

L^AT_EX 표를 tabulararray로 그려보았다

Nova de Hi

2025-12-11 version 0.4

제 1 장 시작하기 전에	3
1.1 이 문서는	3
1.2 표와 행간격	3
1.3 셀 안의 \\	4
제 2 장 L^AT_EX4WP의 예제	8
2.1 첫 번째 표	8
2.2 booktabs 표	10
2.3 heading row의 정렬과 색상	12
2.4 엄청난 표	14
2.5 기울어진 표	16
2.6 rowcolor와 columncolor	18
2.7 row의 높이 제어	20
2.8 수치 정렬	21
2.9 diagbox: 셀에 대각선	22
제 3 장 <faq:Tabular 환경>의 표 그려보기	24
3.1 multicolumn과 multirow	24
3.2 raisebox	26
3.3 표 안의 list 환경	27
3.4 표 안에 대각선 그리기	28
3.5 p{...}와 가운데 정렬	29
3.6 booktabs	30
3.7 colortbl	31
3.8 hhline	32
3.9 arydshln	34
3.10 패션의 굵기	35
3.11 Progress 예제 (1)	37
3.12 Progress 예제 (2)	38

제 4 장	실전 응용	41
4.1	문제	41
4.2	시도	42
4.2.1	표의 본체를 먼저 입력한다	42
4.2.2	multicolumn과 multirow를 먼저 처리한다	43
4.2.3	colspec을 정돈한다	44
4.2.4	표의 수직 길이를 조정한다	45
4.2.5	그림을 넣는다	46
4.2.6	패선과 폰트	47
제 5 장	여러 페이지 표	50
5.1	여러 페이지에 걸치는 표	50
5.2	Progress, 케이터 연혁	50

시작하기 전에

1.1 이 문서는

KTUG 게시판의 다음 두 글로 발표한 두 개의 문서를 하나로 합친 것이다.

- KTUG:latex4wp의 표 그리기
- KTUG:tabular 환경의 표를 tabulararray로 그려보자

첫 번째 글은 [LATEX4WP](#) 문서의 “제6절 Tables”의 예제를 tabulararray로 그리는 방법을 다루고 있고 두 번째 글은 현재는 열람만 가능한 예전 KTUG Faq 위키 문서 [《Tabular환경》](#)에 올라 있는 표 그리기 예제를 tabulararray의 <tblr>로 그려보는 내용이다. 다음 인용은 둘째 글의 시작 부분에 적어두었던 것이다.

현재는 운영되지 않고 열람만이 가능한 KTUG Faq 위키에 [《Tabular환경》](#)이라는 페이지가 있다.
그 당시까지 표 그리기에 관련된 온갖 팁들을 모아 놓았는데, 어떤 문제에 어떤 해결책이 모색되었던
가를 감상해볼 수 있다.

이제 tabulararray의 <tblr>로 표를 그리게 되었으니, 이 페이지가 제공하는 샘플을 <tblr>로
그려보면 재미있지 않겠는가 하는 것이 이 문서를 작성한 동기이다.

해당 페이지에 나와 있는 코드를 먼저 실행해보고 같은(또는 유사한) 표를 <tblr>로 그려본다.

아래 이어지는 두 개의 토막글은 첫째 글의 첫머리에 있던 것인데 별도의 섹션으로 제시한다. 이 내용 일부는 작성자의 개인적 판단과 경험을 포함한다.

0.3 버전에서 하나의 장(41 페이지)을 추가하였다. 이것은 KTUG 게시판에 올라온 질문에 답변한
경험을 재구성한 것이다.

1.2 표와 행간격

한글 문서는 표준 L^AT_EX 문서에 비하여 행간격이 넓어진다. 그 결과 <tabular>로 그린 표에서도 행 사이의 간격이 넓어진다. 다음은 *oblivoir* 문서에서 아무런 조치를 취하지 않았을 때 표가 그려지는 모양을 표준 L^AT_EX 문서에서의 모양과 비교해 본 것이다.

1	2	3
a		c
one	two	three

1	2	3
a		c
one	two	three

한편, *oblivoir*에서 “좁은 행간격”을 1로 설정하였거나¹ *setspace*를 로드한 경우²에는 `<table>` 또는 `<figure>` 환경 안에서 표의 행간격이 1로 맞춰진다. 다음 보기지를 보라.

1	2	3
a		c
one	two	three

그림 1.1: 플로트 환경에서 행간격 예시

그러므로 모든 도표가 *float* 환경 안에 있으면 표 내부 행간격이 문제되는 경우가 없고, 이것이 일반적으로 권장되는 것이다.

그런데 문서에 따라 도표를 *float* 안에 넣지 않는 때가 있다. 이럴 적에 `<tabular>`의 행간격을 보기 좋게 만드는 데 꽤 심력을 소모해야 하는 경우가 있었다. *tabulararray*는 이 고민을 상당 부분 해소해주는데, 행(row) 사이의 간격을 제어하는 다양한 매크로를 제공하기 때문이다. 그 가운데 하나로서 *stretch*라는 옵션 key가 있다. 대체로 1.33 *linespread* 문서에서 이 값을 0.5 내지 0.6 정도로 주면, *float* 밖의 *tabular*에 대해서도 적당히 보기 좋은 표 행간격을 얻을 수 있다.

다음과 같이 하면 문서 안(또는 선언이 주어진 이후) 모든 `<tblr>`에 대하여 *stretch*를 일괄 설정할 수 있는데, 위에 설명했듯이 *float* 안에 들어가면 행간이 변해서 곤란할 수 있으므로 주의해야 한다. 이와 같은 global 설정은 “문서 안의 모든 표가 *float* 밖에 있다는 확신이 있을 때”만 하도록 하고, 그렇지 않다면 필요에 따라 *stretch* 값을 각각 부여하는 것이 좋다.

```
\SetTblrInner[tblr]{stretch=.6}
```

이 글에서는 표를 *float* 안에 넣지 않을 것이기 때문에 모든 `<tblr>`들은 *stretch*=.6 상태로 작성하였다.³ 강조해두지만 문서 전체의 행간격 설정과 표의 행간격을 잘 고려하여 이 값을 설정해야 할 것이다.

1.3 셀 안의 \\

*LATEX*에서 도표(`<tabular>`)에 대해 뭔가 항상 개운치 않은 느낌을 받았던 이유는 이것이 (*LATEX* 철학과는 달리) 모양(style)을 나타내는 코드와 내용(contents)을 나타내는 코드를 뒤엉킨 채로 입력해야 하기 때문이었다. 예를 들어

¹“좁은 행간격”이란 `\SetHangulSpace` 명령의 두 번째 인자로 주는 값을 말한다. 디폴트는 1보다 조금 큰 값이다.

²*memoir*/*oblivoir*에서는 *setspace*를 별도로 로드하지 않는다.

³대체로 이보다 더 줄이는 것은 좋지 않다. 여러분의 간격을 조절하는 파라미터가 *tabulararray*에 잘 정의되어 있으므로 이보다 좁은 간격을 원한다면 이런 파라미터를 잘 설정하는 것이 좋다.

```
a & b & c \\ \hline
```

에서 `\hline`은 표의 내용의 일부가 아니고 양식적 장식을 위한 코드인데도 매 라인마다 이것을 달고 다녀야 하는 것이다.

`tabulararray`를 발견한 후 가장 기뻤던 점 중 하나가, `\hline` 명령을 표의 내용에서 완전히 제외하여도 표와 패션을 다 잘 그릴 수 있다는 점이었다.

그리고 다음과 같은 문제를 생각해 보자.

첫 번째 문제. 셀 안에 오는 텍스트에 대해서 문단 정렬을 할 수 있는가?

만약 셀 정렬자를 l, c, r만을 쓴다면 셀 안의 텍스트는 “정렬”되지 않는다는 사실을 알 것이다.

```
\begin{tabular}{l|c|r}
\hline
100년 후에도 똑같은 모양으로 인쇄될 문서를 작성하고 싶으세요? &
잘되는군요 &
그러면 \LaTeX{} 을 써보세요. 100년 후에도 똑같은 모양으로 인쇄된답니다. \\ \hline
\end{tabular}
```

100년 후에도 똑같은 모양으로 인쇄될 문서를 작성하고 싶으세요?	잘되는군요	그러면 <code>\LaTeX</code> 을 써보세요. 100년
--------------------------------------	-------	--------------------------------------

셀 안의 텍스트가 정렬되게 하려면, p 지시자를 쓰고 폭을 지정해주어야 한다.

```
\begin{tabular}{p{5cm}|c|p{6cm}}
\hline
100년 후에도 똑같은 모양으로 인쇄될 문서를 작성하고 싶으세요? &
잘되는군요 &
그러면 \LaTeX{} 을 써보세요. 100년 후에도 똑같은 모양으로 인쇄된답니다. \\ \hline
\end{tabular}
```

100년 후에도 똑같은 모양으로 인쇄될 문서를 작성하고 싶으세요?	잘되는군요	그러면 <code>\LaTeX</code> 을 써보세요. 100년 후에도 똑같은 모양으로 인쇄된답니다.
--------------------------------------	-------	---

얼핏 보면 당연한 일인데, 때로 이 컬럼 너비를 정해주는 것이 문제가 되기도 한다. 컬럼 너비를 절대치로 지정해두면 나중에 판면이 바뀌거나 했을 때 통제할 수 없는 일이 생길 수 있다. 표 전체 길이에 대하여 상대적으로 그 값을 자동 설정하게 할 수 있으면 좋을텐데. 그래서 `tabularx`와 X 정렬자가 나오게 된 것이다.

```
\begin{tabularx}{\textwidth}{X|c|X}
\hline
100년 후에도 똑같은 모양으로 인쇄될 문서를 작성하고 싶으세요? &
잘되는군요 &
그러면 \LaTeX{} 을 써보세요. 100년 후에도 똑같은 모양으로 인쇄된답니다. \\ \hline
\end{tabularx}
```

100년 후에도 똑같은 모양으로 인쇄될 문서를 작성하고 싶으세요?	잘되는군요	그리면 LATEX을 써보세요. 100년 후에도 똑같은 모양으로 인쇄된답니다.
--------------------------------------	-------	--

이러고 나면 이제 생기는 의문은 “정렬된 문단”을 내용으로 가진 셀들끼리 vertical 정렬을 할 수 없을까 하는 것이다. 이에 대해서 여러 해법이 나와 있는데, 우리는 다음 <tblr> 예시에서 이를 보이는 것에 그치려 한다. tabulararray는 X 정렬자를 잘 지원한다. 물론 width 옵션으로 <tblr>의 전체 폭을 지정해 주어야 한다.

```
\begin{tblr}{  
    width = \textwidth,  
    colspec = {X[1,b]Q[c,m]X[1.2,t]},  
    hlines,  
    vline{2-Y} = {solid}  
}  
100년 후에도 똑같은 모양으로 인쇄될 문서를 작성하고 싶으세요? &  
잘되는군요 &  
그리면 \LaTeX 을 써보세요. 100년 후에도 똑같은 모양으로 인쇄된답니다. \\  
\end{tblr}
```

100년 후에도 똑같은 모양으로 인쇄될 문서를 작성하고 싶으세요?	잘되는군요	그리면 LATEX을 써보세요. 100년 후에도 똑같은 모양으로 인쇄된답니다.
--------------------------------------	-------	--

두 번째 문제. 셀 안의 텍스트에 \\가 들어 있다면 어떻게 되나?

위의 예문에 나온 (뭔가 좀 과장광고같은) 텍스트를 다음과 같이 분질러서 조판하고 싶다는 것이다.

```
그리면 \LaTeX 을 써보세요.\\  
100년 후에도 똑같은 모양으로\\  
인쇄된답니다.
```

그냥은 안 되는 것이 당연한 게, \\는 행(row)의 종료를 나타내는 부호이므로, 두 가지 서로 다른 의미를 같은 부호로 표시하려는 것이기 때문이다. 이에 대해서도 여러 해법이 있고 나름 사연이 있지만, 아무튼 결국 makecell 패키지의 \makecell{}가 고민을 해결해주었다. 다음 예문을 보자. 텍스트를 강제로 절단하는 케이스이므로 문단의 “정렬”이 필요없고, 따라서 l, c, r만으로 표를 작성할 수 있다.

```
\begin{tabular}{l|c|r}  
\hline  
\makecell{100년 후에도\\ 똑같은 모양으로\\ 인쇄될 문서를\\ 작성하고 싶으세요?} &  
잘되는군요 &  
\makecell[l]{그리면 \LaTeX 을 써보세요.\\ 100년 후에도\\ 똑같은 모양으로\\ 인쇄된  
답니다.} \\ \hline  
\end{tabular}
```

100년 후에도 똑같은 모양으로 인쇄될 문서를 작성하고 싶으세요?	잘되는군요	그러면 L ^A T _E X을 써보세요. 100년 후에도 똑같은 모양으로 인쇄된답니다.
--------------------------------------	-------	---

<tblr>은 여기서 한 걸음 더 나아가서, 아예 하나의 셀을 중괄호로 둘러싸기만 하면 그 안에서 \\를 써도 되게 되어 있다. 이것은 상당히 감동적(?)인 것이었다. 한글로 문서를 적을 때 makecell의 아주 미묘한 어긋남 때문에 그것을 보정하는 데 들인 노력을 생각하면 세상이 좋아진 것이 틀림없다.

```
\begin{tblr}{  
    width=.7\textwidth,  
    colspec={Q[c,m]|X[c,m]|Q[l,m]}, hlines  
}  
{100년 후에도\\ 똑같은 모양으로\\ 인쇄될 문서를\\ 작성하고 싶으세요?} &  
잘되는군요 &  
{그러면 \LaTeX 을 써보세요.\\ 100년 후에도\\ 똑같은 모양으로\\ 인쇄된답니다.} \\  
\end{tblr}
```

100년 후에도 똑같은 모양으로 인쇄될 문서를 작성하고 싶으세요?	잘되는군요	그러면 L ^A T _E X을 써보세요. 100년 후에도 똑같은 모양으로 인쇄된답니다.
--------------------------------------	-------	---

LATEX4WP의 예제

2.1 첫 번째 표

LATEX4WP-KO p.38의 첫 번째 예제는 다음과 같다.

```
\begin{tabular}{|l|c r|}
\hline
1 & 2 & 3 \\ \hline
a & & c \\
one & two & three \\ \hline
\end{tabular}
```

1	2	3
a		c
one	two	three

이 코드를 그대로 <tblr> 안에 넣어도 표는 그려질 것이다.

```
\begin{tblr}{|l|c r|}
\hline
1 & 2 & 3 \\ \hline
a & & c \\
one & two & three \\ \hline
\end{tblr}
```

1	2	3
a		c
one	two	three

그런데, 우리가 <tblr>을 쓰는 중요한 이유 중의 하나가 표의 컨텐츠 부분의 코딩에서 \hline을 쓰지 않으려는 것이다. \hline와 같은 추가적인 매크로들은 나중에 표를 수정하려 하거나 소스를 읽으려 할 적에 상당히 방해가 된다. <tblr>이 표의 스타일(모양)을 (대부분) 옵션으로 설정할 수 있고 표의 컨텐츠는 깨끗한 상태로 유지할 수 있어서, 훗날 스타일을 바꿀 필요가 생겨도 귀찮은 수작업을 크게 요구하지 않는다는 것이 한 장점이다.

```
\begin{tblr}[
  colspec={|l|c r|}, rowspec={|Q|QQ|}
]
1 & 2 & 3 \\
```

```
a &   & c \\
one & two & three \\
\end{tblr}
```

1	2	3
a		c
one	two	three

또는, hline과 vline을 별도로 제어하는 방법도 있다. Z는 마지막 줄이라는 의미이다.

```
\begin{tblr}{}
  colspec={lcr},
  hline{1,2,Z}=\{\lightrulewidth\},
  vline{1,2,Z}=\{\lightrulewidth\},
}
1 & 2 & 3 \\
a &   & c \\
one & two & three \\
\end{tblr}
```

1	2	3
a		c
one	two	three

위의 예제에서 사용된 \lightrulewidth와 \heavyrulewidth는 booktabs 명령인데,¹ 각각 .05em과 .08em으로 정의되어 있다. 한편 memoir의 \normalrulethickness는 .4pt이다.

이 문서의 preamble에 다음과 같이 정의하여, \heavyrulethickness라는 매크로를 만들어서 쓰겠다. 물론 그냥 .4pt와 같이 길이값을 적어넣어도 무방하다.

```
\let\heavyrulethickness\heavyrulewidth % 0.08em
```

참고로, 모든 패션을 다 그리려면, 간단히 hlines, vlines로 충분하다.

```
\begin{tblr}{}
  colspec={lcr},hlines,vlines
}
1 & 2 & 3 \\
a &   & c \\
one & two & three \\
\end{tblr}
```

¹memoir는 booktabs를 기본으로 불러들인다.

1	2	3
a		c
one	two	three

연습문제 1. 가로 패선만 그리고 세로 패선은 하나도 안 그리려면 어떻게 하면 되겠는가?

연습문제 2. 가로 패선을 첫 row의 상하에 대해서와 마지막에만 그리고 세로 패선은 그리지 않으려면 어떻게 하면 되겠는가?

연습문제 3. 첫 가로 패선과 마지막 가로 패선의 굵기를 `heavyrulethickness`로 하고 두 번째 가로 패선은 정상 굵기로 연습문제 연습문제 2.의 표를 그리려면 어떻게 하면 되겠는가?

연습문제 풀이: 1. 잘 하면 된다. 2. 간단하다. 3. 쉽다.

2.2 booktabs 표

booktabs 표는 heading row와 표의 마지막 row 다음에 가로줄을 긋고 세로줄은 전혀 없는 표를 말한다. 첫 가로줄을 `\toprule`, 마지막 가로줄을 `\bottomrule`, 그리고 heading row와 표의 body를 구분하는 선을 `\midrule`이라고 한다. 가로줄과 표의 cell 사이에 약간의 간격을 준다. 이 양식은 많은 타이포그래피 교과서에서 권장하는 것으로 서양에서는 사실상 표 그리기의 표준이다. LATEX4WP-KO p.38의 예제는 다음과 같다.

```
\begin{tabular}{lcr}
\toprule
1 & 2 & 3 \\
\midrule
a& &c\\
one & two & three \\
\bottomrule
\end{tabular}
```

1	2	3
a		c
one	two	three

단순하고 무식하게(이전 절의 연습문제에서 했던 것처럼) 가로줄의 두께만을 조절해서 비슷한 모양을 만들어보자.

```
\begin{tblr}[
  colspec={lcr},
  hline{1,Z}=\{\heavyrulewidth\}, hline{2}=\{\lightrulewidth\},
]
1 & 2 & 3 \\
a & &c\\
one & two & three \\
\end{tblr}
```

1	2	3
a		c
one	two	three

뭐, 그럭저럭 봐줄 만은 하지만 이것은 booktabs 표가 아니다. 특히 toprule과 row 사이에 추가로 들어가야 할 간격이 배제되어 있다. tabulararray는 \UseTblrLibrary 명령으로 “library”를 로드할 수 있게 하고 있고, 마침 booktabs라는 라이브러리가 있다.

```
\begin{booktabs}[
    colspec={lcr}, stretch=1
]
\toprule
1 & 2 & 3 \\
\midrule
a & & c \\
one & two & three \\
\bottomrule
\end{booktabs}
```

1	2	3
a		c
one	two	three

<tblr> 환경으로 작성할 때에도 \toprule 등을 쓸 수 있다. 다음 보기와 같이 row와 rule의 spec을 사용자가 완전히 자신의 뜻대로 지정하는 것도 가능하다.

```
\begin{tblr}[
    colspec={lcr}, rowspec={|Q|QQ|}, rowsep=1pt,
    hline{1,Z}=.8pt, hline{2}=.4pt,
    row{1}={abovesep=3pt,belowsep=3pt},
    row{2}={abovesep=3pt}, row{Z}={belowsep=3pt}
]
\toprule
1 & 2 & 3 \\
a & & c \\
one & two & three \\
\end{tblr}
```

1	2	3
a		c
one	two	three

2.3 heading row의 정렬과 색상

표에서 첫 번째 행(row)은 표 본체(body)와는 구별되도록 작성하는 때가 있다. <booktabs>도 그런 종류 중의 하나일 것이다.

먼저, 첫 행의 정렬만을 모두 가운데(c)로 해보자. 1번 row의 속성을 설정하는 방법으로 간단히 된다.

```
\begin{tblr}{  
    colspec={lcr}, hlines,  
    row{1}={c}  
}  
1 & 2 & 3 \\  
a & & c \\  
left & center & right \\  
one & two & three \\  
\end{tblr}
```

1	2	3
a		c
left	center	right
one	two	three

이번에는 첫 행의 배경색과 전경색을 바꾸고 폰트도 다른 걸로 해보자. 가로줄은 모두 없애겠다.

```
\begin{tblr}{  
    colspec={lcr},  
    row{1}={c, fg=white, bg=azure2, font=\sffamily}  
}  
one & two & three \\  
1 & 2 & 3 \\  
a & & c \\  
left & center & right \\  
\end{tblr}
```

one	two	three
1	2	3
a		c
left	center	right

첫 행의 상단에는 조금 여분이 필요하겠다. 마지막 행의 하단에 여분을 조금 주고 굵은 글자를.

```
\begin{tblr}{  
    colspec={lcr},  
    row{1}={c, abovesep=3pt,  
            fg=white, bg=azure2,  
            font=\sffamily},  
    row{2}={belowsep=2pt},
```

```

    hline{Z}=\{\heavyrulethickness\}
}
one & two & three \\
1 & 2 & 3 \\
a & & c \\
left & center & right \\
\end{tblr}

```

one	two	three
1	2	3
a		c
left	center	right

이런 식으로 하면 표가 길어질 때 잘 알아보기 어려워지는 경우가 있다. 이를 위해서 한 행 건너 하나씩 살짝 배경색을 칠하기도 한다.

```

\begin{tblr}{}
  colspec={lcr},
  row{odd}={bg=azure9},
  row{1}={c, abovesep=3pt,
          fg=white, bg=azure2,
          font=\sffamily},
  row{Z}={belowsep=2pt},
  hline{Z}=\{\heavyrulethickness\}
}
A & B & C \\
one & two & three \\
1 & 2 & 3 \\
a & & c \\
left & center & right \\
this & last & row \\
\end{tblr}

```

A	B	C
one	two	three
1	2	3
a		c
left	center	right
this	last	row

연습문제 1. 이 절의 마지막에 완성한 표의 전체 너비(width)를 text block의 50%로 하고 각 열(column)의 폭을 균등하게 배분해 보아라.

연습문제 2. 표의 내용이 두 페이지를 넘어가는 데이터로 이루어진 경우에 <booktabs> 표를 쓸 방법이 있는가? tabulararray 매뉴얼에서 해당 사항을 찾아보고 실제 샘플을 연습삼아 작성해 보라.

2.4 엄청난 표

LATEX4WP-KO에 이 예제가 이미 실려 있으므로 여기서는 인용하고 약간의 해설을 붙인다.

```
\begin{tblr}{  
    colspec={Q[1]Q[c]Q[r]Q[4cm,1]},  
    hline{1,Z}=.8pt,hline{2}=\heavyrulethickness,  
    vline{3}=\normalrulethickness,  
    vline{3}={4}{dashed},  
}  
\SetRow{font=\bfseries,fg=olive5} Left & Center & Right & 4cm \\  
row 1,col 1 & row 1,col 2 & row 1,col 3 & row 1,col 4\\ \cline{1-2}  
row 2,col 1 & row 2,col 2 & row 2,col 3 & row 2,col 4\\  
\cline[dashed]{1-2}\rline[dashed]{4}  
\SetCell[c=2]{bg=gray!30,c} spanning two columns & row 3,col 3 & row 3,col  
→ 4\\  
\cline[dashed]{1-2}\cline{3}  
row 4,col 1 & row 4,col 2 & row 4,col 3 & \SetCell{r} right\\  
row 5,col 1 & row 5,col 2 & row 5,col 3 & left \\  
row 5,col 1 & row 5,col 2 & row 5,col 3 & \SetCell{c} centre\\  
\end{tblr}
```

Left	Center	Right	4cm
row 1,col 1	row 1,col 2	row 1,col 3	row 1,col 4
row 2,col 1	row 2,col 2	row 2,col 3	row 2,col 4
spanning two columns		row 3,col 4	
row 4,col 1	row 4,col 2	row 4,col 3	right
row 5,col 1	row 5,col 2	row 5,col 3	left
row 5,col 1	row 5,col 2	row 5,col 3	centre

colspec은 다음과 같이 쓸 수도 있다.

```
colspec={lcrQ[1,4cm]},
```

hline, vline 설정을 colspec에 같이 하기로 하면, 다음처럼 할 수 있다.

```
colspec={lc|rQ[1,4cm]},  
rowspec={| [.8pt]Q|QQQQQQ| [.8pt]},
```

\cline은 그대로 입력해도 구현되지만 만약 테이블 컨텐츠를 클린하게 유지하는 것이 목표라서 별도의 스타일링 매크로를 가급적 억제하고 싶다면, 다음처럼 \cline 명령을 모두 hline 설정으로 바꿀 수 있다.

```
\begin{tblr}{  
    colspec={Q[1]Q[c]Q[r]Q[4cm,1]},
```

```

    hline{1,Z}={.8pt},hline{2}={solid},
    hline{4}={1-2}{dashed}, hline{5}={1-2}{dashed},
    hline{5}={3}{solid},
    vline{3}={solid},
    vline{1,3}={4}{dashed},
}
\SetRow{font=\bfseries,fg=olive5} Left & Center & Right & 4cm \\
row 1,col 1 & row 1,col 2 & row 1,col 3 & row 1,col 4\\ \cline{1-2}
row 2,col 1 & row 2,col 2 & row 2,col 3 & row 2,col 4\\
\SetCell[c=2]{bg=gray!30,c} spanning two columns & row 3,col 3 & row 3,col
\rightarrow 4\\
row 4,col 1 & row 4,col 2 & row 4,col 3 & \SetCell{r} right\\
row 5,col 1 & row 5,col 2 & row 5,col 3 & left \\
row 5,col 1 & row 5,col 2 & row 5,col 3 & \SetCell{c} centre\\
\end{tblr}

```

Left	Center	Right	4cm
row 1,col 1	row 1,col 2	row 1,col 3	row 1,col 4
row 2,col 1	row 2,col 2	row 2,col 3	row 2,col 4
spanning two columns		row 3,col 4	
row 4,col 1	row 4,col 2	row 4,col 3	right
row 5,col 1	row 5,col 2	row 5,col 3	left
row 5,col 1	row 5,col 2	row 5,col 3	centre

이제, 나아가 \SetRow를 body에서 제거할 방법을 생각하자. 다만, \SetCell은 그대로 써도 크게 문제가 되지 않는 경우가 많다. 왜냐하면 이런 임의적 매크로가 아주 자주 나오지 않을 것이기 때문이다. \SetRow는 row 속성 지정으로 대부분 대체할 수 있다.

```
row{1} = {font=\bfseries,fg=olive5},
```

이것을 추가하고 \SetRow 명령 부분을 삭제해도 된다.

```

\begin{tblr}{}
  colspec={Q[1]Q[c]Q[r]Q[4cm,1]},
  hline{1,2,Z}={.8pt},
  hline{4}={1-2}{dashed}, hline{5}={1-2}{dashed},
  hline{3}={1-2}{solid},
  hline{5}={3}{solid},
  vline{3}={solid},
  vline{1,3}={4}{dashed},
  row{1}={font=\bfseries,fg=olive5},
}
Left & Center & Right & 4cm \\
row 1,col 1 & row 1,col 2 & row 1,col 3 & row 1,col 4\\
row 2,col 1 & row 2,col 2 & row 2,col 3 & row 2,col 4\\

```

```
\SetCell[c=2]{bg=gray!30,c} spanning two columns & row 3,col 3 & row 3,col
↪ 4\\
row 4,col 1 & row 4,col 2 & row 4,col 3 & \SetCell{r} right\\
row 5,col 1 & row 5,col 2 & row 5,col 3 & \SetCell{l} left \\
row 5,col 1 & row 5,col 2 & row 5,col 3 & \SetCell{c} centre\\
\end{tblr}
```

Left	Center	Right	4cm
row 1,col 1	row 1,col 2	row 1,col 3	row 1,col 4
row 2,col 1	row 2,col 2	row 2,col 3	row 2,col 4
spanning two columns		row 3,col 4	
row 4,col 1	row 4,col 2	row 4,col 3	right
row 5,col 1	row 5,col 2	row 5,col 3	left
row 5,col 1	row 5,col 2	row 5,col 3	centre

그런데, `\multicolumn`이나 `\multirow` 대신 쓰는 `\SetCell`이나, 특정 셀 하나만 정렬을 바꾸려 하는 경우에는 `\SetCell` 명령을 해당 셀에 두는 것이 좋다. 위의 예에서 그렇게 했다.

연습문제 1. 이 절의 표를 다음과 같은 모양으로 만들어라. 2,3번 컬럼의 너비는 1번 컬럼의 너비의 120%이고, 4번 컬럼의 너비는 4cm이다. 표 전체의 너비는 `\textwidth`의 80%이다.

Left	Center	Right	4cm
row 1,col 1	row 1,col 2	spanning three rows	row 1,col 4
row 2,col 1	row 2,col 2		row 2,col 4
spanning two columns			
row 4,col 1	row 4,col 2	row 4,col 3	right
row 5,col 1	row 5,col 2	row 5,col 3	left
	(row 5,col 2)		
row 5,col 1	has long, long long long text	row 5,col 3	centre

2.5 기울어진 표

LATEX4WP-KO p.40의 예제를 회전시키지 않고 `<tblr>`로 그리면 다음과 같다.

```
\begin{tblr}{}
  width=7.5cm,
  colspec={Q[1,m]XX},
  vlines,
  hlines,
}
normal & titled & wider \\
normal & I'm titled! & I'm wider \\
\end{tblr}
```

normal	titled	wider
normal	I'm titled!	I'm wider

LATEX4WP에서는 한 개의 셀을 기울이는 데는 \rotatebox, 표 전체를 회전시키는 데는 rotating의 <sideways> 환경을 쓰고 있다. 실은 adjustbox로 이 모든 게 다 해결 가능하다.²

먼저 셀 하나만 회전시킨다.

```
\begin{tblr}{  
    width=7.5cm,  
    colspec={Q[l,m]XX},  
    vlines, hlines,  
}  
    \SetRow{font=\bfseries} normal  
        & titled & wider \\  
    normal  
        & \adjustbox{rotate=30}{I'm titled!}  
        & I'm wider \\  
    \end{tblr}
```

normal	titled	wider
normal	I'm titled!	I'm wider

셀 하나가 기울어지면서 두 번째 행의 위치 관계에 변화가 생겼다. 두 번째 행의 각 셀들을 되도록 나란히 배열하려면 어떻게 할 것인가?

```
\begin{tblr}{  
    width=7.5cm,  
    colspec={Q[l,m]XX},  
    vlines, hlines,  
}  
    \SetRow{font=\bfseries} normal  
        & titled & wider \\  
    normal  
        & \adjustbox  
            {rotate=30, valign=m}{I'm titled!}  
        & I'm wider \\  
    \end{tblr}
```

²\rotatebox는 graphicx가 제공하는 것이지만 adjustbox는 이를 조금 개선하고 있다.

normal	titled	wider
normal	I'm titled!	I'm wider

그리고 표 전체를 회전시킨다.

```
\adjustbox{rotate=90}%
\begin{tblr}{width=7.5cm,
colspec={Q[l,m]XX},
vlines, hlines,
}
\SetRow{font=\bfseries} normal & titled &
\wider \\
normal & \adjustbox{rotate=30,valign=m}{I'm titled!} & I'm wider \\
\end{tblr}
```

normal	titled	wider
normal	I'm titled!	I'm wider

연습문제 1. 이 절의 마지막 예제에서 첫 줄(row)의 모든 셀을 가운데 정렬하려 한다. 폰트의 속성은 그대로 유지한다. \SetRow 명령을 표의 body에 남기지 말고 모두 옵션에서 설정하라.

연습문제 2. 회전된 “I'm titles!” 셀에 텍스트 대신 이미지가 들어간다면 어떻게 되는지 실제로 테스트하여 보아라. adjustbox에 [export] 옵션을 주어 \includegraphics 명령의 옵션으로 이 그림의 크기, 정렬, 회전 등을 제어해 보아라.

2.6 rowcolor와 columncolor

row와 column의 색상 설정은 이미 보았으므로, 다음 예를 보이는 것으로 충분하겠다.

```
\begin{tblr}{colspec={lcr},
rowspec={Q[cyan8]Q[green]Q[yellow]},
}
one & two & three \\
one & two & three \\
one & two & three
\end{tblr}
```

one	two	three
one	two	three
one	two	three

```
\begin{tblr}{%
    colspec={Q[l,cyan8]Q[c,green]Q[r,yellow]},
}
one & two & three \\
one & two & three \\
one & two & three
\end{tblr}
```

one	two	three
one	two	three
one	two	three

```
\begin{tblr}{%
    colspec={lQ[c,green]r},
}
one & two & three \\
\SetRow{green} one & two & three \\
one & two & three
\end{tblr}
```

one	two	three
one	two	three
one	two	three

tabulararray로 간편해진 것이, 표 안의 패션에도 색을 입힐 수 있다는 것이다. 실제 `<tabular>`에서는 패션에 색을 입히는 것은 아주 어렵거나 거의 불가능하다. 다음에서 보듯이 그냥 색상 이름만 지정하는 것으로 이것이 그냥 이루어진다.

```
\begin{tblr}{%
    colspec={l|[2pt,teal6]cr},
    rowspec={| [.8pt,cyan6]Q| [.4pt,green]Q%
              | [.4pt,magenta4]Q| [.8pt,red5]},
}
one & two & three \\
one & two & three \\
one & two & three
\end{tblr}
```

one	two	three
one	two	three
one	two	three

혹은, 다음과 같이 코딩할 수 있다.

```
\begin{tblr}{}
  colspec={lcr},
  hline{1}=\{\heavyrulethickness,cyan6\},
  hline{2}=\{\normalrulethickness,green\},
  hline{3}=\{\normalrulethickness,magenta4\},
  hline{4}=\{\heavyrulethickness,red5\},
  vline{2}=2pt,teal6\},
}
one & two & three \\
one & two & three \\
one & two & three
\end{tblr}
```

one	two	three
one	two	three
one	two	three

<tblr>을 처음 다루던 시기에는 |[red] 이런 식으로 colspec과 rowspec에서 설정하는 것이 편해 보였는데, 시간이 갈수록 hline과 vline 속성으로 지정하는 것이 더 판독하기 쉽고 제어하기도 쉽다는 느낌이 듈다.

연습문제 1. colortbl의 예제로 다음과 같은 것이 있다.

최대한 비슷한 도표를 어떻게 만들 수 있겠는지 생각해보자.

2.7 row의 높이 제어

LATEX4WP는 row 사이에 간격을 주려 할 때,

- \\에 길이 옵션을 추가하는 방법,
- 수직 높이를 갖는 strut (\rule)를 넣는 방법

이 두 가지를 소개하고 있다. <tblr>에서는 그런 뭔가 꼼수스러운 방법을 쓰지 않아도 옵션으로 제공되는 key를 이용해서 편하게 해결할 수 있다.

```
\begin{tblr}{  
    colspec={l1l},  
    row{2}={belowsep=.3cm},  
    row{5}={abovesep=1.2cm},  
}  
one & two & three \\  
0.3 centimeters & \SetCell{font=\bfseries}after  
    & this line \\  
one & two & three \\  
one & two & three \\  
1.2 centimeters & \textbf{before} & this line  
\end{tblr}
```

one	two	three
0.3 centimeters	after	this line
one	two	three
one	two	three

1.2 centimeters **before** this line

연습문제 1. 이 절에서 만든 도표를 <taking>으로 만든다면 어떻게 가능하겠는가?

연습문제 2. 표에 보기 좋을 정도의 패선을 그려서 완성하여라.

2.8 수치 정렬

LATEX4WP에서는 소수점 정렬을 위해 정수부와 소수부를 다른 컬럼으로 배치하는, 유구한 역사를 가진 트릭(LSHORT에도 소개되고 있음)과 더불어 dcolumn을 이용하는 방법을 소개한다. 그것도 크게 나쁘지 않은데 문제는 heading row를 텍스트 모드로 입력하기 위해서 \mbox나 \text 명령을 꼭 써야 한다는 게 불편하다.

tabulararray는 이런 상황을 위해서 siunitx 패키지를 로드하는 방법을 사용한다.

```
\UseTblrLibrary{siunitx}
```

이 명령을 통하여 siunitx 방식의 key=value 방식의 수치 표현 설정을 할 수 있게 되어 있다.

```
\begin{tblr}{  
    colspec={  
        Q[si={table-format=4.2,output-decimal-marker={,}},1]
```

```

    Q[si={table-format=4.5,table-number-alignment=center},c]
    Q[si={table-format=4.5,table-number-alignment=left},r]
},
row{1} = {guard},
}
One & Two & Three \\
10.33 & 10.33 & 10.33 \\
1000 & 1000 & 1000 \\
5.1 & 5.1 & 5.1 \\
3.14 & 3.14159 & 3.14159 \\
\end{tblr}

```

One	Two	Three
10,33	10.33	10.33
1000	1000	1000
5,1	5.1	5.1
3,14	3.14159	3.14159

Q 셀에서는 si= 지정자를 이용할 수 있고, (라이브러리를 로드하면 쓸 수 있는) S 셀 지시자로 siunitx 옵션을 바로 지정할 수도 있다.

이 글에서는 siunitx 옵션에 대하여 상세히 설명하지 않겠다. 옵션의 세목에 대해서는 siunitx 패키지 문서를 보라. 우리는 preamble에

```
\sisetup{group-separator={}}
```

를 선언하여 숫자 그루핑에 추가되는 간격을 없앴다.

특기할 것으로 row 속성으로 지정할 수 있는 guard라는 것이다. 이것이 지정된 행은 텍스트 모드이다.

2.9 diagbox: 셀에 대각선

```
\UseTblrLibrary{diagbox}
```

설명이 필요하지 않겠다.

```

\begin{tblr}{}
  colspec={lll},
  rowspec={| [.8pt]Q|QQQ| [.8pt]},
  vlines,
}
\diagbox{Lesson}{Date} & Monday & Tuesday \\
Stratigraphy & room A & room A \\
Chemistry & room B & Lab $\alpha$ \\

```

```
Physics & room C & Lab $\delta$  

\end{tblr}
```

Lesson \diagdown Date	Monday	Tuesday
Stratigraphy	room A	room A
Chemistry	room B	Lab α
Physics	room C	Lab δ

slashbox에 얹힌 (슬픈?) 역사를 우리는 기억한다. 대각선 각도가 미묘하게 안 맞아서 고생하던 것, 그리고 `slashbox.sty` 자체가 없어져서 어디서 찾아야 하는지 헤매던 일, 뭐 그런 것이다. 전자는 `diagbox` 가 `pict2e`를 통해 `LATEX`에서 그을 수 있는 선의 기울기 제한을 없앰으로써 이제 옛말이 되었다. `slashbox` 자체는 CTAN에 남아 있으나 `texlive`에도 `miktex`에도 포함되지 않는다. 그러나 `diagbox`가 예전 명령도 잘 지원하고 있고(그래도 새로운 명령 `\diagbox`를 쓸 것을 권장한다), 따라서 `slashbox.sty`는 없어도 그만이다.

셀에 대각선을 그은 표는 아무리 해도 그다지 세련된 모양이 되지 않는다. 대각선따위 안 그리는 것이 정답이다.

연습문제 1. 표에 관하여, 이 문서에서 다루지 않은 주제로 (`LATEX4WP`와는 상관없이) 추가하면 좋을 것 같은 주제를 신고하자.

연습문제 2. 이 문서를 읽고 느낀 점을 KTUG에 보고하자.

〈faq:Tabular 환경〉의 표 그려보기

3.1 multicolumn과 multirow

<tabular>에서 \multicolumn은 당연히 쓸 수 있다. multirow에 대해서는 multirow 패키지를 이용하였다.

```
\usepackage{multirow}
```

```
\renewcommand{\multirowsetup}{\centering} % 기본값은 \raggedright
\begin{tabular}{c|c|c|c|c|c} \hline
    \multicolumn{2}{c|}{\multirow{2}{15mm}{선택지들}} & & & & \\
    & \multirow{2}{10mm}{경우 1} & & \multirow{2}{10mm}{경우 2} & & \\
    & & \multicolumn{2}{c}{고르시오} & & \\ \cline{5-6}
    \multicolumn{2}{c|}{\multirow{2}{15mm}{}} & & \hline & 높음 & 낮음 \\ \cline{5-6}
    \multirow{3}{S}{R} & 2 & 3 & 4 & 5 & \\ & 4 & 1 & 6 & 9 & 7 \\ & R & 2 & 3 & 4 & 5 \\ \hline
\end{tabular}
```

선택지들	경우 1	경우 2	고르시오	
			높음	낮음
S	R	2	3	4
	4	1	6	9
	R	2	3	5

tabulararray에서는 \SetCell의 옵션 인자로 [r=n] 또는 [c=n]과 같은 방식으로 지정하면 되니까 아주 간단하다. multi-columns 또는 rows로 span하고 나서 비는 셀 위치의 & 개수를 빼먹지 않도록 주의 한다.

```
\begin{tblr}{%
    colspec={*[6]{Q[c,m]}},
    }
```

```

    hlines, vlines,
    vline{1,Z}={0pt}
}
\SetCell[c=2,r=2]{} 선택지들 & & % c=2이므로 하나 비움
\SetCell[r=2]{} 경우 1 & \SetCell[r=2]{} 경우 2 &
\SetCell[c=2]{} 고르시오 & \\ % c=2이므로 하나 비움
& & & 높음 & 낮음 \\ \hline % r=2인 셀이 위에 있으므로
% 비는 셀 개수 확인
\SetCell[r=3]{} S & R & 2 & 3 & 4 & 5 \\
& 4 & 1 & 6 & 9 & 7 \\
& R & 2 & 3 & 4 & 5 \\
\end{tblr}

```

선택지들	경우 1	경우 2	고르시오	
			높음	낮음
S	R	2	3	4
	4	1	6	9
	R	2	3	5

단순히 hlines만 선언하여도, multi-rows 또는 columns로 span된 셀은 <tblr>이 알아서 패션을 잘 그어준다. 위의 코드에서 \cline이 한번도 쓰이지 않았음에 주목할 것. <tblr>에서는 r=n, c=n으로 span한 셀의 가려져야 하는 패션을 잘 처리해준다. 그래서 이런 경우의 \cline을 쓸 필요가 없다.

한편, \SetCell을 표의 body에 써넣지 않고 다음처럼 하는 방법도 있다. 이왕 이런 예를 보이는 김에, \hline을 표의 body에 적어넣었던 것도 옵션 설정으로 이동하였다. 이런 방식의 장점은 표의 내용 자체를 한눈에 파악하기 쉽고 나중에 모양을 쉽게 바꿀 수 있다는 점이다.

```

\begin{tblr}{}
  colspec={*{6}{Q[c,m]}},
  hlines,
  hline{3}={1}{-}{.4pt},
  hline{3}={2}{-}{.4pt},
  vlines,
  vline{1,Z}={0pt},
  cell{1}{1}={r=2,c=2}{},
  cell{1}{2-4}={r=2}{},
  cell{1}{Y}={c=2}{},
  cell{3}{1}={r=3}{}
}
선택지들 & & 경우 1 & 경우 2 & 고르시오 & \\
& & & 높음 & 낮음 \\
S & R & 2 & 3 & 4 & 5 \\
& 4 & 1 & 6 & 9 & 7 \\
& R & 2 & 3 & 4 & 5 \\
\end{tblr}

```

선택지들	경우 1	경우 2	고르시오	
			높음	낮음
S	R	2	3	4 5
	4	1	6	9 7
	R	2	3	4 5

3.2 raisebox

```
\begin{tabular}{|c|c|}\hline
1&\cline{1-1}
10&\cline{1-1}
100&\cline{1-1}
1000&\cline{1-1}
10000&\raisebox{2.8ex}[0pt]{\includegraphics[width=1.5cm]{example-image}}\hline
\end{tabular}
```

1	
10	
100	
1000	
10000	

이 예제는 \raisebox 명령을 재치있게 사용하여 원하는 모양을 구현한 것이다. 그러나 끌어올리는 길이를 시행착오를 통해서 얻어야 하기 때문에 좀 불편하다.

이제 우리는 다음과 같이 간단히 이 모양을 그릴 수 있다. 이 코드에서는 adjustbox의 valign까지 사용했는데 이렇게 하면 더 정확한 “가운데 위치”를 지정할 수 있다. valign을 사용하지 않아도 “대략” m 위치에는 올 것이다.

```
\begin{tblr}[
  colspec={Q[c,m]Q[c,m]},
  hlines,vlines
]
1 & \SetCell[r=5]{} \includegraphics[width=1.5cm,valign=M]{example-image} \\
10 & \\
100 & \\
1000 & \\
10000 & \\
\end{tblr}
```

1	
10	
100	
1000	
10000	

3.3 표 안의 list 환경

표의 한 셀 내부에 리스트(<enumerate>, <itemize>)가 들어가는 경우에 대한 (원시적인) 해법은 하나의 셀에 (폭(width)이 주어지는) \parbox로 넣는다는 것이었다. 다음은 그런 아이디어의 코드이다.

```
\begin{tabular}{l|c|c|c} \hline
No. # & T1 & T2 & Remarks \\ \hline
\#~46 & 0.14 & 0.63 & \\ \hline
\#~47a & 0.14 &
\parbox{20em}{%
\begin{itemize}
\item SSSSSSS
\item FFFFFFF
\end{itemize}
} & \\ \hline
\#~48 & 0.15 & 0.67 & \\ \hline
\end{tabular}
```

No. #	T1	T2	Remarks
# 46	0.14	0.63	
# 47a	0.14	<ul style="list-style-type: none"> • SSSSSSS • FFFFFFF 	
# 48	0.15	0.67	

이 방식의 의의는, “어쨌든 <itemize>가 들어갔다”는 것밖에 없다. 단점을 지적하자면 너무 많은데, 가장 큰 것은 보기에 거슬리는 행간격, leftmargin 등이다. 물론 이것은 enumitem을 이용하여 줄일 수는 있다. 일일이 이런 세세한 것을 손보는 것은 제법 지루한 일이다.

조금 나은 방식의 발전을 여러 단계 거치지만, tabulararray의 방법을 바로 보기로 하자. 다행히 설명서에 이 문제에 관한 상세한 설명과 예제가 있다.

```
\begin{tblr}{%
hlines,vlines,rowspec={l l l l},
measure=vbox,stretch=-1,
row{1}={c},row{3}={m}}
```

```

}

No. \# & T1 & T2 & Remarks \\
\#~46 & 0.14 & 0.63 & \\
\#~47a & 0.14 & {\begin{itemize}[nosep, leftmargin=1em]
    \item SSSSSS
    \item FFFFFFF
\end{itemize}} & \\
\#~48 & 0.15 & 0.67 & \\
\end{tblr}

```

No. #	T1	T2	Remarks
# 46	0.14	0.63	
# 47a	0.14	<ul style="list-style-type: none"> • SSSSSS • FFFFFFF 	
# 48	0.15	0.67	

앞에 보인 `\parbox`를 쓴 예에서 이 셀의 width를 강제로 지정하지 않으면 안 되던 것과 달리, 이 해법으로는 natural width를 쓸 수 있다.

몇 가지 주의할 점은, `measure=vbox`를 쓰기 위해서 다음 행을 preamble에 두어야 하는 점이다.

`\UseTblrLibrary{varwidth}`

`measure=vbox`라는 옵션이 필요한 이유는 이 셀의 width를 (`varwidth`의 도움을 받아) `\hbox`가 아니라 `\vbox`로 계산해내기 위해서이다. 리스트 내용물이 들어간 셀에서 특히 필요하다. 그리고 `nosep`, `leftmargin`과 같이 `<itemize>`의 모양을 조절하는 옵션을 위해 `enumitem`을 `\usepackage`해주어야 한다.¹

3.4 표 안에 대각선 그리기

LATEX4WP의 TABLE에서 이미 다루었다(22페이지). 핵심은 `diagbox`. Faq:DiagBoxPackage에 나와 있는 샘플이다.

`\usepackage{diagbox}`

```

\begin{tabular}{|c|ccc|}
\hline
\diagbox{\footnotesize 시간대}{\footnotesize 요일} & 월 & 화 & 수 \\
\hline
아침 & 일정있음 & 일정있음 & 일정없음 \\ \hline
오후 & 일정있음 & 일정있음 \\

```

¹memoir/oblivoir에서는 클래스 내부의 복잡한 설정과 부딪치지 않도록 `enumitem`을 되도록 preamble의 뒷쪽에 배치하는 것이 좋다.

```
\hline  
\end{tabular}
```

시간대	요일	월	화	수
아침		일정있음	일정있음	일정없음
오후		일정있음	일정있음	

tabulararray의 diagbox 라이브러리를 로드하고 <tblr>로 그려보자.

```
\UseTblrLibrary{diagbox}
```

```
\begin{tblr}{  
  colspec={cccc},  
  hlines,  
  vline{1,2,Z}={solid},  
}  
\diagbox{\footnotesize 시간대}{\footnotesize 요일} & 월 & 화 & 수 \\  
아침 & 일정있음 & 일정있음 & 일정없음 \\  
오후 & 일정있음 & 일정있음 & 일정있음 \\  
\end{tblr}
```

시간대	요일	월	화	수
아침		일정있음	일정있음	일정없음
오후		일정있음	일정있음	

3.5 p{...}와 가운데 정렬

이 텁은 p 정렬자로 폭(width)을 지정하였을 경우 그 열(column)의 정렬을 제어하는 방법에 대한 것이다.

```
\usepackage{array}
```

```
\begin{tabular}{|>{\centering}p{40mm}|p{15em}|}  
\hline  
40mm & 15em \\ \hline  
가운데 정렬 & p로 되어 있으면 어디로 붙을까? \\ \hline  
항상 가운데라네 & \multicolumn{1}{|c|}{임시 가운데라네} \\ \hline  
\end{tabular}
```

40mm	15em
가운데 정렬	p로 되어 있으면 어디로 붙을까?
항상 가운데라네	임시 가운데라네

지금은 이렇게 복잡한 것이 필요하지 않다. 실제로 p 정렬자를 쓸 일이 거의 없다.

```
\begin{tblr}{|c|c|}
  \hlines
  \SetCell[c,40mm]{c} 40mm & \SetCell[l,15em]{c} 15em \\
  가운데 정렬 & p로 되어 있으면 어디로 붙을까? \\
  항상 가운데라네 & \SetCell[c]{c} 임시 가운데라네 \\
\end{tblr}
```

40mm	15em
가운데 정렬	p로 되어 있으면 어디로 붙을까?
항상 가운데라네	임시 가운데라네

3.6 booktabs

LATEX4WP의 TABLE에서 충분히 다루었으나(10페이지), Faq 페이지에 샘플 코드가 있으므로 시도해본다.

```
\usepackage{booktabs} %preamble
\usepackage{multirow} %preamble
```

```
\renewcommand{\multirowsetup}{\centering}
\begin{tabular}{cccc}
\toprule
Case & Box A & Box B & Remarks \\
\midrule
\multirow{2}{10mm}{A} & $+X$ & $-1.0$ & \multirow{2}{10mm}{B} \\
& $-X$ & $-7.9$ & \\
\bottomrule
\end{tabular}
```

Case	Box A	Box B	Remarks
A	$+X$	-1.0	B

특별할 것은 없어 보인다.²

```
\UseTblrLibrary{booktabs}
```

```
\begin{booktabs}[
    colspec={cccc},
    stretch=1
]
\toprule
Case & Box A & Box B & Remarks \\
\midrule
\SetCell[r=2]{c,m} A & $+X$ & $-1.0$ & \SetCell[r=2]{c,m} B \\
\cmidrule[lr]{2-3}
& $-X$ & $-7.9$ & \\
\bottomrule
\end{booktabs}
```

Case	Box A	Box B	Remarks
A	+X	-1.0	B
	-X	-7.9	

3.7 colortbl

tabulararray가 등장하기 전에 tabular에 색을 입히는 것은 colortbl이 대표적이었다. xcolor 패키지도 [table] 옵션을 주면 비슷한 일을 했는데 colortbl과 명령이 중복되었고 현재는 tabulararray가 있어서 잘 안 쓰게 되었다. 이 도표는 그림으로 결과를 보이겠다.

```
\usepackage{colortbl}      % preamble
```

```
\begin{tabular}{cccc} \hline
Case & Box A & Box B & Remarks \\ \hline
A & +X & -1.0 & \\
\rowcolor[gray]{0.9} B & -X & -7.9 & \\
\end{tabular}
```

Case	Box A	Box B	Remarks
A	+X	-1.0	
B	-X	-7.9	

표의 색상 처리는 tabulararray의 장점이므로 더 설명할 것이 없다.

²예시 코드에서 stretch=1을 선언한 것은 이 문서에서 미리 선언해둔 다른 값이 있기 때문이다. 일반적으로 이 옵션은 주지 않아도 좋다.

```
\begin{tblr}{  
    colspec={cccc},  
    hlines,  
    row{Z}={gray!50}  
}  
Case & Box A & Box B & Remarks \\  
A & +X & -1.0 & \\  
B & -X & -7.9 & \\  
\end{tblr}
```

Case	Box A	Box B	Remarks
A	+X	-1.0	
B	-X	-7.9	

3.8 hline

```
\usepackage{hhline}
```

```
\begin{tabular}{||cc||c|c||} % ||는 겹세로줄  
 \hline{|t==:t==:t|}  
 a & b & c & d \\  
 \hline{|:==:|~|~||}  
 1 & 2 & 3 & 4 \\  
 \hline{#==#~|=#}  
 i & j & k & l \\  
 \hline{||--||--||}  
 w & x & y & z \\  
 \hline{|b==:b==:b|}  
\end{tabular}
```

a	b	c	d
1	2	3	4
i	j	k	l
w	x	y	z

hhline은 겹迨선이 많은 표를 그릴 때 유용하다. 특히 겹가로줄과 겹세로줄이 겹치는 때 선이 연속되게 할 것인지 여부를 비교적 간단하게 제어할 수 있다.

<tblr>에서 이런 종류의 선을 정교하게 그리려면 약간의 아이디어가 필요하다.

일단 겹가로줄을 그리는 것부터 해보자. 다음과 같은 두 가지 방법이 있는데, 위에 예시된 표와 같은 것을 얻어내려는 상황이면 rowspec으로 지정하지 않는 것이 낫다.

```

\begin{tblr}{}
  colspec={cccc},
  hlines={1}{-}{solid}, hlines={2}{-}{solid}
}
a & b & c & d \\
1 & 2 & 3 & 4 \\
i & j & k & l \\
w & x & y & z \\
\end{tblr}
\quad
\begin{tblr}{}
  colspec={cccc},
  rowspec={||Q||Q||Q||Q||}
}
a & b & c & d \\
1 & 2 & 3 & 4 \\
i & j & k & l \\
w & x & y & z \\
\end{tblr}

```

a	b	c	d
1	2	3	4
i	j	k	l
w	x	y	z

a	b	c	d
1	2	3	4
i	j	k	l
w	x	y	z

그런데 문제로 주어진 표에서 4번째 가로줄은 외줄이고 2, 3번 줄은 빈 곳이 있다.

```

\begin{tblr}{}
  colspec={cccc},
  hline{1,Z}={1}{-}{solid}, hline{1,Z}={2}{-}{solid},
  hline{2}={1}{1-2}{solid}, hline{2}={2}{1-2}{solid},
  hline{3}={1}{1-2,4}{solid}, hline{3}={2}{1-2,4}{solid},
  hline{4}={solid},
}
a & b & c & d \\
1 & 2 & 3 & 4 \\
i & j & k & l \\
w & x & y & z \\
\end{tblr}

```

a	b	c	d
1	2	3	4
i	j	k	l
w	x	y	z

세로줄을 어떻게 세팅할 것인지 생각해본다.

```

\begin{tblr}{|c||c|c||}
  \hline{1,Z}={1}{-}{solid}, \hline{1,Z}={2}{-}{solid}, \\
  \hline{2}={1}{1-2}{solid}, \hline{2}={2}{1-2}{solid}, \\
  \hline{3}={1}{1-2,4}{solid}, \hline{3}={2}{1-2,4}{solid}, \\
  \hline{4}={solid, leftpos=0, rightpos=0}, \\
  \hline{1,2}={2}{1}{leftpos=0}, \hline{2}={1}{1}{leftpos=0}, \\
  \hline{1}={2}{Z}{rightpos=0}, \hline{5}={1}{1}{leftpos=0}, \\
  \hline{5}={1}{Z}{rightpos=0}, \hline{5}={2}{1,Z}{leftpos=1, rightpos=1}, \\
  \hline{1}={2}{2}{rightpos=0}, \hline{1}={2}{3}{leftpos=0}, \\
  \hline{Z}={1}{2}{rightpos=0}, \hline{Z}={1}{3}{leftpos=0}, \\
  \hline{1}={1}{1-Z}{abovepos=1, belowpos=1}, \\
  \hline{1,3}={2}{2-Y}{belowpos=1}, \hline{3,Z}={1}{2-Y}{belowpos=1}, \\
  \hline{4}={1}{1-Y}{solid, belowpos=1}, \\
  \hline{Z}={1}{1-Y}{belowpos=1}, \hline{Z}={2}{1-Z}{abovepos=1, belowpos=1}, \\
\end{tblr}
a & b & c & d \\
1 & 2 & 3 & 4 \\
i & j & k & l \\
w & x & y & z \\
\end{tblr}

```

a	b	c	d
1	2	3	4
i	j	k	l
w	x	y	z

얼핏 보면 뭔가 복잡하고 정신없어보여도 막상 코딩을 해보면 꽤 논리적이라 그다지 어렵지 않다. 그렇지만 `hhline`보다 간편하다고 하기는 좀 망설여진다. `tabulararray`와 `hhline`을 함께 쓰기는 조금 곤란하니까, 이 방법에 익숙해지면 좋을 것이다. 이만큼의 패션을 복잡하게 가지는 도표가, 이렇게 샘플을 만들 때를 제외하고 있을까?

3.9 arydshln

패션을 점선으로 만들어주는 패키지가 `arydshln`(이 이상한 패키지 이름은 아마도, array dashed line에서 온 듯하다)이다.

두 개의 예가 있는데 그 중에 더 복잡해보이는 두 번째 샘플은 다음과 같다. 안타깝지만 `hhline`과 `arydshln`은 정의가 충돌하는 부분이 있어 하나의 문서에 둘 다 쓰는 데 약간 문제가 있다. 에러는 없지만 둘 중 하나의 모양이 원하는 대로 나타나지 않을 수 있다. 다음 예제의 실행 결과는 그림으로 보이겠다.

```
\usepackage{array,arydshln}
```

```
\renewcommand\arraystretch{1.3333}% extra space evenly
                                % distributed
\setlength\dashlinedash{1pt}
\setlength\dashlinegap{1pt}

\begin{tabular}{;{5pt/2pt}c::c:c;{5pt/2pt}}
\hline
A & B & C \\ \hline
X & Y & Z \\ \hline[5pt/2pt]
100 & 10 & 1 \\ \hline
\end{tabular}
```

A	B	C
X	Y	Z
100	10	1

tabulararray는 패션의 모양, 굵기, 색상을 아주 잘 제어하기 때문에 아마 이 패키지는 더이상 필요하지 않아 보인다. 다음 보기에서는 위의 arydshln 표의 dashed line과 비슷한 모양을 만들기 위해 mydashed라는 새로운 dash style을 정의하였다. 보통은 그냥 dashed로 충분할 것이라 생각한다.

```
\NewDashStyle {mydashed} {on 3pt off 1.5pt}
\begin{tblr}{}
  colspec={ccc},
  hlines={dotted},
  hline{3}={mydashed},
  vline{1,Z}={mydashed},
  vline{2}={1}{-}{dotted}, vline{2}={2}{-}{dotted},
  vline{3}={dotted},
  rulesep=1pt
}
A & B & C \\
X & Y & Z \\
100 & 10 & 1 \\
\end{tblr}
```

A	B	C
X	Y	Z
100	10	1

원래 샘플에서 \dashlinegap을 1pt로 하고 있기 때문에 여기서도 rulesep을 조절해주었다.

3.10 패션의 굵기

booktabs가 미리 정해진 조금 두꺼운 가로선과 조금 얇은 가로선을 제공하지만, 만약 사용자가 가로 세로 패션의 굵기를 마음대로 바꾸고 싶을 때 어떻게 하는가? 지금은 그냥 간단히 굵기를 주면 그만이지만 한때는 이게 쉬운 일이 아니었다.

워크어라운드 해결책을 대략 정리하면 다음과 같다. 세로선은 mdwtab을 이용한다. 그리고 가로선의 굵기는 새로운 명령을 정의하여 해결하는 것이다.

```
\usepackage{array,mdwtab}
```

```
\makeatletter
\def\hlinewd#1{%
  \noalign{\ifnum0=\`}\fi\hrule \@height #1 \futurelet \reserved@a\@xhline}
\makeatother
\begin{tabular}{|c|c!{\vrule width 3pt}c||c|}
  \hlinewd{3pt}
  Model & $E_k / E_p$ & Non-dim. & Ratio. \\
  A1 & 3.62 & 20.12 & 0.912 \\
  A5 & 7.10 & 22.75 & 1.139 \\
\end{tabular}
```

Model	E_k/E_p	Non-dim.	Ratio.
A1	3.62	20.12	0.912
A5	7.10	22.75	1.139

얼마나 쉬워졌는지 보자.

```
\begin{tblr}{}
  colspec={|c|c|[3pt]c||c|},
  rowspec={| [3pt]Q|[3pt]Q|Q|[3pt]}
}
Model & $E_k / E_p$ & Non-dim. & Ratio. \\
A1 & 3.62 & 20.12 & 0.912 \\
A5 & 7.10 & 22.75 & 1.139 \\
\end{tblr}
```

Model	E_k/E_p	Non-dim.	Ratio.
A1	3.62	20.12	0.912
A5	7.10	22.75	1.139

물론 rowspec이나 colspec 안의 패션 명령(|)에 옵션을 쓰지 않고, 앞서 보인 대로 hline, vline 설정을 해도 상관없다. 그러나 이런 단순한 예에서는 이것이 더 간단할 것이다.

3.11 Progress 예제 (1)

```
\begin{tabular}{|b{40mm}|m{40mm}|p{40mm}|}
\hline
\centering\textrcolor{blue}{가나다라} & 답답한 마음 & 질문 \\
\hline
b 옵션을 줬으니 맨 위로 붙어라!!
가나다라 마바사아 자차카타파하 가나다라 마바사아 자차카타파하 &
\centering\textrcolor{blue}{너는 m옵션을 줬으니 가운데에 있으렴!! } \\ \\답답한
마음 답답한 마음 답답한 마음 답답한 마음 } & 너는 p옵션
偈으니 바닥에 붙어라!! 질문 질문 질문 질문 질문 질문 질문
→ 질문 질문 \\
질문 질문 \\
\hline
가나다라 & 답답한 마음 & \multicolumn{1}{c|}{\textrcolor{blue}{질문}} \\
\hline
가나다라 & 답답한 마음 & \multicolumn{1}{r|}{\textrcolor{blue}{질문}} \\
\hline
가나다라 & 답답한 마음 & 질문 \\
\hline
\end{tabular}
```

가나다라	답답한 마음	질문
b 옵션을 줬으니 맨 위로 붙어라!! 가나다라 마바사아 자차카타파하 가나다라 마바사아 자차카타파하	너는 m옵션을 줬으니 가운데에 있으렴!! 답답한 마음 답답한 마음 답답한 마음 답답한 마음 답답한 마음	너는 p옵션 줬으니 바닥에 붙어라!! 질문
가나다라	답답한 마음	질문
가나다라	답답한 마음	질문
가나다라	답답한 마음	질문

```
\begin{tblr}{width=133mm,
colspec={XXX},
hlines,vlines,
vline{Z}={4}{white}}
}
\SetCell{fg=blue,c} 가나다라 & 답답한 마음 & 질문 \\
\SetCell{b} {b 옵션을 줬으니 맨 위로 붙어라!!}
가나다라 마바사아 자차카타파하 가나다라 마바사아 자차카타파하 } &
\SetCell{m,fg=blue,c} {너는 m옵션을 줬으니 가운데에 있으렴!! } \\ \\답답한
마음 답답한 마음 답답한 마음 답답한 마음 } &
```

```

\SetCell{t} {너는 t 옵션
줬으니 바닥에 붙어라!! 질문 질문 질문 질문 질문 질문 질문 질문 질문 질문
→ 질문
질문 질문} \\
가나다라 & 답답한 마음 & \SetCell{c,fg=blue} 질문 \\
가나다라 & 답답한 마음 & \SetCell{r,fg=blue} 질문 \\
가나다라 & 답답한 마음 & 질문 \\
\end{tblr}

```

3.12 Progress 예제 (2)

다음 샘플을 컴파일하려면 array, multirow, tabulary가 필요하다. 일부 패키지의 충돌 때문에 이 샘플의 결과는 그림으로 보인다.

```
\begin{tabulary}{\textwidth}%
  필요하다면 100mm 형태로 표전체길이 지정
  {|p{5em}|C|R|L|J|} % 첫번째 칼럼이 유난히 짧아서 비례할당한 결과는 주어진 텍
  ↵ 스트가 모두 들어가지
  % 못할 정도로 좁으므로, 5em을 강제할당함

\hline
짧은 칼럼 & \textcolor{blue}{가나다라} & 답답한 마음 & 질문 & 맨오른쪽 컬럼
↪ \tabularnewline

\hline
여기에 텍스트가 &
C 옵션을 줬으니 가운데 정렬을 해야겠지!!!!
가나다라 마바사아 자차카타파하 가나다라 마바사아 자차카타파하 &
\textcolor{blue}{너는 R옵션을 줬으니 오른쪽으로 가렴!!} \par
답답한 마음 답답한 마음 답답한 마음 답답한 마음 &
너는 L옵션 줬으니 왼쪽에 붙어라!!
질문 질문
질문 질문 &
J 옵션은 어떻게 동작할지 생각해보자. J는 Justification의 약자가 틀림없을지도
몰라. \tabularnewline
```

```

\hline
온다면 &
\multirow{2}{*[-11ex]}{가나다라} & \multicolumn{2}{c}{\textcolor{blue}{질}}
    ↳ 문}} &
그러니까 표가 만들어지네 신기해라 \tabularnewline
\cline{1-1}\cline{3-5}
잘되겠죠 & & 답답한 마음 &
\multicolumn{1}{r|}{\textcolor{blue}{질문}} &
100년 후에도 똑같은 모양으로 인쇄될 문서를 작성하고 싶으세요? \tabularnewline
\hline
잘되는군요 &
가나다라 & 답답한 마음 & 질문 &
그러면 \LaTeX{}을 써보세요. 그렇지만 100년후까지는 사실 장담 못하지 않을까요?
    ↳ \tabularnewline
\hline
\end{tabulary}

```

짧은 칼럼	가나다라	답답한 마음	질문	맨오른쪽 컬럼
여기에 텍스트가	C 옵션을 줬으니 가운데 정렬을 해야겠지?!!! 가나다라 마바사아 자차카타파하 가나다라 마바사아 자차카타파하	너는 R옵션을 줬으니 오른쪽으로 가렵!! 답답한 마음 답답한 마음 답답한 마음 답답한 마음 답답한 마음	너는 L옵션 줬으니 왼쪽에 붙어라!! 질문 질문 질문 질문 질문 질문 질문 질문 질문	J 옵션은 어떻게 동작할지 생각해보자. J는 Justification의 약자가 틀림없을지도 몰라.
온다면			질문	그러니까 표가 만들어지네 신기해라
잘되겠죠	가나다라		답답한 마음 질문	100년 후에도 똑같은 모양으로 인쇄될 문서를 작성하고 싶으세요?
잘되는군요	가나다라	답답한 마음	질문	그러면 \LaTeX{}을 써보세요. 그렇지만 100년후까지는 사실 장담 못하지 않을까요?

```

\begin{tblr}{%
width=\textwidth,
colspec={Q[5em]X[c]X[r]XX},
hlines,vlines
}
짧은 칼럼 & \SetCell{fg=blue} 가나다라 & 답답한 마음 & 질문 & 맨오른쪽 컬럼 \\
여기에 텍스트가 &
C 옵션을 줬으니 가운데 정렬을 해야겠지?!!!
가나다라 마바사아 자차카타파하 가나다라 마바사아 자차카타파하 &

```

```

{\textcolor{blue}{너는 R옵션을 줬으니 오른쪽으로 가렴!!}}\\
답답한 마음 답답한 마음 답답한 마음 답답한 마음 \\
& 너는 L옵션 줬으니 왼쪽에 붙어라!!
질문 \\
질문 질문 &
J 옵션은 어떻게 동작할지 생각해보자. J는 Justification의 약자가 틀림없을지도 몰라. \\
온다면 &
\SetCell[r=2]{c,m} 가나다라 & \SetCell[c=2]{fg=blue,c} 질문 & &
그러니까 표가 만들어지네 신기해라 \\
잘되겠죠 & & 답답한 마음 &
\SetCell{fg=blue,r} 질문 &
100년 후에도 똑같은 모양으로 인쇄될 문서를 작성하고 싶으세요?\\ 잘되는군요 &
가나다라 & 답답한 마음 & 질문 &
그러면 \LaTeX{} 을 써보세요. 그렇지만 100년후까지는 사실 장담 못하지 않을까요? \\
\end{tblr}

```

짧은 칼럼	가나다라	답답한 마음	질문	맨오른쪽 컬럼
여기에 텍스트가	C 옵션을 줬으니 가운데 정렬을 해야겠지?!!! 가나다라 마바사아 자차카타파하 가나다라 마바사아 자차카타파하	너는 R옵션을 줬으니 오른쪽으로 가렴!! 답답한 마음 답답한 마음 답답한 마음 답답한 마음 답답한 마음 답답한 마음	너는 L옵션 줬으니 왼쪽에 붙어라!! 질문 질문 질문 질문 질문 질문	J 옵션은 어떻게 동작할지 생각해보자. J는 Justification의 약자가 틀림없을지도 몰라.
온다면	가나다라	질문		그러니까 표가 만들어지네 신기해라
잘되겠죠		답답한 마음	질문	100년 후에도 똑같은 모양으로 인쇄될 문서를 작성하고 싶으세요?
잘되는군요	가나다라	답답한 마음	질문	그러면 \LaTeX{} 을 써보세요. 그렇지만 100년후까지는 사실 장담 못하지 않을까요?

4

실전 응용

4.1 문제

KTUG 질문 게시판의 [269842](#) 질문글이 올라왔다. 이 질문에 대한 답변을 작성한 과정을 기술하여, 실전 응용의 예로 삼고자 한다.

질문																																			
다음 그림과 같은 표를 그리시오.																																			
Table 8.6-2 Section Properties - Gross Composite Section																																			
<table border="1"><thead><tr><th>Section</th><th>A_i (in.²)</th><th>y_i (in.)</th><th>$A_i(y_i)$ (in.³)</th><th>I_o (in.⁴)</th><th>$A_i(Y-y_i)^2$ (in.⁴)</th></tr></thead><tbody><tr><td>Deck</td><td>391</td><td>46.5</td><td>18,182</td><td>1,681</td><td>79,956</td></tr><tr><td>Haunch</td><td>14.7</td><td>42.5</td><td>625</td><td>1</td><td>1,560</td></tr><tr><td>Girder</td><td>474</td><td>20</td><td>9,480</td><td>95,400</td><td>70,550</td></tr><tr><td>Total</td><td>879.7</td><td>-</td><td>28,287</td><td>97,082</td><td>152,066</td></tr></tbody></table>						Section	A_i (in. ²)	y_i (in.)	$A_i(y_i)$ (in. ³)	I_o (in. ⁴)	$A_i(Y-y_i)^2$ (in. ⁴)	Deck	391	46.5	18,182	1,681	79,956	Haunch	14.7	42.5	625	1	1,560	Girder	474	20	9,480	95,400	70,550	Total	879.7	-	28,287	97,082	152,066
Section	A_i (in. ²)	y_i (in.)	$A_i(y_i)$ (in. ³)	I_o (in. ⁴)	$A_i(Y-y_i)^2$ (in. ⁴)																														
Deck	391	46.5	18,182	1,681	79,956																														
Haunch	14.7	42.5	625	1	1,560																														
Girder	474	20	9,480	95,400	70,550																														
Total	879.7	-	28,287	97,082	152,066																														
$A_c = 879.7 \text{ in.}^2$																																			
$Y_{BC} = \frac{\sum A_i y_i}{\sum A_i} = \frac{28,287}{879.7} = 32.2 \text{ in.}$																																			
$Y_{TC} = 50 - 32.2 = 17.8 \text{ in.}$																																			
$I_c = 97,082 + 152,066 = 249,148 \text{ in.}^4$																																			
$S_{BC} = \frac{I_c}{Y_{BC}} = \frac{249,148}{32.2} = 7,735 \text{ in.}^3$																																			

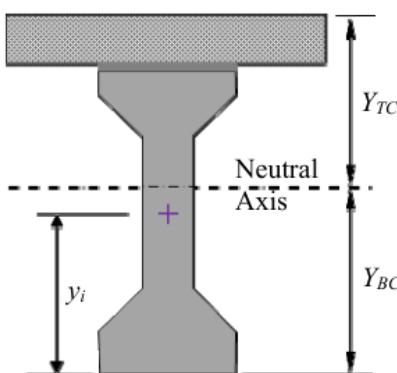


Diagram showing a gross composite section composed of a top flange, two haunches, and a girder. The girder is U-shaped. The total width of the section is labeled Y_{TC} . The height of the girder is labeled Y_{BC} . The haunch height is labeled y_i . A dashed horizontal line represents the Neutral Axis, marked with a purple cross symbol (+).

4.2 시도

4.2.1 표의 본체를 먼저 입력한다

일체의 조판 요소를 모두 나중으로 미루고, 일단 표의 본체만을 먼저 입력하는 것이다. 이 때, 표의 컬럼 수를 먼저 확인하고 (임시로) 컬럼은 전부 1 정렬하도록 해둔다. 그림이 들어갈 cell을 미리 표시해두면 좋다. 여기서는 **로 하였다. 그리고 셀의 위치를 확인하기 위하여 패션은 일단 모두 그어둔다.

```
\begin{tblr}{  
    width=\textwidth,  
    colspec={llllll},  
    hlines,vlines  
}  
Section & {$A_i$ \\ (in.$^2$)} & {$y_i$\\ (in.$^2$)} &  
→ {$A_i(y_i$\\(in.$^3$)} & {$I_o$ \\(in.$^4$)} &  
→ {$A_i(Y-y_i)^2$\\(in.$^4$)} \\  
Deck & 391 & 46.5 & 18,182 & 1,681 & 79,956 \\  
Haunch & 14.7 & 45.2 & 625 & 1 & 1,560 \\  
Girder & 474 & 20 & 9,480 & 95,400 & 70,550 \\  
Total & 879.7 & -- & 28,287 & 97,082 & 152,066 \\  
{$A_c=879.7 \mathrm{in}.^4$} & & & & & \\  
{$\mathrm{Y}_{BC} = \sum \frac{A_i y_i}{A_i} = \frac{28,287}{879.7} = 32.2 \mathrm{in}$} & & & & & \\  
{$Y_{TC} = 50 - 32.2 = 17.8 \mathrm{in}$} & & & & & \\  
{$I_c = 97,082 + 152,066 = 249,148 \mathrm{in}^4$} & & & & & \\  
{$S_{BC} = \frac{I_c}{Y_{BC}} = \frac{249,148}{32.2} = 7,735 \mathrm{in}^3$} & & & & & \\  
\end{tblr}
```

이렇게만 입력한 결과는 다음 그림과 같다.

Section	A_i (in. ²)	y_i (in. ²)	$A_i(y_i)$ (in. ³)	I_o (in. ⁴)	$A_i(Y - y_i)^2$ (in. ⁴)
Deck	391	46.5	18,182	1,681	79,956
Haunch	14.7	45.2	625	1	1,560
Girder	474	20	9,480	95,400	70,550
Total	879.7	–	28,287	97,082	152,066
$A_c = 879.7 \mathrm{in}^4$			**		
$Y_{BC} = \sum \frac{A_i y_i}{A_i} = \frac{28,287}{879.7} = 32.2 \mathrm{in}$					
$Y_{TC} = 50 - 32.2 = 17.8 \mathrm{in}$					
$I_c = 97,082 + 152,066 = 249,148 \mathrm{in}^4$					
$S_{BC} = \frac{I_c}{Y_{BC}} = \frac{249,148}{32.2} = 7,735 \mathrm{in}^3$					

첫 번째 열(row)의 두 줄로 된 것은, 그 셀을 중괄호로 둘러싸고 그 안에서 \\하면 행을 나눌 수 있다.

4.2.2 `multicolumn`과 `multirow`를 먼저 처리한다

표의 내용에 이상이 없으면, 이제 “셀을 합치는” 작업을 해야 한다.

- (1) $(6, 1), (7, 1), \dots, (10, 1)$ 셀은 모두 `multicolumns = 3`로 합쳐야 한다.
- (2) $(6, 4), (7, 4), \dots, (10, 4)$ 부터 $(6, 6), (7, 6), \dots, (10, 6)$ 까지 `multicolumns = 3`으로 합쳐야 한다.
- (3) $(6, 4), (6, 5), (6, 6)$ 은 `multirows = 5`로 합쳐야 한다.

이것을 코딩하면 다음과 같다. 다음 코드에서 표의 본체 부분은 앞서 입력한 그대로이므로 preamble 부분에서 어떻게 조절하는가를 유심히 보기 바란다.

```
\begin{tblr}{  
    width=\textwidth,  
    colspec={llllll},  
    hlines,vlines,  
    cell{6-Z}{1}={c=3}{},  
    cell{6-Z}{4}={c=3}{},  
    cell{6}{4-Z}={r=5}{}}  
    }  
Section & {$A_i$ \\ (in.$^2$)} & {$y_i$ \\ (in.$^2$)} &  
    & {$A_i(y_i$ \\ (in.$^3$)} & {$I_o$ \\ (in.$^4$)} &  
    & {$A_i(Y-y_i)^2$ \\ (in.$^4$)} \\  
Deck & 391 & 46.5 & 18,182 & 1,681 & 79,956 \\  
Haunch & 14.7 & 45.2 & 625 & 1 & 1,560 \\  
Girder & 474 & 20 & 9,480 & 95,400 & 70,550 \\  
Total & 879.7 & -- & 28,287 & 97,082 & 152,066 \\  
{$A_c=879.7 \mathrm{in}.^4$} & & & & & \\  
{$\displaystyle Y_{BC} = \sum \frac{A_i y_i}{A_i} = \frac{28,287}{879.7} = 32.2 \mathrm{in}.$} & & & & & \\  
{$Y_{TC} = 50 - 32.2 = 17.8 \mathrm{in}$} & & & & & \\  
{$I_c = 97,082 + 152,066 = 249,148 \mathrm{in}^4$} & & & & & \\  
{$\displaystyle S_{BC} = \frac{I_c}{Y_{BC}} = \frac{249,148}{32.2} = 7,735 \mathrm{in}^3$} & & & & & \\  
\end{tblr}
```

Section	A_i (in. ²)	y_i (in. ²)	$A_i(y_i)$ (in. ³)	I_o (in. ⁴)	$A_i(Y - y_i)^2$ (in. ⁴)
Deck	391	46.5	18,182	1,681	79,956
Haunch	14.7	45.2	625	1	1,560
Girder	474	20	9,480	95,400	70,550
Total	879.7	-	28,287	97,082	152,066
$A_c = 879.7 \text{ in.}^4$			**		
$Y_{BC} = \sum \frac{A_i y_i}{A_c} = \frac{28,287}{879.7} = 32.2 \text{ in.}$			**		
$Y_{TC} = 50 - 32.2 = 17.8 \text{ in.}$			**		
$I_c = 97,082 + 152,066 = 249,148 \text{ in.}^4.$			**		
$S_{BC} = \frac{I_c}{Y_{BC}} = \frac{249,148}{32.2} = 7,735 \text{ in.}^3$			**		

4.2.3 colspec을 정돈한다

문제로 제시된 표는 모든 셀을 다 중앙정렬(c)하고 있고, 각 열의 폭이 조금씩 다르다. 이것을 반영하여 colspec을 조절하자. 이 “상대 폭 설정 X”는 미리 정확하게 길이를 재어 설정하더라도 그 뒤에 합쳐진 셀이 있으면 길이가 약간 어긋나기도 하므로 한두 번 정도 시행착오를 거칠 수도 있다.

```
\begin{tblr}{}
  width={\textwidth},
  colspec={X[c,m,2]X[c,m,1.5]X[c,m,1.5]X[c,m,1.5]X[c,m,1.5]X[c,m,2]},
  hlines,vlines,
  cell{6-Z}{1}={c=3}{},
  cell{6-Z}{4}={c=3}{},
  cell{6}{4-Z}={r=5}{}
}
Section & {$A_i$ \\ (in.$^2$)} & {$y_i$ \\ (in.$^2$)} &
→ {$A_i(y_i$ \\ (in.$^3$)} & {$I_o$ \\ (in.$^4$)} &
→ {$A_i(Y-y_i)^2$ \\ (in.$^4$)} \\
Deck & 391 & 46.5 & 18,182 & 1,681 & 79,956 \\
Haunch & 14.7 & 45.2 & 625 & 1 & 1,560 \\
Girder & 474 & 20 & 9,480 & 95,400 & 70,550 \\
Total & 879.7 & -- & 28,287 & 97,082 & 152,066 \\
{$A_c=879.7 \mathrm{in.}^4$} & & & & & \\
{$\displaystyle Y_{BC} = \sum \frac{A_i y_i}{A_c} = \frac{28,287}{879.7} = 32.2 \mathrm{in.}$} & & & & & \\
{$\displaystyle Y_{TC} = 50 - 32.2 = 17.8 \mathrm{in.}$} & & & & & \\
{$\displaystyle I_c = 97,082 + 152,066 = 249,148 \mathrm{in.}^4$} & & & & & \\
{$\displaystyle S_{BC} = \frac{I_c}{Y_{BC}} = \frac{249,148}{32.2} = 7,735 \mathrm{in.}^3$} & & & & & \\
\end{tblr}
```

Section	A_i (in. ²)	y_i (in. ²)	$A_i(y_i)$ (in. ³)	I_o (in. ⁴)	$A_i(Y - y_i)^2$ (in. ⁴)
Deck	391	46.5	18,182	1,681	79,956
Haunch	14.7	45.2	625	1	1,560
Girder	474	20	9,480	95,400	70,550
Total	879.7	-	28,287	97,082	152,066
$A_c = 879.7 \text{ in.}^4$					
$Y_{BC} = \sum \frac{A_i y_i}{A_c} = \frac{28,287}{879.7} = 32.2 \text{ in.}$					
$Y_{TC} = 50 - 32.2 = 17.8 \text{ in.}$					
$I_c = 97,082 + 152,066 = 249,148 \text{ in.}^4.$					
$S_{BC} = \frac{I_c}{Y_{BC}} = \frac{249,148}{32.2} = 7,735 \text{ in.}^3$					

4.2.4 표의 수직 길이를 조정한다

대체로 행간격에 해당하는 것은 쉽게 조절할 수 있다. 이 표에서 어려운 점은 6행부터는 이전의 5행까지 보다 행의 수직 높이가 늘어난다는 점이다. 이것을 처리하는 방법을 다음 코드를 통해서 익혀두자.

```
\begin{tblr}{width=\textwidth,
colspec={X[c,m,2]X[c,m,1.5]X[c,m,1.5]X[c,m,1.5]X[c,m,1.5]X[c,m,2]},
hlines,vlines,
cell{6-Z}{1}={c=3}{},
cell{6-Z}{4}={c=3}{},
cell{6}{4-Z}={r=5}{},
stretch=1.5,
row{6-Z}={abovesep+=3pt,belowsep+=3pt}
}
Section & {$A_i$ \\ (in.$^2$)} & {$y_i$ \\ (in.$^2$)} & & & \\
& {$A_i(y_i)$ \\ (in.$^3$)} & {$I_o$ \\ (in.$^4$)} & & & \\
& {$A_i(Y-y_i)^2$ \\ (in.$^4$)} \\
Deck & 391 & 46.5 & 18,182 & 1,681 & 79,956 \\
Haunch & 14.7 & 45.2 & 625 & 1 & 1,560 \\
Girder & 474 & 20 & 9,480 & 95,400 & 70,550 \\
Total & 879.7 & -- & 28,287 & 97,082 & 152,066 \\
{$A_c=879.7 \mathit{in.^4}$} & & & & & \\
{$\displaystyle Y_{BC} = \sum \frac{A_i y_i}{A_c} = \frac{28,287}{879.7} = 32.2 \mathit{in.}$} & & & & & \\
{$Y_{TC} = 50 - 32.2 = 17.8 \mathit{in.}$} & & & & & \\
{$I_c = 97,082 + 152,066 = 249,148 \mathit{in.^4}$} & & & & & \\
{$\displaystyle S_{BC} = \frac{I_c}{Y_{BC}} = \frac{249,148}{32.2} = 7,735 \mathit{in.}^3$} & & & & & \\
\end{tblr}
```

Section	A_i (in. ²)	y_i (in. ²)	$A_i(y_i)$ (in. ³)	I_o (in. ⁴)	$A_i(Y - y_i)^2$ (in. ⁴)
Deck	391	46.5	18,182	1,681	79,956
Haunch	14.7	45.2	625	1	1,560
Girder	474	20	9,480	95,400	70,550
Total	879.7	-	28,287	97,082	152,066
$A_c = 879.7 \text{ in.}^4$					
$Y_{BC} = \sum \frac{A_i y_i}{A_c} = \frac{28,287}{879.7} = 32.2 \text{ in.}$					
$Y_{TC} = 50 - 32.2 = 17.8 \text{ in.}$			**		
$I_c = 97,082 + 152,066 = 249,148 \text{ in.}^4$					
$S_{BC} = \frac{I_c}{Y_{BC}} = \frac{249,148}{32.2} = 7,735 \text{ in.}^3$					

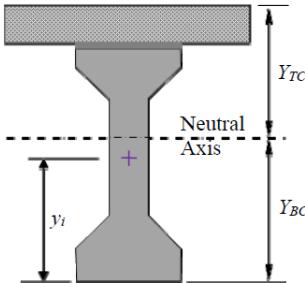
4.2.5 그림을 넣는다

그림은 **한 셀에 넣으면 된다. c, m 정렬이 주어져 있기 때문에 “대체로” 원하는 위치에 올 것이다. 문제는 그림의 적당한 크기를 정하는 일이다.

```
\begin{tblr}{width=\textwidth,
  colspec={X[c,m,2]X[c,m,1.5]X[c,m,1.5]X[c,m,1.5]X[c,m,1.5]X[c,m,2]},
  hlines, vlines,
  cell{6-Z}{1}={c=3}{},
  cell{6-Z}{4}={c=3}{},
  cell{6}{4-Z}={r=5}{},
  stretch=1.5,
  row{6-Z}={abovesep+=3pt,belowsep+=3pt}
}
Section & {$A_i$ \\ (in.$^2$)} & {$y_i$ \\ (in.$^2$)} &
\rightarrow {$A_i(y_i$ \\ (in.$^3$)} & {$I_o$ \\ (in.$^4$)} &
\rightarrow {$A_i(Y-y_i)^2$ \\ (in.$^4$)} \\
Deck & 391 & 46.5 & 18,182 & 1,681 & 79,956 \\
Haunch & 14.7 & 45.2 & 625 & 1 & 1,560 \\
Girder & 474 & 20 & 9,480 & 95,400 & 70,550 \\
Total & 879.7 & -- & 28,287 & 97,082 & 152,066

```

Section	A_i (in. ²)	y_i (in. ²)	$A_i(y_i)$ (in. ³)	I_o (in. ⁴)	$A_i(Y - y_i)^2$ (in. ⁴)
Deck	391	46.5	18,182	1,681	79,956
Haunch	14.7	45.2	625	1	1,560
Girder	474	20	9,480	95,400	70,550
Total	879.7	—	28,287	97,082	152,066
$A_c = 879.7 \text{ in.}^4$					
$Y_{BC} = \sum \frac{A_i y_i}{A_c} = \frac{28,287}{879.7} = 32.2 \text{ in.}$					
$Y_{TC} = 50 - 32.2 = 17.8 \text{ in.}$					
$I_c = 97,082 + 152,066 = 249,148 \text{ in.}^4.$					
$S_{BC} = \frac{I_c}{Y_{BC}} = \frac{249,148}{32.2} = 7,735 \text{ in.}^3$					



그림의 width가 그림 위의 세 열(col)의 폭을 넘어선다면 마지막 열이 확대된다. 마찬가지로 그림의 height가 왼쪽의 다섯 행(row)의 높이를 넘어선다면 마지막 행이 확대된다. 현재 나타난 결과는 별도의 조정이 필요없을 정도로 충분히 만족스러우므로 더이상 손댈 것이 없다.

4.2.6 패션과 폰트

끝으로, 폰트를 조금 조절하자. 헤더에 해당하는 first row는 bfseries로 식자하게 하려 하는데, 첫 번째 셀을 제외한 다른 셀은 모두 수식이다. 따라서 (1, 1) 셀의 폰트만 바꾸면 되겠다. 그리고 문제에서는 이 첫 셀의 수직 정렬을 상단에 맞추고 있으므로, 다음과 같이 하면 될 듯하다. (**mathbold**는 다른 문제이므로 이 예시에서는 반영하지 않겠다.)

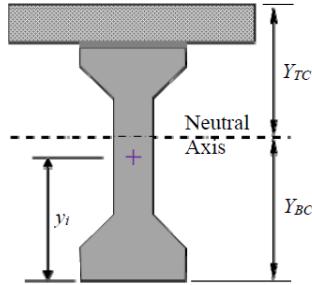
```

\begin{tblr}{%
    width=\textwidth,
    colspec={X[c,m,2]X[c,m,1.5]X[c,m,1.5]X[c,m,1.5]X[c,m,2]},
    % hlines,vlines,
    cell{6-Z}{1}={c=3}{},
    cell{6-Z}{4}={c=3}{},
    cell{6}{4-Z}={r=5}{},
    stretch=1.5,
    row{6-Z}={abovesep+=3pt,belowsep+=3pt},
    hline{1,2,Z}=.8pt,
    hline{3-Y}={solid},
    vline{2-Y}={solid},
    cell{1}{1}={}{{font=\bfseries}}
}
{Section\\~} & {$A_i$ \\ (in.$^2$)} & {$y_i$\\ (in.$^2$)} &
\rightarrow {$A_i(y_i$\\(in.$^3$)} & {$I_o$ \\(in.$^4$)} &
\rightarrow {$A_i(Y-y_i)^2$\\(in.$^4$)} \\
Deck & 391 & 46.5 & 18,182 & 1,681 & 79,956 \\
Haunch & 14.7 & 45.2 & 625 & 1 & 1,560 \\
Girder & 474 & 20 & 9,480 & 95,400 & 70,550 \\
Total & 879.7 & -- & 28,287 & 97,082 & 152,066 \\
{$A_c$=879.7 \mathrm{in.^4}} & & & & & \\
\rightarrow & & & & & \\
{$Y_{BC} = \sum \frac{A_i y_i}{A_i} = \frac{28,287}{879.7} = $} & & & & & \\
\rightarrow 32.2 \mathrm{in.} & & & & & \\
{$Y_{TC} = 50 - 32.2 = 17.8 \mathrm{in.}$} & & & & & \\
{$I_c = 97,082 + 152,066 = 249,148 \mathrm{in.^4}$} & & & & & \\
{$S_{BC} = \frac{I_c}{Y_{BC}} = \frac{249,148}{32.2} = 7,735 \mathrm{in.^3}$} & & & & & \\
\end{tblr}

```

Section	A_i (in. ²)	y_i (in. ²)	$A_i(y_i)$ (in. ³)	I_o (in. ⁴)	$A_i(Y - y_i)^2$ (in. ⁴)
Deck	391	46.5	18,182	1,681	79,956
Haunch	14.7	45.2	625	1	1,560
Girder	474	20	9,480	95,400	70,550
Total	879.7	-	28,287	97,082	152,066

$A_c = 879.7 \text{ in.}^4$
$Y_{BC} = \sum \frac{A_i y_i}{A_c} = \frac{28,287}{879.7} = 32.2 \text{ in.}$
$Y_{TC} = 50 - 32.2 = 17.8 \text{ in.}$
$I_c = 97,082 + 152,066 = 249,148 \text{ in.}^4.$
$S_{BC} = \frac{I_c}{Y_{BC}} = \frac{249,148}{32.2} = 7,735 \text{ in.}^3$



첫 번째 열의 행간격과 sep이 조금 좁은 것이 원래 문제로 제시된 표인데, 이것까지 일치시키는 것이 무슨 의미가 있겠는가 싶어서 여기서는 생략한다.

이로써, `\hline`도, `\multicolumn`도, `\multirow`도 없이, 아름다운 표를 그릴 수 있었다. 이를 통하여 강조하고자 하는 것은, “표의 내용과 스타일 지시 사항을 되도록 분리하여 표의 본체를 클린하게 유지하면서 preamble(문서의 *preamble*이 아니라, `<tblr>` 환경의 preamble을 가리킨다)을 통하여 스타일을 지정한다”는 관점을 유지하라는 점이다.

여러 페이지 표

5.1 여러 페이지에 걸치는 표

`tabulararray`는 `<longtabr>` 환경을 제공한다. 여러 페이지에 걸치는 표를 그리기 위한 도구이다. 전부터 있던 `longtable` 패키지가 이에 해당하는 기능을 제공하였다.

긴 표의 경우 고려해야 할 사항이 몇 가지 있다.

- (1) 이어지는 페이지에 별도의 캡션이 붙을 수 있다. 주로 ... (계속)과 같은 문자열을 포함하는 캡션 인데 표의 카운터를 증가시키거나 하지는 않는다.
- (2) 표의 최상단 한 행 또는 몇 행이 이어지는 페이지의 표에서 반복되어야 할 수 있다.
- (3) 표의 최하단 한 행 또는 몇 행이 특별하게 처리되거나, 페이지가 잘라지는 곳에서 다음 페이지에 계속과 같은 표지를 두어야 할 수 있고 이것은 표의 마지막 페이지에서 별도로 처리되어야 한다.
- (4) 표 안의 각주 등은 모아두었다가 마지막 페이지 표의 말미에 한꺼번에 출력한다.

5.2 Progress, 케이터 연혁

“KTUG 및 KTS 연혁: 2001~2021” 글에 `Progress`께서 `<longtblr>`의 예를 보여주셨으므로 이를 공부해보자. 표의 preamble 부분은 다음과 같다.

```
\DefTblrTemplate{contfoot-text}{normal}{\textsf{\small(다음 쪽에 계속)}}
\SetTblrTemplate{contfoot-text}{normal}
\DefTblrTemplate{conthead-text}{normal}{\small(계속)}
\SetTblrTemplate{conthead-text}{normal}

\begin{longtblr}[
    caption = {KTUG 및 KTS 연혁},
    label = {tab:history_of_KTUG_and_KTS},
]%
    [%
        colspec = {XX[1.5]X[10]},
        hline{1,Z} = {1pt},
```

```

hline{2} = {.5pt},
hline{3-Y} = {dashed},
width = \linewidth,
row{1}={c,m,gray!15,font=\sffamily},
rowhead = 1,% 표머리 되풀이
column{1}={c, font=\sffamily},
column{2}={font=\sffamily},
}
\SetCell[c=2]{c}날짜 & & 주요 사건 \\

```

처음 네 줄은 `contfoot-text`와 `conthead-text`를 한글화한 것이다.

`caption`과 `label`을 선택인자 안에 넣어두는 점에 주의하여야 한다. 여기에는 이밖에도 `theme`, `entry` (짧은 캡션), `note`, `remark` 등이 올 수 있다. `note`와 `remark`은 주석으로 긴 표의 맨 마지막에 놓인다.

첫 인자는 `preamble`이다. `<longtblr>`에만 해당하는 것을 들자면 `rowhead`과 `rowfoot`이 있는데 페이지마다 반복되는 행의 수를 지정한다.

헤더에는 주로 캡션 텍스트가 오는데, 만약 페이지마다 헤더행이 달라져야 한다면 `firsthead`, `middlehead`, `lasthead`를 별도로 지정할 수 있다.

이들에 대한 상세한 사항은 `tabulararray` 매뉴얼 Chapter 4에서 다루고 있으므로 이를 참고하여야 한다.

Progress께서 그리신 결과를 감상하기로 하자. 소스 코드는 동봉한 `progress_ktughist.tex`을 보라.

표 5.1: KTUG 및 KTS 연혁

날짜		주요 사건
2001	11. 8.	한글 \TeX 사용자 모임(The Korean \TeX Users Group) 활동 시작 (www.ktug.or.kr)
2002	11.	dvipdfmx 개발 (조진환) 세계 \TeX 사용자 모임(TUG)의 LUG (Local Users Group)에 등록된다 lshort-kr, Graphics FAQ, METAPOST 활용 등 사용자 설명서 지원 작업 (계속)
2003	3.	KTUG FAQ 위키 서비스 시작 (박원규)
	7.	TUG 2003 컨퍼런스 참석(조진환, dvipdfmx에 관한 Talk 진행, 세계 무대에서 처음 KTUG 소개)
2004	8.	은글꼴 트루타입폰트 공개 (은광희·박원규)
	12	HPack 프로젝트 (홍석호, MiK \TeX 에서 안정적인 한글 환경 지원) Hangul-ucs 제작 (김도현·김강수, 유니코드 기반의 한글 환경, DHhangul은 중세 국어를 비롯한 모든 한글 구현)
2005	4.	KTUG Collection 2005 제작·배포 (MiK \TeX 기반) H \TeX 1.0.1 발표 (은광희)
2006	7.~12.	KTUG Collection 2006 (ver. 1 & 2) 제작·배포 (W32 \TeX 기반)
	10.~11.	공주대학교 문서작성 워크숍 2006 개최 지원

(다음 쪽에 계속)

표 5.1: KTUG 및 KTS 연혁 (계속)

날짜	주요 사건
11.	안정적인 서버 환경 구축 및 faq/ project/ ftp 서비스 개시 (남수진의 도움으로 (주)다음커뮤니케이션·(주)컨텐츠플러그에서 후원)
2007 1. 27.	KTUG 설립 5주년 기념 학술대회 개최 한국TeX학회(The Korean TeX Society) 설립 (www.ktug.kr) 초대 회장: 조인성 교수(공주대) 선출
4. 30.	학술지 <i>The Asian Journal of TeX</i> 창간호(vol.1 no.1) 발간
7. 20.	koTeX 발표 (HETEX과 dhucs 통합)
8. 29.	KTUG Collection 2007 배포 (W32TeX 기반)
2008 1. 25.~26.	Asian TeX Conference 2008 개최
1. 26.	KTS 제1차 정기총회
9. 13.	KTUG Collection 2008 배포 (TeXLive 기반)
2009 2. 14.	2009 한국TeX학회 학술대회 및 정기총회 (한국외대) 제2대 회장: 조인성 교수 선출(연임)
12. 23.	koTeXLive 2009 배포 (TeXLive 기반)
2010 1. 30.	2010 한국TeX학회 학술대회 및 정기총회 (고려대)
11. 7.	koTeXLive 2010 배포 (TeXLive 기반)
2011 2. 12.	2011 한국TeX학회 학술대회 및 정기총회 (동국대) 제3대 회장: 김영록 교수(한국외대) 선출
7. 29.	koTeXLive 2011 배포 (TeXLive 기반)
2012 2. 4.	2012 한국TeX학회 학술대회 및 정기총회 (한국외대)
2013 2. 16.	2013 한국TeX학회 학술대회 및 정기총회 (KIAS) 제4대 회장: 김영록 교수 선출(연임)
3. 9.	koTeXLive 2012 배포 (TeXLive 기반)
3. 13.	KTUG/ KTS 서버 노후화에 따른 서비스 장애 발생
3. 23.~24.	cjk-ko, CTAN 및 TeXLive 등록 (김도현)
3. 27.~28.	xetex-ko, CTAN 및 TeXLive 등록 (김도현)
4. 20.	서버 교체 및 서비스 재개 (조진환, 이기황, 윤석천, 목봉균) KTUG/ KTS 대표 도메인 변경 http://www.ktug.org
5. 10.~11.	luatex-ko, CTAN 및 TeXLive 등록 (김도현)
9. 30.	koTeX v.2.0 배포
10. 12.	koTeXLive 2013 배포 (TeXLive 기반)
10. 23.~26.	TUG 2013 in Tokyo 참석 (김영록, 조진환)
10. 28.	koTeX, CTAN에 등록 (김도현, 김강수, 이기황)
11. 2.	koTeX, MiKTeX 2.9에 등록

(다음 쪽에 계속)

표 5.1: KTUG 및 KTS 연혁 (계속)

날짜		주요 사건
	11. 9.	공주대 문서작성 워크숍 2013 개최 지원 koTeX, TeXLive에 등록
2014	2. 14.	2014 한국텍학회 정기총회 및 KTUG 미팅 (한국외대)
	6.	TeXLive 2014에 koTeX 관련 패키지 대거 등재(except kotex-euc) *koTeXLive 2014 제작 불필요
	10. 8.	공주대학교 문서작성 워크숍 2014 A 개최 지원
	11. 1.	공주대학교 문서작성 워크숍 2014 B 개최 지원
연중		KTUG 사설 저장소: koTeX 관련 패키지의 꾸준한 개발 및 유지보수
2015	1. 31.	2015 한국텍학회 학술대회 및 정기총회 (NIMS) 제5대 회장: 김도현 교수(동국대) 선출
	10. 27.	공주대 문서작성 워크숍 2015 Pre-Workshop session A 개최 지원
	11. 3.	공주대 문서작성 워크숍 2015 Pre-Workshop session B 개최 지원
	11. 7.	공주대 문서작성 워크숍 2015 개최 지원
연중		지속적인 koTeX 관련 패키지 개발 및 유지보수 지속적인 KTUG 홈페이지 및 위키 관리
2016	1. 30.	2016 한국텍학회 학술대회 및 정기총회 (고려대)
	11. 5.	공주대 문서작성 워크숍 2016 개최 지원 *공주대 문서작성 워크숍 10주년 (since 2006)
	11. 29.	《KTS 설립 10주년 기념문집》발간 사업 준비
연중		지속적인 koTeX 관련 패키지 개발 및 유지보수 (XeTeX-ko: 문장부호, 자동 조사 등 관련 업데이트 LuaTeX-ko: 루아텍 버전업에 맞춘 core 업데이트 등) 지속적인 KTUG 홈페이지 및 위키 관리(Xpress Engine 계시판 및 MathJax 모듈 업데이트, 계시판 Data Base 정리, web server 캐싱 점검 등)
2017	2. 11.	2017 한국텍학회 학술대회 및 정기총회 (동국대) *한국텍학회 설립 10주년 (since 2007) 제6대 회장: 김도현 교수(동국대) 선출(연임) 《TeX: 조판, 그 이상의 가능성》(KTS 설립 10주년 기념문집) 발간 http://conf.ktug.org/2017/
	5. 4.	KTUG 서버 이전 및 복구: (공주대학교 → 한국컨텐츠인프라) (2017. 3.15. 국가사이버안전센터의 침해사고 관련 긴급조치로 외부 접속 차단)
	10.	네이버, CTAN 미러링 서비스 개시 http://mirror.navercorp.com/CTAN/
	11. 18.	공주대 문서작성 워크숍 2017 개최 지원

(다음 쪽에 계속)

표 5.1: KTUG 및 KTS 연혁 (계속)

날짜		주요 사건
연중		지속적인 koTeX 관련 패키지 개발 및 유지보수 (XeTeX-ko, LuaTeX-ko, oblivious 등) 지속적인 KTUG 홈페이지 및 위키 관리(미러링 서비스 속도 조정 등)
2018	2. 3.	2018 한국텍학회 학술대회 및 정기총회 (국가수리과학연구소 산업수학혁신센터)
	11. 17.	공주대 문서작성 워크숍 2018 개최 지원
연중		지속적인 koTeX 관련 패키지 개발 및 유지보수 지속적인 KTUG 홈페이지 및 위키 관리
2019	2. 16.	2019 한국텍학회 학술대회 및 정기총회 (고등과학원) 제7대 회장: 조명철 교수(고려대) 선출
	5. 20.	카카오, CTAN 미러링 서비스 개시 https://mirror.kakao.com/CTAN/
	11. 16.	공주대 문서작성 워크숍 2019 개최 지원
연중		지속적인 koTeX 관련 패키지 개발 및 유지보수 지속적인 KTUG 홈페이지 및 위키 관리
2020	2. 15.	2020 한국텍학회 학술대회 및 정기총회 (고려대)
	5. 30.	공주대 문서작성 워크숍 2020 개최 지원
	7. 29.	CRAN (Comprehensive R Archive Network), CTAN 및 KTUG 사설저장소 미러링 서비스 개시 https://cran.asia/tex/ , https://cran.asia/KTUG/texlive/tlnet/ *thanks to 배성호(CRAN 미러 동아시아 권역(0-Cloud-East-Asia) 관리자)
연중		지속적인 koTeX 관련 패키지 개발 및 유지보수 지속적인 KTUG 홈페이지 및 위키 관리
2021	1. 4.	2021 한국텍학회 학술대회 및 정기총회 미개최 결정 (COVID-19 팬데믹)
	11. 27.	공주대 문서작성 워크숍 2021 개최 지원
연중		지속적인 koTeX 관련 패키지 개발 및 유지보수 지속적인 KTUG 홈페이지 및 위키 관리