

T_EX 사용의 입문

신주형

Sejong Academy of Science and Arts

April 14, 2018

What is T_EX?

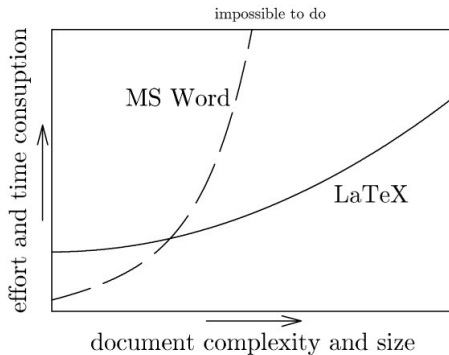
- Donald Knuth가 만든 문서 조판 언어.
- WYSIWYG 방식의 워드 프로세서가 아님.
- 명령어를 치고 컴파일하면 예쁘게 조판해 주는 형식.

Why T_EX is Good?

- 수식 환경의 간편함
- 문서의 질이 상승
- 생각보다는 간편하다.
 - ▶ 매크로의 활용
 - ▶ 패키지의 활용
 - ▶ 문서 양식의 활용(Internet에서 get)
 - ktug wiki에 자세한 내용이 있다.
 - google에 물어봐도 자세한 내용이 있다.
 - 그 외의 세부적 내용은 해당 패키지의 도움말을 찾아볼 것.

Why T_EX is Good?

다시 말하면요...



Starting T_EX

- 프로그래밍의 요소가 섞여 있다는 것을 유념할 것.
- 특히 T_EX에서는 '환경' 개념이 아주 중요함
 - ▶ 환경이란 문서 내부에서 특징적으로 지정해주는 것을 의미
(ex: 수식 환경, 가운데 정렬 환경, ...)

T_EX 문서의 구조

- 1 Preamble(전처리구): 문서의 설정, 양식, Package 등을 첨부
 - ▶ `\documentclass[10pt,a4papersize]{article}`: 문서의 종류를 설정
 - ▶ `\usepackage{tikz}`: Package 추가(`\include <stdio.h>`)
 - ▶ 기타 여러 가지가 있으나 (ex: `\usecolortheme{beaver}`) 여기까지는 몰라도 됨 ...
- 2 Body(본문): 본문
 - 전처리구를 깔끔히 한 파일로 만들고(즉, 문서 양식을 만들고) 본문은 `\include{file}`을 통해 다른 파일에 쉽게 쓸 수 있다.(우리의 목표)

문서 만들기

Hello World 쓰기

- 목표: Hello World를 출력해보자.
- Hint: 문서를 만들기 위해서는 `\documentclass{...}`로 문서 종류를 정하고 문서 환경을 `\begin{document} ... \end{document}`로 만들어야 함.

Source Code

Hello, World!

```
1 \documentclass{article}
2 \begin{document}
3   Hello, World!
4 \end{document}
```


문서 만들기

안녕하세요 쓰기

- 목표: 안녕하세요를 출력해보자.
- Hint: 아까와 똑같이 하면... 될까?

Source Code

안녕하세요.

```
1 \documentclass{article}
2 \usepackage{kotex}
3 \begin{document}
4   안녕하세요.
5 \end{document}
```

문서 만들기

안녕하세요 쓰기

- `\kotex` 패키지를 쓰면 한글을 쓸 수 있음.
- 한글 전용 환경인 `\oblivoir` 환경을 쓰는 것을 추천.

Source Code

안녕하세요.

```
1 \documentclass{oblivoir}  
2 \begin{document}  
3   안녕하세요.  
4 \end{document}
```

문서 만들기

'큰' 안녕하세요 쓰기

- 목표: 아까보다 크기가 큰 안녕하세요를 출력해보자.
- Hint: `\large` 명령어를 사용해보자.

Source Code

안녕하세요. 안녕하세요. 안녕하세요.

안녕하세요. 안녕하세요.

```
1 \documentclass{oblivoir}  
2 \begin{document}  
3   안녕하세요.  
4   \large{안녕하세요.}  
5   \Large{안녕하세요.}  
6  
7   \huge{안녕하세요.}  
8   \Huge{안녕하세요.}  
9 \end{document}
```

문서 만들기

'큰' 안녕하세요 쓰기

- T_EX에서는 폰트 크기 변경이 자유롭지는 않음.
- 문서를 결정할 때 기본 폰트 크기를 같이 결정할 수 있음
(`\documentclass[12pt]{oblivoir}`)
 - ▶ 단, 10pt, 11pt, 12pt만 가능함.

문서 만들기

노래 가사 쓰기

■ 다음을 써 보자.

- ▶ 바보야 오늘은
안된다고 말하지마
오늘만큼은 내게도
꼭 기회를 줘
사랑스럽게 웃는 것도
예쁘게 말하는 것도
많이 연습했어
보고 싶어도 참으라고 하지 마

- ▶ (다음 문단)
Baby 난 좀 억지 부리는 것도
맞아
오늘도 내 눈물 연기로
받아낸 너와의 데이트
보고 싶어도
매일 꼭 참고 있어도
나는 네가 안아주기만 하면
사르르르르 녹아 Yeah

문서 만들기

노래 가사 쓰기

바보야 오늘은
안된다고 말하지마
오늘만큼은 내게도
꼭 기회를 줘
사랑스럽게 웃는 것도
예쁘게 말하는 것도
많이 연습했어
보고 싶어도 참으라고 하지 마

Baby 난 좀 억지 부리는 것도 맞아
오늘도 내 눈물 연기로
받아낸 너와의 데이트
보고 싶어도
매일 꼭 참고 있어도
나는 네가 안아주기만 하면
사르르르르 녹아 Yeah

- 결과적으로 우리는 이런 형태의 글을 쓰기를 원한다.
- 일단 한 번 시도해보자.

Source Code

바보야 오늘은 안된다고 말하지만 오늘날만큼 내게도 꼭 기회를 줘 사랑스럽게 웃는 것도 예쁘게 말하는 것도 많이 연습했어 보고 싶어도 참으라고 하지 마
Baby 난 좀 억지 부리는 것도 맞아 오늘도 내 눈물 연기로 받아낸 너와의 데이트 보고 싶어도 매일 꼭 참고 있어도 나는 네가 안아주기만 하면 사르르르 녹아
Yeah

```
1 \documentclass{oblivoir}
2 \begin{document}
3 바보야 오늘은
4 안된다고 말하지만
5 오늘날만큼은 내게도
6 꼭 기회를 줘
7 사랑스럽게 웃는 것도
8 예쁘게 말하는 것도
9 많이 연습했어
10 보고 싶어도 참으라고 하지 마
11
12 Baby 난 좀 억지 부리는 것도 맞아
13 오늘도 내 눈물 연기로
14 받아낸 너와의 데이트
15 보고 싶어도
16 매일 꼭 참고 있어도
17 나는 네가 안아주기만 하면
18 사르르르 녹아 Yeah
19 \end{document}
```

문서 만들기

노래 가사 쓰기-해결1

- 안타깝게도, 엔터 한 번으로는 줄바꿈이 되지 않음.
- 엔터 두 번이면 새로운 문단으로 인정하기 때문에 우리가 원하는 띄어쓰기와는 다름.
- 엔터 세 번부터는 엔터 두 번과 같은 효과
- 어떻게 하면 좋을까?

문서 만들기

노래 가사 쓰기-해결1

- T_EX에는 강제 줄바꿈 명령어가 있다.
- `\\`와 `\newline`이 그것
 - ▶ `\newline`은 진짜로 '강제' 줄바꿈이다.
 - ▶ `\\`를 이용하자.

Source Code

바보야 오늘은
안된다고 말하지마
오늘만큼은 내게도
꼭 기회를 줘
사랑스럽게 웃는 것도
예쁘게 말하는 것도
많이 연습했어
보고 싶어도 참으라고 하지 마

Baby 난 좀 억지 부리는 것도 맞아
오늘도 내 눈물 연기로
받아낸 너와의 데이트
보고 싶어도
매일 꼭 참고 있어도
나는 네가 안아주기만 하면
사르르르 녹아 Yeah

```

1 \documentclass{oblivoir}
2 \begin{document}
3 바보야 오늘은\\
4 안된다고 말하지마\\
5 오늘만큼은 내게도\\
6 꼭 기회를 줘\\
7 사랑스럽게 웃는 것도\\
8 예쁘게 말하는 것도\\
9 많이 연습했어\\
10 보고 싶어도 참으라고 하지 마\\
11
12 Baby 난 좀 억지 부리는 것도 맞아\\
13 오늘도 내 눈물 연기로\\
14 받아낸 너와의 데이트\\
15 보고 싶어도\\
16 매일 꼭 참고 있어도\\
17 나는 네가 안아주기만 하면\\
18 사르르르 녹아 Yeah
19 \end{document}

```

문서 만들기

노래 가사 쓰기-해결2

- T_EX에서는 문단을 만들 때마다 들여쓰기를 실행
- `\noindent`명령어를 통해 이를 제거 가능

Source Code

바보야 오늘은
 안된다고 말하지마
 오늘날만큼은 내게도
 꼭 기회를 줘
 사랑스럽게 웃는 것도
 예쁘게 말하는 것도
 많이 연습했어
 보고 싶어도 참으라고 하지 마

 Baby 난 좀 억지 부리는 것도 맞아
 오늘도 내 눈물 연기로
 받아낸 너와의 데이트
 보고 싶어도
 매일 꼭 참고 있어도
 나는 네가 안아주기만 하면
 사르르르 녹아 Yeah

```

1 \documentclass{oblivoir}
2 \begin{document}
3 \noindent 바보야 오늘은\\
4 안된다고 말하지마\\
5 오늘날만큼은 내게도\\
6 꼭 기회를 줘\\
7 사랑스럽게 웃는 것도\\
8 예쁘게 말하는 것도\\
9 많이 연습했어\\
10 보고 싶어도 참으라고 하지 마\\
11
12 \noindent Baby 난 좀 억지 부리는 것도 맞아\\
13 오늘도 내 눈물 연기로\\
14 받아낸 너와의 데이트\\
15 보고 싶어도\\
16 매일 꼭 참고 있어도\\
17 나는 네가 안아주기만 하면\\
18 사르르르 녹아 Yeah
19 \end{document}
  
```

문서 만들기

노래 가사 쓰기-해결3

- 가운데 정렬을 하고 싶다.
- `\begin{center}...\end{center}` 명령을 이용하자.

Source Code

바보야 오늘은
 안된다고 말하지마
 오늘만큼은 내게도
 꼭 기회를 줘
 사랑스럽게 웃는 것도
 예쁘게 말하는 것도
 많이 연습했어
 보고 싶어도 참으라고 하지 마
 Baby 난 좀 억지 부리는 것도 맞아
 오늘도 내 눈물 연기로
 받아낸 너와의 데이트
 보고 싶어도
 매일 꼭 참고 있어도
 나는 네가 안아주기만 하면
 사르르르 녹아 Yeah

```

1 \documentclass{oblivoir}
2 \begin{document}
3 \begin{center}
4 \noindent 바보야 오늘은\\
5 안된다고 말하지마\\
6 오늘만큼은 내게도\\
7 꼭 기회를 줘\\
8 사랑스럽게 웃는 것도\\
9 예쁘게 말하는 것도\\
10 많이 연습했어\\
11 보고 싶어도 참으라고 하지 마\\
12 \vspace{10pt}
13
14 \noindent Baby 난 좀 억지 부리는 것도 맞아\\
15 오늘도 내 눈물 연기로\\
16 받아낸 너와의 데이트\\
17 보고 싶어도\\
18 매일 꼭 참고 있어도\\
19 나는 네가 안아주기만 하면\\
20 사르르르 녹아 Yeah
21 \end{center}
22 \end{document}
  
```

문서 만들기

노래 가사 쓰기-정리

- T_EX에서는 두 개 이상의 엔터를 문단 변화로 인식.
- 강제 줄 이동은 `\\`나 `\newline`으로
- `\noindent`를 통해 문단의 들여쓰기를 제거할 수 있음.
- `\begin{center}...\end{center}`로 가운데 정렬 가능
 - ▶ 오른쪽 정렬은 `\begin{flushright}...\end{flushright}`
 - ▶ 왼쪽 정렬은 `\begin{flushleft}...\end{flushleft}`
- `\vspace{length}`를 통해 줄 간격을 만들 수 있음
 - ▶ 칸 사이의 간격은 `\,`, `\;`, `\quad`, `\qquad`등으로 만들 수 있음 (꼼수다!)
 - ▶ 띄어쓰기 한 칸은 `\` 로 해결 가능.
 - ▶ 그 외는 `` 명령어나 `\hspace{10pt}`로 띄어쓰기를 하자.

제목 만들기

- 목표: 제목은 T_EX is SO Easy, 이름은 모두의 T_EX 동아리로 제목을 만들어 보자.
- Hint: 아까 배운 것들을 활용하여 만들 수 있다. (T_EX은 \T_EX으로 조판할 수 있는데, 이런 특수 문자 뒤의 띄어쓰기는 무시된다. 강제로 칸 사이 간격을 만들어야 할 것이다.)

Source Code

T_EX is SO EASY

모두의 T_EX 동아리

```
1 \documentclass{oblivoir}
2 \begin{document}
3   \begin{center}
4     {\Huge\TeX \hspace{10pt}is SO EASY} \\
5     \vspace{10pt}
6     {\large 모두의 \TeX\phantom{ }동아리}
7   \end{center}
8 \end{document}
```

제목 만들기

- 가운데 정렬 환경을 통해 제목을 만들 수 있음.
- `\maketitle`을 통해 편하게 제목을 만들 수 있음.

Source Code

T_EX is SO EASY

모두의 T_EX 동아리

2018년 4월 2일

```
1 \documentclass{oblivoir}
2 \author{모두의 \TeX \, 동아리}
3 \title{\Huge{\TeX \, is SO EASY}}
4
5 \begin{document}
6   \maketitle
7 \end{document}
```

강조하기

- 목표: 내용을 **이렇게**, *이렇게*,
이렇게, 이렇게, 이렇게 강조해보자.
- Hint: 각각 `\textbf{...}`, `\textit{...}`, `\emph{...}`,
`\underline{...}`, `\dotemph{...}`로 표현할 수 있다.

Source Code

We can use many fonts, including **Boldface**, *Italic*, Sans Serif, Typewriter, Roman, and some else.

강조할 때에는 중요한 부분에 밑줄을 긋거나, 이탤릭체로 쓰거나, 점을 찍어 강조할 수 있습니다.

`\emph` usually works by *italicizing* what you want to emphasize, *but it works as opposite when it is in italicized fonts.*

```

1 \documentclass{oblivoir}
2 \begin{document}
3   We can use many fonts, including \textbf{Boldface},
4   \textit{Italic}, \textsf{Sans Serif},
5   \texttt{Typewriter}, \textrm{Roman},
6   and some else.
7
8   강조할 때에는 중요한 부분에
9   \underline{밑줄을 긋거나},
10  \emph{이탤릭체로 쓰거나},
11  \dotemph{점을 찍어 강조}할 수 있습니다.
12
13  \verb|\emph| usually works by \emph{italicizing}
14  what you want to emphasize,
15  \textit{but it works as opposite
16  when it is in \emph{italicized} fonts.}
17 \end{document}

```


강조하기

- `\textbf{...}`, `\textit{...}` 등은 폰트를 바꾸는 것이고, `\emph{...}`는 상황에 맞춰 단어를 강조함.
- `oblivoir`에서는 `\textit{한국어}`가 잘 적용되지 않음.
- `oblivoir`를 사용하면 `\dotemph{...}`으로 단어를 강조할 수 있음.

수식 만들기

수식 환경 사용하기

- T_EX에서는 수식 환경이라는 특별한 환경이 존재
- 목표: 수식 환경을 이용해서 근의 공식($x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$)를 써보자.
- Hint: `\begin{eqnuation}...\end{eqnuation}`을 이용해보자.
- Hint: 분수는 `\frac{분자}{분모}`로, 루트 기호는 `\frac{...}`으로 쓸 수 있다. \pm 기호는 `\pm`으로 쓴다.

Source Code

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

(1)

```
1 \documentclass{oblivoir}
2 \begin{document}
3   \begin{equation}
4     x=\frac{-b\pm\sqrt{b^2-4ac}}{2a}
5   \end{equation}
6 \end{document}
```

수식 만들기

수식 환경 사용하기

- `\begin{eqaution}...\end{eqaution}`은 우리를 수식 환경의 세계로 인도함.
- `equation` 환경은 수식을 하나밖에 적을 수 없고, 뒤에 수식 번호가 적혀 나옴.

수식 만들기

수식 참조하기

- 목표: 아까 쓴 근의 공식 아래에 '식 (공식 번호)는 근의 공식이다'라고 써 보자.

Source Code

식 [1](#)은 근의 공식이다.

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

```
(1) 1 \documentclass{oblivoir}
2 \usepackage{amsmath}
3 \begin{document}
4 \begin{equation}
5 x=\frac{-b\pm\sqrt{b^2-4ac}}{2a} \label{1}
6 \end{equation}
7 식 \ref{1}는 근의 공식이다.
8 \end{document}
```

수식 만들기

수식 참조하기

- `\label{key}`와 `\ref{key}`를 통해 수식 참조 가능
- 수식 번호를 삭제하고 싶으면 `\nonumber` 명령어 사용.
- `\(...\)`는 수식 번호가 없는 행 내 수식을 만들고, `\[...\]`는 수식 번호가 없는 표시형 수식을 만든다.

수식 만들기

여러 개의 수식 쓰기(I)

- 다음을 써 보자.



$$\begin{aligned}\alpha + \beta &= -\frac{b}{a} \\ \alpha\beta &= \frac{c}{a}\end{aligned}\tag{1}$$

- Hint: `\usepackage{mathtools}`를 넣고,
`\begin{equation} \begin{gathered}...`
`\end{gathered} \end{equation}`로 식을 삽입하자.

Source Code

$$\alpha + \beta = -\frac{b}{a}$$

$$\alpha\beta = \frac{c}{a}$$

$$\alpha + \beta = -\frac{b}{a}$$

$$\alpha\beta = \frac{c}{a}$$

```

(1) 1 \documentclass{oblivoir}
      2 \usepackage{mathtools}
      3 \begin{document}
      4 \begin{equation}
      5 \begin{gathered}
      6 \alpha + \beta = -\frac{b}{a} \\
      7 \alpha\beta = \frac{c}{a}
      8 \end{gathered}
      9 \end{equation}
(2) 10 %using "gather"
      11 \begin{gather}
      12 \alpha + \beta = -\frac{b}{a} \\
      13 \alpha\beta = \frac{c}{a}
      14 \end{gather}
(3) 15 \end{document}

```

수식 만들기

여러 개의 수식 쓰기(II)

- 다음을 써 보자.

$$\begin{aligned} 1 + 2 + 3 + \cdots + n &= \sum_{k=1}^n k \\ &= \frac{1}{2} \sum_{k=1}^n (k + (n + 1 - k)) \\ &= \frac{1}{2} \sum_{k=1}^n (n + 1) \\ &= \frac{n(n + 1)}{2} \end{aligned} \tag{2}$$

- Hint: \cdots 는 `\cdots`으로 만들고, $\sum_{k=1}^n k$ 는 `\sum_{k=1}^n k`로 만든다.

Source Code

$$\begin{aligned}
 1 + 2 + 3 + \cdots + n &= \sum_1^n k \\
 &= \frac{1}{2} \sum_1^n (k + (n + 1 - k)) \\
 &= \frac{1}{2} \sum_1^n (n + 1) \\
 &= \frac{n(n + 1)}{2}
 \end{aligned}$$

(1)

```

1 \documentclass{oblivoir}
2 \usepackage{mathtools}
3 \begin{document}
4   \begin{align}
5     1+2+3+\cdots+n &= \sum_{1}^n k \nonumber\\
6     &= \frac{1}{2} \sum_{1}^n (k+(n+1-k)) \nonumber\\
7     &\nonumber\\
8     &= \frac{1}{2} \sum_{1}^n (n+1) \nonumber\\
9     &= \frac{n(n+1)}{2}
10   \end{align}
11 \end{document}

```

수식 만들기

여러 개의 수식 쓰기(II)

- `\usepackage{mathtools}`을 첨부하고
`\begin{align}...\end{align}`을 이용하면 `...&=...\backslash...&=...`
을 통해 묶어줄 위치를 선정할 수 있음.

수식 만들기

글 안에 수식 넣기

- 다음을 써 보자.
- 물체의 변위를 \vec{x} 라고 한다면 $\frac{d^2\vec{x}}{dt^2}$ 는 가속도를 의미하게 된다.
이 때 $\vec{F}_{net} = m\frac{d^2\vec{x}}{dt^2}$ 가 성립한다.
- Hint: $\$...\$$ 로 수식 환경을 글 안에서 만들 수 있다. 벡터 표시는 $\texttt{\textbackslash vec\{a\}}$ 로 구현할 수 있다.

Source Code

물체의 변위를 x 라고 한다면 $\frac{d^2x}{dt^2}$ 는 가속도를 의미하게 된다. 이 때 $\vec{F}_{net} = m\frac{d^2x}{dt^2}$ 가 성립한다.

```
1 \documentclass{oblivoir}
2 \usepackage{mathtools}
3 \usepackage{physics}
4 \begin{document}
5   물체의 변위를  $\vec{x}$ 라고 한다면
6    $\frac{d^2\vec{x}}{dt^2}$ 는 가속도를 의미하게 된다.
7   이 때  $\vec{F}_{net}=m\frac{d^2\vec{x}}{dt^2}$ 가
8   성립한다.
9 \end{document}
```

수식 만들기

글 안에 수식 넣기

- 글 안의 수식은 $\$...\$$ 를 통해 넣을 수 있음.
- `\usepackage{physics}`를 이용해 미분을 더 편하게 사용할 수 있음.

수식 만들기

기타 수식에 관한 정보

- 만약 수식 번호를 붙이지 않는 환경을 쓰고 싶으면 `align*`, `gather*` 등 뒤에 `*`을 붙이면 됨.
- 기타 수식 기호들
 - ▶ `\sqrt[root]{arg}`: 제곱근 기호
 - ▶ `\int_{text}^{max}`: 적분 기호
 - ▶ `\delta`, `\Delta`: 그리스 문자
 - ▶ `a^{n}`, `a_{n}`: 위 첨자, 아랫 첨자
 - ▶ `\begin{cases} ... \end{cases}`: 중괄호로 경우 나누기
 - ▶ `\left(... \right)`: 수식 크기에 맞는 괄호 제작
 - ▶ 기타 기호들은 T_EXStudio의 편집창 옆의 바에서 쉽게 삽입할 수 있음

수식 만들기

examples

$$\begin{cases} f(x) = \int_{-\infty}^{\infty} g(\alpha) e^{i\alpha x} d\alpha, \\ g(\alpha) = \frac{1}{2\pi} \int_{-\infty}^{\infty} f(x) e^{-i\alpha x} dx \end{cases}$$

$$i\hbar \frac{\partial \Psi}{\partial t} = \left[-\frac{\hbar^2}{2m} \nabla^2 + V \right] \Psi$$

마치며

- 지금 배운 게 T_EX의 빙산의 일각이라는 사실.
 - ▶ 그림 첨부, 벡터 그래픽, 매크로, 등등 아직 해야 할 것들이 많다...
- 그러나 이만큼으로도 내가 원하는 문서의 90%는 만들 수 있다.
 - ▶ 나머지 10%는 인터넷에 검색해서 배우도록 하자.
- 142분 동안 익히는 L^AT_EX 2_ε에 설명이 잘 되어 있다. 추가로 찾아볼 것.

남은 10%에 관한 설명

- 남은 10%는 심하면 문서에서 사용하지 않을 가능성도 크다.
- 그럼에도 이런 것이 있다- 정도를 보여주기 위해 다음을 소개할 예정.
 - ▶ tabular, minipage, etc(문서 분할하기)
 - ▶ 이미지 추가
 - ▶ 벡터 그래픽 제작
 - ▶ 매크로 지정

tabular 환경

- 다음의 표를 만들어보자.

	T _E X	MS Word
Difficulty	Hard	Easy
Type	WYSIWYM	WYSIWYG
Easy Document	Easy	
Hard Document	Easy	Hard

Source Code

	T _E X	MS Word
Difficulty	Hard	Easy
Type	WYSIWYM	WYSIWYG
Easy Document	Easy	
Hard Document	Easy	Hard

```

1 \documentclass{oblivoir}
2 \begin{document}
3   \begin{tabular}{c|c|c}
4     \multicolumn{1}{c}{ } & \TeX & MS Word \\
5     \hline \hline
6     Difficulty & Hard & Easy \\
7     \hline
8     Type & WYSIWYM & WYSIWYG \\
9     \hline
10    Easy Document & \multicolumn{2}{c}{Easy} \\
11    \hline
12    Hard Document & Easy & Hard
13  \end{tabular}
14 \end{document}

```

tabular 환경

- tabular 환경은 표를 제작해줌.
- 수식의 경우 array 환경을 사용(matrix 환경도 존재.)

minipage 환경

- 문서를 절반으로 쪼개서 각각 글을 채워 넣자.
- 다양한 명령어가 존재 (box류 명령어, minipage, `\begin{multicol}...\end{multicol}(\usepackage{multicol})`) 등이 존재

Source Code

Nulla non mauris vitae wisi posuere convallis. Sed eu nulla nec eros acelerisque pharetra. Nullam varius. Etiam dignissim elementum metus. Vestibulum faucibus, metus sit amet mattis rhoncus, sapien dui laoreet odio, nec ultricies nisl augue a enim. Fusce in ligula. Quisque at magna et nulla conmodo consequat. Proin accumsan imperdiet sem. Nunc porta. Donec frugiat mi at justo. Phasellus facilisis ipsum quis ante. In ac elit eget ipsum pharetra faucibus. Maureris viverra nulla in massa.

Nulla ac nisl. Nullam urna nulla, ullamcorper in, interdum sit amet, gravida ut, risus. Aenean ac enim. In luctus. Phasellus eu quam vitae turpis viverra pellentesque. Duis frugiat felis ut enim. Phasellus pharetra, sem id portitor sodales, magna nunc aliquet nisl, nec blandit nisi mauris at pede. Suspendisse risus risus, lobortis eget, semper at, imperdiet sit amet, quam. Quisque acelerisque dapibus nisl. Nam enim. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nunc ut metus. Ut metus justo, auctor at, ultrices eu, sagittis ut, purus. Aliquam aliquam.

Etiam pede massa, dapibus vitae, rhoncus in, placerat posuere, odio. Vestibulum luctus conmodo luctus. Morbi luctus dui, tempor sed, euismod eget, condimentum at, tortor. Phasellus aliquet odio ac luctus tempor faucibus. Praesent sed sem. Praesent iaculis. Cras rhoncus tellus sed justo ullamcorper sagittis. Donec quis arcu. Sed ut tortor quis tellus euismod tincidunt. Suspendisse congue nisl eu elit. Aliquam tortor diam, tempus id, tristique eget, sodales vel, nulla. Praesent tellus mi, condimentum sed, viverra at, consectetur quis, lectus. In auctor vehicula orci. Sed pede sapien, euismod in, aenean in, pharetra placerat, metus. Vivamus conmodo dui non odio. Donec et felis.

```

1 \documentclass{oblivoir}
2 \usepackage{lipsum}
3 \usepackage{multicol}
4 \begin{document}
5 \lipsum[19]
6 \vspace{10pt}
7 \noindent
8 \framebox{
9   \begin{minipage}{0.45\linewidth}
10    \vspace{10pt}
11    \begin{center}
12      \begin{minipage}{0.965 \linewidth} \lipsum[20] \end{minipage}
13    \end{center}
14    \vspace{0pt}
15  \end{minipage}
16  \hspace{0.02\linewidth}\vrule\hspace{0.02\linewidth}
17  \framebox{
18    \begin{minipage}{0.45\linewidth}{0.45\linewidth}
19    \vspace{10pt}
20    \begin{center}
21      \begin{minipage}{0.965 \linewidth} \lipsum[21] \end{minipage}
22    \end{center}
23    \vspace{0pt}
24    \end{minipage}
25  \end{document}

```


minipage 환경

- `\framebox[width]{text}` 이외에도 `\colorbox`, `\shadowbox`, `\doublebox`(`fancybox` 패키지 필요) 등을 사용할 수 있음.
- `minipage` 환경은 `\parbox{width}{text}`와 비슷한 효과(문서 안의 박스를 설정)
- 자세한 이론은 우선 생략.

이미지 추가

- 이미지를 하나 추가해보자.

Source Code



```
1 \documentclass{oblivoir}
2 \usepackage{graphics}
3 \begin{document}
4   \noindent
5     \begin{figure}
6       \centering
7       \includegraphics{donaldknuth.jpg}
8       \caption{Donald Knuth}
9     \end{figure}
10 \end{document}
```

그림 1: Donald Knuth

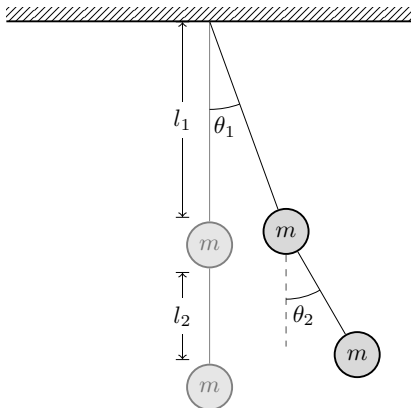
이미지 추가

- graphics나 graphicx 패키지를 이용해 그래픽을 문서 안에 추가할 수 있음.
- figure 환경 안에 그림을 첨부하고 caption을 달 수 있다.
- 문서랑 어울리게 하는 것이 어려움. (minipage를 열심히 이용하거나, kswrapfig 패키지를 이용하여 할 수 있음)

벡터 그래픽 제작

- 물리 문제에 사용할 진자 그림을 그려보자.
- tikz 패키지를 이용해 벡터 그래픽을 직접 만들 수 있음.

Source Code



```

1 \documentclass{standalone} \usepackage{tikz} \usetikzlibrary{patterns,calc}
2
3 \def\centerarc[#1](#2)(#3:#4:#5){#6}{#7}{
4   \draw[#1] (#2)+({#5*cos(#3)}:{#5*sin(#3)}) arc (#3:#4:#5) node[#6]{#7}; }
5 \begin{document}
6   \begin{tikzpicture}
7     \node[circle,draw=black!50,thick,fill=black!10,
8           minimum width=6mm,minimum height=6mm,text opacity=0.5] (A) at (0,1.0) {m$};
9     \node[circle,draw=black!50,thick,fill=black!10,
10          minimum width=6mm,minimum height=6mm,text opacity=0.5] (A') at (0,-1.1) {m$};
11     \fill [pattern = north east lines] (-3,4.3) rectangle (3,4.5);
12     \draw[thick] (-3,4.3) -- (3,4.3);
13     \draw[draw=black!50] (0,4.3) -- (A); \draw[draw=black!50] (A) -- (A');
14     \draw[|<->|] (-0.4,4.3) -- (-0.4,1.4) node[pos=0.5,fill=white]{l$_1$};
15     \draw[|<->|] (-0.4,0.6) -- (-0.4,-0.7)
16     node[pos=0.5,fill=white]{l$_2$};
17     \begin{scope}[rotate around=(20:(0,4.3))]
18       \node[circle,draw=black,thick,fill=black!15,minimum width=6mm,minimum height=6mm,text opacity=1]
19       (B) at (0,1.0) {m$};
20       \draw[draw=black] (0,4.3) -- (B);
21       \begin{scope}[rotate around=(10:(B))]
22         \node[circle,draw=black,thick,fill=black!15,minimum width=6mm,minimum height=6mm,text opacity=1]
23         (B') at (0,-1.1) {m$};
24         \draw[draw=black] (B) -- (B');
25       \end{scope}
26     \end{scope}
27     \draw[dashed,draw=black!70] (B) -- +(0,-1.7);
28     \centerarc[|] (0,4.3) (-90-90+20:1.3) [below,midway] ({\theta_1}$)
29     \centerarc[|] (B) (-90-90+30:1) [below,midway] ({\theta_2}$)
30 \end{tikzpicture}
\end{document}

```

벡터 그래픽 제작

- Tikz 패키지 말고 PSTricks도 많이 유명함.(둘 다 배우는 것은 비추)
- 그림 그릴 때는 standalone 환경에서 작업하는 것도 좋다.
- 사실 좀 어려움...
 - ▶ lpe, TikzEdit 등 벡터 그래픽 편집 프로그램을 사용하면 조금 쉬워진다.
 - ▶ 지오지브라에도 벡터 그래픽 제작 매크로가 존재한다.
 - ▶ 가끔은 그림판과 타협하는 것이 더 좋을지도 모른다.

매크로 지정

- T_EX에는 매크로 기능이 존재(아마 프로그래밍으로서의 T_EX을 가장 잘 드러내는 부분)
- 매크로를 사용하여 항등행렬을 간단히 출력해보자.

Source Code

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

```

1 \documentclass{article}
2 \usepackage{amsmath,amssymb,mathtools}
3 \usepackage{ifthen}
4 \newcommand{\forLoop}[5][1]{
5   \setcounter{#4}{#2}
6   \ifthenelse{ \value{#4} < #3 }
7     {#5\addtocounter{#4}{#1}\forLoop{#1}{\value{#4}}{#3}{#4}{#5}}
8     {\ifthenelse{\value{#4} = #3}{#5}{}}
9   }
10  \newcounter{identRow}
11  \newcounter{identCol}
12  \newcommand{\idmatrixn}[1]
13  {
14    \left[\begin{array}{*{#1}c}
15      \forLoop{1}{#1}{identRow}
16      {
17        \forLoop{1}{#1}{identCol}{
18          \ifthenelse{\equal{\value{identRow}}{\value{identCol}}}{1}{0}
19          \ifthenelse{\equal{\value{identCol}}{#1}}{&}{&}
20        }\\
21      }
22    \end{array}\right]
23  }
24  \begin{document}
25    \begin{equation}
26      \idmatrixn{5}
27    \end{equation}
28  \end{document}

```

매크로 지정

- 새로운 명령어를 지정하는 것은
`\newcommand{cmd}{def}`, `\def`,
`\newenvironment{text}{begdef}{enddef}` 등이 존재
- 다만 C처럼 함수를 만든다고 생각하면 조금 곤란함(가능은 하다만...)
- 알아두면 글 작업이 편리해짐.(특히 이미지 첨부, 벡터 그래픽 제작에서)