

How To LaTeX

1강 – LaTeX 기초

201811206 황인탁

September 23, 2022

- TeX, LaTeX이 무엇인지 안다.
- 텍으로 과제를 만들어서 제출할 수 있다.

한 강의당 1시간-1시간 30분 정도의 분량으로 진행될 예정이다. 각 강의의 주제는 다음과 같다.

1강 LaTeX 기초

2강 수식, 그림, 그리고 표

3강 LaTeX 프로그래밍

TeX이 뭔가요?

TeX, LaTeX, pdfTeX... 어려운 말들이 너무 많다!

TeX Donald Knuth가 1978년에 만든 조판 시스템(언어 + 컴파일러).

LaTeX Leslie Lamport가 1984년에 만든 조판 시스템. TeX 언어로 작성되어 있으며, TeX과도 당연히 호환된다.

pdfTeX, XeTeX, LuaTeX TeX 언어의 구현체들.

TeX을 왜 쓰나요??

TeX의 장점...

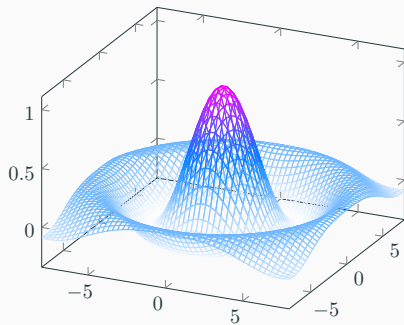
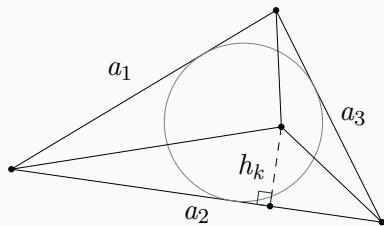
- 수식이 포함된 문서를 작성하는 데 최고다.
- 문서의 관리가 편하다.
- 내용과 스타일을 분리할 수 있다.
- 간지난다.

그럼에도 불구하고, TeX의 단점...

- 배우기 어렵다.
- 안돼도 왜 안되는지 모른다. / 돼도 왜 되는지 모른다.

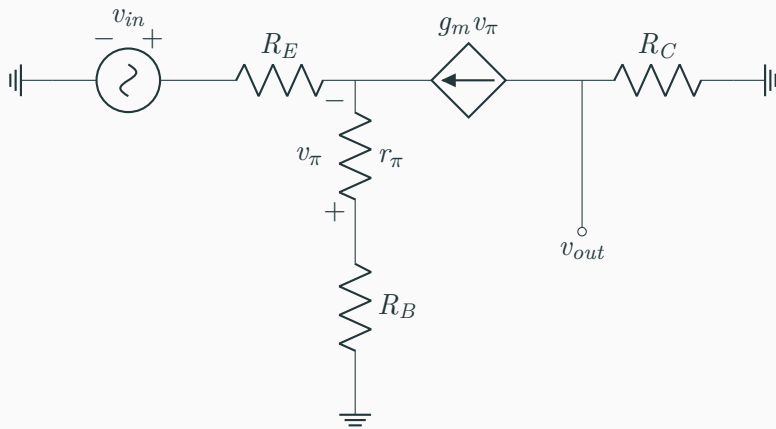
TeX으로 무엇을 할 수 있나요?

TikZ, pgfplots를 이용한 그림과 플로팅...



TeX으로 무엇을 할 수 있나요?

다이어그램...



TeX으로 무엇을 할 수 있나요?

Syntax Highlighting...

```
package main
```

```
import "fmt"
```

```
func main() {  
    fmt.Println("Hello, World!")  
}
```


따라서 한 번 쳐보자.

```
\documentclass{article}
```

```
\usepackage{kotex}
```

```
\begin{document}
```

```
Hello, World! 안녕, 세계!  $a^2 + b^2 = c^2$ 
```

```
\[ E = mc^2 \]
```

```
\end{document}
```

`\begin{document}` 이전까지의 부분은 **preamble**이라고 부른다.

preamble에는 보통 다음의 명령어들이 온다.

- `\documentclass`: 이 문서의 클래스를 결정한다. 클래스에 따라서 문서의 모양과 형식이 달라진다. 유명한 클래스로는 article, book, beamer, 그리고 앞으로 사용할 oblivoir 등이 있다.
- `\usepackage`: 사용할 패키지들을 임포트한다. 텍에서의 패키지는 다른 언어의 패키지들과 유사하다.
- 기타 매크로 및 설정 (자세한 것은 3강에서 다룬다.)

LaTeX에서는 이어지는 화이트스페이스를 전부 하나의 빈 칸으로 대체한다.

문단을 나눌 때는 반드시 한 줄을 비워놓는다. 화이트스페이스와 마찬가지로, 여러 줄을 비워놓아도 한 줄을 비운 것으로 인식한다.

```
So,soyouthink  
youcantell
```

```
Heavenfromhell?
```

```
Blueskiesfrompain?
```

```
So, so you think you can tell
```

```
Heaven from hell?
```

```
Blue skies from pain?
```

다음의 문자들은 특수한 용도가 있으므로 일반적으로 입력할 수 없다.

\$ % ^ & _ { } ~ \

입력하고 싶다면, 앞에 \를 붙이면 된다. (\#, \\$, ...)

\는 \textbackslash로 입력한다.

LaTeX의 명령어

LaTeX은 프로그래밍 언어이므로, 명령어의 개념이 있다. LaTeX의 명령어는 `command`와 `environment`로 분류할 수 있다.

`command`는 다음과 같은 형식을 가진다.

`\command[option1, option2, ...]{arg1}{arg2}...`

argument가 하나일 경우, 대괄호를 생략할 수 있다. 이 경우 토큰을 하나만 먹는다.

<code>\sqrt {2}</code>	$\sqrt{2}$
<hr/>	
<code>\sqrt 2</code>	$\sqrt{2}$
<hr/>	
<code>\sqrt 23</code>	$\sqrt{23}$

environment는 다음과 같은 형식을 가진다.

```
\begin{environment}
```

```
...
```

```
\end{environment}
```

environment는 `\begin`과 `\end`로 둘러싸인 구간에서 작동한다.

다음은 쳐 보자.

```
\begin{center}
  \textbf{Centered Text}:  $\sqrt{2}$ ,  $\sqrt[3]{2}$ 
\end{center}
```

다음과 같이 텍스트에 효과를 넣을 수 있다.

<code>\textbf {text}</code>	text
-----------------------------	-------------

<code>\textit {text}</code>	<i>text</i>
-----------------------------	-------------

<code>\textsl {text}</code>	<i>text</i>
-----------------------------	-------------

<code>\texttt {text}</code>	text
-----------------------------	------

다음과 같이 텍스트의 크기를 바꿀 수 있다.

<code>\tiny</code> text	text
<code>\small</code> text	text
<code>\normalsize</code> text	text
<code>\Large</code> text	text
<code>\Huge</code> text	text

이 효과는 문서 전체에 적용된다. 텍스트를 원래 크기로 되돌리려면 다시 `\normalsize` 명령어를 사용하여야 한다.

정렬을 변경하려면, flushleft, flushright, center 환경을 사용한다.

```
\begin{flushleft}
```

```
left aligned
```

```
\end{flushleft}
```

```
\begin{center}
```

```
center aligned
```

```
\end{center}
```

```
\begin{flushright}
```

```
right aligned
```

```
\end{flushright}
```

left aligned

center aligned

right aligned