если хотите от этой команды предсказуемых результатов, употребляйте ее непосредственно после `\newpage`.

5.3 Поля страницы

Приведем здесь простой (но не единственный) способ задания полей страницы.

В преамбуле пишем:

```
\usepackage{geometry}
\geometry{top=20mm}
\geometry{bottom=20mm}
\geometry{left=25mm}
\geometry{right=15mm}
```

5.4 Колонтитулы

Укажем удобный способ настройки колонтитулов.

Пишем в преамбуле:

```
usepackage{fancyhdr} %загрузим пакет
\pagestyle{fancy} %применим колонтитул
\fancyhead{} %очистим хидер на всякий случай
\fancyhead[LE,R0]{\thepage} %номер страницы слева сверху на четных и справа на нечетных
\fancyhead[CO]{текст-центр-нечетные}
\fancyhead[LO]{текст-слева-нечетные}
\fancyhead[CE]{текст-центр-четные}
\fancyfoot{} %футер будет пустой
% C,L,R --- расположение текста в колонтитуле
% E, O --- четные и нечетные страницы
```

Не забываем: если колонтитулы на четных и нечетных страницах отличаются, нужно включить опцию `twoside`.

Если для одной страницы нужен особый колонтитул, то можно создать в преамбуле новый стиль для страницы:

```
\fancypagestyle{firststyle} новый стиль
{
\fancyfootoffset[R]{-12cm} так можно регулировать ширину колонтитула
\fancyfoot[L]{текст-слева}
\renewcommand{\footrulewidth}{0.3 mm} толщина отделяющей полосы снизу
\renewcommand{\headrulewidth}{0.3 mm} толщина отделяющей полосы сверху
}
```

5.5 Рубрикация документов

Рассмотрим рубрикацию на примере команды `\section`.

Пусть вам нужно начать раздел документа, озаглавленный «Кое-что о слонах». Для этого в исходном тексте можно написать так: `\section{Кое-что о слонах}`. Команда `\section` принимает один обязательный аргумент — название раздела (это же название пойдет в колонтитулы, если таковые предусмотрены классом, и в оглавление, если вы дадите команду «создать оглавление»). Промежутки между разделами, их нумерация, те же колонтитулы — все это делается автоматически. Кроме обязательного аргумента, у команды `\section` предусмотрен и необязательный. Необязательный аргумент идет перед обязательным; в нем записывается вариант заголовка, предназначенный для оглавления и колонтитулов (если класс предусматривает, что заголовок войдет в колонтитул). Обычно этот вариант — просто сокращенный заголовок.

Пример: `\section[0 слонах]{Кое-что о слонах}`

У команды `\section` есть вариант «со звездочкой». Команда `\section*` начинает новый раздел, не нумеруя его; на оглавлении и колонтитулах наличие раздела, вводимого этой командой, никак не отразится. У команды `\section*` предусмотрен только обязательный аргумент.

Для оформления разделов существуют такие команды:

```
\part \chapter \section \subsection \subsubsection \paragraph \subparagraph.
```

В этом перечне каждая последующая команда обозначает более мелкий подраздел, чем предыдущая. Следует иметь ввиду, что команда `\chapter` («глава») в классах `pros` и `article` не определена (благодаря этому обстоятельству статью легко переделать в главу книги), остальные команды определены в четырех основных классах.

Все то, что мы говорили про необязательный аргумент и вариант «со звездочкой» у команды `\section`, применимо и к командам, перечисленным в этом разделе.

5.6 Список литературы

Список литературы оформляется как окружение `thebibliography`. Это окружение имеет обязательный аргумент — номер источника, занимающий больше всего места на печати (в стандартных шрифтах все цифры имеют одинаковую ширину, так что достаточно привести в качестве аргумента, например, номер 99, если источников будет заведомо меньше 100).

Каждый источник вводится командой `\bibitem`. У нее есть один обязательный аргумент — ваше условное обозначение. В качестве такого обозначения можно использовать любую последовательность из букв и цифр.

В тексте ссылка на источник делается с помощью команды `\cite`. У нее есть обязательный аргумент — условное обозначение того источника, на который вы ссылаетесь. Можно сослаться сразу на несколько источников — для этого в аргументе команды `\cite` надо указать их обозначения через запятую. Приведем пример:

`B~\cite[гл.~1]{Winnie}` описана встреча Винни-Пуха с несколькими пчелами.

`B~\cite{voevoda,med3}` приведены другие сведения о медведях.

```
\begin{thebibliography}{99}
```

```
\bibitem{voevoda}
```

```
М.\,Е.\,Салтыков-Щедрин.
```

```
Медведь на воеводстве.
```

```
\bibitem{med3}
```

```
Л.\,Н.\,Толстой.
```

```
Три медведя.
```

```
\bibitem{Winnie}
```

```
А.\,А.\,Милн. Винни-Пух.
```

```
\end{thebibliography}
```

В тексте это выглядит так:

В [3, гл. 1] описана встреча Винни-Пуха с несколькими пчелами. В [1, 2] приведены другие сведения о медведях.

Список литературы

[1] М. Е. Салтыков-Щедрин. Медведь на воеводстве. 6

[2] Л. Н. Толстой. Три медведя. 6

[3] А. А. Милн. Винни-Пух. 6

5.7 Аннотация

Пишется в окружении `abstract`

6 Кликабельные ссылки

Просто пишем в преамбуле `\usepackage[pdftex,unicode, bookmarks, pagebackref]{hyperref}`