

# HzGuide, expl3와 함께 확장하다

이 호재

2015-11-07

# 차례

HzGuide 클래스	4
HzIllust.sty	5
HzGuide.cls	7
조판 언어 (Markup Language)	9
Programmable Markup 언어이기 때문에 가능한	10
문서화 솔루션으로서의 HzGuide	11
다국어 조판	12
서식 위에 서식	13
셀 스크립트	15
Tortoise Tagger: 번역 전처리	19
Expl3의 미덕	20
미완의 L <sup>A</sup> T <sub>E</sub> X3	21

Xparse와 Expl3 . . . . .	22
매뉴얼들 . . . . .	24
Xparse로 매크로 만들기 . . . . .	25
Expl3의 낫선 양식: 공백, 밑줄 문자, 쌍점 . . . . .	27
Xparse 매크로 인자 지정자들 . . . . .	29
필수 (mandatory)와 선택 (optional) . . . . .	30
간편 스위치: 별 (star) . . . . .	32
그대로 그렇게 (verbatim) . . . . .	33
다른 인자 지정자들 . . . . .	34
Xparse 도구들 . . . . .	36
Expl3의 새로운 시도: 함수 (function)와 변수 (variable) . . . . .	37
Expl3 변수들 . . . . .	38
Expl3 함수 지정자들 . . . . .	41
예: 그림 넣기 . . . . .	42
전역 (global) 변수와 지역 (local) 변수 . . . . .	52
해묵은 것들과의 조화 또는 불화 . . . . .	53
Expl3가 대체하기 어려운 것들 . . . . .	55
맺음말 . . . . .	56

# HzGuide 클래스

---

- `hzlayout.sty`
- `hzillust.sty`
- `hzheading.sty`
- `hzmisc.sty`
- ...
- `hzlang.sty`
  - `hzlang-DE.sty`
  - ...
  - `hzlang-TC.sty`

## HzIllust.sty

```
\newcommand\placefigure[2] [] {%
  \resetplacefigure%
  \setkeys{placefigure}{#1}%
  \ifPF@figure%
    \ifPF@label\else%
      \renewcommand\PFlabel{#2}%
    \fi%
    \ifPF@frame%
      \setlength\fbboxrule{\PFframerule}%
      \setlength\fbboxsep{\PFframesep}%
      \ifPF@reflect%
        \sbox{\PFbox}{\fcolorbox{\PFframecolor}{white}{%
          \reflectbox{\includegraphics[scale=\PFscale]{#2}}}}%
      \else%
        \sbox{\PFbox}{\fcolorbox{\PFframecolor}{white}{%
          \includegraphics[scale=\PFscale]{#2}}}%
      \fi%
    \else%
      \ifPF@reflect%
```

```

        \sbox{\PFbox}{\reflectbox{\includegraphics[scale=\PFscale]{#2}}}%
    \else%
        \sbox{\PFbox}{\includegraphics[scale=\PFscale]{#2}}%
    \fi%
\fi%
\else%
    \sbox{\PFbox}{#2}%
\fi%
\ifPF@float%
    \begin{figure}[!htb]\putPF\end{figure}%
\else%
    \PFTopskip%
    \begin{minipage}{\linewidth}\putPF\end{minipage}%
    \PFbottomskip%
\fi}

```

## HzGuide.cls

```
\NewDocumentCommand \placeimage { 0 { } m }
{
  \group_begin:
  \placeimage_clear:
  \PlaceImageSetup{#1}
  \tl_if_empty:NT \l_placeimage_label
  {
    \tl_set:Nn \l_placeimage_label {#2}
  }
  \bool_if:NTF \l_placeimage_frame_bool
  {
    \hbox_set:Nn \l_image_box
      { \imageframe{\includegraphics[scale=\l_placeimage_scale]{#2}} }
  }
  {
    \hbox_set:Nn \l_image_box
      { \includegraphics[scale=\l_placeimage_scale]{#2} }
  }
  \bool_if:NTF \l_placeimage_float_bool
```

```
{
  \begin{figure}
  \placeimage_put:
  \end{figure}
}
{
  \object_skip_before:n {\l_placeimage_beforeskip}
  \begin{minipage}{\linewidth}
  \placeimage_put:
  \end{minipage}
  \object_skip_after:n {\l_placeimage_afterskip}
}
\group_end:
}
```



# 조판 언어 (Markup Language)

---

What You See Is What You Get.

**Presentational:** 워드 프로세서

**Procedural:** troff,  $\text{\LaTeX}$ , PostScript (그렇다면 PDF도?)

**Descriptive:** HTML, XML, Markdown, AsciiDoc

What You See Is What You Mean.

## Programmable Markup 언어이기 때문에 가능한

- 내용과 서식의 분리
- 기능의 지속적인 추가와 확장
- 외부 프로그램을 이용한 조작

# 문서화 솔루션으로서의 HzGuide

---

다수가 사용하는 윈도우즈에서

HzGuide

```
\documentclass[12pt, language=korean, verbatim=true]{hzguide}  
\LayoutSetup{paper=slide}
```

- + 다층의 서식을 허용하는 방법론
- + PowerShell
- + Tortoise Tagger

## 다국어 조판

다국어 조판에서 레이텍을 독보적 존재로 만드는

- Polyglossia

```
\verb{\setmainlanguage[variant=german, spelling=new, latesthyphen=true]{german}}
```

- Bidi
- XeCJK

# 서식 위에 서식

## 1. foo.tex

```
\documentclass[language=korean, styleset=../../HermesStyle]{hzguide}}
```

## 2. HermesStyle.tex

```
\NewDocumentCommand \pathinput { m }  
{  
  \InputIfFileExists {../#1} {} {}  
  \InputIfFileExists {#1} {} {}  
}  
  
\pathinput{PresetStyle.tex}  
  
\IfDefinedTF{\AdhocLayout}  
{ \AdhocLayout }  
{ \LayoutSetup[paper=A4,column=vartwo] }  
  
\pathinput{PostsetStyle.tex}  
\pathinput{DocInfo.tex}
```

### 3. PresetStyle.tex

```
\newcommand\AdhocLayout{
  \LayoutSetup{paper=B5,column=one}
  \makeheadrule{common}{\textwidth}{\normalrulethickness}
  \makeevenhead{common}{\hffont\leftmark}{}{\hffont\evenheadinner}
  \makeoddhead{common}{\hffont\oddheadinner}{}{\hffont\rightmark}
  \makeevenfoot{common}{\hffont\hffont\thepage}{}{}
  \makeoddfoot{common}{}{}{\hffont\thepage\hffont}
  \makeheadrule{chapter}{0pt}{0pt}
  \makeevenhead{chapter}{}{}{}
  \makeoddhead{chapter}{}{}{}
}
```

# 셀 스크립트

```
c:\>c.ps1 -a
```



c.ps1:

```
1 $option = $Args[0]
2
3 $job = "OnTW"
4 $TexFile = $job + ".tex"
5 $PdfFile = $job + ".pdf"
6 $BakFile = $TexFile + ".bak"
7 $today = get-date -format "yyyy-MM-dd"
8
9 function MakeSlide () {
10 rep.ps1 -o "\\articletrue" -n "\\articlefalse" -t $TexFile
11 rep.ps1 -o "\\DisableOverlay" -n "\\EnableOverlay" -t $TexFile
12 remove-item $BakFile
13 xpub $TexFile -s
14 xpub $TexFile -s
15 $output = $job + "_slide_" + $today + ".pdf"
16 copy-item $PdfFile $output -force
17 delaux.ps1
18 }
19
20 function MakeCopy () {
```



```

21 rep.ps1 -o "\\articletrue" -n "\\articlefalse" -t $TexFile
22 rep.ps1 -o "\\EnableOverlay" -n "\\DisableOverlay" -t $TexFile
23 remove-item $BakFile
24 xpub $TexFile -s
25 $tex = "\documentclass{minimal}
26 \usepackage[a4paper]{geometry}
27 \usepackage{pdfpages}
28 \includepdfset{templatesize={130mm}{110mm},delta=0mm -10mm,nup=1x2,frame}
29 \begin{document}
30 \includepdf[pages=-]{$PdfFile}
31 \end{document}"
32 $output = $job + "_copy_" + $today + ".pdf"
33 set-content _tmp.tex -encoding UTF8 $tex
34 xpub _tmp.tex -s
35 copy-item _tmp.pdf $output -force
36 remove-item _tmp.*
37 delaux.ps1
38 }
39
40 function MakeHandout () {
41 rep.ps1 -o "\\articlefalse" -n "\\articletrue" -t $TexFile
42 remove-item $BakFile

```

```
43 xpub $TexFile -n
44 $output = $job + "_handout_" + $today + ".pdf"
45 copy-item $PdfFile $output -force
46 }
47
48 switch ($option) {
49 "-s" {MakeSlide}
50 "-c" {MakeCopy}
51 "-h" {MakeHandout}
52 "-a" {MakeSlide; MakeCopy; MakeHandout}
53 default {xpub $TexFile}
54 }
```

```
c:\>xpub.ps1 foo.tex -e -rus
```

```
c:\>open.ps1 memoir.cls -s
```

```
c:\>resize.ps1 *.jpg 125 75
```

# Tortoise Tagger: 번역 전처리기

```
\section{Channel List}\label{sec:ChannelList}
```

↵

Use the channel list to change channels more easily.

Press the `\mi{OK}` button to open the channel list.

↵

```
\para{Schedule a programme to record}
```

Select a programme entry and choose either `\ui{Once Record}` or `\ui{Series Record}`.

For details, see `\titleref{sec:SchedulingRecordings}` on page `\pageref{sec:SchedulingRecordings}`.

|

```
\begin{UI}
```

```
\item[Service preview] With this enabled, selecting a channel entry displays the channel's picture in the inset window on the upper right of the screen. Select it once more to switch to the channel.
```

With this option disabled, selecting a channel entry immediately switches to the channel.

↵

```
\item[Keep viewing service when exit] With this disabled, pressing the \mi{EXIT} button switches to the channel that was focused on.
```

With this option enabled, pressing the `\mi{EXIT}` button closes the channel list without changing channels.

```
\end{UI}
```

## Expl3의 미덕

- 변경과 확장이 용이한 기능 그룹으로서의 매크로 모듈들을 만들 수 있다.
- 간결하여 읽고 이해하기 쉽다.
- 그래서 나 아니어도

# 미완의 L<sup>A</sup>T<sub>E</sub>X3

---

Plain T <sub>E</sub> X	<code>\def</code>	<code>plain.tex</code>	<code>tex.fmt</code>
L <sup>A</sup> T <sub>E</sub> X2 <sub>ε</sub>	<code>\newcommand</code>	<code>latex.ltx</code>	<code>latex.fmt</code>
L <sup>A</sup> T <sub>E</sub> X3	<code>\NewDocumentCommand</code>	<code>xparse.sty</code>	?

---

# Xparse와 Expl3

---

<b>xparse.sty</b>	<b>expl3.sty</b>
<code>\NewDocumentCommand</code>	<code>\cs_new:Nn</code>
High-level interface	Programming interface
사용자 매크로	사용자 매크로가 사용하는 매크로
<code>\begin{document}</code> 뒤에서만	<code>\begin{document}</code> 앞에서만

Xparse에 의해 정의된 매크로는 expl3 함수와 변수를 포함할 수 있고, 그 반대도 가능하다. 그러나 expl3에 의해 정의된 매크로는 사용자가 직접 사용할 수 없고, 오로지 xparse 매크로만이 사용할 수 있다.

따라서 다음과 같은 코딩 방식이 이상적이다.

```
\NewDocumentCommand \foo { m }  
{  
  \foo_foo:n {#1}  
}  
  
\cs_new:Npn \foo_foo:n #1  
{  
  ... #1 ...  
}
```

# 매뉴얼들

---

- xparse.pdf
- expl3.pdf
- interface3.pdf



# Xparse로 매크로 만들기

---

- `\DeclareDocumentCommand`  $\approx$  `\def`
- `\NewDocumentCommand`  $\approx$  `\newcommand`
- `\RenewDocumentCommand`  $\approx$  `\renewcommand`
- `\ProvideDocumentCommand`  $\approx$  `\providecommand`
- `\DeclareDocumentEnvironment`
- `\NewDocumentEnvironment`  $\approx$  `\newenvironment`
- `\RenewDocumentEnvironment`  $\approx$  `\renewenvironment`
- `\ProvideDocumentEnvironment`
- `\DeclareExpandableDocumentCommand`

```
\def\foo#1#2{%  
... #1 %  
... #2 ...%  
}  
  
\newcommand\foo[2]{%  
... #1 %  
... #2 ...%  
}  
  
\NewDocumentCommand \foo { m m }  
{  
... #1  
... #2 ...  
}
```

## Exp13의 낫선 양식: 공백, 밑줄 문자, 쌍점

---

Exp13가 취하는 기본 양식은

- 정의부에서 공백(space)을 무시하고,
- 매크로 이름에 밑줄 문자(underscore)와 쌍점(colon)을 사용하는 것이다.

```
\NewDocumentCommand \foreign {s o m}
{
  \group_begin:
    \WithHangul
    \ifx\f@family\rmdefault
      \IfNoValueTF {#2} {\foreignrmfamily} {\fontspec{#2}}
    \else
      \IfNoValueTF {#2} {\foreignsfamily} {\fontspec{#2}}
    \fi
    \IfBooleanTF {#1} {\char"#3} {#3}
  \group_end:
}
```

```
\DeclareDocumentCommand \foo { m }
{ \textcolor{#1}{This color is #1.} }
\foo{blue}
```

```
\ExplSyntaxOn
```

```
\DeclareDocumentCommand \foo { m }
{ \textcolor{#1}{That color is #1.} }
\ExplSyntaxOff
\foo{orange}
```

```
\ExplSyntaxOn
```

```
\DeclareDocumentCommand \foo { m }
{ \textcolor{#1}{That\space color~is~#1.} }
\ExplSyntaxOff
\foo{red}
```

This color is blue.  
Thatcolorisorange.  
That color is red.

# Xparse 매크로 인자 지정자들

---

Exp13 없이 xparse만으로도 쓸모있는가? 글썄 ...

**m**: mandatory

**o**: optional

**O**: optional with default

**s**: star

**v**: verbatim

```
\NewDocumentCommand \foo { s o o m m o }  
{  
  ... #1  
  ... #2 ... #3 ...  
  ... #4 ... #5 ...  
  ... #6 ...  
}
```

## 필수 (mandatory) 와 선택 (optional)

```
\DeclareDocumentCommand \foo { o m }  
{  
  \IfNoValueTF {#1}  
  { \includegraphics{#2} }  
  { \includegraphics[scale=#1]{#2} }  
}  
\foo{rabbit1} \foo[.5]{rabbit1}
```



```
\DeclareDocumentCommand \foo { 0 {1} m }  
{  
  \includegraphics[scale=#1]{#2}  
}  
\foo{rabbit1} \foo[1.5]{rabbit1}
```



```
\DeclareDocumentCommand \foo { o o m m }
{
  \IfValueT {#1} { #1~>~ }
  \IfValueT {#2} { #2~>~ }
  #3~>~#4
}
\menu{메신저}{카카오}\
\menu[다음]{메신저}{카카오}\
\menu[포털][다음]{메신저}{카카오}
```

메신저 > 카카오

다음 > 메신저 > 카카오

포털 > 다음 > 메신저 > 카카오

## 간편 스위치: 별 (star)

```
\section{Introduction}
\section*{Introudction}

\newcommand\foo{\@ifstar{\fooa}{\foob}}
\newcommand\fooa[1]{ ... #1 ... }
\newcommand\foob[1]{ ... #1 ... }
```

```
\DeclareDocumentCommand \foo { s m }
{
#2
\IfBooleanTF {#1}
{ \includegraphics{dog1} }
{ \includegraphics{dog2} }
}
\foo{나의 기분이} \
\foo*{너의 기분이}
```

나의 기분이



너의 기분이





## 그대로 그렇게 (verbatim)

```
\verb|C:\Users\Hugh\Desktop\My_Documents|
```

C:\Users\Hugh\Desktop\My\_Documents

```
\DeclareDocumentCommand \foo {v}
{ \textcolor{violet}{\textsl{#1}} }

\foo|yihoze@icloud.com| \\
\foo+https://github.com/YiHoze/HzGuide+\\
\foo{C:\Users\Hugh\Desktop\My_Documents}
```

*yihoze@icloud.com*

*https://github.com/YiHoze/HzGuide*

*C:\Users\Hugh\Desktop\My\_Documents*

## 다른 인자 지정자들

그 밖에 l, r, R, u, d, D, t, g, G가 있는데 ...

```
\DeclareDocumentCommand \foo {lm}
{ \includegraphics{#2} #1 }

\foo "엄마, 말할 게 있는데..."{rabbit2}

\foo "뭐?"{rabbit3}

\foo "성적 오르면 폰 줌..."{rabbit4}
```



“엄마, 말할 게 있는데...”



“뭐?”



“성적 오르면 폰 줌...”

```
\DeclareDocumentCommand \foo {u{}}  
{ \textcolor{blue}{#1} }
```

```
\foo I am happy to join with you today| in  
↪ what will go down in history as the  
↪ greatest demonstration for freedom in the  
↪ history of our nation.
```

I am happy to join with you today in what will go  
down in history as the greatest demonstration for  
freedom in the history of our nation.

# Xparse 도구들

---

- \ReverseBoolean
- \SplitArgument
- \SplitList
- \ProcessList
- \TrimSpaces

```
\DeclareDocumentCommand \foo { > {  
  ↪ \SplitList{;} } m }  
{  
  \ProcessList {#1} { \MakeUppercase }  
}  
  
\foo{a;b;c;d}
```

ABCD

|

# Exp13의 새로운 시도: 함수 (function)와 변수 (variable)

---

- Exp13로 매크로를 만들려면 \Exp1Syntax0n을 선언해야 한다.
- Exp13가 권장하는 명명 방식에 따라 함수와 변수를 만드는 것이 좋다.
- 함수 이름과 변수 이름에 밑줄(underscore)이 포함된다.
- 함수 이름에 쌍점(colon)이 포함된다.

함수: 모듈 그룹 + 기능

```
\foo_skip:n
```

변수: 영역 + 모듈 그룹 + 기능 + 유형

```
\l_foo_width_dim
```

# Exp13 변수들

---

**bool:** `\g_tmpa_bool, \l_tmpa_bool, \g_tmpb_bool, \l_tmpb_bool`

**box:** `\l_tmpa_box`

**clist:** 콤마 분리 목록, `\l_tmpa_clist`

**dim:** 치수, `\l_tmpa_dim`

**tl:** 토큰 목록 `\l_tmpa_tl`

**int:** 정수, `\l_tmpa_int`

**fp:** 소수, `\l_tmpa_fp`

**skip:** 고무줄, `\l_tmpa_skip`

“0.5” 따위를 전달하기 위해 **fp** 변수가 필요할까?

```
X \includegraphics[scale=\l_tmpa_fp]{foo}
0 \includegraphics[scale=\l_tmpa_tl]{foo}
\end{latex}

\topic{Exp13 함수들}
```

```

\begin{itemize}
\item \texcode{\tl_new:N}
\item \texcode{\tl_set:Nn}
\item \texcode{\tl_gset:Nn}
\item \texcode{\tl_if_empty:NTF}
\item \texcode{\int_zero:N}
\item \texcode{\int_add:Nn}
\item \texcode{\int_incr:N}
\item \texcode{\int_compare:nTF}
\item \texcode{\str_case:nn}
\item \texcode{\bool_if:NTF}
\item \texcode{\keys_define:nn}
\end{itemize}

```

```

\topic{Exp13 함수 만들기}

```

```

\begin{minted}{latex}
\cs_new:Nn \foo:
{
  ...
}

```

```
\cs_new:Npn \foo:n #1
{
  ... #1 ...
}

\cs_new:Npn \foo:nn #1 #2
{
  ... #1 ...
  ... #2 ...
}

\foo:nn {#1}{#2}
```



## Exp13 함수 지정자들

---

**D:** 사용 금지 (Do not use)

**N:** 단일 토큰 (No manipulation)

**n:** 토큰 무리

**c:** csname

```
\foo:c {Arg} = \foo:N \Arg
```

**V와 v:** 변수 값

```
\foo:V {\Var} = \foo:v {Var}
```

**o:** 한 번만 확장 (expansion Once)

**x:** 끝까지 확장 (eXhaustive expansion)

**f:** 가능한 데까지 확장 (Full expansion)

**T:** 참 (True)

**F:** 거짓 (False)

**p:** 매개 변수 (Parameter)

**w:** Weird?

## 예: 그림 넣기

---

```
\keys_define:nn { placeimage }
{
  beforeskip    .skip_set:N = \l_placeimage_beforeskip,
  afterskip     .skip_set:N = \l_placeimage_afterskip,
  float         .bool_set:N = \l_placeimage_float_bool,
  frame         .bool_set:N = \l_placeimage_frame_bool,
  scale         .tl_set:N = \l_placeimage_scale,
  landscape     .bool_set:N = \l_placeimage_landscape_bool,
  align         .tl_set:N = \l_placeimage_align,
  caption       .tl_set:N = \l_placeimage_caption,
  label         .tl_set:N = \l_placeimage_label,
  legend        .bool_set:N = \l_placeimage_legend_bool,
  showfilename  .bool_set:N = \l_placeimage_showfilename_bool,
}

\NewDocumentCommand \PlaceImageSetup { m }
{
  \keys_set:nn { placeimage } {#1}
```

```
}
```

```
\NewDocumentCommand \placeimage { 0 { } m }  
{  
  \group_begin:  
  \placeimage_clear:  
  \PlaceImageSetup{#1}  
  \tl_if_empty:NT \l_placeimage_label  
  {  
    \tl_set:Nn \l_placeimage_label {#2}  
  }  
  \bool_if:NTF \l_placeimage_frame_bool  
  {  
    \hbox_set:Nn \l_image_box  
      { \imageframe{\includegraphics[scale=\l_placeimage_scale]{#2}} }  
  }  
  {  
    \hbox_set:Nn \l_image_box  
      { \includegraphics[scale=\l_placeimage_scale]{#2} }  
  }  
  \bool_if:NTF \l_placeimage_float_bool  
  {
```

```

\begin{figure}
\placeimage_put:
\end{figure}
}
{
\object_skip_before:n {\l_placeimage_beforeskip}
\begin{minipage}{\linewidth}
\placeimage_put:
\end{minipage}
\object_skip_after:n {\l_placeimage_afterskip}
}
\group_end:
}

```

```

\cs_new:Nn \placeimage_put:
{
\bool_if:NTF \l_placeimage_landscape_bool
{ \placeimage_put_landscape: }
{ \placeimage_put_portrait: }
}

```

```

\cs_new:Nn \placeimage_put_portrait:

```

```

{
  \dim_compare:nT { \g_protrude_dim > Opt }
  {
    \dim_set_towidth:Nn \l_tmpa_dim { \box_use:N \l_image_box }
    \object_locate:N \l_tmpa_dim
  }
  \l_placeimage_align\nil
  \box_use:N \l_image_box
  \placeimage_captions:
}

\cs_new:Nn \placeimage_put_landscape:
{
  \dim_compare:nT { \g_protrude_dim > Opt }
  {
    \dim_set_towidth:Nn \l_tmpb_dim { \box_use:N \l_image_box }
    \dim_set_toheight:Nn \l_tmpa_dim { \box_use:N \l_image_box }
    \object_locate:N \l_tmpa_dim
  }
  \l_placeimage_align\nil
  \rotatebox{90}{
    \begin{minipage}{\l_tmpb_dim}

```

```

    \box_use:N \l_image_box
    \placeimage_captio:n:
\end{minipage}
}
}

```

```

\cs_new:Nn \placeimage_captio:n:
{
  \bool_if:NTF \l_placeimage_showfilename_bool
  {
    \tl_set:Nx \l_tmpa_tl {\l_placeimage_label}
    \legend{\tl_to_str:N \l_tmpa_tl}
  }
  {
    \tl_if_empty:NF \l_placeimage_captio:n
    {
      \bool_if:NTF \l_placeimage_legend_bool
      {
        \legend{\l_placeimage_captio:n}
      }
      {
        \imgcaption{\l_placeimage_captio:n}
      }
    }
  }
}

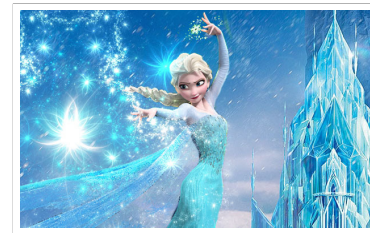
```

```
\label{\l_placeimage_label}  
}  
}  
}  
}
```

```
\placeimage{elsa}
```

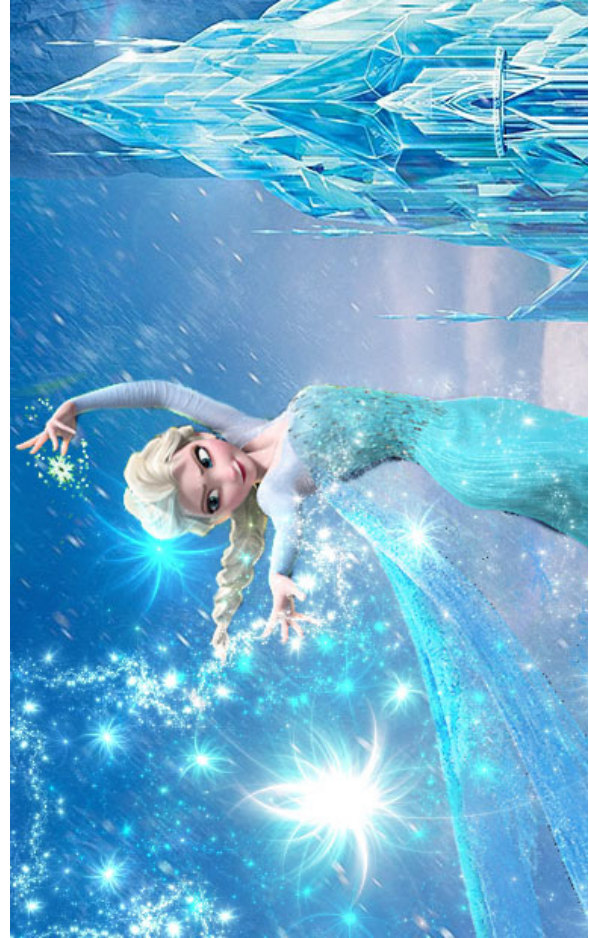


```
\placeimage[frame=true, scale=0.75]{elsa}
```





```
\placeimage[scale=2, landscape=true]{elsa}
```



```
\placeimage[caption=Elsa]{elsa}
```



그림 1: Elsa

```
\PlaceImageSetup{legend=true}  
\placeimage[caption=Elsa]{elsa}
```



Elsa

```
\placeimage[showfilename=true]{elsa.jpg}
```



elsa.jpg

# 전역 (global) 변수와 지역 (local) 변수

---

```
\ExplSyntaxOn
\DeclareDocumentCommand \foo { m m }
{
\tl_set:Nn \l_tmpa_tl {#1}
\l_tmpa_tl \quad
\group_begin:
\tl_set:Nn \l_tmpa_tl {#2}
\l_tmpa_tl \quad
\group_end:
\l_tmpa_tl
}
\foo{A}{B}
\ExplSyntaxOff
```

A B A

|

## 해묵은 것들과의 조화 또는 불화

---

이것은 괜찮지만,

```
\ExplSyntaxOn
\dim_gset:Nn \paperwidth {210mm}
\ExplSyntaxOff
```

이것은 진퇴양난처럼 보이는데, 틸데(~)를 써서 해결할 수 있고,

```
\ExplSyntaxOn
\NewDocumentCommand \foo {m}
{
  \tikz \draw[thick,rounded~corners=#1](\l_tmpa_tl)--(\l_tmpb_tl)
}
\ExplSyntaxOff
```

이것은 오류를 일으킨다. (xkeyval 때문이다.)

```
\usepackage{polyglossia}
\ExplSyntaxOn
\setmainlanguage[spelling=new]{german}
\ExplSyntaxOff
```

Expl3가 @를 만났을 때

```
\makeatletter
\ExplSyntaxOn
...
\ExplSyntaxOff
\makeatother
```

# Exp13가 대체하기 어려운 것들

---

- `\documentclass` 전에 올 수 있는 것들. 예를 들자면, `\newif`로 만들어진 것들.
- `\newcounter{my}[chapter]`

옛 것은 옛 방식으로, 새 것은 새 방식으로.