

고급스러운 텍 사용의 첫 걸음: 환경과 커맨드의 이해

권현우

Sogang Univ.

November 30, 2017

- 커맨드와 환경 개념을 구분할 수 있다.
- 커맨드와 환경을 이용해서 프로그래밍을 할 수 있다.

커맨드와 환경

Definition (커맨드)

커맨드 (command)란 어떤 내용을 실행하기 위한 코드를 말한다.

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

Definition (환경)

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

예: `\begin{center}...\end{center}`

커맨드를 정의하기 위한 방법

```
\newcommand{이름}{내용}
```

예시:

```
\newcommand{\mynotes}{오늘은 공주대학교 라텍 워크샵입니다.}
```

....

```
\mynotes
```

커맨드를 정의하기 위한 방법: 파라미터 이해

파라미터는 매크로에 개수를 넣는 것이다.

```
\verb|\newcommand{이름} [파라미터개수] [기본] {내용}
```

예시:

```
\newcommand{\mynotes}[2] [오늘] {#1은 공주대학교 라텍 #2입니다.}
```

...

```
\mynotes{워크샵}
```

```
\mynotes [내일] {워크샵}
```

환경 정의

```
\newenvironment{이름}{내용전}{내용뒤}
```

예시:

```
\newenvironment{proof}{\noindent \textbf{증명}}  
{\vskip 10pt}
```

환경에도 파라미터 지정이 가능하다.

Remark: *의 유무

```
\newcommand{\foo}{...}
```

```
\newcommand*{\foo}{...}
```

```
\newenvironment{\foo}{...}{...}
```

```
\newenvironment*{\foo}{...}{...}
```

기타커맨드 관련 명령어

`\renewcommand`

`\providecommand`

`\DeclareRobustCommand`

예시:

```
\renewcommand{\emph}[1]{\sffamily #1}
```


* 명령어 정의

같은 철학을 가지고 있지만 *를 부여함으로써 다르게 출력되는 명령어를 만들고자 할 때는 다음과 같이 코딩하면 된다.

```
\makeatletter
```

```
\newcommand\foo{\@ifstar{\foo@xii}{\foo@xi}}
```

```
\newcommand\foo@xii[1]{Hi #1}
```

```
\newcommand\foo@xi[1]{Hello #1}
```

```
\makeatother
```

Optional argument

```
\newcommand*{\emphasis}[2] [\@empty] {%  
  \ifx#1\@empty  
    \emph{#2}  
  \else  
    \textsf{#2}  
  \fi
```

```
\emphasis{Hello} \emphasis[*]{안녕}
```

참고: ifthen, ttwopt, pgfkeys....

카운터에 대한 이해

라텍의 주요강점을 하나 든다면 ‘상호참조’이다. 상호참조를 하기 위해서는 카운터에 대한 이해가 필요하다.

- `\newcounter{카운터명}`
- `\stepcounter{카운터명}`
- `\refstepcounter{카운터명}`
- `\setcounter{카운터명}{숫자}`
- `\addtocounter{카운터명}{숫자}`

카운터 출력 관련 명령어

- `\the카운터이름`
- `\value, \arabic, \alph, \Alph`

`\roman, \Roman, \fnsymbol`