

xparse

ktug wiki

2015년 7월 19일

- texdoc xparse
- Beyond `\newcommand` with `xparse`¹
- From `\newcommand` to `\NewDocumentCommand` with `xparse`²
- `\usepackage{xparse}`가 필요함.

차 례

1	<code>\newcommand</code> 에서 <code>\NewDocumentCommand</code> 로...	2
1.1	<code>syntax</code>	2
1.2	<code>\newcommand</code> 대체하기: basics	3
1.3	<code>\newcommand</code> 대체하기: 하나의 옵션이 있는 경우	3
1.4	<code>\newcommand</code> 넘어서기: 두 개 이상의 옵션이 있는 경우	4
2	argument specifiers	4
2.1	<code>m & l</code>	4
2.2	<code>r</code>	5
2.3	<code>R</code>	5
2.4	<code>u</code>	6
2.5	<code>v</code>	6
3	optional argument specifiers	6

¹<http://tug.org/TUGboat/tb31-1/tb97wright-xparse.pdf>

²<https://tug.org/TUGboat/tb31-3/tb99wright.pdf>

3.1	o	6
3.2	O	6
3.3	d	7
3.4	D	7
3.5	g	7
3.6	G	7
3.7	s	8
3.8	t	8
4	xparse 명령 함수	9
5	Examples	9
5.1	text color	9
5.2	$\frac{b}{1/b}$, $\frac{a}{b}$ \rightarrow a/b	9
5.3	옵션을 활괄호 안에 넣기	10
5.4	ProcessList, SplitList, xparse	10

1 \newcommand에서 \NewDocumentCommand로...

1.1 syntax

- `\newcommand{<이름>}[<옵션1>][<옵션2>]{<명령 수행 내용>}`
 - 이름: {foo}와 같이 활괄호로 둘러싸지 않고 \foo로 써도 된다.
 - 옵션1: \foo가 받아들이는 옵션/인자의 개수
 - 옵션2: \foo가 받아들이는 옵션(#1)의 디폴트 값
 - 예: `\newcommand\ktug{Korean TeX Users Group}`
- `\NewDocumentCommand{<이름>}{<옵션/인자 지시자>}{<명령 수행 내용>}`
 - <이름>과 <명령 수행 내용>은 \newcommand와 같다.
 - 두번째 인자인 <지시자>(specifier, 아래에서 설명)를 조절하여 다양한 형태의 명령을 정의한다
 - 예: `\NewDocumentCommand\ktug{{Korean TeX Users Group}}`

1.2 \newcommand 대체하기: basics

- 단순 대체

```
%\newcommand\bsh{\textbackslash}
\NewDocumentCommand\bsh{}{\textbackslash}

\bsh\texttt{newcommand} 대체하기.
```

- 인자 하나 있는 명령... m: mandatory argument

```
%\newcommand\hi[1]{Hello, #1}
\NewDocumentCommand\hi{m}{Hello, #1}

\hi{Tom}.
```

- 인자 두개 있는 명령... mm

```
%\newcommand\hii[2]{Hello, #1 and #2}
\NewDocumentCommand\hii{mm}{Hello, #1 and #2}

\hii{Tom}{Judy}. Long time no see.
```

- \renewcommand -> \RenewDocumentCommand

```
%\newcommand\hii[2]{Hello, #1 and #2}
\renewcommand\hii[2]{Hi, #1 and #2}
\RenewDocumentCommand\hii{mm}{Hi, #1 and #2}
\hii{Tom}{Judy}. Long time no see.
```

1.3 \newcommand 대체하기: 하나의 옵션이 있는 경우

- o: 옵션, m: mandatory argument

```
%\usepackage{kotex,xcolor}
%\newcommand\myword[2][\textcolor{#1}{#2}]
\NewDocumentCommand\myword{om}{\textcolor{#1}{#2}}

\myword[black]{검정 글씨}, \myword[blue]{파랑 글씨}
```

- 0{<default>}: 디폴트값이 있는 옵션 지시자

```
%\usepackage{kotex,xcolor}
%\newcommand\Myword[2][black]{\textcolor{#1}{#2}}
\NewDocumentCommand\Myword{0{black}m}{\textcolor{#1}{#2}}

\Myword{검정 글씨}, \Myword[blue]{파랑 글씨}
```

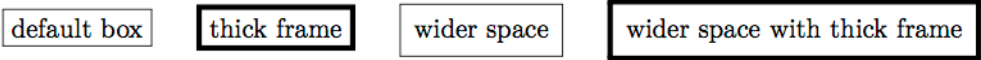
1.4 \newcommand 넘어서기: 두 개 이상의 옵션이 있는 경우

- 0{<default>}m0{<default>}

```
\NewDocumentCommand\myfbox{0{.4pt}m0{3pt}}
{
  {\setlength{\fboxrule}{#1} \setlength{\fboxsep}{#3} \fbox{#2}}
}

\myfbox{default box}          \myfbox[2pt]{thick frame}
\myfbox{wider space}[5pt]    \myfbox[2pt]{wider space with thick frame}[5pt]
```

output:



2 argument specifiers

2.1 m & l

m: mandatory argument 꼭 있어야 하는 인자... 위 아래의 예 참조

l: 첫 여는 활괄호 "{" 를 만날 때까지 적힌 모든 것을 인자로 받는다. (다음 예에서 l은 첫번째 인자, m은 두번째 인자)

```
%\usepackage{xcolor}
\NewDocumentCommand\arglm{lm}{\textcolor{#2}{#1}}

\arglm Do I know you?{red} \arglm Excuse me?{blue} \arglm Should I know you?{red}
```

Do I know you? Excuse me? Should I know you?

output:

‘love’ version³

2.2 r

r: 구분자가 필요한(required) 인자. 바로 이어 나오는 토큰 두개를 구분자로 간주.

```
\NewDocumentCommand\argr{r<>m}{`#1' and \textbf{#2}}  
  
\argr<in between>{outside}
```

output:

‘in between’ and **outside**

- 위 예에서 구분자 “<”와 “>”는 “(”와 “)”처럼 다른 것으로 각각 대체해도 된다.
- 좀 더 과격하게는 “@”와 “^” 따위로 대체해도 결과는 같다. (아래 예)

2.3 R

R: “required” with default

```
\NewDocumentCommand\argrR{R@^{default}m}{`#1' and \textbf{#2}}  
  
\argrR@in between^{outside}
```

output:

‘in between’ and **outside**

- “@”과 “^” 사이에 오는 모든 것을 첫번째 인자로 간주하도록 정의되었다.

³http://www.ktug.org/xe/index.php?mid#KTUG_open_board&order_type#desc&sort_index#regdate&page#3&document_srl#202894

2.4 u

u: 지정하는 토큰까지 (until) 적힌 모든 것을 인자로 받는다.

```
\NewDocumentCommand\argu{u{+}}{\` \textsc{#1}'}  
  
\argu Thakj Lkd Pks Alkj+ is what you might want to see.
```

output:

‘THAKJ LKD PKS ALKJ’ is what you might want to see.

2.5 v

v: verbatim 처럼 프린트되는 인자

```
\NewDocumentCommand\vttext{v}{\fbox{\texttt{#1}}}  
  
\vttext+${\ln xy}$는 \vttext{\(\ln xy\)}와 같은 결과를 보여주지만,  
\vttext{\v_{$}^text}는 예러...
```

output:

`${\ln xy}$` 는 `\(\ln xy\)` 와 같은 결과를 보여주지만, `\v_{$}^text` 는 예러...

- +...+나 |...|처럼 짝지어진 구분자 사이에 인자를 넣으면 된다.
- 다른 함수의 인자 안에서는 작동하지 않는다.

3 optional argument specifiers

3.1 o

o: optional argument

- 위 newcommand 대체하기 (subsection 1.2)의 예 참조

3.2 O

O: optional argument with default

- 위 newcommand 대체하기 (subsection 1.2)의 예 참조

3.3 d

d: delimited option

```
\NewDocumentCommand\tzdot{r()d()}{\tikz{\filldraw (#1) circle (#2);}}
\tzdot(0,0)(2pt)
```

3.4 D

D: delimited option with default

```
\NewDocumentCommand\tzDotx{0{overlay}r()D(){3pt}}
{\tikz{\filldraw [#1] (#2) circle (#3);}}

\tzDotx(0,0)\tzDotx(1,0)(4pt)\tzDotx[overlay,red](2,0)
\tzDotx[overlay,blue](3,0)(5pt)
```

output:



3.5 g

g: option delimited by “{” and “}” (group tokens)

```
\NewDocumentCommand\tzdotg{r()g}{\tikz{\filldraw (#1) circle (#2);}}
\tzdotg(0,0){2pt}
```

3.6 G

G: {option} with default

```
\NewDocumentCommand\tzDotG{0{overlay}r()G{3pt}}
{\tikz{\filldraw [#1] (#2) circle (#3);}}

\tzDotG(0,0)\tzDotG(1,0){4pt}\tzDotG[overlay,orange](2,0)
\tzDotG[overlay,green](3,0){5pt}
```

output:



3.7 s

s: starred version

```
\NewDocumentCommand\tzcirc{sr(){  
\IfBooleanTF#1  
{\tikz{\draw [fill,overlay,blue!80] (#2) circle (3pt);}} }  
{\tikz{\draw [overlay,blue!80] (#2) circle (3pt);}} }  
}  
  
\verb+\tzcirc*(0,0)+: \tzcirc*(0,0),  
\quad\verb+\tzcirc(0,0)+: \tzcirc(0,0)
```

output:

`\tzcirc*(0,0):` ● , `\tzcirc(0,0):` ○

3.8 t

t: more general token

```
\DeclareDocumentCommand\tzcirc{t#r(){  
\IfBooleanTF#1  
{\tikz{\draw [fill,overlay,blue!80] (#2) circle (3pt);}} }  
{\tikz{\draw [overlay,blue!80] (#2) circle (3pt);}} }  
}  
  
\verb|\tzcirc#(0,0)|: \tzcirc#(0,0), \quad\verb|\tzcirc(0,0)|: \tzcirc(0,0)
```

output:

`\tzcirc=(0,0):` ● , `\tzcirc(0,0):` ○

- s 옵션에서 를 사용하는 것과 같이, t 옵션에서는 뿐만 아니라 #,+,!,@,#,/,^,알파벳 등 다양한 토큰 사용 가능

4 xparse 명령 함수

- `NewDocumentCommand\Foo...`
 - 같은 이름을 가진 명령이 있는지 검사하여, 없으면 새로운 명령 `\Foo`를 정의한다
 - 있으면, 'already defined' 에러를 낸다.
- `RenewDocumentCommand\Foo...`
 - 같은 이름을 가진 명령이 있는지 검사하여, 있으면 기존 명령을 대체하여 새로 정의한다.
 - 없으면, 'not yet defined' 에러를 낸다.
- `ProvideDocumentCommand\Foo...`
 - 같은 이름을 가진 명령이 있는지 검사하여, 없으면 새로운 명령을 정의한다.
 - 있으면, 아무 일도 하지 않는다. 에러도 내지 않는다.
- `DeclareDocumentCommand\Foo...`
 - 같은 이름을 가진 명령이 있는지 검사하지 않고, 항상 명령을 새롭게 정의한다.

5 Examples

5.1 text color

```
\documentclass{article}
\usepackage{kotex}
\usepackage{xparse}
\usepackage{xcolor}
\usepackage{jiwonlipsum}
\NewDocumentCommand\txtcol{lm}{\textcolor{#2}{#1}}
\begin{document}
\txtcol \jiwon[1]{blue!30}
\txtcol \jiwon[2]{blue!50}
\txtcol \jiwon[3-4]{blue!70}
\end{document}
```

5.2 `\frac{b}->1/b`, `\frac{a}{b}->a/b`

출처⁴

⁴<http://www.texdev.net/2010/05/23/from-newcommand-to-newdocumentcommand/>

```

\documentclass{article}
\usepackage{xparse}
\let\RealFrac\frac
\RenewDocumentCommand\frac{mg}{%
  \IfNoValueTF{#2}
    {\RealFrac{1}{#1}}
    {\RealFrac{#1}{#2}}%
}
\begin{document}
One arg: \verb|\frac{10}$| prints $\frac{10}$ .
Two args: \verb|\frac{2}{30}$| prints $\frac{2}{30}$ .
\end{document}

```

5.3 옵션을 활괄호 안에 넣기

출처⁵

```

\documentclass{article}
\usepackage{xparse}
\NewDocumentCommand\en{g}{%
  \IfNoValueTF{#1}{\epsilon}{\epsilon_{#1}}%
}
\begin{document}
\verb|\en| prints \(\ \en \) and \verb|\en{stuff}| prints \(\ \en{stuff} \) .
\end{document}

```

5.4 ProcessList, SplitList, xparse

```

\documentclass{article}
\usepackage{xparse}
\NewDocumentCommand\mylist{>{\SplitList{;}}m}
{
  \begin{itemize}
    \ProcessList {#1} { \insertitem}
  \end{itemize}
}
\newcommand\insertitem[1]{\item #1}

```

⁵<http://www.texdev.net/2010/05/22/promoting-xparse/>

```
\begin{document}
Short list:
\mylist{a;b}

Longer list:
\mylist{a;b;c;d}

List within a list:
\mylist{a;b\mylist{A;B;C;D};c;d}
\end{document}
```