

## L<sup>A</sup>T<sub>E</sub>X소개 2부 : 수학, BibTeX, 사용자 정의

공개 통계학 개론 (OpenIntro Stat.) 저작 학습용

David Diez <sup>1</sup>      이광춘(번역) <sup>2</sup>

<sup>1</sup>OpenIntro [openintro.org](http://openintro.org)

<sup>2</sup>xwMOOC [xwmooc.net](http://xwmooc.net)

# 목차

- $\text{\LaTeX}$ 수학
- BibTeX:  $\text{\LaTeX}$ 참고문헌
- 사용자 정의 명령어와 환경 만들기
- 기타 비법

# Guide to LaTeX

*Guide to LaTeX* 책에는  $\text{\LaTeX}$ 에 대한 멋진 안내가 나와 있고, 이번 학습에서 이 책에서 나온 예제 일부를 충실히 따라간다:

## 7 수학

### 11,12 BibTeX

### 10 사용자 정의 명령어와 환경

$\text{\LaTeX}$ 에 관한 학습교재를 찾는다면, *Guide to LaTeX* 책은 훌륭한 대안이 될 수 있다.

# 사용자 정의 명령어 재료

- 계수기(Counters)
- 명령어 생성하기
- 환경 생성하기

# 기존 계수기

$\text{\LaTeX}$ 은 계수기(변수)를 사용해서 적절하게 번호를 매긴다.

- $\text{\LaTeX}$ 에서 사용되는 계수가 다음에 나와있다: `part`, `chapter`, `section`, `subsection`, `subsubsection`, `page`, `footnote`, `equation`, `figure`, `table`. 이러한 계수기는 상응하는 명령어에 대응된다.
- 각 수준별 열거할 때 다른 계수기가 사용된다: `enumi`, `enumii`, `enumiii`, `enumiv`.
- 다른  $\text{\LaTeX}$  계수기 몇가지: `paragraph`,  `subparagraph`, `mpfootnote`.

# 신규 계수기 생성하기

본인 목적에 맞춰서 자신만의 계수기를 생성할 수도 있다. 아마도 번호 매기고 싶은 예제가 있을 수 있다.

`\newcounter{counterName}[inCounter]`

상기 명령어는 `counterName` 으로 불리는 새로운 계수기를 생성한다. `inCounter` 인자와 + 꺾쇠 괄호는 선택옵션이다. `inCounter`가 증가할 때마다 `inCounter` 를 사용해서 `counterName` 을 재설정한다. (예를 들어, `subsection` 은 “`inCounter`” section 을 갖고 있다).

# 변형하기

기존 혹은 신규 계수기를 변형할 수 있다.

`\setcounter{counter}{n}`

`\addtocounter{counter}{n}`

`\stepcounter{counter}`

`\refstepcounter{counter}`

만약 `\label` 을 붙여놓는다면, `\refstepcounter` 명령어를 사용해서 계수기 값을 참조한다.

# 출력하기

계수기를 생성하고, 변형하고, 참조할 수 있다. 하지만, 문서에서 계수기를 출력할 필요도 있다. 계수기를 다음 명령어 중 하나를 호출해서 출력한다:

\arabic{chapter} (4, 아라비아 숫자)

\Roman (IV, 대문자 로마 숫자)

\roman (iv, 소문자 로마 숫자)

\Alph (D, 영문 대문자)

\alph (d, 영문 소문자)

\fnsymbol (§, 각주 기호)

사용자 정의 명령어와 환경에 계수기를 두고 사용한다.

## 간단한 명령어

$x_1, \dots, x_n$  같은 흔한 문장은 다음 new 명령어로 축약할 수 있다.

```
\newcommand{\xvec}{x_1,\dots,x_n}
```

상기 명령어를 삽입하고 나서 (나중에 문서에)  $\$ \xvec \$$  을 타이핑하면,  $x_1, \dots, x_n$  을 얻게 된다. 만약 달려 부호를 생략하면 문제가 된다. 이것을 해결하는데 추가 명령어를 사용한다.

```
\newcommand{\xvec}{\ensuremath{x_1,\dots,x_n}}
```

두번째 정의 말미에 추가 공백을 두어서, 공백 문제를 방지하게 했다. 좀더 멋진 해결책은 **xspace** 팩키지를 사용하는 것이다 (*Guide to LaTeX*, 186쪽을 참조한다).

# 인자를 갖는 명령어

만약 작성한 명령어를 일반화하려고 하면, 인자 두개를 추가한다.

```
\newcommand{\subvec}[2]{\ensuremath{#1_1,\dots,#1_{\#2}}}
```

`\subvec{y}{m}`로부터  $y_1, \dots, y_m$  을 생성할 수 있다.

$n^{th}$  인자에 대해서 `#n` 을 통해 추가 인자를 생성하고 참조할 수 있다. 디폴트 선택옵션 인자도 활용할 수 있다 (*Guide to LaTeX*, 188쪽을 참조한다).

# 일반화

신규 명령어에 대한 일반적인 프레임워크는 다음과 같다.

\newcommand{\commandName}[n]{the commands}

여기서

- `commandName`은 명령어 명칭이 된다.
- `n`은 인자 갯수가 된다.
- 인자를 `the commands` 에 #1, #2, ..., #n 으로 참조한다.

이미 존재하는 명령어를 다시 정의하려면, 상기와 동일한 형식으로

\renewcommand 명령어를 사용한다.

## 견본 환경

환경은 begin과 end 태그를 사용한다 (예를 들면, itemize). `\begin` 과 `\end` 태그에 발생하는 것만 정의할 필요가 있다. 예를 들어,  
`\newenvironment{예제}`

`{\small\textbf{예제.} \hspace{2mm}} % 시작하는 것`  
`{\hspace{-2mm}} % 끝나는 것`

견본 환경 호출:

`\begin{예제}`

나머지 연산의 덧셈은 신비롭게 동작한다:  $2+2=1$   $(\text{mod } 3)$ .

`\end{예제}`

결과:

예제. 나머지 연산의 덧셈은 신비롭게 동작한다:  $2 + 2 = 1$   $(\text{mod } 3)$ .

## General environment

Generally environments take the form

```
\newenvironment{environmentName}{begin stuff}{end stuff}
```

We can also declare that there will be  $n$  arguments.

```
\newenvironment{environmentName}[n]{begin stuff}{end  
stuff}
```

As before, we refer to the arguments as  $\#1, \dots, \#n$  in the **begin stuff** and **end stuff**.

To redefine an environment that already exists, use

```
\renewenvironment with the same format as above.
```

## Environment + counter

```
\newcounter{example}  
\setcounter{example}{0}  
\newenvironment{example}  
{\refstepcounter{example}\small  
 \textbf{Example \arabic{example}.}\hspace{2mm}}  
{\\}
```

## Organizer and time saver

The `\include` command is useful for long documents:

```
\include{otherDocName}
```

For instance, this presentation actually calls three separate documents: one for each big section. Thus I would not take time typesetting parts of the document I was not working on while keeping organized:

```
\include{math/math} % "math" document in the "math"  
folder
```

```
%\include{bibtex/bibtex}
```

```
%\include{comenv/comenv}
```

# Wrap-up

After this class, you should have a general idea of

- using the math modes in LaTeX,
- creating bibliographies using BibTeX, and
- creating your own commands and environments.

Any questions?