

L^AT_EX 연습 기록

ischo

2006년 9월 29일

요약

$\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 에서 워드프로세서의 문서작성 방식을 구현하는 방법을 설명한 *L^AT_EX for Word Processor Users*를 보면서 연습하던 기록이다. 정확히 기억이 나지는 않지만, 2년쯤 전에 공부하던 것이어서, $\text{H}^{\text{E}}\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 으로 작성했던 것을 언제인가 dhucs로 컴파일할 수 있게 바꾸었던 것으로 보인다. 처음 사용자에게 부분적으로나마 도움이 되기를 바란다.

차례

제 1 장	Wordprocessor 문서작성을 L^AT_EX에서 구현한다.	2
제 1 절	item 머리부호	2
제 2 절	enumerate 패키지	2
제 3 절	pifont 패키지	3
제 4 절	footnote	3
제 5 절	또 다른 원문자	4
제 6 절	내 번호만들기 (그리고 list 환경)	4
제 2 마당	latex4wp 계속	6
제 1 저절	장.절 이름바꾸기	6
제 2 저절	index 만들기	6
제 3 절	난외주(margin note) 만들기	7
제 4 절	글상자 만들기	7
제 5 절	color 패키지로 색 있는 글 만들기	8
제 6 절	tabbing 환경의 거의 모든 것	9
제 7 절	tabular 환경으로 표 그리기	10
제 8 절	myulem 패키지로 한글문서 밑줄 긋기	10
제 9 절	상하 수식	11
제 10 절	수식 지우기	11
제 11 절	표의 대각선	12

제 1 장

Wordprocessor 문서작성을 L^AT_EX 에서 구현한다.

제 1 절 item 머리부호

여러가지 item 머리부호를 시도해 보자.

- ⊙ `\item[{\circledcirc}]`
- `\item[{\circ}]`
- `\item[{\textbullet}]`
- `\item[{\tiny\circ}]`
- ※ `\item[{\textreferencemark}]`
- ★ `\item[{\bigstar}]`
- * `\item[{\ast}]`

제 2 절 enumerate 패키지

(예 1) `\begin{enumerate}[(예 1)]`와 같이 `enumerate` 뒤에 옵션을 준다.¹

(예 2) 로마자 번호를 원하면: `\begin{enumerate}[항목 i.] \item ... \end{enumerate}`

항목 i. 편리한가?

항목 ii. 편리한데.

(예 3) `\renewcommand{\labelenumii}{항목 \roman{enumii}.}`를 써도 같은 결과를 얻는다.

¹[(예 1)]을 입력하면 그 후의 번호는 자동으로 처리된다.

제 3 절 pifont 패키지

dingbat 문자
'빈칸'

...dingbat 문자

- ① 원문자 번호: `\begin{dingautolist}{172} ... \item ... \end{dingautolist}`
- ② pifont에서의 variation: `\ding{172}` 대신 `{182}`, `{192}`, `{202}`도 try해 볼 것.⁷
- ③ `\renewcommand{\labelenumi}{\onum\enumi}`로도 구현가능. ~ [아래의 예]를 보라.²
 ▷ `\ding`으로 작성한 원번호를 ③처럼 불러올 수 있다.

~
▶ `\RHD`
각주 번호

[아래의 예] ... verbatim 환경: `\fbox{What I typed.}`와 같이 보이게 하는 방법

- ① `\verb*+\fbox{What I typed.}+` ('+'는 '*'와 '빈칸' 이외의 문자로 대체 가능.)
- ② `\begin{verbatim*} \fbox{What I typed.} \end{verbatim*}`³
`\fbox{What I typed.}`라고 typeset하면 아래의 output처럼 나온다.

output: What I typed.

- ③ `verbatim` 또는 `verb` 뒤에 오는 '*'를 빼면 빈칸 표시 ' '를 하지 않는다.
 ▶ `\RHD \renewcommand{\labelenumi}{\onum\enumi}`로 작성한 원번호도 불러올 수 있는데,
 그 때는 `\textcircled{\ref{num:onum}}`로 불러와야 ③처럼 제대로 된 원번호를 얻는다.
 주의할 점은 그냥 `\ref{num:onum}`으로 불러오면 3처럼 원하는 것과 다른 결과를 얻는다.

제 4 절 footnote

- ④ 이 항목은 `\begin{dingautolist}{185}`으로 시작했다.⁴

typeset: `\ding{182} \ding{188} \ding{178} \ding{198} \ding{208}`

output: ① ⑦ ⑦ ⑦ ⑦

▶ 위 예는 `\begin{tabbing} ... \end{tabbing}`으로 작성되었다.

- ⑤ 다른 모양의 각주 번호를 표시해 보자.⁵

`\renewcommand{\thefootnote}{\Roman{footnote}}\footnote{어떻게 나올까?}`

▶ 실제로 이 문서에서는, 먼저 preamble에서 `\newfootnote`를 다음과 같이 정의했다.

`\def\newfootnote {\renewcommand{\thefootnote}{\Roman{footnote}}}`

그리고, 이 곳에서 `\newfootnote\footnote{어떻게 나올까?}`를 입력하였다.

⁷`\footnote[n]{ text }`와 같이 임의의 각주번호 [n]를 줄 수 있지만 자동count되지는 않는다.
²`\onum`은 `hangul`에서만 작동된다. `pifont`에서는 10번까지, `hangul`에서는 15번까지 numbering이 가능하다.
³`verbatim`환경 또는 `verb`명령은 다른 명령인자 안에서 사용할 수 없다!!!
⁴`\ding{182}`로 하면 1번부터 시작한다.
⁵어떻게 나올까?

- ⑥ 맞춤식 각주표시^{Read me!}입니다.

```
\renewcommand{\thefootnote}{\sf Read me!}\footnote{...}
```

맞춤식
각주표시

- ⑦ 또 다른 맞춤식 각주표시[‡]입니다.

```
\renewcommand{\thefootnote}{\$ \ddagger$}\footnote{...}
```

- ⑧ 이것은?^{viii}

```
\myfootnotenum{\roman{footnote}}\footnote{어떻게 나올까?}
```

▶ \myfootnotenum은 preamble에서 다음과 같이 정의되었다.

```
\newcommand{\myfootnotenum}[1]{\renewcommand{\thefootnote}{#1}}
```

- ⑨ 또 이것은?^{**}

```
\myfootnotenum{\$ \ast \ast$}\footnote{이건 또 어떻게 나올까?}
```

- ⑩ \fnsymbol{footnote}로 각주표시를 * † ‡ § ¶ || ** [8] [9]로 순서가 정해진 9가지 문자로 바꿀 수 있다.[¶]

```
\renewcommand{\thefootnote}{\fnsymbol{footnote}}%
```

```
\footnote[5]{다섯번째 부호인 \P는 [5]로 옵션을 넣어 얻는다.}
```

제 5 절 또 다른 원문자

- ① is typeset by \textcircled{a} ... ②③④⑤⑥⑦
- ①②③④⑤ \textcircled{1} ...
- ①②③④⑤ \ding{172} ...

제 6 절 내 번호만들기 (그리고 list 환경)

내 번호 A 나 보기가 역겨워 가실 때에는 사뿐히 즈려밟고 가시옵소서. 영변에 약산 진달래꽃 아름
따다 가실 길에 뿌리오리다. 나 보기가 역겨워 가실 때에는 죽어도 아니 눈물 흘리오리다.

내 번호 B 김소월의 시입니다.

```
\newcounter{myone}
\begin{list}{\bf 내 번호 \Alph{myone}}
{\usecounter{myone}}
\item%=====
```

Read me! 이것이 맞춤식 footnote 표시이다.

‡ 이것이 또 다른 맞춤식 footnote 표시이다.

viii 어떻게 나올까?

** 이건 또 어떻게 나올까?

¶ 다섯번째 부호인 ¶는 [5]로 옵션을 넣어 얻는다.

나 보기가 역겨워 가실 때에는 아니 눈물 흘리오리다.

`\item%=====`

김소월의 시입니다.

`\end{list}`

제 2 마당

latex4wp 계속

제 1 저얼 장·절 이름바꾸기

- 장(chapter)과 절(section)의 이름은 다음과 같이 바꾼다.

```
\renewcommand{\chaptername}{마당}
\renewcommand{\sectionname}{저얼}
\chapter{latex4wp 계속}
\section{장$\cdot$절 이름 바꾸 기}
. . . . .
\section{index 만들 기}
. . . . .
\renewcommand{\chaptername}{장}
\renewcommand{\sectionname}{절}
```

- 다음의 이름을 바꿀 수 있다.

<code>\abstractname</code>	<code>\appendixname</code>	<code>\bibname</code>	<code>\contentsname</code>
<code>\chpatername</code>	<code>\sectionname</code>	<code>\figurename</code>	<code>\headpagename</code>
<code>\indexname</code>	<code>\listfigurename</code>	<code>\listtablename</code>	<code>\pagename</code>
<code>\partname</code>	<code>\refname</code>	<code>\tablename</code>	<code>\alsoseenname</code>

제 2 저얼 index 만들기

- 1) 기초작업: 색인을 만들기 위해서는 preamble에 다음을 넣는다.

```
\usepackage{makeidx}
\makeindex
\newcommand{\myindex}[1]{\index{#1}#1}
%\index{...}만 하면 본문에서 {...}가 보이지 않는다.
```

그리고 색인인쇄 명령 `\printindex`를 원하는 위치(보통 맨 뒤)에 넣는다.

2) 색인단어 선택: `\myindex{색인 만들 기}`를 입력하면 ‘색인만들기’가 색인에 나타난다.

색인만들기
안보이는 단어
보이는 단어

▶ `\index{색인 만들 기}`를 입력해도 색인은 만들어지지만 본문의 내용에는 ‘색인만들기’출력되지 않는다. 다음의 예를 보자.

예 1) `\index`를 사용하면 `[[[]]]`처럼 색인단어가 안보인다.

`\verb+\index+`를 사용하면 `[[\index{안보이는 단어}]]`처럼 색인단어가 안보인다.

예 2) `\myindex`를 사용하면 `[[보이는 단어]]`처럼 색인단어가 안보인다.

`\verb+\myindex+`를 사용하면 `[[\myindex{보이는 단어}]]`처럼 색인단어가 보인다.

3) 색인파일 만들기: 색인을 인쇄하기 위해서는 `\printindex`가 필요하지만 이것만으로는 충분하지 않다. 색인을 얻으려면 먼저 `[filename].idx`를 만든 후 `latex`을 실행해야 한다. 따라서 색인을 얻으려면 command prompt에서 다음과 같은 순서로 컴파일한다.

```
latex filename      ... 페이지 등 모든 정보를 update한다.
makeindex filename ... 색인파일 filename.idx를 얻는다.
latex filename      ... \printindex가 있는 위치에 idx파일을 삽입하여 문서를 완성한다.
```

4) `\usepackage{makeidx,showidx}`: `makeidx`와 함께 `showidx` 패키지를 사용하면 색인단어로 선택된 단어를 해당 페이지의 여백에 표시해 주므로 색인작업을 확인하는 데 매우 편리하다.

제 3 절 난외주(margin note) 만들기

- 굉장히 쉽다고 한다.

```
\marginpar{\fbox{\footnotesize 정말 쉬울 까?}}
```

(이 위치에 아래의 typeset에 의한 두 개의 난외주가 들어가 있다.)

난외주는 `\marginpar{...}`로 작성하면 된다.

```
\marginpar{\footnotesize \sf \자모소라 쉽네.
```

```
그러나 \\ margin note를 \\ 상자 안에 넣는 등의 작업도 \\ 쉬운 일일까?}
```

아래와 같이 선을 그어 필요한 효과를 얻을 수 있다.

```
\marginpar{\rule\marginparwidth{.5pt}\vskip 1pt 여백글}
```

정말 쉬울까?

쉽네. 그러나 margin note를 상자 안에 넣는 등의 작업도 쉬운 일일까?

여백글

제 4 절 글상자 만들기

- boxedminipage 환경을 이용한 글상자 만들기 예입니다.

안녕하십니까? 블랑카입니다. 우리나라에는 많은 외국인 노동자가 있습니다. 그 분들이 좋은 추억만 가지고 돌아가게 되기를 바라겠습니다.

```
\begin{boxedminipage}[c]{0.5\linewidth}
안녕하십니까? 블랑카입니다.
우리나라에는 많은 외국인 노동자가 있습니다.
그 분들이 좋은 추억만 가지고 돌아가게 되기를 바랍니다.
\end{boxedminipage}
```

- 글상자 만들기를 연습하기로 한다.

$f(x) = ax^2 + bx + c$	나 보기가 역겨워 가실 때에는 사뿐히 즈려밟고 가시옵소서. 영변에 약산 진달래꽃 아름따다 가실 길에 뿌리오리다. 나 보기가 역겨워 가실 때에는 죽어도 아니 눈물 흘리오리다.
------------------------	--



```
\begin{boxedminipage}[t]{0.35\linewidth}
\vspace{-5mm}
\[ f(x)=ax^2+bx+c \]
\end{boxedminipage}
%-----
\begin{boxedminipage}[t]{0.65\linewidth}
나 보기가 역겨워 가실 때에는 사뿐히 즈려밟고 가시옵소서.
영변에 약산 진달래꽃 아름따다 가실 길에 뿌리오리다.
나 보기가 역겨워 가실 때에는
죽어도 아니 눈물 흘리오리다.
\end{boxedminipage}
```

- \framebox 또는 \fbox로 글을 상자로 둘러싸기:

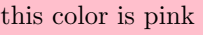

\framebox로 쓴 글상자	<code>\framebox{\backslash}</code> <code>\texttt{\framebox}로 쓴 글상자</code>
4.5cm 글상자	<code>\framebox[4.5cm]{4.5cm 글상자}</code>
4.5cm 글상자	<code>\framebox[4.5cm][r]{4.5cm 글상자} %오른 정렬</code>
\fbox 글상자: variation	<code>\setlength\fboxrule{1.5pt} %box 테두리 굵기 조정</code> <code>\fbox{\backslash}</code> <code>\texttt{\fbox} 글상자: variation</code>
\framebox 글상자: variation	<code>\setlength\fboxsep{4mm} %상자와 글 사이 간격</code> <code>\framebox{\backslash}</code> <code>\texttt{\framebox} 글상자: variation</code>

제 5 절 color 패키지로 색 있는 글 만들기

- 색깔의 기본값은 black, white, red, green, blue, cyan, magenta, yellow이다.

	
<pre>\colorbox{red}{red} \colorbox{cyan}{cyan} \colorbox{magenta}{magenta} \colorbox{yellow}{yellow}</pre>	<pre>\fcolorbox{black}{red}{red} \fcolorbox{black}{blue}% {\bf\color{yellow}{blue}} \fcolorbox{red}{yellow}% {yellow with red frame}</pre>

- You define new color:

 	<pre>\definecolor{pink}{rgb}{1, .75, .8} \colorbox{pink}{this color is pink} \\ \definecolor{color1}{rgb}{.5, .5, .5} \colorbox{color1}{\color{white}{this color is color1}}</pre>
--	--

제 6 절 tabbing 환경의 거의 모든 것

- | | | |
|----------------------------------|----------------------|---------------------------|
| x. 이 위에 | 1. killed 탭 정의가 | 2. 있다. |
| x. push-x | 1. original | 2. original |
| w. | 1. | 2. space 2.1 new--새로 탭 설정 |
| w. 바로 위 탭들을 | 1. 그대로 | 2. 이용 2.1 한다. |
| y. push-y (1.a) new--바로 위 탭1. 조정 | 2. old 2.1.a new--조정 | 2.b new--설정 |
| y. (1.a) 그대로 이용 | 2. 2.1.a | 2.b |
| y. (1.a) | 2. 2.1.a | 2.b |
| y. (1.a) | | |
| 0. 모든 탭을 새로 설정한다 | A. new--설정 | B. new--설정 |
| 0. 위의 탭을 | A. 그대로 | B. 이용 |
| y. pop-y (1.a) 최근 저장 탭 불러오기 | 2. 2.1.a | 2.b |
| x. pop-x | 1. 이전 것 불러오기 | 2. original |

```
\begin{tabbing}
\hspace*{35mm}   \= 안보이는 wide column   \= \hspace{10mm}   \kill % suppresses the print
x. 이 위에      \> 1. killed 탭 정의가   \> 2. 있다.  \\
x. push-x      \> 1. original             \> 2. original \\
\pushtabs % saves tab positions
w.             \> 1.                       \> 2. space \=2.1 new--새로 탭 설정 \\
w. 바로 위 탭들을 \> 1. 그대로             \> 2. 이용 \>2.1 한다.  \\
y. push-y \= (1.a) new--바로 위 탭1. 조정 \> 2. old \= 2.1.a new--조정 \=2.b new--설정 \\
\pushtabs % saves TAB
y.             \> (1.a) 그대로 이용 \> 2.   \> 2.1.a \> 2.b  \\
y.             \> (1.a)                   \> 2.     \> 2.1.a       \>2.b  \\
y.             \> (1.a)                   \\
0. 모든 탭을 새로 설정한다 \= A. new--설정 \= B. new--설정 \\
0. 위의 탭을             \> A. 그대로   \> B. 이용       \\
\poptabs % restore TAB --- Nearest First
y. pop-y \> (1.a) 최근 저장 탭 불러오기 \> 2.     \>2.1.a       \>2.b  \\
\poptabs % restore saved tab positions
```

```
x. pop-x      \> 1. 이전 것 불러오기 \> 2. original \\
\end{tabbing}
```

제 7 절 tabular 환경으로 표 그리기

- 한 예를 보자.

가	나	다	라
<code>\abstractname</code>	<code>\appendixname</code>	<code>\bibname</code>	<code>\contentsname</code>
<code>\chpatername</code>	<code>\sectionname</code>	<code>\figurename</code>	<code>\headpagename</code>
<code>\partname</code>	<code>\refname</code>	<code>\tablename</code>	<code>\pagename</code>

이 표는 아래와 같이 작성한 것이다.

```
\begin{tabular}[htb]{|l|l|l|l|} \hline
가 & 나 & & 다 & & 라 & \\ \hline \hline
\verb+\abstractname+ & & & \verb+\appendixname+ & & & \\
& \verb+\bibname+ & & \verb+\contentsname+ & & & \\ \hline
\verb+\chpatername+ & & & \verb+\sectionname+ & & & \\
& \verb+\figurename+ & & \verb+\headpagename+ & & & \\ \hline
\verb+\partname+ & & & \verb+\refname+ & & & \\
& \verb+\tablename+ & & \verb+\pagename+ & & & \\ \hline
\end{tabular}
```

제 8 절 myulem 패키지로 한글문서 밑줄 긋기

- H_AT_EX에서 아래와 같이 밑줄을 그으려면 (ulem package는 한글에서 작동하지 않으므로) 도은이 아빠가 ulem 패키지를 한글환경에 맞게 고친 myulem.sty을 preamble에서 다음과 같이 불러와야 한다.¹

```
\usepackage[normalem]{myulem}
```

underline은 <code>\huline</code> 으로 긋고	<code>\huline{underline은 \huline으로 긋고}\</code>
겹밑줄은 <code>\huuline</code> 으로 긋고	<code>\huuline{겹밑줄은 \huuline으로 긋고}\</code>
파도밑줄은 <code>\hwave</code> 로 긋고	<code>\hwave{파도밑줄은 \hwave로 긋고}\</code>
자운 흔적은 <code>\hsout</code> 으로,	<code>\hsout{자운 흔적은 \hsout으로,}\</code>
삭제는 \hxout으로 나락낸다.	<code>\hxout{삭제는 \hxout으로 나락낸다.}</code>

영어 문장에 밑줄을 칠 때는, myulem 패키지에서와 같이 해도 되고, ulem package에서 사용하는 방법대로, `\uline`, `\xout`등과 같이 첫자 'h'를 빼고 사용해도 된다.

¹<http://faq.ktug.or.kr/mywiki/Karnes>에서 myulem 패키지를 구하면 된다.

<p>underline for English</p> <p><u>this is double underline</u></p> <p>this is wave underline</p> <p>this is wrong</p> <p>You can cross out.</p>	<pre>\uline{underline for English}\ \uuline{this is double underline}\ \uwave{this is wave underline}\ \sout{this is wrong}\ You can \xout{cross out}.</pre>
--	--

제 9 절 상하 수식

- `\stackrel`:

$$\vec{x} \stackrel{\text{def}}{=} (x_1, x_2, \dots, x_n)$$

$$A \xrightarrow{\alpha'} B \xleftarrow{\beta'} C$$

```
\[ \vec{x} \stackrel{\rm def}{=}
(x_1, x_2, \dots, x_n) b \]
\[ A \stackrel{\alpha'}{\longrightarrow} B
\stackrel{\beta'}{\longleftarrow} C \]
```

- `\choose`:

$$\binom{n+1}{k} = \binom{n}{k} + \binom{n}{k-1}$$

```
\[ {n+1 \choose k}
={n \choose k} + {n \choose k-1} \]
```

- `\atop`:

$$\sum_{\substack{k_0, k_1, \dots \geq 0 \\ k_0 + k_1 + \dots = 0}} a_{k_0} a_{k_1} \dots$$

```
\[ \sum_{k_0, k_1, \dots \ge 0}
\atop k_0 + k_1 + \dots = 0}
a_{k_0} a_{k_1} \dots \]
```

제 10 절 수식 지우기

- 다음과 같이 한 문자를 지운다.

$$a + b - a = b$$

```
\[ a \hspace{-5pt} / + b - a \hspace{-5pt} / = b \]
```

- 패키지 `cancel`을 쓰면 다양한 표현이 가능하다. 필요로 한다.

※ `\usepackage[Smaller]{cancel}` 옵션 `thicklines`을 쓰면 상쇄 표시가 진하게 됨 옵션 `Smaller`를 쓰면 `\cancelto{B}{A}`의 B의 글씨가 `scriptsize`가 됨

$$\begin{aligned}
 & \cancel{(a-b)^2} + b - \cancel{(a^2 - 2ab + b^2)} = b \\
 & \cancel{(a-b)^2} + b - \cancel{(a^2 - 2ab + b^2)} = b \\
 & \cancel{(a-b)^2} + b - \cancel{(a^2 - 2ab + b^2)} = b \\
 & b + \cancel{(a^2 - 2ab + b^2)} \xrightarrow{(a-b)^2} = b + (a-b)^2
 \end{aligned}$$

```

\begin{align*}
\cancel{(a-b)^2} + b \ \cancel{-(a^2-2ab+b^2)} = b \quad \backslash\ [.5em]
\bcancel{(a-b)^2} + b \ \bcancel{-(a^2-2ab+b^2)} = b \quad \backslash\ [.5em]
\xcancel{(a-b)^2} + b \ \xcancel{-(a^2-2ab+b^2)} = b \quad \backslash\ [.5em]
b + \cancelto{(a-b)^2}{(a^2-2ab+b^2)} = b + (a-b)^2 \quad \backslash\
\end{align*}

```

제 11 절 표의 대각선

- 대각선을 그리려면 slashbox 패키지를 이용한다.

slashbox.sty

$i \backslash j$	4	5	6	7	v_i
1	* 6	7 7	* 3	5 7	0
2	-1 8	* 3	* 6	* 5	-3
3	-5 5	0 4	-2 5	* 6	-4
v_j	6	0	3	2	

```

\centerline{\texttt{slashbox.sty}}
\bigskip
\begin{tabular}{c|c|c|c|c|c} \hline
\backslashslashbox{$i$}{$j$} & 4 & 5 & 6 & 7 & \\
1 & \slashbox{$*$}{6} & \slashbox{7}{7} & \slashbox{$*$}{3} & \slashbox{5}{7} & \\
2 & -1 & * & * & * & -3 \\
3 & -5 & 0 & -2 & * & -4 \\
\hline
v_j & 6 & 0 & 3 & 2 & \\
\hline
\end{tabular}

```

```
2 & \slashbox{-1$}{8} & \slashbox{*$}{3} & \slashbox{*$}{6}
      & \slashbox{*$}{5} & $-3$ \\ \hline
3 & \slashbox{-5$}{5} & \slashbox{0}{4} & \slashbox{-2$}{5}
      & \slashbox{*$}{6} & $-4$ \\ \hline\hline

&&&&& \\[-1ex]
$v_j$ & 6 & 0 & 3 & 2 & \\[-1ex]
&&&&& \\ \hline
\end{tabular}
```

참고 문헌

- [1] Gonzato, Guido. *LaTeX for Word Processor Users*. CTAN, 2001.
- [2] Gonzato, G. *LaTeX for Word Processor Users*. CTAN, 2001.

찾아보기

【 기호 】	
▶ \RHD.....	3
.....	3
‘빈칸’.....	3
【 D 】	
dingbat 문자.....	3
【 E 】	
enumerate.....	2
【 I 】	
item 머리부호.....	2
【 ㄱ 】	
각주 번호.....	3
【 □ 】	
맞춤식 각주표시.....	4
【 ㅂ 】	
보이는 단어.....	7
【 ㅅ 】	
색인만들기.....	7
【 ○ 】	
안보이는 단어.....	7