

L^AT_EX 프로그래밍

권현우

서강대학교

학습목표

- \LaTeX 2e 프로그래밍 개요를 익힌다.
- xparse와 expl3 문법을 익힌다.

프로그래밍(수학) 실력이 늘려면

1. 내가 해결하고자 하는 문제를 찾는다.
2. 삽으로 맨틀까지 뚫을 각오로 땅을 판다.
3. 남의 코드(논문)을 보며 어떻게 내 문제를 효과적으로 풀 수 있는지를 고민한다.
4. 자신의 생각을 담아 코딩을 한다(글을 쓴다).
5. 문법만 공부한다고 빌게이츠가 되는건 아니다(교과서만 공부한다고 테렌스타오가 되는건 아니다).

그런 의미에서 오늘 우리가 해결하고자 하는 문제는...

- 강의록 조판
- 피보나치 수열
- 학회 자료만들기

커맨드와 환경

정의 (커맨드)

커맨드(command)란 어떤 내용을 실행하기 위한 코드를 말한다.

예: `\color{red}`, `\newline`, `\vspace{...}`

정의 (환경)

환경(environment)이란 `\begin{...} .. \end{..}`을 이용해 구역을 구분해 일정한 틀을 만들어내는 코드를 말한다.

예: `\begin{center} ... \end{center}`

```
\newcommand{\mynote}{레이텍 특강}
```

```
...
```

```
\mynote
```

```
\newcommand{\mynote}[1]{#1\은 레이텍 특강입니다.}
```

...

```
\mynote{오늘} \mynote{내일} \mynote{모레}
```

[]안의 숫자는 파라미터의 개수를 의미한다.

```
\newenvironment{이름}{환경 전}{환경 뒤}
```

```
\newenvironment{myproof}{\bigskip \noindent \textbf{증명} }{\bigskip }
```



```
\renewcommand
```

```
\providecommand
```

```
\DeclareRobustCommand
```

- `renewcommand` : 재정의
- `providecommand`: 같은 이름의 명령이 이미 정의되어 있다면 새로 정의하는 것을 무시한다.
- `DeclareRobustCommand`: `providecommand`와 비슷해보이나 명령어가 풀리지 않도록 지정하는 명령어

```
\newcommand{\AAA}{SNU}
```

```
\DeclareRobustCommand{\BBB}{서울대학교}
```

```
\section{\AAA}, \section{\BBB}
```

NewDocumentCommand

`\usepackage{xparse}`

`\NewDocumentCommand{명령어}{옵션/인자 지시자}{명령 수행 내용}`

- 옵션/인자 지시자 : m(mandatory argument) o(optional argument) O(optional argument with default)
- 그외 인자 지시자
 - l(중괄호 {가 나올때까지 적힌 모든 것을 인자로 받아들임)
 - r(구분자가 필요한 인자)
 - R(r에 default가 있는 것)
 - u(지정된 토큰까지 적힌 모든 것을 인자로 받음)
 - v(verbatim처럼 프린트되는 인자)

NewDocumentCommand

`\usepackage{xparse}`

`\NewDocumentCommand{명령어}{옵션/인자 지시자}{명령 수행 내용}`

- 옵션/인자 지시자 : m(mandatory argument) o(optional argument) O(optional argument with default)
- 그외 인자 지시자
 - d: delimited option
 - D: delimited option with default
 - g: option delimited by '{' and '}'
 - G: option delimited by '{' and '}' with default
 - s: starred version (자주 씀)

NewDocumentCommand

```
\usepackage{xparse}
```

```
\NewDocumentCommand{\mybold}{m}{\textbf{#1}\index{#1}}
```

```
\mybold{공룡}
```

NewDocumentCommand

```
\usepackage{xparse}
```

```
\usepackage{imakeidx}
```

```
\makeindex[name=korean,title={찾아보기}]
```

```
\makeindex[name=english,title={Index}]
```

```
\NewDocumentCommand{\term}{m m}{\textbf{#1} (#2)\index[korean]{#1}\index[english]{#2}}
```

```
\term{공룡}{Dinosaur}
```

```
\printindex[korean]
```

```
\printindex[english]
```

NewDocumentCommand

```
\usepackage{xparse}
```

```
\NewDocumentCommand{\mybold}{m}{\textbf{#1}\index{#1}}
```

```
\mybold{공룡}
```

NewDocumentCommand

```
\usepackage{xparse}
```

```
\usepackage{imakeidx}
```

```
\makeindex[name=korean,title={찾아보기}]
```

```
\makeindex[name=english,title={Index}]
```

```
\NewDocumentCommand{\term}{m m}{\textbf{#1} (#2)\index[korean]{#1}\index[english]{#2}}
```

```
\term{공룡}{Dinosaur}
```

```
\printindex[korean]
```

```
\printindex[english]
```

NewDocumentCommand

```
\usepackage{xparse}
\usepackage{imakeidx}
...

\NewDocumentCommand{\term}{o m o m}{\textbf{#2} (#4) \index{#2}%
\IfNoValueTF {#1}
{ \index[korean]{#2} }{ \index[korean]{#1} }%
\IfNoValueTF {#3}
{ \index[english]{#4} }{ \index[english]{#3} }%
}

\term{공룡}{Dinosaur}
\term{공룡!티라노사우루스}{티라노사우루스}[Dinosaur!Tyrannosaurus]{Tyrannosaurus}
\term{공룡!브라키오사우루스}{브라키오사우루스}[Dinosaur!Brachiosauridae]{Brachiosauridae}
...
```


NewDocumentCommand– *starred

```
\NewDocumentCommand \norm {s m m }  
{  
  \IfBooleanTF {#1}  
  { \Vert {#2} \Vert_{#3}}  
  { \left\Vert {#2} \right\Vert_{#3}}  
}
```

xparse가 없었다면... @에 xkeyval에...

NewDocumentEnvironment (2.tex)

```
\NewDocumentEnvironment{환경명}{옵션/인자 지시자}{환경 이전}{환경 이후}
```

```
\newcounter{mytheorem}
```

```
\NewDocumentEnvironment{mytheorem}{o}{\refstepcounter{mytheorem}}%
```

```
\IfNoValueTF{#1}
```

```
{ \begin{tcolorbox}
```

```
↪ [colback=red!5!white,colframe=red!75!black,title={{\bfseries\sffamily 정리
```

```
↪ \themytheorem}}]}
```

```
{ \begin{tcolorbox}[colback=red!5!white,colframe=red!75!black,title={\sffamily\bfseries
```

```
↪ { 정리 \themytheorem} (#1)] }
```

```
}{\end{tcolorbox}}
```

Expl3란?

L^AT_EX 프로그래밍의 여러 제약점을 해소할 수 있음

- @을 안 써도 된다.
- 함수와 변수가 구분됨
- 프로그래밍을 위해 불러들였던 패키지를 부르지 않아도 된다.

주요문서

- >texdoc interface3
- >예제로 배우는 Expl3 (KTUG)

Expl3란?

```
\usepackage{expl3}
```

```
\ExplSyntaxOn
```

```
...
```

```
\ExplSyntaxOff
```

유의사항

- 스페이스는 모두 무시된다. 스페이스를 입력 문자열로 남기고자 할 때에는 ~(틸데)를 남겨야 한다.
- :과 _가 명령의 일부로 쓰인다.

Expl3란? (3.tex)

```
\usepackage{xparse,expl3}
```

```
\ExplSyntaxOn
```

```
\NewDocumentCommand{\hellocmd}{m}{ Hello,~#1! }
```

```
\cs_new:Npn \hello_fn:n #1
```

```
{
```

```
    Hello,~#1!
```

```
}
```

```
\NewDocumentCommand{\hellocmdt}{m}
```

```
{\hello_fn:n {#1}}
```

```
\ExplSyntaxOff
```

정수 계산 (4.tex)

```
\NewDocumentCommand{\evaluation}{m}{  
  \int_eval:n { #1 }  
}
```

정수 계산 (4.tex)

```
\NewDocumentCommand{\evalsec}{m m}{  
  \tl_new:N \l_my_tl  
  \tl_set:Nn \l_my_tl { #1 }  
  \int_new:N \l_my_int  
  \int_set:Nn \l_my_int { #2 }  
  \int_eval:n { \l_my_tl + \l_my_int }  
}
```

정수 계산 (4.tex)

```
\NewDocumentCommand{\modcalc}{m m}{  
  \int_new:N \l_mod_mynumber  
  \int_set:Nn \l_mod_mynumber {\int_mod:nn {#1} {#2}}  
  \int_eval:n {\l_mod_mynumber}  
}
```


forloop (5.tex)

- `\int_gadd`
- `\int_step_inline:nn`

Fibonacci (6.tex)

```
\NewDocumentCommand \Fibo { m }
{
    \Fibo_defn:n { #1 }
}

\cs_new:Npn \Fibo_defn:n #1
{
    \int_compare:nTF { #1 < 3}
    {
        1
    }
    {
        \int_eval:n { \Fibo_defn:n {#1 - 2} + \Fibo_defn:n {#1 - 1} }
    }
}
}
```

Fibonacci (7.tex)

```
\usepackage{pythontex}
\begin{pycode}
def fib(n):
    a,b = 1,1
    if n==1 or n==2:
        return 1
    for i in range(1,n):
        a,b = b, a+b
    return a

result = str(" ")

for i in range(1,20):
    result = str(result) + str("$F_{") +str(i) + str("}$") +str("&")+ str(fib(i)) + str("\\tabularnewline") + str(" ")
\end{pycode}
\begin{tabular}{c | c}
\py{result}
\end{tabular}
```

References

- >texdoc interface3
- >texdoc xparse
- 예제로 배우는 Expl3, KTUG
- xparse / ktug wiki
- <https://www.alanshawn.com/latex3-tutorial>