

L^AT_EX HowTo

홍찬의
(R-Project & Open Statistics Korea)

June 26, 2010

목적

이 문서는 Open Statistics Korea 회원님들이 단시간에 L^AT_EX문서 작성법을 익힐 수 있도록 작성된 문서입니다. 이 문서는 아주 기본적인 기능만을 소개하고 있으니 더 자세한 내용은 참고문헌의 문서를 참고하십시오.

Chapter 1

L^AT_EX 설치와 실행

L^AT_EX는 그 하나로 독립된 프로그램이 아니라 여러가지 매크로와 패키지의 조합으로 이루어져 있습니다. 따라서 사용자의 용도와 환경에 맞춰 필요한 매크로와 패키지를 설치해야 하는데 처음 이용자에게는 아주 어려운 작업입니다. 따라서 여러가지 통합환경들이 배포되고 있습니다만 이 문서에서는 윈도우 환경에서 ko.TeXLive 2009를 기준으로 설명하겠습니다.

‘ko.TeXLive 2009’는 <http://faq.ktug.or.kr/faq/KoTeXLive> 여기서 얻을 수 있습니다. 전체설치 버전을 받으시면 됩니다. 다운로드가 완료된 이후에는 위 링크의 설치방법을 따라가시면 됩니다.(아주 간단합니다!) ‘ko.TeXLive 2009’가 정상적으로 설치되었다면 바탕화면에 ‘TeXworks Editor’라는 바로가기가 생성됩니다. 이 바로가기를 실행하면 그림 ??과 같은 편집화면을 볼 수 있습니다. 위의 과정을 모두 마쳤다면 L^AT_EX문서를 작성하기 위한 준비가 다 된 것입니다.

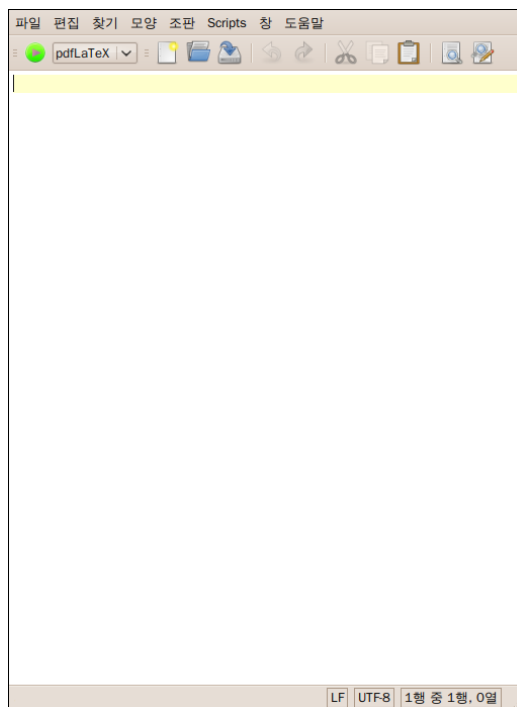


Figure 1.1: TeXworks Editor 실행화면

Chapter 2

첫 문서 작성하기

L^AT_EX는 우리가 일반적으로 많이 사용하는 아래아한글이나 MS Word와 다르게 WYSIWYG 방식이 아닙니다. 따라서 결과물을 보며 작업하는 것이 아니라 문서의 내용을 L^AT_EX의 문법에 맞게 작성한 후에(.tex 파일) L^AT_EX를 이용해 변환(컴파일)해 pdf로 결과물을 받게 됩니다. 여기선 기본적인 L^AT_EX파일(.tex)를 작성하는 방법과 구조 그리고 변환방법을 예제를 가지고 설명하겠습니다.

2.1 L^AT_EX의 기초 구조와 한글입력

우선 가장 간단한 형태의 T_EX파일을 보겠습니다. 아래 내용을 TeXwork 편집기에 입력한 후 임의의 폴더에 저장한 후 Ctrl+t 혹은 편집기 좌측 상단의 연두색 ‘조판’ 버튼을 눌러보세요.

```
\documentclass{report}
\begin{document}
Hello World!!
\end{document}
```

익숙한 Hello World!! 문구를 볼 수 있을겁니다. 생성된 pdf 파일은 .tex 파일이 저장된 경로와 동일한 경로에 저장됩니다. .tex 파일과 pdf 파일 이외에 파일들은 지금은 뭔지 모르셔도 됩니다.

위의 T_EX명령어에서 \documentclass{report}은 작성중인 문서가 어떤 종류(class)인지 정의하는 부분입니다. 이 문서에서는 report 만을 다루겠지만 letter, memoir, beamer, oblovoir, article, book 등의 다양한 클래스가 있습니다. 다음 예제에서 설명하겠지만 class 이외에 용지, 폰트사이즈 등도 여기서 설정하게 됩니다.

다음으로 `\begin{document}`와 `\end{document}`는 문서 내용의 시작과 끝을 선언하는 명령어입니다. $\text{\TeX}$ 명령어는 문서를 아주 구조적으로 작성할 것을 요구합니다. 따라서 `\begin{...}`와 `\end{...}`는 개체를 삽입하거나 어떤 설정을 할 때 아주 빈번하게 사용됩니다.

2.1.1 ko. $\text{\TeX}$ 패키지를 이용한 한글입력

여기까지 따라오셨다면 한글은 어떻게 입력하는지 궁금하실 겁니다. $\text{\LaTeX}$ 에서 한글을 사용하기 위해서는 ko. $\text{\TeX}$ 패키지를 불러와야합니다. 아래 명령어에 두번째 줄에 `\usepackage{kotex}` 명령이 ko. $\text{\TeX}$ 패키지를 불러오라는 명령어입니다. 아래 명령어를 입력하고 변환해보세요.

```
\documentclass{report}
\usepackage{kotex}
\begin{document}
안녕하세요!
\end{document}
```

한글로 ‘안녕하세요!’가 출력되는 것을 보실 수 있을겁니다. $\text{\TeX}$ 문서에서는 `\begin{document}`위의 부분을 ‘전처리부’라고 하고 필요한 옵션과 패키지를 정의해줍니다. 그렇기 때문에 문서의 전체적인 틀은 모두 전처리부에서 결정한다고 생각하시면 됩니다.(물론 문서의 한 부분만 글씨를 키운다거나 글꼴을 바꾼다거나 할 수 있습니다.)

2.2 용지설정, 제목과 저자 그리고 줄바꿈

이번 예제에서는 제목과 저자를 쓰는 법 그리고 글씨크기, 용지크기를 설정하는 방법과 줄넘기기에 대해 설명하겠습니다. 아래 명령으로 그림??과 같은 결과를 얻을 수 있습니다.

```
\documentclass[latter paper, 12pt]{report}
\usepackage{kotex}
\begin{document}
\title{애국 가}
\author{저자불명}
\maketitle{}
동해물과 백두산이
마르고 닳도록\\
```

애국가

저자불명

March 2, 2010

동해물과 백두산이 마르고 닳도록
하나님이 보후하사

우리나라 만세
무궁화 삼천리 화려강산

대한사람 대한으로 길이 보전하세

Figure 2.1: 예제2의 결과물

```
하나님이 보후하사\\  
  
우리나라 만세  
  
무궁화 삼천리 화려강산\\  
  
대한사람 대한으로 길이 보전하세  
\end{document}
```

2.2.1 용지설정과 글자크기

첫줄의 `\documentclass[latter paper, 12pt]{report}`을 보면 앞의 예제 `\documentclass{report}`와 다른 것을 볼 수 있습니다. 우선 `[]`안의 `letter paper`는 용지설정이 `latter` 용지임을 나타냅니다. `a4paper`, `b5paper` 등의 옵션이 있습니다. 이어서 `12pt`는 글씨크기가 `12point`임을 의미합니다.(이 문서는 `letter paper`, `12pt`와 `report` 클래스로 작성되었습니다.)

2.2.2 제목과 저자 삽입

이어서 `title{애국 가}`와 `author{저작불명}`는 문서의 제목과 저자를 정의합니다. 이렇게 정의된 제목과 저자는 `maketitle{}` 명령으로 문서가 작성된 날짜와 함께 문서의 첫머리에 나타나게 됩니다.

2.2.3 줄바꾸기.

이제 줄바꿈에 대해서 설명하겠습니다. $\text{\TeX}$ 는 명령어 내의 줄바꿈 한번을 무시합니다.(수식 환경에서는 줄바꿈은 물론 띄어쓰기까지 무시합니다. 추후에 수식환경에서 띄어쓰기를 하려면 $\sim$ (틸다)를 띄어쓰기할 부분에 삽입하면 됩니다.) 위의 명령어를 보면 ‘...백두산과’와 ‘마르고...’ 사이의 줄바꿈이 결과물에선 무시된 것을 알 수 있습니다.

명령어의 ‘...닿도록’과 ‘하나님이...’ 사이에는 $\backslashbackslash$ 로 구분이 되었는데 이는 줄바꿈을 의미하는 명령어입니다. 줄바꿈을 원하는 부분에 $\backslashbackslash$ 를 삽입하면 줄바꿈이 됩니다. 다만 $\backslashbackslash$ 는 이어지는 문장에 들여쓰기가 되지 않습니다. 만약 이어지는 문장에 들여쓰기를 하려면 ‘...만세’와 ‘무궁화...’ 사이처럼 두줄띄기를 하면 됩니다. 나머지 띄어쓰기 방식에 대해선 코드와 결과를 비교하며 연구해보시기 바랍니다.

2.3 장, 절과 주석 삽입하기

여기서는 장, 절 구조와 주석을 포함한 완결된 형태의 문서를 만드는 방법을 설명하겠습니다. 우선 아래 $\text{\TeX}$ 명령어를 변환해 보십시오.

```
\documentclass[letter paper, 12pt]{report}
\usepackage{kotex}

\begin{document}

\title{훈민정음}
\author{세종대왕}
\maketitle{}
```

우리 나라 말이 중국과 달라 한자와는 서로 잘 통하지 아니한다. 이런 까닭으로 어리석은 백성들이 말하고자 하는 바 있어도 마침내 제 뜻을 펴지 못하는 사람이 많다. 내가 이것을 가엾게 생각하여 새로 스물여덟 글자를 만드니. 모든 사람들로 하여금 쉬이 익혀서 날마다 쓰

는 데 편하게 하고자 할 따름이니라.\footnote{실제 훈민정음 현대어 번역본은 이 형태로 정렬되지 않았습니다. 장과 절 기능을 설명하기 위해 임의로 정렬한 것입니다.}

\section{ㄱ, ㄲ, ㅋ, ㅇ}

기억은 어금닛소리니 굳자의 처음에서 나는 소리와 같은데, 나란히 쓰면 끝자 처음에서 나는 소리와 같다. 키억은 어금닛소리니 꿩자의 처음에서 나는 소리와 같다. 이응은 어금닛소리니 업자의 처음에서 나는 소리와 같다.

\subsection{ㄷ, ㄸ, ㅌ, ㄴ, ㅂ, ㅃ}

디글은 헛소리니 두자의 처음에서 나는 소리와 같은데, 나란히 쓰면 땀자의 처음에서 나는 소리와 같다. 티글은 헛소리니 탄자의 처음에서 나는 소리와 같다. 니은은 헛소리니 낭자의 처음에서 나는 소리와 같다. 비읍은 입술소리니 별자의 처음에서 나는 소리와 같은데, 나란히 쓰면 뽕자의 처음에서 나는 소리와 같다.

\subsubsection{ㅍ, ㅑ, ㅓ, ㅕ, ㅗ}

피윽은 입술소리니 풍자의 처음에서 나는 소리와 같다. 미윽은 입술소리니 밍자의 처음에서 나는 소리와 같다. 지윽은 잇소리지 즉자의 처음에서 나는 소리와 같은데, 나란히 쓰면 짱자의 처음에서 나는 소리와 같다. 치윽은 잇소리니 침자의 처음에서 나는 소리와 같다.

\end{document}

2.3.1 장절 구분

변환한 결과를 보면 문서가 절(section)과 소절(subsection) 그리고 소절보다 작은 절(subsubsection)로 구분된 것을 볼 수 있습니다. 실제 문서에서 이런 구분은 문서를 구조화해 글읽기가 수월해집니다. 예제에서 \section{A}, \subsection{B}와 \subsubsection{C}는 각각 절, 소절, 작은 소절을 구분하라는 명령입니다. 절, 소절, 작은 소절의 제목은 A, B, C 순서로 번호와 함께 나타나게 됩니다. 번호는 자동으로 부여됩니다. 다만 번호를 붙이지 않으려면 \section*{A}, \subsection*{B}와 \subsubsection*{C}와 같이 {앞에 asterisk(*)를 넣으면 됩니다.

2.3.2 주석 삽입하기

예제의 결과를 보면 첫번째 문단의 마지막에 주석이 삽입된 것을 볼 수 있습니다. 주석이 삽입될 장소에 `\footnote{주석내용}`라고 입력하면 작은 번호가 붙고 페이지의 밑에 번호와 주석내용이 나타납니다.

Chapter 3

그 외의 기능

3.1 삽입

여기서는 하이퍼링크, 이미지, 소스코드, 표 등 다양한 개체를 문서에 포함시키는 방법을 설명하겠습니다. 주의할 점은 이미지와 표는 사용자가 지정한 위치에 정확히 위치하지 않습니다. 그 이유는 L^AT_EX의 조판방식에 있습니다. L^AT_EX는 문서를 조판할 때 작성자가 입력한 문서의 내용과 개체를 정해진 규칙에 따라 배열합니다.(배열 방식은 문서의 가장 처음에 정의한 class에 따라 달라집니다.) 이때 이미지나 표가 지정한 위치에 삽입되기에 너무 크거나 배열규칙을 위배하면 개체를 지정한 위치 주변의 삽입 가능한 위치에 삽입합니다. 따라서 개체의 위치가 작성자가 의도한 위치와 조금씩 어긋날 수 있습니다.

3.1.1 이미지 삽입

문서에 그림을 포함하려면 `graphix` 패키지를 사용해야 합니다. 전처리부에 `\usepackage[pdftex]{color, graphix}`를 삽입하면 됩니다. 여기서 `pdftex`는 이미지를 읽어오는 driver를 나타냅니다(`pdftex` 드라이버는 `.png`, `.pdf`, `.jpg`, `.mps` 만을 읽을 수 있습니다. 만약 `.eps` 포맷을 사용하고 싶다면 드라이버를 `dvips`로 두면 됩니다). 그리고 `color`는 웹에서 색상을 정확히 표현하기 위한 패키지입니다. 필요에 따라 드라이버만 바꿔 사용하고 나머지는 그대로 두시면 됩니다.

이렇게 `graphix` 패키지를 불러왔으면 삽입할 이미지를 `.tex` 파일이 있는 폴더에 옮겨둡니다. 그 후에 이미지를 삽입하기 원하는 위치에 아래와 같이 입력합니다.

```
\includegraphics[...]{image_file_name}
```

여기서 `image_file_name`는 삽입할 이미지의 파일명을 의미합니다. 만약 이미지가 저장된 위치에 동일한 이름을 가진 파일이 없다면 확장자는 생략 가능합니다.

이어서 ...부분에는 이미지를 어떤 방식으로 삽입할지를 지정합니다. 여기에 들어가는 옵션에는 `width`, `height`, `angle`이 있습니다. 각각의 사용법은 아래와 같습니다.

```
\includegraphics[width=0.5\textwidth]{image_file_name}  
\includegraphics[height=0.8\textheight]{image_file_name}  
\includegraphics[angle=90]{image_file_name}
```

첫번째 줄은 그림의 폭을 문서의 폭의 0.5배 크기로 삽입하라는 의미입니다. 이미지의 가로세로 비율은 유지됩니다. 두번째는 문서 높이의 0.8배 크기로 삽입하라는 의미이며 마지막 `angle=90`은 이미지를 시계방향으로 90도 회전하라는 의미입니다. 각각의 옵션은

```
\includegraphics[width=0.5\textwidth,angle=90]{image_file_name}
```

과 같이 ,로 구분해 중첩해서 사용할 수 있습니다.

마지막으로 한가지 예를 더 보여드리겠습니다.

```
\begin{figure}  
\centering  
\includegraphics[width=0.5\textwidth]{image_file_name}  
\caption{캡션이 들어갑니다.}  
\end{figure}
```

문서의 원하는 위치에 이렇게 입력하면 Figure 1.1과 같이 중앙정렬과 캡션이 적용됩니다. `\begin{figure}... \end{figure}` 환경은 개체를 입력할 때 여러가지 옵션을 적용할 수 있게 합니다. 예제에서 `\centering`은 개체의 가운데 정렬을 지정하는 명령입니다. 그리고 `\caption{캡션 내용}`은 개체에 캡션을 추가하는 옵션입니다.

3.1.2 표삽입

문서에 표를 작성하기 위해서 `tabular` 환경을 사용합니다. 아래의 예제를 봅시다.

Student Name	Student Serial	Major
John Doe	05	Statistics
Jane Doe	06	Economics

Figure 3.1: tabular 환경으로 작성된 표

```
\begin{tabular}{|l|r|c|}
\hline
Student Name & Student Serial & Major \\ \hline
John Doe & 05 & Statistics \\ \hline
Jane Doe & 06 & Economics \\ \hline
\end{tabular}
```

이렇게 작성된 명령어의 결과는 Figure ??입니다. `\begin{tabular}{...}`와 `\end{tabular}`는 tabular 환경을 선언하는 부분입니다.

`\begin`에서 `{...}` 부분은 표의 열수와 열별 정렬형태를 결정합니다. 예제에서 보면 `|`로 구분된 `l`, `r`, `c`라는 문자열이 있습니다. 이 문자의 갯수로 열의 수를 결정하고 `l`로 지정된 열은 왼쪽정렬, `r`로 지정된 열은 오른쪽 정렬, `c`로 지정된 열은 중앙정렬이 됩니다. 예제에선 문자가 3개이기 때문에 열이 3개고 왼쪽부터 좌측, 우측, 중앙 정렬이 됩니다. 만약 모두 우측정렬을 하는 5열짜리 표를 만들고 싶다면 `{...}` 안에 `|r|r|r|r|r|`을 넣으면 됩니다. 열을 구분하는 선은 기본으로 넣어집니다.

행을 작성하는데 있어 우선 `\hline`은 가로선을 삽입하는 명령입니다. 만약 `\hline`을 넣지 않는다면 가로 구분선이 넣어지지 않습니다.

3.1.3 하이퍼링크 삽입

`hyperref` 패키지를 이용하여 문서 내에 하이퍼링크를 삽입할 수 있습니다. 문서의 전처리부에 다음과 같은 명령으로 `hyperref` 패키지를 불러옵니다.

```
\usepackage[pdftex]{hyperref}
```

이 명령을 입력할 때 주의할 것은 전처리부의 모든 내용중에 가장 아래에 입력되어야 한다는 것입니다. 그러니까 `\begin{document}` 바로 위에 입력하는 것입니다.

자주 사용하는 하이퍼링크 형태는 두가지가 있습니다. “<http://www.openstatistics.net>”과 같이 주소를 직접 표현하는 방법, “[R Project&Open Statistics Korea의 홈페이지는 여기입니다.](#)”과 같이 텍스트에 연결하는 방법이 있습니다. 첫번째 방법은 하이퍼링크를 삽입하기 원하는 위치에 아래의 명령을 입력하면 됩니다.

`\url{http://www.openstatistics.net}`

두번째 방법은 위와 마찬가지로 원하는 위치에

`\href{http://www.openstatistics.net}{여기}`

로 입력하면 됩니다. 두번째 브라켓에 링크가 연결될 텍스트를 입력하는 것입니다.

3.1.4 소스코드 삽입:verbatim 환경

L^AT_EX문서를 작성하다보면 프로그램의 소스코드를 삽입해야할 경우가 있습니다. 여기서 문제는 소스코드 안의 여러 기호나 명령이 L^AT_EX명령과 중복되는 것입니다. 이럴 경우 문서가 변환이 안되거나 의도되로 되지 않습니다. 따라서 모든 기호와 명령을 무시하고 그대로 출력 시키는 환경이 필요합니다. verbatim 환경이 그것입니다.

이 환경은 두가지 사용방법이 있습니다. 행중에 삽입하는 방법과 새로운 문단으로 삽입하는 방법입니다. 첫번째 방법은 원하는 위치에 아래와 같이 입력하면 됩니다.

`\verb|내용|`

그리고 위와 같이 새 문단을 내서 소스코드를 삽입할 때는 `\begin{verbatim}` 과 `\end{verbatim}` 사이에 내용을 삽입하면 됩니다. 이 문서에서 삽입된 모든 L^AT_EX명령어는 이런 방식으로 입력된 것입니다.

3.1.5 item 삽입

문서 안에 여러 항목을 나열할 경우에 itemize 환경을 사용합니다.

- 병아리 삐약삐약
- 강아지 멍멍
- 오리 꺽꺽

이런 내용을 작성하기 위해 아래와 같은 명령어를 사용합니다.

```
\begin{itemize}
\item 병아리 삐약삐약
\item 강아지 멍멍
\item 오리 꺽꺽
\end{itemize}
```

`\begin{itemize}`와 `\end{itemize}` 사이에 목록을 나열하는 것입니다. 각 항목의 앞에 `\item` 이라고 두고 항목의 끝에서 줄바꿈을 함으로서 항목의 시작과 끝을 나타냅니다. 만일 항목의 머리부분을 예제의 채워진 원이 아닌 다른 것으로 바꾸고 싶다면 각 항목의 시작을 `\item[원하는 기호 혹은 글자]`로 바꿔주시면 됩니다. `itemize`환경 안에는 다른 `itemize`환경을 중첩해서 사용할 수 있습니다.

3.2 기호

L^AT_EX문서는 내용 이외에 많은 기호와 명령이 사용되기 때문에 HWP나 MS Word 처럼 기호를 입력하면 문제가 발생하게 됩니다. 따라서 기호를 입력하기 위해 `\`를 기호앞에 써야합니다.(대부분의 경우 이 방법으로 기호를 입력하지만 다르게 입력하는 기호도 있다.) 즉 `# $ % & _ ^` 등을 입력하기 위해선

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

와 같이 입력해야합니다. `^`의 경우 `{}`를 입력하는데 `^`가 강제 부호로 사용되기 때문입니다. `{}` 사이에 글자를 입력하면 첫글자에 `test`와 같이 강제가 붙습니다.(한글은 안됨) 대부분의 기호는 이런 방식으로 작성됩니다. 다만 이 문서에서 모든 기호를 나열할 수 없으므로 L^AT_EX 2_εCheat Sheet를 참고하세요.

3.3 수식

L^AT_EX는 본래 수학, 과학 문서를 아름답게 작성하기 위해 만들어졌습니다. 이에따라 당연하게도 수식을 아주 아름답고 미려하게 작성할 수 있습니다. 아래의 Navier-Stoke 방정식을 봅시다. 다른 어떤 수식입력기도 이정도 품질의 수식을 표현하진 못합니다.

$$\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} = f_i - \frac{1}{\rho} \frac{\partial p}{\partial x_i} + \nu \frac{\partial^2 u_i}{\partial x_j \partial x_j}$$

본래 L^AT_EX는 자체적으로 수식표현이 가능하지만 그 기능을 확장하기 위해 미국수학회에서 배포하는 `amsmath` 패키지를 많이 사용합니다. `\usepackage{amsmath}`를 전처리부에 삽입하면 됩니다. 여기서는 `amsmath` 패키지가 지원하는 내용을 따로 구분하지 않고 설명하겠습니다.

3.3.1 환경

L^AT_EX에서 수식을 입력하기 위한 환경 중 많이 사용되는 환경은 `$...$`, `equation`, `align` 등 입니다.

먼저 `$...$` 환경은 행중에 직접 수식을 입력할 때 사용합니다. 예를들어 “다음 조건을 만족하는 x 가 존재한다.”를 입력하려면 아래와 같이 입력해야 합니다.

다음 조건을 만족하는 x 가 존재한다.

다음으로 `equation` 환경은 수식을 새로 문단을 내어 중앙정렬로 작성할 때 사용합니다. 예를들어

$$\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} = f_i - \frac{1}{\rho} \frac{\partial p}{\partial x_i} + \nu \frac{\partial^2 u_i}{\partial x_j \partial x_j} \quad (3.1)$$

이렇게 말이죠. 사용법은 수식을 쓰기 원하는 곳에 아래와 같이 작성하면 됩니다.

```
\begin{equation}
삽입할 원하는 수식
\end{equation}
```

다만 위의 식에서 우측에 번호가 매겨지게 되는데 번호가 매겨지지 않길 원한다면 아래와 같이 `equation` 뒤에 `asterisk`를 붙여주시면 됩니다.

```
\begin{equation*}
삽입할 원하는 수식
\end{equation*}
```

많은 수식을 쓸때마다 저 긴 명령어를 작성하기 어렵기 때문에 `\[ 수식 \]`과 같은 생략형으로 사용 가능합니다.

위의 두 환경은 여러줄에 걸쳐서 수식을 작성할 수 없습니다. 따라서 `align` 환경을 사용해 다음과 같이 여러줄에 걸친 수식을 작성하게 됩니다.

$$L(\theta) = \prod_{i=1}^n f(x_i : \theta) \quad (3.2)$$

$$= \prod_{i=1}^n \frac{1}{\theta} e^{-\frac{x_i}{\theta}} \quad (3.3)$$

위의 수식을 작성하기 위해


```

\begin{align}
L(\theta)&=\prod_{i=1}^n f(x_i:\theta)\\
&=\prod_{i=1}^n \frac{1}{\theta} e^{-\frac{x_i}{\theta}}
\end{align}

```

과 같은 명령어를 작성했는데 자세히 봐야할 부분은 `\\`와 `&`입니다. `&`는 각 행을 정렬하기 위한 기준점이고 `\\`은 행을 나누기 위한 명령입니다. 위의 예제를 보면 두 식을 행바꿈 하기 위해 첫번째 식의 마지막에 `\\`를 썼고, =왼쪽을 기준으로 정렬하기 위해 =앞에 `&`를 삽입했습니다. `align` 환경역시 그냥 사용하게 되면 예제와 같이 수식마다 번호가 붙게되는데 `equation` 환경과 마찬가지로 `align` 뒤에 `asterisk`를 삽입해주면 번호가 붙지 않게됩니다.

3.3.2 명령어

띄어쓰기 먼저 알아두어야 할 점은 어떤 수식환경 내에서도 명령어 사이의 `enter`와 띄어쓰기는 무시된다는 점입니다. 이런점을 이용해 띄어쓰기와 `enter`를 적절히 섞어 복잡한 수식을 잘 구분해서 쓸 수 있습니다. 수식 섹션의 가장 처음에 예로 든 Navier-Stoke 방정식의 경우 수식을 작성하기 위해 아래와 같이 복잡한 명령어를 입력해야합니다. 만약이때 띄어쓰기와 `enter`를 사용하지 못했다면 아주 복잡해지겠죠?

```

\frac{ \partial u_i }{ \partial t }
+
u_j \frac{ \partial u_i }{ \partial x_j }
=
f_i -\frac{ 1 }{ \rho } \frac{ \partial p }{ \partial x_i }
+
\nu \frac{ \partial^2 u_i }{ \partial x_j \partial x_j }

```

그리스문자 수식을 사용하다보면 영어 알파벳 이외에도 그리스 문자를 많이 사용하게 됩니다. 알파벳은 특별한 경우(명령어와 겹치는 경우)가 아니면 그대로 사용해도 되지만 그리스 문자는 `\alpha`, `\beta`, `\gamma`와 같이 입력해야 α , β , γ 를 쓸 수 있습니다.

분수 분수는 간단히 `\frac{분자}{분모}`의 형태로 입력하면 됩니다.

이항 연산자 기본적인 사칙연산자 `+` `-` `\times` `\div`는 `+`, `-`, `\times`, `\div`과 같이 입력합니다. 그 외의 이항연산자는 참고문헌을 참고하세요.

이항 관계연산자 자주 사용되는 이항관계연산자는 아래와 같이 입력할 수 있습니다.

$<$	<code><</code>	$>$	<code>></code>	$=$	<code>=</code>
$\leq$	<code>\leq or \le</code>	$\geq$	<code>\geq or \ge</code>	$\equiv$	<code>\equiv</code>
$\subset$	<code>\subset</code>	$\supset$	<code>\supset</code>	$\neq$	<code>\neq</code>
$\subseteq$	<code>\subseteq</code>	$\supseteq$	<code>\supseteq</code>	$\approx$	<code>\approx</code>
$\in$	<code>\in</code>	$\ni$	<code>\ni</code>	$\sim$	<code>\sim</code>

괄호 L^AT_EX는 $\{ \}$, $[\]$, $(\)$ 이렇게 세 종류의 괄호를 사용할 수 있습니다. 각각은 `\left\{ 내용 \right\}`, `\left[ 내용 \right]`, `\left( 내용 \right)` 과 같이 사용합니다. $\{ \}$ 만이 나머지와 다르게 `\` 를 $\{ \}$ 앞에 삽입하는데 수식을 묶어주는 $\{ \}$ 와 혼동을 피하기 위해서입니다. 그 사용예는 아래와 같습니다. 괄호의 크기는 내용의 크기에 맞춰 자동으로 조절됩니다.

$$\left\{ \sum_{i=1}^n f(x_i) \right\} \quad (f(x)) \quad \left[\int_a^b f(x_i) dx \right]$$

`\left\{ \sum_{i=1}^n f(\{x\}_{i}) \right\}`
`\left( f(x) \right)`
`\left[ \int_a^b f(\{x\}_{i}) dx \right]`

첨자 x^y 와 x_y 같은 위, 아래첨자는 간단히 $\{x\}^{\{y\}}$, $\{x\}_{\{y\}}$ 와 같이 입력할 수 있습니다. $\{ \}$ 는 꺾이 입력할 필요가 없으나 후에 복잡한 수식을 입력할 때 혼동을 피하기 위해 입력하는 습관을 들이는 것이 좋습니다. a_{x_i} 와 같이 첨자에 첨자를 붙이는 것도 가능합니다. ($\{a\}_{\{x\}_{i}} \}$)

대형연산자 summation과 product 그리고 integral과 같은 대형 연산자를 설명합니다.

$$\sum_{i=1}^n f(x_i) \quad \prod_{i=1}^n f(x_i) \quad \int_a^b f(x_i) dx \quad \int_a^b f(x_i) d x$$

우선 summation은 `\sum_{i=1}^n f(\{x\}_{i})` 와 같이 입력했는데 `\sum` 이 기본이고 마치 아래첨자와 위첨자를 붙이듯이 `_{i=1}^n` 로 범위를 지정합니다. 그리고 마지막으로 더하려는 함수를 써주시면 됩니다. 다음으로 product 연산자는 `\sum` 을 `\prod` 로 바꿔주시기만 하면 됩니다. 마지막으로 적분은 `\int_a^b f(\{x\}_{i}) dx` 로 입력해서 작성했습니다.

`\int_{a}^{b}f(x)_i`)로 적분과 그 범위 그리고 함수를 써주는 것은 동일하나 마지막에 적분하려는 변수만 `dx`로 써주시면 됩니다. 적분이나 미분을 쓸 때 `dx`보다 `d x`를 선호하신다면 `d`대신 `\operatorname{d}`를 사용하면 됩니다. 미분연산자는 따로 없고 분수를 이용해 `\frac{d}{dx}f(x)`와 같이 사용합니다.

대형연산자에서 주의할 점은 `equation`이나 `align`에서 표시되는 모양이 `$...$`환경에서의 모양과 조금 다릅니다. `$...$`환경에서 대형연산자를 사용하면 $\sum_{i=1}^n f(x_i)$, $\prod_{i=1}^n f(x_i)$, $\int_a^b f(x_i)dx$ 과 같이 연산자가 작아지고 범위가 오른쪽으로 표시됩니다.

행렬

참고문헌

- Tobias Oetiker, Hubert Partl, Irene Hyna and Elisabeth Schegl, The Not So short Introduction to L^AT_EX 2_ε. ‘한국 T_EX사용자 그룹’에서 번역서를 제공하는 L^AT_EX 입문서. <http://faq.ktug.or.kr/faq/lshort-kr>에서 최신버전을 내려받을 수 있다.
- Guido Gonzato, 워드 프로세서 사용자를 위한 L^AT_EX, 김강수. ‘한국 T_EX사용자 그룹’에서 번역서를 제공하는 또다른 L^AT_EX 입문서. <http://faq.ktug.or.kr/faq/LaTeX4Wp>에서 최신버전을 내려받을 수 있다.
-