



فهرست

۴	۱	تایپ مقدماتی در $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}2_{\epsilon}$
۴	۱.۱	چگونه آغاز کنیم.
۶	۱.۲	چگونه متن را تایپ کنیم.
۶	۱.۲.۱	قلمها
۶	۱.۲.۲	اندازه‌ها
۶	۱.۲.۳	ایجاد فاصله
۸	۱.۲.۴	Box یا جعبه
۱۰	۱.۲.۵	نمادها و حروفها
۱۱	۱.۳	فرمول نویسی در $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$
۱۴	۱.۴	قلمهای ریاضی
۱۶	۱.۵	محیطهای متنی
۲۰	۱.۶	محیطهای ریاضی
۲۲	۱.۷	واحدهای قابل قبول در $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$
۲۳	۲	تایپ پیشرفته در $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}2_{\epsilon}$
۲۳	۲.۱	جداول
۲۷	۲.۲	آرایه و ماتریس
۳۱	۲.۳	شکل بندی قضایا
۳۳	۲.۴	دیاگرام‌های جایجائی
۳۴	۲.۵	دیاگرام‌های ساده
۳۶	۲.۶	عنوان بندی
۳۷	۲.۷	چگونه سربرگ درست کنیم

۳۹	چگونه يك مقاله را تايپ كنيم	۲.۸
۴۱	چگونه يك نامه را تايپ كنيم	۲.۹
۴۲	امكانات ویژه در $\text{\LaTeX}2\epsilon$	۳
۴۲	ارجاعات و ارجاعات جهشی يا hiper refrens	۳.۱
۴۴	چگونه يك تصوير را در متن فراخوانيم.	۳.۲
۴۶	چگونه رنگی بنويسيم.	۳.۳
۵۳	چگونه فارسی بنويسيم	۳.۴
۵۴	امكانات PostScript	۳.۵

فصل ۱

تایپ مقدماتی در $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X} 2_{\epsilon}$

۱.۱ چگونه آغاز کنیم.

برای تایپ به فرم $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X} 2_{\epsilon}$ باید در ابتدا نوع متن خود را مشخص کنیم. متنهای مرسوم عبارتند از مقاله (article)، گزارش (report) و کتاب (book). بنابر این سطر اول خود را چنین آغاز می‌کنیم.

```
\documentclass{article}
```

در دستور فوق می‌توانیم اندازه فونت مورد نظر را نیز درج کنیم، این کار را می‌توانیم با اضافه کردن یک کرشه حاوی اندازه مورد نظر انجام دهیم.

```
\documentclass[12pt]{article}
```

اندازه‌های موجود 10pt, 11pt, 12pt هستند که پیش فرض آن 10pt است.

توجه: انواع دیگری از متنها مانند {`phdthesis`, `a5paper`, `a4paper`, `amsart`, `letter`, ...} نیز موجودند و اگر از متن `foils` استفاده کنیم حتی می‌توانیم اندازه 30pt را نیز اختیار کنیم. همچنین همراه اندازه می‌توانیم دستورات [`12pt, twoside`] و [`12pt, twocolumn`] را بکار گیریم که اولی شماره صفحه را دوطرفه می‌کند مانند کتاب و دومی بصورت دستونی در صفحه می‌نویسد مانند روزنامه.

در نسخه‌های قدیمی $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ دستور نوع متن بصورت `\documentstyle[12pt]{article}` بکار می‌رفت و در نسخه‌های $\text{T}_{\text{E}}\text{X}$ و $\text{A}_{\text{M}}\text{S}-\text{T}_{\text{E}}\text{X}$ حتی این دستور موجود نیست. لذا با دیدن این دستور در یک متن متوجه می‌شویم که این متن به فرم $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ است.

بعد از انتخاب نوع و اندازه متن، ابتدا و انتهای متن را مشخص می‌کنیم. این عمل توسط دو دستور زیر انجام می‌پذیرد.

`\begin{document}`

`\end{document}`

حال می‌توانیم متن خود را شروع کنیم. متن را بین دو دستور فوق تایپ می‌کنیم. کلیدهایی را که برای تایپ بکار می‌بریم به سه دسته کلی تقسیم می‌شوند.

(۱) کلیدهای پایه‌ای که آنها را می‌توان بصورت معمولی در متن درج کرد.

(۱-۱) ۵۲ کلید حروف `ABC...Z abc...z`

(۱-۲) ۱۰ عدد `1 2 3 4 5 6 7 8 9 0`

(۱-۳) علائم تأکیدی `- ' ' ! : ; ,`

(۱-۴) پرانتز و کروشه `() []`

(۱-۵) علائم ریاضی `/ * + = - < >`

(۲) کلیدهای ویژه که خود جزء علائم دستوری هستند یعنی بعنوان یک دستور و یا قبل از یک دستور `LATEX` بکار می‌روند. مانند `\` که قبل از هر دستور بکار می‌رود و `-` که در فرمول به معنای اندیس است. این کلیدها عبارتند از:

`# $ % ~ - ^ { } " | @ \`

این علائم را نمی‌توانیم بطور معمولی در متن درج کنیم، مگر `{ @ | " }`. در بقیه موارد لازم است که یک `\` قبل از هر کدام قرار دهیم و برای خود `\` باید تایپ کنیم `\backslash $` زیرا `\\` به معنای یک سطر به پایین رفتن است.

(۳) کلیدهایی که ممکن است در صفحه کلید ما موجود نباشند. این گونه حروف و علائم را باید با دستورات خاصی تایپ کنیم، مانند:

`\alpha → α \Re → ℝ \mathbb{Z} → ℤ`

۱.۲ چگونه متن را تایپ کنیم.

ابتدا" باید با فونتها و اندازه های مختلف متن آشنا شویم.

۱.۲.۱ قلمها

در \LaTeX 2_ϵ چندین قلم برای تایپ کردن در نظر گرفته شده است که با دستور خاص روی متن مورد نظر اعمال می شود مثلاً " **Sans serif** که میتوان به صورت زیر بکار گیریم.

`\textsf{sans serif document font family}` → `sans serif document font family`

برای رؤیت انواع قلمهای متن و دستور آنها لطفاً " **کلیک** کنید.

۱.۲.۲ اندازهها

به غیر از اندازه ای که در `documentclass` تعیین می شود می توانیم اندازه يك حرف تا چند جمله را در متن تغییر دهیم. این اندازهها عبارتند از `tiny` ، `scriptsize` ، `footnotesize` ، `small` ، `normalsize` ، `large` ، `Large` ، `LARGE` ، `huge` و `Huge` . در زیر این اندازهها کاملاً مشخص هستند.

<code>\tiny</code>	sampletext
<code>\scriptsize</code>	sample text
<code>\footnotesize</code>	sample text
<code>\small</code>	sample text
<code>\normalsize</code>	sample text
<code>\large</code>	sample text
<code>\Large</code>	sample text
<code>\LARGE</code>	sample text
<code>\huge</code>	sample text
<code>\Huge</code>	sample text

۱.۲.۳ ایجاد فاصله

برای ایجاد فاصله افقی در متن از دستورهایی `\hspace{npt}` و `\hspace*{npt}` و برای ایجاد فاصله عمودی از دستورهایی `\vspace{npt}` و `\vspace*{npt}` استفاده می کنیم، که در آن `n` اندازه

فاصله مورد نظر است. بجای `pt` می‌توانیم از `cm, mm` و `in` نیز استفاده کنیم. لازم به ذکر است که در بعضی از موارد $\LaTeX 2\epsilon$ این دستورات را نادیده می‌گیرد لذا دو دستور `\basestretch` و بی‌ستاره ایجاد شده است که اگر دستور ستاره‌دار را بکار بگیرید $\LaTeX 2\epsilon$ حتماً این فاصله را منظور خواهد کرد. همچنین سه فاصله `\vspace{12pt}`، `\vspace{0.5in}` و `\vspace{1.5cm}` بین فاصله‌های عمودی استاندارد بوده و به ترتیب با سه دستور `\smallskip`، `\medskip` و `\bigskip` فراخوانده می‌شوند.

از دستورات دیگر میتوان از `\[` برای رفتن به سطر پایین و از `\]` برای رفتن به چند سانتیمتر پایین استفاده کرد. برای ایجاد فاصله افقی در متن جدول زیر کاملاً گویا است.

فاصله مثبت			فاصله منفی		
شکل کوتاه	شکل کامل	نتیجه	شکل کوتاه	شکل کامل	نتیجه
<code> \, </code>	<code> \thinspace </code>		<code>A \!A</code>	<code>A \negthinspaceA</code>	AA
<code> \: </code>	<code> \medspace </code>			<code>A \negmedspaceA</code>	AA
<code> \; </code>	<code> \thickspace </code>			<code>A \negthickspaceA</code>	AA
	<code> \quad </code>				
	<code> \qquad </code>				

اگر می‌خواهیم فقط به اندازه یک کلمه مشخص فاصله بگیریم از دستور `` استفاده می‌کنیم.

`|\alpha||\gamma||` → `|\alpha| |gamma| |`
`||\beta||\delta|` → `| |beta| |delta|`

اگر بین متن یک خط فاصله ایجاد کنیم، این امر یک پاراگراف جدید در متن ایجاد می‌کند. برای ایجاد پاراگراف از دستور `\par` نیز می‌توان استفاده کرد.

از نمونه‌های دیگر فاصله می‌توان از فاصله‌های بسط یافته نام برد که با دستورات `\dotfill`، `\hfill` و `\hrulefill` ایجاد می‌شوند. به نمونه‌های زیر دقت کنید.

فهرست `\dotfill` ۶۵ که تایپ شده آن می‌شود:

فهرست ۶۵

ردیف `\hfill` عنوان `\hfill` شماره صفحه که تایپ شده آن می‌شود:

نام `\hrulefill` نام خانوادگی `\hrulefill` شماره شناسنامه `\hrulefill` که تایپ شده آن می‌شود:

نام خانوادگی شماره شناسنامه

توجه: وقتی يك پاراگراف جدید شروع می‌شود خط اول آن بطور استاندارد کمی تورفتگی پیدا می‌کند. اگر بخواهیم از چنین تورفتگی جلوگیری کنیم باید دستور `\noindent` را در ابتدای سطر اول پاراگراف اضافه کنیم. بدیهی است که برای عکس این عمل یعنی برای ایجاد تورفتگی در يك خط دستور `\indent` را در ابتدای آن خط اضافه می‌کنیم. و اگر منظورمان این باشد که سطر اول تورفتگی نداشته باشد ولی از سطر دوم به بعد دارای تورفتگی باشد باید دستور `\hangindent` را قبل از دستور `\noindent` در سطر اول پاراگراف بکار ببریم. البته در دستور `\hangindent` اندازه تورفتگی نیز قابل تعیین است و دستور کلی آن بصورت زیر می‌باشد.

```
setlength{\hangindent}{npt}
```

توضیحات بالا يك مثال از این دستور با `n=30pt` می‌باشد. دستور دیگری که در اینجا می‌توانیم بکار گیریم `\hangafter` است که مشخص می‌کند، از چند سطر بعد دستور `\hangindent` اجرا شود و مقدار پیش فرض آن 1 است که با دستور `\setlength{\hangafter}{m}` مقدار آن به `m` قابل تغییر است.

از دستورات دیگر می‌توان از `\break` برای شکستن خط و `\newpage` برای شروع متن در يك صفحه جدید را نام برد. در نهایت می‌توانیم فاصله خطوط در متن را از اندازه پیش فرض آن تغییر دهیم و به چند برابر برسانیم. بعنوان مثال دستور زیر فاصله خطوط را به 1.5 برابر مقدار پیش فرض آن تغییر می‌دهد. این دستور را می‌توان قبل از `\begin{document}` بکار گرفت.

```
\renewcommand{\baselinestretch}{1.5}
```

توجه: دستور `\setlength` برای تغییر اندازه هر دستور اندازه‌پذیر بکار می‌رود.

۱.۲.۴ Box یا جعبه

در `LATEX` دو نوع جعبه بندی داریم يك نوع خطی و نوع دیگر چند خطی یا پاراگرافی است. از دستورات نوع اول و ساده‌ترین آنها `\mbox{text}` است که در يك سطر، می‌تواند از کشیده شدن متن تحت خود و یا شکسته شدن آن جلوگیری کند. در واقع `LATEX` به متن تحت `\box` بعنوان يك حرف بزرگ نگاه می‌کند.

دستور `\mbox` شکل خلاصه شده دستور `\makebox` است که فرم کلی آن بصورت زیر می‌باشد.

```
\makebox[طول][وضعیت]{متن}
```

اگر طول را حذف کنید طول به اندازه مورد نیاز متن محسوب می‌شود.

وضعیت شامل یکی از حروف **c-Center** (که پیش فرض است) ، **l-left** ، **r-Right** یا **s-Stretch** است. در حالت **s** اگر متن قابل بسط‌دادن باشد آن را بسط می‌دهد. به مثالهای زیر دقت کنید:

```
\makebox{Short title.}| Short title.|
\makebox[2in][l]{Short title.}| Short title. |
\makebox[2in]{Short title.}| Short title. |
\makebox[2in][r]{Short title.}| Short title.|
\makebox[2in][s]{Short title.}| Short title.|
```

همچنین می‌توان بجای طول از چهار دستور از پیش تعریف شده **\height** ، **\depth** ، **\totalheight** و **\width** نیز می‌توان استفاده کرد که مثالهای آن به خواننده واگذار می‌شود. دستورات دیگر **\frambox** است که دستور کلی آنها به صورت زیر است.

{ متن } [وضعیت] [طول] \frambox

در این دستور وضعیت همانند **\makebox** است با این تفاوت که می‌تواند **\makebox** را هم در بر بگیرد. شکل خلاصه شده آن **\fbox** است. توجه در شکل‌های خلاصه تنظیمات قابل ارائه نیست.

\framebox{\makebox[2in][s]{Shot title}} Shot title

از دستورات نوع دوم می‌توان به **\parbox** اشاره کرد که شکل کلی آن به صورت زیر است.

{ متن } { طول } [وضعیت داخلی] [عرض] [وضعیت] \parbox

وضعیت شامل چهار حالت **b** ، **c** ، **t** و **s** است که بترتیب حالات پایین، وسط، بالا و بسط یافته را مشخص می‌کنند.

\fbox{\parbox[c][1.5in][s]{3in}} {وضعیت داخلی}



دستور بعدی **minipage** است که دستور کلی آن بصورت زیر است.

{ طول } { وضعیت داخلی } [عرض] [وضعیت] \begin{minipage}

متن

\end{minipage}

تنظیمات همانند **\parbox** می‌باشد.

دو دستور بعدی را با مثال توضیح می‌دهیم.

`\rule{1.6ex}{1.6ex}` → ■

`\raisebox{0.5ex}{top}\raisebox{-0.5ex}{bottom}center` → ^{top}_{bottom}centre

آخرین دستور ارائه شده در این بخش `\savebox` می‌باشد. با این دستور می‌توان جعبه مورد نظر را ذخیره کرد و در موقع نیاز با دستور `\usebox` آن را فراخواند. شکل کلی آن به فرم زیر است.

`\savebox{نام جعبه}[طول][وضعیت]{متن}`

`\usebox{نام جعبه}`

وضعیت و طول همانند `\makebox` می‌باشد.

۱.۲.۵ نمادها و حروفها

همان طور که قبلاً اشاره شد. بعضی از نمادها و حروفها در صفحه کلید ما موجود نیستند. برای تایپ اینگونه نمادها بسته به این که آن نماد ریاضی و یا متنی است، از دستورات متفاوتی استفاده می‌کنیم. دستورات ریاضی در بخش ۱.۲ بیان می‌شوند. برای بکار بردن نمادهای مختلف متنی می‌توانیم از دستور `\symbol{ASCII-CODE}` استفاده کنیم. مثلاً `\symbol{26}` نماد `æ` را ایجاد می‌کند.

از جمله دستوراتی که می‌توان با آن نماد جدید ایجاد کرد دستور `\stackrel{پایین}{بالا}` می‌باشد. بعنوان مثال برای درج نماد $\stackrel{?}{=}$ تایپ می‌کنیم: `\stackrel{?}{=} $`. در این دستور نماد بالائی اندازه superscript را داراست و نماد را بعنوان يك رابطه درج می‌کند یعنی يك فضائی را طرف نماد ایجاد می‌کند. دستورات مشابه عبارتند از `\overset{نماد}{بالا}` و دستور متقابل آن `\underset{نماد}{پایین}` که در این دستورات نماد اول اندازه small را داراست و فضای اضافی نیز حذف شده است. اگر شما از زبانهای دیگری استفاده می‌کنید مثلاً "آلمانی" می‌توانید این حروف ویژ آن را با دستوراتی ایجاد کنید. مانند `ö` که با `\ddot{o}` ایجاد می‌شود و جزء نمادهای ریاضی است. البته این گونه حروف را می‌توانید مستقیماً در متن خود تایپ کنید. برای این منظور ابتدا باید این زبان را فراخوانید. این کار توسط دستور زیر که قبل از `\begin{document}` تایپ خواهید کرد صورت می‌پذیرد.

`\usepackage{german}`

حال می‌توانید مستقیماً حروف `Ä ä Ë ë Ì ï Ö ö Ü ü Ÿ Ç ç È è É é Ê ê Ò ò Ó ó Ô ô` را در متن خود بکار گیرید. تایپ این حروف بسته به ویراستاری دارد که بکار می‌گیرید. ویراستار WinEdt این توانائی را داراست.

۱.۳ فرمول نویسی در \LaTeX

\LaTeX زیباترین و ساده‌ترین روش را برای فرمول نویسی داراست. برای فرمول نویسی ابتدا باید محیط ریاضی را ایجاد کرد. محیط ریاضی در \LaTeX توسط `\$ \$` برای حالت درون خطی و `$$ $$` برای یک خط مستقل (وسط چین) و `\[\]` برای حالت پاراگرافی، ایجاد می‌شود. هر نماد ریاضی توسط یک نام مشخص می‌شود که آن را در محیط ریاضی مربوطه تایپ می‌کنیم. بعنوان مثال `\sum_{i=1}^n i` را برای $\sum_{i=1}^n i$ تایپ می‌کنیم.

توجه: برای دیدن اسامی نمادهای ریاضی می‌توانید به راهنماهای مختلف موجود مراجعه کنید و یا از ویراستارهای مخصوص \LaTeX مانند WinEdt یا WinShell استفاده کنید، که این اسامی را در داخل خودشان دارند.

به مثالهای زیر دقت کنید.

اگر تایپ کنید: `\left(\sum_{i=1}^n i\right)^2`

$$\sum_{i=1}^n i^3 = \left(\sum_{i=1}^n i\right)^2$$

نتیجه می‌شود:

همان طور که ملاحظه می‌کنید حدود \sum در دو حالت درون خطی و وسط چین فرق می‌کند. شما می‌توانید با دستورات `\limits` و `\nolimits` آنها را بدخواه تغییر دهید. برای مثال به صفحه ۱۳ نگاه کنید.

شکل نهائی	شکل اولیه
$3x^2$	<code>\$ 3x^{2}\$</code>
k_γ	<code>\$ k_{\gamma}\$</code>
$G \times H$	<code>\$ G \times H \$</code>

در $\LaTeX 2_\epsilon$ همه نمادهای ریاضی بطور طبیعی قابل اجرا نیستند و باید پیشاپیش آنها را فراخواند. این عمل توسط درج دستور زیر قبل از `\begin{document}` صورت می‌پذیرد.

`\usepackage{amsmath, amssymb}`

بعنوان مثال دیگر دستور کسر تعمیم یافته را مورد بررسی قرار می‌دهیم. شکل کلی آن به فرم زیر است.

`\genfrac{left-delim}{right-delim}{thickness}{mathstyle}{numerator}{denominator}`

thickness اندازه خط کسری را مشخص می‌کند که بر حسب pt می‌باشد.

mathstyle چهار مقدار ۰، ۱، ۲، ۳ را شامل است که بترتیب بیان کننده حالت `\displaystyle`، `\textstyle`، `\scriptstyle` و `\scriptscriptstyle` می‌باشند. به مثالهای زیر دقت کنید.

$$\frac{a+b}{c} \quad \left(\frac{a+b}{c}\right) \quad \left[\frac{a+b}{c}\right] \quad \left|\frac{a+b}{c}\right| \quad \frac{a+b}{c} \quad \frac{a+b}{c}$$

تایپ شده آنها بصورت زیر است.

`\genfrac{}{}{1pt}{}{a+b}{c}`

`\genfrac{()}{0pt}{}{a+b}{c}`

`\genfrac{[]}{1}{0pt}{}{a+b}{c}`

`\genfrac{||}{1}{1pt}{1}{a+b}{c}`

`\genfrac{}{}{1pt}{3}{a+b}{c}`

`\genfrac{}{}{1.5pt}{1}{a+b}{c}`

`\genfrac{}{}{1pt}{0}{a+b}{c}`

برای ایجاد کسره‌های مُسلسل از دستور `\cfrac` استفاده می‌کنیم بعنوان مثال:

$$\frac{1}{2 + \frac{1}{3 + \dots}}$$

که تایپ شده آن بصورت `\[\cfrac{1}{2+\cfrac{1}{3+\cdots}}\]` است. البته این دستور دارای عملگر وضعیت `[l]` و `[r]` است که وضعیت اعداد را در صورت کسر تعیین می‌کند. اگر مثال بالا را با `\cfrac[l]` بنویسیم، بصورت زیر خواهد بود.

$$\frac{1}{2 + \frac{1}{3 + \dots}}$$

مسئله دیگر در فرمول نویسی، تعیین حدود و زیرنویس است. يك دستور کلی اندیس گذاری دستور `\sideset` است که بصورت زیر بکار می‌رود.

`\sideset{_{ll}^{\ul}}{_{lr}^{\ur}}{large_op}`

بعنوان مثال اگر تایپ کنیم، $\prod_{i=1}^n i$ خواهیم داشت:

$$\prod_{i=1}^n i$$

برای زیرنویسهای چند خطی از دو دستور `\subarray` و `\substack` استفاده می‌کنیم.

$$\sum_{\substack{i < n \\ i \text{ even}}} i$$

$$\sum_{\substack{i < n \\ i \text{ even}}} x_i^2$$

اگر بجای `{1}` از `c,r` استفاده کنیم آنگاه اندیسه‌ها وسط چین و یا از راست مرتب خواهند شد. مطلب بعدی که به آن اشاره می‌کنیم تایپ انتگرالهای چند گانه است.

$$\int \iint \iiint \int \dots \int \phi$$

نمونه بالا بصورت زیر تایپ شده است.

`\int \iint \iiint \dots \int`

به دو روش می‌توانیم برای انتگرال حدود قرار دهیم، یکی بصورت $\int_{-\infty}^{\infty} e^{-x^2} dx = \sqrt{\pi}$ و دیگری بصورت $\int_{-\infty}^{\infty} e^{-x^2} dx = \sqrt{\pi}$ ، که تایپ شده آنها بصورت زیر است.

$$\int_{-\infty}^{\infty} e^{-x^2} dx = \sqrt{\pi}$$

$$\int_{-\infty}^{\infty} e^{-x^2} dx = \sqrt{\pi}$$

دستور `\smallint` نیز برای تایپ $\int_{-\pi}^{\pi} \cos(x) dx$ که از لحاظ اندازه کوچک است، بکار می‌رود. حال دستورات همنهشتی را بررسی می‌کنیم. جدول زیر این دستورات را بطور کامل بیان می‌کند.

دستور	نتیجه
<code>\a \equiv v \pmod{\theta}</code>	$a \equiv v \pmod{\theta}$
<code>\a \equiv v \bmod{\theta}</code>	$a \equiv v \pmod{\theta}$
<code>\a \equiv v \pmod{\theta}</code>	$a \equiv v \pmod{\theta}$
<code>\a \equiv v \pod{\theta}</code>	$a \equiv v (\theta)$

از نقطه نظر تایی توابع و عملگرها داخل متن نباید بصورت کج یا ایتالیک تایپ شوند. $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ توابعی مانند توابع مثلثاتی، لگاریتمی و حدی را درون خود دارد، که آنها را با دستوراتی مانند $\backslash\lim$ ، $\backslash\limsup$ ، $\backslash\log$ ، $\backslash\det$ ، $\backslash\hom$ و $\backslash\arcsin$ به اجرا در می‌آورد.

$$\lim_{x \rightarrow 0} f(x), \limsup_{x \in A}, \log x, \hom(A, B), \det(A), \arcsin(\pi/2)$$

اگر نیازمند به درج توابع جدیدی داخل متن هستیم، باید آن را بعنوان یک تابع معرفی کنیم که این عمل توسط دستور زیر که قبل از $\backslash\begin\{document\}$ نوشته می‌شود، صورت می‌پذیرد.

$\backslash\DeclareMathOperator\{\backslash\text{Op-Command}\}\{\text{Op-Name}\}$

مانند Aut که با $\backslash\DeclareMathOperator\{\backslash\text{Aut}\}\{\text{Aut}\}$ معرفی می‌شود.

دستور نهائی که نام می‌بریم \boxed است. این دستور برای جعبه کردن در محیط ریاضی بکار می‌رود. به مثال زیر دقت کنید.

$\backslash\boxed{\int_{-\infty}^{\infty} e^{-x^2} dx = \sqrt{\pi}}$

$$\int_{-\infty}^{\infty} e^{-x^2} dx = \sqrt{\pi}$$

برای دیدن نمونه‌ای از دستورات لطفاً "کلیک" کنید.

۱.۴ قلمهای ریاضی

برای استفاده از قلمهای گوناگون در متن ریاضی ابتدا پکیجهای زیر را فرامی‌خوانیم.

$\backslash\usepackage\{amsfonts\}$

$\backslash\usepackage[mathscr,mathcal]\{eucal\}$

حال به مثالهای زیر دقت کنید.

$\backslash\mathfrak\{A B C D\}$ $\mathfrak{A B C D}$

$\backslash\mathfrak\{a b c d\}$ $\mathfrak{a b c d}$

$\backslash\mathscr\{A B C D\}$ $\mathscr{A B C D}$

$\backslash\mathcal\{A B C D\}$ $\mathcal{A B C D}$

$\backslash\mathbb\{A B C D\}$ $\mathbb{A B C D}$

اگر شما فقط از قلمهای `mathbb` و `mathcal` استفاده می‌کنید، کفایت `amsmath` را فراخوانید. همانند حالت متنی می‌توانیم از دستورات `mathbf`، `mathsf`، `mathtt`، `mathit` و `mathrm` استفاده کنیم و یا آنها را با هم بکار گیریم. مانند:

```
\mathrm{ABCabc}   ABCabc
\mathbf{ABCabc}   ABCabc
\mathsf{ABCabc}   ABCabc
\mathtt{ABCabc}   ABCabc
\mathit{ABCabc}   ABCabc
\mathit{\mathbf{ABCabc}}  ABCabc
```

برای پُررنگ کردن حروف از دستور `mathbf` می‌توانیم استفاده کنیم ولی برای پُررنگ کردن نمادها این دستور کارساز نیست برای این منظور از دستور `\boldsymbol` استفاده می‌کنیم مانند " σ " ولی نمادهایی مانند \sum هستند که بدین صورت پُررنگ نمی‌شوند. برای پُررنگ کردن اینگونه نمادها از دستور `\pmb` که توسط پکیج `amsbsy` فراخوانده می‌شود استفاده می‌کنیم. البته این پکیج بطور خودکار توسط پکیج `amsmath` فراخوانده می‌شود. دستور دیگری نیز بنام `\mathop` موجود است که نمادها را درشت تر می‌کند. به مثالهای زیر توجه کنید.

$$\sum_{i=1}^n i^2 \quad \boldsymbol{\sum}_{i=1}^n i^2 \quad \pmb{\sum}_{i=1}^n i^2$$

که بترتیب زیر تایپ می‌شوند.

```
\sum_{i=1}^n i^2
```

```
\pmb{\sum}_{i=1}^n i^2
```

```
\mathop{\pmb{\sum}}_{i=1}^n i^2
```

برای بزرگ کردن نمادها میتوان از دستورهایی `\big`، `\Big`، `\bigg`، `\Bigg` و `\Biggl` استفاده کرد. البته این دستورات را برای نمادهای چپ مانند " $($ "، بصورت `\bigl` و برای نمادهای راست " $)$ " بصورت `\bigr` نوشته می‌شود. دستور `\bigm` را هم برای نمادهایی که به عنوان یک رابطه نوشته می‌شوند، بکار می‌رود. اختلاف این دستورات معمولاً چندان مخصوص نیست و در اکثر موارد می‌توان از حالت نوشتاری `\big` استفاده کرد. ولی برای زیبا نویسی بهتر است هر کام را جای خود بکار برد. به مثال فوق توجه کنید. $($ $($ $($ $($ $($ ، که تایپ شده آن بصورت زیر است.

```
$ (\quad \bigl(\quad \Bigl(\quad \biggl(\quad \Biggl(\quad $
```

دستورات `\left, \right` نیز در بعضی موارد بکار می‌روند، و نماد مورد نظر را نسبت به نمادهای دیگر بزرگ می‌کنند. مانند:

$$\left\{ x \mid \int_0^x t^2 dt \leq 5 \right\}$$

که تایپ شده آن بصورت زیر است.

```
[\left\{, x \biggm| \int_{0}^x t^2 dt \leq 5, \right\}]
```

توجه داشته باشید که دو دستور `\left` و `\right` را باید باهم بکار ببرید. اگر فقط به `\left\{` نیاز دارد، در انتهای فرمول دستور `\right.` را بنویسید و بلعکس.

۱.۵ محیط‌های متنی

در $\text{\LaTeX} 2\epsilon$ چندین دستور برای ایجاد محیط‌های متنی موجود است که همه آنها بصورت کلی زیر می‌باشند.

```
\begin{list-name}
```

```
\item ...
```

```
\item ...
```

```
\item ...
```

```
\end{list-name}
```

که در آن `list-name` یکی از موارد `enumerate`، `itemize`، `discription` و `list` می‌باشد. در هر خط `item` می‌تواند شامل یک [عبارت] باشد که این عبارت می‌تواند شماره یا نام را دربر بگیرد. البته این عبارت قابل حذف است و در هر مورد \LaTeX به نوعی علامت گذاری می‌کند. بعنوان مثال `enumerate` به هر خط یک شماره می‌دهد و `itemize` هر خط را با یک نقطه سیاه آغاز می‌کند. ولی اگر عبارتی نوشته شود هر خط با آن عبارت شروع خواهد شد. به مثال زیر دقت کنید که در خط اول آن عبارت حذف و در بقیه خطوط مشابه عبارت `\item` [فردوسی] نوشته شده .

۱. از شعرای بنام ایران چهار تن را می‌توان نام برد که سرآمد دیگر شعرا هستند.

مولوی: مثنوی سُرای بنام ایران.

سعدی: قصیده سُرای بنام ایران.

حافظ: غزل سُرای بنام ایران.

فردوسی: حماسی سُرای بنام ایران.

به این نکته هم باید دقت داشت که بین `\item` –ها نباید هیچ خط اضافی درج کرد. از موارد دیگر می‌توان به `quote` و `quotation` اشاره کرد. این دو دستور مشابه هم هستند با این تفاوت که `quotation` می‌تواند شامل چند پاراگراف باشد که با `\begin{list-name}` شروع و به `\end{list-name}` ختم می‌شوند ولی `quote` برای نقل قول‌های کوتاه است.

<code>\begin{quote}</code>	<code>\begin{quotation}</code>
Text	Text 1
<code>\end{quote}</code>	<code>\begin{quotation}</code>
	Text 2
	<code>\end{quotation}</code>
	Text 3
	<code>\end{quotation}</code>

از موارد دیگر می‌توان به `verse` اشاره کرد. `verse` دارای این تفاوت است که انتهای هر خط آن با `\\` مشخص می‌شود.

```
\begin{verse}
Text ...\\
Text ...\\
Text ...\\
\end{verse}
```

به مثال زیر دقت کنید.

أبله و فرزانه را فرجام خاك
 جایگاه هر دو اندر يك مَغاك
 * * *
 هر که نآموزد از گذشت روزگار
 نیز نآموزد ز هیچ آموزگار
 * * *
 هیچ شادی نیست اندر این جهان
 برتر از دیدار روی دوستان
 هیچ تلخی نیست بر دل تلخ تر
 از فراق دوستان پُر هنر

رودکی

اگر دستور `\usepackage{enumerate}` را قبل از `\begin{document}` درج کنیم، آنگاه محیط متنی `enumerate` را با امکانات عالی در اختیار خواهیم داشت. این دستور دارای عملگری میباشد که امکان درج هر نوع نما و یا عدد را برای شمارش سطرها فراهم می‌کند. شکل کلی آن بصورت زیر است.

```
\begin{enumerat}[Counter]
\end{enumerate}
```

اگر در `Counter` تایپ کنیم `a` ، `A` ، `i` ، `I` و یا `1` آنگاه شمارنده سطور بترتیب بصورت: حروف کوچک الفبائی `{a, b, c, ...}` ، بزرگ الفبائی `{A, B, C, ...}` و حروف کوچک یونانی `{i, ii, iii, iv, ...}` ، بزرگ یونانی `{I, II, III, IV, ...}` و یا اعداد ریاضی یا عربی درج خواهند شد. با تایپ هر نمادی به غیر از موارد بالا خود آن نماد عیناً تکرار می‌شود. مثلاً" با تایپ `[{A-}1]` سطرها با `(A-1)` ، `(A-2)` ، ... شماره گذاری می‌شوند.

دستور بعدی `list` است که امکانات بیشتری دارد. شکل کلی آن بصورت زیر است.

```
\begin{list}{default-label}{declarations}
\item ...
\item ...
\item ...
\end{list}
```

در `default-label` شما هر نمادی را می‌توانید قرار دهید. و در `declarations` شما می‌توانید تنظیمات مختلف از جمله فاصله `item` -ها از یکدیگر و از جوانب و موارد دیگر را بکار گیرید. دستوراتی که در این قسمت می‌توانید قرار دهید بصورت زیر میباشند.

```
\begin{list}{}{\setlength{\leftmargin}{30pt}
\setlength{\rightmargin}{0pt}
\setlength{\itemindent}{14pt}
\setlength{\labelwidth}{40pt}
\setlength{\labelsep}{5pt}
\setlength{\parsep}{0.5ex plus 0.2ex minus 0.1ex}
\setlength{\itemsep}{0ex plus 0.1ex minus 0ex}}
```

آخرین موردی که در این فصل به آن می‌پردازیم `thebibliography` است. شکل کلی آن بصورت زیر است.

```
\begin{thebibliography}{99}{text}
\bibitem ...
\bibitem ...
\bibitem ...
\end{thebibliography}
```

همانند `itemize` مقابل `bibitem` يك كروشه باز می‌کنیم که شامل نام نویسنده است در صورت حذف آن، سطرها بترتیب شماره گذاری خواهد شد. برای مثال لطفاً **کلید** کنید.

۱.۶ محیط‌های ریاضی

قبلاً" با محیط $\$ \$$ و $[\backslash]$ آشنا شدیم. علاوه بر این دو مورد، موارد دیگری از محیط‌های ریاضی داریم که با $\begin{\text{name}}$ شروع و به $\end{\text{name}}$ ختم می‌شوند. این محیطها عبارتند از equation ، eqnarray ، align ، gather و multline . خصوصیت این محیطها عبارتند از:

- (۱) هر سطر با $\backslash\backslash$ تمام می‌شود مگر سطر آخر.
- (۲) به هر سطر يك شماره اختصاص داده می‌شود.
- (۳) در هر سطر قبل از $\backslash\backslash$ با دستور $\tag{\text{symbol}}$ می‌توان بجای شماره هر نمادی را درج کرد. این نماد داخل يك جفت پرانتز خواهد بود. اگر از دستور $\tag*{\text{symbol}}$ استفاده شود نماد بدون پرانتز درج می‌شود.
- (۴) اگر دستور \notag را قبل از اتمام سطر بکار گیریم، شماره سطر حذف می‌شود.
- (۵) با دستور $\raisetag{\text{npt}}$ که در آن n يك عدد است می‌توان فاصله نماد یا شماره را از فرمول تغییر داد.
- (۶) می‌توان با دستور $\intertext{\text{Text}}$ بعد از هر سطر، بین آن سطر و سطر بعدی متن مورد نظر را درج کرد.
- (۷) درون این محیطها سطر اضافی نباید درج کرد.
- (۸) اگر این دستورات را ستاره دار بکار گیریم، شماره سطرها حذف خواهد شد.
- (۹) با قرار دادن عملگر $[\text{leqno}]$ یا $[\text{reqnu}]$ در پکیج amsmat می‌توان سمت شماره‌ها را تغییر داد (مانند: $\usepackage[\text{reqno}]{\text{amsmath}}$). پیش فرض آن leqno است. البته این دستور را در تعریف نوع متن نیز می‌توان قرار داد. عملگر دیگر flqno است که موجب می‌شود شماره در انتهاالیه خط درج شود. این عملگر را می‌توان همراه دو عملگر قبل تایپ کرد. همچنین با عملگر $[\text{tbtags}]$ (top - or - bottom tags) می‌توان نمادهای درج شده را به بالا یا پایین فرمول انتقال داد. پیش فرض آن centertags می‌باشد.

۱۰) با دستور `\numberwithin{equation}{section}` می‌توان شماره‌ها را برحسب بخش، زیربخش یا فصل درج کرد. بجای section می‌توان نوشت `subsection` یا `chapter`.

این محیطها تفاوت‌هایی با هم دارند. `equation` مانند `[]` عمل می‌کند، با این تفاوت که فرمول شماره دار است. `eqnarray` و `align` هر دو می‌توانند همانند جدول، هر سطر را ستون بندی کرده و هر قسمت را دقیقاً زیر هم درج کنند. با این تفاوت که فاصله این ستونها در این دو دستور متمایز است. و تعداد ستونها در `eqnarray` کمتر از سه ستون است. مانند جدول هر ستون با "&" شروع می‌شود. به مثال زیر دقت کنید.

$$\begin{aligned} x + y + z g(x) = & \quad \quad \quad x + y z f(x) = & \quad \quad \quad) ۱.۱(\\ (x + y)(x + z)(y + z) k(x) = & \quad \quad \quad xy + xz + yz h(x) = & \quad \quad \quad \end{aligned}$$

$$\begin{aligned} g(x) = x + y + z & \quad \quad \quad f(x) = x + yz & \quad \quad \quad) ۱.۲(\\ k(x) = (x + y)(x + z)(y + z) & \quad \quad \quad h(x) = xy + xz + yz & \quad \quad \quad \end{aligned}$$

تایپ آنها بصورت زیر است.

```
\begin{align}
f(x) &= x + yz & g(x) &= x + y + z \\
h(x) &= xy + xz + yz & k(x) &= (x + y)(x + z)(y + z) \notag
\end{align}
```

```
\begin{eqnarray}
& f(x) = x + yz & g(x) = x + y + z \\
& h(x) = xy + xz + yz & k(x) = (x + y)(x + z)(y + z) \notag
\end{eqnarray}
```

دستور `flalign` نیز موجود است که فرمول سمت چپ را در انتها لیه سمت چپ درج می‌کند و فرمول سمت راست را نیز متشابهاً انتها لیه سمت راست درج می‌کند. `gather` مانند دستورات پیش ستون بندی نمی‌کند، ولی فرمولها را وسط چین می‌کند. `multline` برای یک فرمول طولانی بکار میرود و می‌توان هر قسمت از فرمول را در یک سطر نوشت. این دستور به فرمول فقط یک شماره اختصاص می‌دهد. همچنین اولین سطر را در سمت چپ، آخرین

سطر را در سمت راست و سطرهای دیگر را بین این دو می‌چیند. در این دستور می‌توان از دو دستور $\text{\shoveleft\{formul}}$ و $\text{\shoveright\{formul}}$ در هر سطر استفاده کرد که موجب می‌شود آن سطر به سمت راست یا چپ کشیده شود.

با دستور $\text{\renewcommand\{thesection\}\{number-format}}$ می‌توان نوع درج شماره‌ها را تغییر داد. اگر بجای number-format بنویسیم:

$\text{\arabic\{chapter\}-\roman\{section\}.\alph\{subsection}}$

شماره زیر بخش ۲ از بخش ۳ از فصل ۳ بصورت (3-ii.b) درج می‌شود. با تایپ:

$\text{\theparentequation.\roman\{equation}}$

شماره‌ها بصورت (2.i) ، (2.ii) و نظایر آن درج می‌شوند.

\LaTeX به align بعنوان يك نماد بزرگ نگاه می‌کند. از اینرو اگر در انتهای صفحه فضای کافی برای درج آن نداشته باشد آن را نمی‌شکند و به صفحه بعد انتقال می‌دهد. برای رفع این مشکل از دستور زیر استفاده می‌کنیم.

$\text{\allowdisplaybreaks\{\begin\{align} \quad \quad \quad \end\{align}\}}$.

۱.۷ واحد‌های قابل قبول در \LaTeX

پنج واحد اصلی در \LaTeX که کاربری زیادی دارند عبارتند از in-Inch ، cm-Centimetre ، mm-Milimetre ، pc-Pica و pt-Point . ولی واحد‌های دیگری مانند bp-Big Point ، cc-Cicero و dd-Didot Point و sp-Scaled point موجودند که در \LaTeX قابل قبول هستند. علاوه بر اینها سه واحد $\text{mu-Mathmatical Unit}$ ، em برابر طول "M" و ex برابر ارتفاع "x" نیز موجودند که در محیط ریاضی کاربرد دارند. در واقع $1\text{em} = 10.00002\text{pt}$.

ذیلاً نسبت این واحدها را با هم، ارائه می‌کنیم.

$$1 \text{ in} = 72 \text{ bp} = 72.27 \text{ pt} = 2.54 \text{ cm}$$

$$1 \text{ cc} = 12 \text{ dd}$$

$$1 \text{ cm} = 10 \text{ mm}$$

$$1 \text{ pc} = 12 \text{ pt}$$

$$1 \text{ pt} = 65536 \text{ sp}$$

$$1157 \text{ dd} = 1238 \text{ pt}$$

فصل ۲

تایپ پیشرفته در $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$

۲.۱ جداول

جدول نویسی در $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ معمولاً پیچیده و پُرزحمت است. ولی اگر دستورات آن را با دقت و حوصله تایپ کنیم کاری ساده خواهد بود. اولین دستور که یک جدول نویسی ساده است، `tabular` می‌باشد. شکل کلی آن بصورت زیر است.

```
\begin{tabular}[Table Position]{Column Options} \hline
```

```
□ & □ & □ \\
```

```
□ & □ & □ \\
```

```
□ & □ & □ \\
```

```
\end{tabular}
```

`Table Position` شامل `t-Top` و `b-Bottom` است که موقعیت جدول را در صفحه مشخص می‌کند. حالت پیش فرض آن `c-Center` می‌باشد.

در `Column Options` دستورات راجع به ستون‌های جدول را قرار می‌دهیم. این دستور حتماً باید نوشته شود در غیر این صورت خطا حاصل می‌شود. در این دستور اطلاعات هر ستون را وارد می‌کنیم این اطلاعات شامل، نحوه قرار گرفتن متن داخل هر ستون از لحاظ راست چین، وسط چین و چپ چین که با `{r c l}` مشخص می‌شوند می‌باشد. همچنین می‌توان با قرار دادن نماد `"|"` مابین حروف (مانند `{||r|c|l||}`) خطوط عمودی ایجاد

کرد. برای ایجاد خطوط افقی از دستور `\hline` استفاده می‌کنیم که بعد از هر سطر بویژه بعد از `\\` بکار می‌رود. اگر از خطوط افقی استفاده می‌کنید باید سطر انتهائی جدول را با `\hline` به پایان برسانید. همانطور که در بالا مشخص است هر ستون با `&` تمام می‌شود و هر سطر با `\\`. در واقع `tabular` از دستورات درون خطی است و می‌شود

۱	۲	۳
۴	۵	۶
۷	۸	۹

آن را داخل يك سطر نوشت مانند: `۴ ۵ ۶`. از دستوراتی که می‌توان در `Column`

`Options` بکار برد دستور پهنای ستون است، `p{width}` که `width` اندازه پهنای ستون را تعیین می‌کند. مانند `{|| p{2cm} |r|1||}`. اگر تعداد ستونهای جدول بالا باشد، می‌توان از دستور `*{i}{c}` برای ایجاد i کپی از "c" استفاده کرد. بعنوان مثال بجای `{|c|c|c|}` می‌نویسیم `{|*{3}{c|}`. دستور دیگر `@{ text}` میباشد که موجب درج يك متن یا نماد در تمام سطرهاى يك ستون می‌شود. به مثال زیر دقت کنید.

<code>\partial</code>	∂
<code>\aleph</code>	\aleph
<code>\infty</code>	∞

تایپ شده آن بصورت زیر است.

```
\begin{tabular}{|| @ { $ \backslash$ } c c ||} \hline
```

```
partial & $ \partial$ \\
```

```
aleph & $ \aleph$ \\
```

```
infty & $ \infty$ \\ \hline
```

```
\end{tabular}
```

همانطور که مشخص است "||" دو خط عمودی ایجاد می‌کند. فاصله دو خط مجاور در جدول قابل تغییر است و دستور `\doublerulesep` میباشد. برای تغییر آن از دستور `\setlength\doublerulesep{5mm}` استفاده می‌کنیم. همچنین کلفتی خطوط جدول نیز قابل تغییر است، با دستور `\setlength\arrayrulewidth{2pt}`.

بجای `\hline` می‌توان از `\cline{a-b}` استفاده کرد، که يك خط افقی از ستون `a` تا ستون `b` رسم می‌کند. دستور مفید بعدی `\multicolumn` است که محتوای ستونها را در يك سطر تغییر می‌دهد.

`\multicolumn{Column Number}{Column Options}{Text}`

1		
	2	
	3	
	4	
	5	
6	text	

در مثال بالا سطر اول بصورت `\begin{tabular}{|| c c | c | r ||} \hline \hline` تایپ شده و در انتهای سطرهای دوم تا چهارم دستور `\cline{3-4}` ، در انتهای سطر پنجم `\cline{2-4}` تایپ شده و سطر ششم نیز بصورت `6 & \multicolumn{2}{| 1}{ text} & \\\ \hline \hline` تایپ شده است. همچنین اندازه `\doublerulesep` به 0.2pt تغییر داده شده است. دستور بعدی `table` است که بسط یافته `tabular` می باشد.

`\begin{table}`

`\begin{tabular}[Table Position]{Column Options} \hline`

`\end{tabular}`

`\caption{text}`

`\end{table}`

در `caption` عنوان جدول را درج می کنیم. به دلخواه می توانیم این دستور را حذف کنیم. دستور بسیار مفید بعدی `longtable` می باشد که برای استفاده از آن ابتدا باید پکیج آن را که به همین نام است فراخوانید. سپس زیر آن را تایپ کنید.

`\begin{longtable}[Position]{|| c | r | c ||}`

`\caption[Options]{captions text} \\\ \hline`

`□ & □ & □ \\\`

`□ & □ & □ \\\`

`□ & □ & □ \\\ \hline`

`\end{longtable}`

این دستور همانند دستورات قبل است فقط با این تفاوت که در صورت طولانی بودن جدول در حدود چند صفحه و یا اگر جدول در انتهای صفحه باشد این دستور بطور خودکار جدول را بریده و بقیه جدول را در صفحه بعد درج می‌کند. دستوراتی که می‌توان برای نمایش این جدول بکار برد به شرح زیر است.

۱- دستورات موقعیت جدول در صفحه

چهار دستور `\LTleft`، `\LTright`، `\LTpost` و `\LTper` بترتیب فاصله جدول را از چپ، راست، بالا و پایین تعیین می‌کنند. مقادیر پیش فرض آنها چنان است که جدول در وسط صفحه بوده و از هر طرف تا ابعاد صفحه قابل گسترش است. برای تغییر اندازه‌های آنها از دستور `\setlength` استفاده می‌کنیم. مانند `\setlength\LTright{2cm}`. دستور دیگر `\LTcapwidth` است که طول caption را تعیین می‌کند و پیش فرض آن `4in` است.

۲- دستورات شروع جدول

سه مشخصه `[c]`، `[r]` و `[l]` وضعیت جدول را در صفحه مشخص می‌کنند که بترتیب راست، وسط و چپ می‌باشند. اگر این دستور صادر نشود جدول در وضعیت پیش فرض حالت قبل قرار می‌گیرد.

۳- دستورات ابتدای سطر

این دستورات شامل `\pagebreak`، `\newpage` برای شکستن جدول در ابتدای سطر و `\nopagebreak` برای عکس آن می‌باشند.

۴- دستورات بین سطری

این دستورات شامل زیرنویسها است.

`\footnote{ text}` به نماد مورد نظر يك شماره داده و با همان شماره متن تحت خود را زیر نویس می‌کند.

`\footnotetext{ text}` متن تحت خود را زیر نویس می‌کند بدون هیچ شماره گذاری روی نماد.

`\footnotemark{ text}` متن تحت خود را مقابل نماد می‌نویسد.

۵- دستورات انتهای سطر

انتهای جدول را مشخص می‌کند	<code>\\</code>
بین دو سطر فاصله داده شده را ایجاد می‌کند.	<code>\\[width]</code>
مانع شکسته شدن جدول، بعد از سطر فوق می‌شود.	<code>*</code>
مشابه <code>\\</code> عمل می‌کند.	<code>\\tabularnewline</code>
سطر مذکور را حذف می‌کند.	<code>\\kill</code>
سطر مذکور را در ابتدای هر صفحه از جدول تکرار می‌کند.	<code>\\endhead</code>
سطر مذکور را در ابتدای صفحه اول جدول نمایش می‌دهد.	<code>\\endfirsthead</code>
سطر مذکور را در انتهای هر صفحه از جدول تکرار می‌کند.	<code>\\endfoot</code>
سطر مذکور را در انتهای صفحه آخر جدول نمایش می‌دهد.	<code>\\endlastfoot</code>

قبل از دستورات **عنوان** لازم است که دستور `\\listoftable` را معرفی کنیم. شما با بکار گیری این دستور در ابتدا یا انتهای نوشتار، فهرستی از جداول درج شده در نوشتار همراه عنوان، شماره جدول و صفحه آن، را بدست می‌آورید.

۶- دستورات عنوان یا `caption`

`\\caption{ text}` جدول دارای شماره و عنوان بوده و عنوان در فهرست جداول درج می‌شود.
`\\caption[lot]{ text}` جدول دارای شماره و عنوان بوده ولی متن `lot` در فهرست جداول درج می‌شود.

`\\caption[]{ text}` جدول دارای شماره و عنوان بوده ولی در فهرست جداول درج نمی‌شود.

جدول فقط دارای عنوان بوده و متنی در فهرست جداول درج نمی‌شود. `\\caption * { text}`

۲.۲ آرایه و ماتریس

آرایه‌ها مشابه جدولها میباشند، و برای مرتب کردن اعداد و نمادها بکار می‌روند، فقط با این تفاوت که نیاز به محیط فرمول دارد. شکل کلی آن بصورت زیر است.

`\begin{array}[Position]{Column-Option}`

`\square & \square & \square \quad \backslash\backslash`

`\square & \square & \square \quad \backslash\backslash`

`\square & \square & \square`

`\end{array}\backslash`

همانند جدول در Column–Option تنظیمات ستونها با $\{r,c,l\}$ مشخص می‌شود و وضعیت آن با $\{t,c,b\}$ به معنای بالا، وسط و پایین تنظیم می‌گردد. همچنین در انتهای هر سطر می‌توان از دستورات `\hline` و `\cline{a,b}` استفاده کرد. اگر از `\left{(}` و `\right{)}` در ابتدا و انتهای یک آرایه استفاده کنیم، یک ماتریس خواهیم داشت. به مثالهای زیر توجه کنید.

$$\begin{array}{ccc} & & 19 \quad 20 \quad 21 \\ 1 \quad 2 \quad 3 & = & \left(\begin{array}{ccc} 10 & 11 & 12 \\ 13 & 14 & 15 \\ 16 & 17 & 18 \end{array} \right) = \begin{array}{ccc} 22 & 23 & 24 \\ 25 & 26 & 27 \end{array} \\ 4 \quad 5 \quad 6 & & \\ 7 \quad 8 \quad 9 & & \end{array}$$

که تایپ شده آن به صورت زیر است.

`\begin{array}[t]{III}`

`1&2&3 \quad \backslash\backslash 4&5&6 \quad \backslash\backslash 7&8&9`

`\end{array}`

`=\left(\begin{array}[c]{III}`

`10&11&12 \quad \backslash\backslash 13&14&15 \quad \backslash\backslash 16&17&18`

`\end{array}\right)`

`= \begin{array}[b]{III}`

`19&20&21 \quad \backslash\backslash 22&23&24 \quad \backslash\backslash 25&26&27`

`\end{array}\backslash\backslash`

$$|x| = \begin{cases} x & x > 0 \\ 0 & x = 0 \\ -x & x < 0 \end{cases}$$

که تایپ شده آن بصورت زیر است.

```
\[|x| = \left \{\begin{array}{c} x & x > 0 \\ 0 & x = 0 \\ -x & x < 0 \end{array}\right .\]
```

$$\det \begin{bmatrix} b & 0 & \dots & 0 \\ a_2 & \boxed{D} \\ \vdots & \\ a_n & \end{bmatrix} = b(\det D)$$

که تایپ شده آن بصورت زیر است.

```
\newcommand*\tempa{\multicolumn{1}{c}{}}
\newcommand*\tempb{\multicolumn{1}{c}{}}
\[\det\left[ \begin{array}{cccc}
b & 0 & \ldots & 0 \\
a_2 & \tempa & \tempb \\
\vdots & \tempa & D & \tempb \\
a_n & \tempa & \tempb \end{array} \right] = b(\det D).\]
```

البته بجای " " می‌توان از هر نماد دیگر مانند "]" و "" نیز استفاده کرد. همچنان که در مثال بالا مشخص است می‌توان دستورات طولانی را خلاصه کرد. این امر توسط دستور:

```
\newcommand{\new-name}{command}
```

صورت می‌پذیرد. مثال `\newcommand{\sig}{\sigma}` موجب می‌شود که با تایپ `\sig` نماد σ را داشته باشیم. اما با بکار گیری دستور `\newcommand{\sig}{\sigma}` کافیت تایپ کنیم `\sig`. ولی با بکار گیری این دستور داخل یک فرمول دچار خطا می‌شویم، زیرا دو `\` را با فاصله داخل هم قرار داده‌ایم. برای رفع این مشکل از دستور `\ensuremath` استفاده می‌کنیم. ما در این متن از دستور زیر برای `\` استفاده کرده‌ایم.

```
\newcommand{\bk}{\ensuremath{\backslash}}
```

برای تایپ ماتریس مستقیماً می‌توان از دستور ماتریس استفاده کرد. این دستور مشابه array می‌باشد.

```
\[ \begin{matrix}
```

```
□ & □ & □ \\
```

```
□ & □ & □ \\
```

```
□ & □ & □
```

```
\end{matrix} \]
```

دستورهای دیگری برای ماتریس موجود است که اختصاصی می‌باشد مانند، `pmatrix`، `bmatrix`، `vmatrix` و `Vmatrix` که بترتیب بصورت زیر می‌باشند

$$\begin{pmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{pmatrix} \quad \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix} \quad \begin{vmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{vmatrix} \quad \left\| \begin{matrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{matrix} \right\|$$

در مواقعی لازم است که ماتریس را داخل خط یا in line بنویسیم. در این موارد دستور `smallmatrix` بسیار مفید است. اگر تایپ کنیم،

```
\left( \begin{smallmatrix} a+b+c & uv \\ a+b & c+d \end{smallmatrix} \right)
```

خواهیم داشت $\left(\begin{matrix} a+b+c & uv \\ a+b & c+d \end{matrix} \right)$. که یک ماتریس کوچک است.

۲.۳ شکل بندی قضایا

از دستورات بسیار مهمی که برای تایپ مقاله، گزارش و بویژه پایان نامه بکار می‌رود، `\newtheorem` است. شکل کلی آن بصورت زیر است.

```
\newtheorem{short-name}[first-counter]{full-name}[second-counter]
```

حال با تایپ `\begin{short-name} Text \end{short-name}` متن شما با يك شماره و با نام full-name شروع می‌شود. این دستور بیشتر برای نوشتن قضیه، لم، تبصره و تعریف بکار می‌رود. در شمارنده اول نام کوتاه یکی از موارد فوق را می‌نویسیم، مثلاً "در تعریف لم نام کوتاه قضیه را بکار می‌بریم با این کار شماره‌های لم و قضیه با هم مرتب می‌شوند. در شمارنده دوم قرار می‌دهیم `chapter`، `section`، `subsection` و یا `subsection` به این ترتیب شماره ارائه شده يك رقمی، دورقمی و یا سه رقمی خواهد شد. به مثال زیر دقت کنید.

```
\newtheorem{thm}{قضیه}[section]
```

```
\newtheorem{lem}{لم}
```

```
\newtheorem{prop}[lem]{نتیجه}
```

```
\newtheorem{defi}[thm]{تعریف}
```

در این مثال شماره‌های قضایا بر حسب بخش و شماره تعاریف با قضایا مرتب خواهد شد. همچنین شماره لم بدون توجه به بخش و شماره نتیجه با لم مرتب می‌شود. اگر بنویسیم:

```
\begin{thm} \begin{lem} \begin{prop} \begin{defi} \begin{thm} \begin{lem} \begin{defi}
```

```
\end{thm} \end{lem} \end{prop} \end{defi} \end{thm} \end{lem} \end{defi}
```

خواهیم داشت:

قضیه ۲.۱

لم ۱

نتیجه ۲

تعریف ۲.۲

قضیه ۲.۳

لم ۲

تعریف ۲.۴

شما می‌توانید از دستور `\theoremstyl{remark}` نیز برای تعریف مثلاً `Remak` استفاده کنید. لذا با تایپ `\begin{remark} Text \end{remak}` متن تبصره شما با نام `Remark` و با یک شماره تایپ می‌شود.

اگر از دستور `\newtheorem*` استفاده کنیم، شماره‌های مربوط به دستور حذف خواهند شد. اگر از برچسب در انتهای `\begin{short-name}\label{label-name}` استفاده کنیم آنگاه با استفاده از دستور `\ref{label-name}` می‌توانیم شماره مربوطه را درج کنیم بی‌آنکه شماره آن را بدانیم. این روش برای تایپ مقالات بسیار مفید است زیرا با جابجایی قضایا، شماره‌های آنها بطور خودکار در سراسر متن تعویض می‌شود و از اشتباهات و سختی کار در این زمینه می‌کاهد.

با بکار بردن دستور `\swapnumbers` قبل از `\newtheorem` می‌توان جای شماره‌ها و قضایا را جابجا کرد.

همین مطالب راجع به `Proof` نیز برقرار است. ولی اگر از پکیج `amsthm` استفاده کنیم اولاً "Proof" از قبل تعریف شده است ثانياً "بعد از شماره قضایا یک نقطه نیز درج می‌شود که در معیار کنونی مرسوم است.

۲.۴ دیاگرام‌های جابجائی

طبق معمول باید پکیج مربوطه را فراخواند سپس دستورات لازم را اجرا کرد. نام پکیج دیاگرام جابجایی `amscd` است. لذا `\usepackage{amscd}` را قبل از `\begin{document}` اضافه می‌کنیم. حال باید بصورت زیر عمل کنیم.

```
[\begin{CD}
```

```
——— \\\
```

```
——— \\\
```

```
\end{CD}]
```

توجه داریم که بجای `---` باید دیاگرام مربوطه را درج کنیم. جدول زیر نحوه درج فلشهای لازم را توضیح می‌دهد.

دستور	نتیجه	دستور	نتیجه	دستور	نتیجه
<code>@ >>></code>	\longrightarrow	<code>@ > Text >></code>	\xrightarrow{Text}	<code>@ >> Text ></code>	\xrightarrow{Text}
<code>@ <<<</code>	\longleftarrow	<code>@ < Text <<</code>	\xleftarrow{Text}	<code>@ << Text <</code>	\xleftarrow{Text}
<code>@ VVV</code>	\downarrow	<code>@ VText VV</code>	$Text \downarrow$	<code>@ VVText V</code>	$\downarrow Text$
<code>@ AAA</code>	\uparrow	<code>@ AText AA</code>	$Text \uparrow$	<code>@ AAText A</code>	$\uparrow Text$
<code>@ =</code>	$\longequal{\quad}$	<code>@ </code>	\parallel	<code>@ .</code>	جای خالی

به مثال زیر توجه فرمایید.

$$\begin{array}{ccccccc}
 A & \longleftarrow & B & \longequal{\quad} & E & \longrightarrow & F \\
 f \downarrow & & \uparrow g & & & & \parallel \\
 C & \longrightarrow & D & & G & \xleftarrow{\lambda} & H
 \end{array}$$

```
[\begin{CD}
```

```
A @<<< B @= E @>>> F \\\
```

```
@VfVV @AAgA @. @| \\\
```

```
C @>>> D @. G @<< \lambda < H
```

```
\end{CD}]
```

البته روشهای دیگری هم برای رسم دیاگرام موجود است، یکی از آنها استفاده از فلش و حروف بجای اعداد در يك ماتریس است. روش دیگری هم موجود است، که پیچیده ولی بسیار کار آمد است، بطوریکه بوسیله این روش حتی دیاگرام مثلثی را نیز می‌توان رسم کرد. برای رسم با این روش شما می‌بایست فایل `explain`، که يك فایل با پسوند `.tex` است را با دستور `\input{explain}` فراخوانید. چون این پکیج با بقیه دستورات سازگار نیست لذا ما فقط به طرز نوشتن يك دیاگرام در حدّ يك مثال اکتفاء می‌کنیم. برای مثال لطفاً " **کلر** کنید.

اگر بخواهیم يك دیاگرام بصورت:

$$A \xrightarrow{1-1} B \xleftarrow[\alpha \rightarrow \beta]{\text{onto}} C \xleftarrow[\gamma]{} D \leftarrow E$$

را رسم کنیم کافیت از دستورات `\xleftarrow` و `\xrightarrow` استفاده کنیم. شکل کلی این دستورات بصورت زیر است.

`\xleftarrow[متن پایین]{متن بالا}`

مثال بالا بصورت زیر تایپ شده است.

```
$ A \xrightarrow{\text{1-1}} B \xleftarrow[\text{onto}]{\alpha \to \beta} C
\xleftarrow[\gamma]{} D \xleftarrow{} E$
```

۲.۵ دیاگرام‌های ساده

ابتدا باید پکیج `graphics` را فراخوانیم. حال از دستور `figure` که یکی از دستورات رسم يك دیاگرام ساده است استفاده می‌کنیم. برای مثال به نمودار صفحه بعد توجه کنید. که بصورت زیر تایپ می‌شود.

```
\begin{figure}
```

```
  $$\def\fbbox#1{\hbox{\vrule
```

```
\vbox{\hrule\kern5pt\hbox{\kern5pt \hbox{#1} }\kern5pt\hrule}\vrule}}
```

شروع دیاگرام، در واقع کلّ این دیاگرام يك جعبه است که هر خط آن با `\cr` تمام

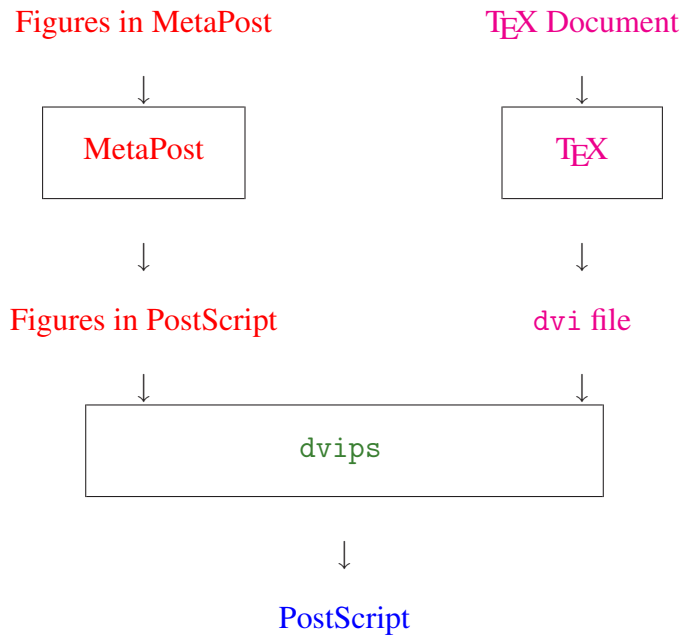


Fig. 1: A diagram of the processing for a TeX document with figures done in MetaPost

می‌شود و معنی آن این است که " برو پایین و در مرکز عبارت بالائی قرار بگیر "

```

\ vbox{ \ halign{ $ \ hfil # \ hfil $ & \ hskip 1in $ \ hfil # \ hfil $ \ cr
\ hbox{ Figures in MetaPost } & \ hbox{ \ TeX \ Document } \ cr
\ bigg \ downarrow & \ bigg \ downarrow \ cr
\ fbox{ \ vrule height.2in depth.133in width0pt \ kern .1in MetaPost \ kern.1in }
&
\ fbox{ \ vrule height.2in depth.133in width0pt \ kern.167in \ TeX \ kern.167in } \ cr
\ bigg \ downarrow & \ bigg \ downarrow \ cr
\ hbox{ Figures in PostScript } & \ hbox{ \ tt dvi file } \ cr
\ bigg \ downarrow & \ bigg \ downarrow \ cr
\ fbox{ \ vrule height.2in depth.133in width0pt \ kern 1in { \ tt dvips } \ kern 1in }
\ span \ omit \ cr
\ bigg \ downarrow \ span \ omit \ cr
\ hbox{ PostScript } \ span \ omit \ cr } }
\ atop \ vbox{ \ noindent Fig. 1: A diagram of the processing for a \ TeX \ document

```

with figures done in MetaPost}}\$

`\end{figure}`

در دستورات بالا ابتدا `\fbox` تعریف شده که می‌توان آن را حذف کرد و از `\boxed` استفاده کرد. دستور دیگری که محیط رسم یک دیاگرام را ایجاد می‌کند `picture` است، که امکانات بالائی را برای رسم خطوط مستقیم و یا میخنی فراهم می‌کند.

۲.۶ عنوان بندی

ابتدا "عنوان بندی ابتدای این راهنما را به عنوان مثال ارائه می‌کنیم.

```
\def\nam{\large حمید موسوی }
```

```
\def\ver{\large سیزدهمین سمینار جبر دانشگاه ارومیه }
```

```
\def\tit{\large کارگاه  $\mathcal{L}^T\mathcal{E}X 2_\epsilon$  }
```

```
\def\dat{\large تیر ۱۳۸۰ }
```

```
\def\maketitle{\noindent\nam\hfill\tit\par
```

```
\hrule height 1.0pt\medskip
```

```
\noindent\ver\hfill\dat \vskip0.3in}
```

حال کفایت که بعد از `\begin{document}` دستور `\maketitle` را اجرا کنید تا عنوان بندی شما ارائه شود. البته در مورد تاریخ باید خاطر نشان کرد که اگر از قبل تعریف نشود، `\maketitle` تاریخ روز را درج می‌کند. اگر می‌خواهید تاریخ حذف شود دستور `\date{}` را بکار گیرید. شما می‌توانید از دستورات `\author{name}` ، `\title{title-}` ، `\keywords{keywords}` ، `\email{email-add}` ، `\address{add-name}` ، `name` ، `\subjclass{subject}` و `\date{date}` استفاده کنید، و سپس دستور `\maketitle` را اجرا کنید. این دستور بطور پیش فرض یک عنوان بندی دارد که آن را بکار می‌گیرد. اگر این دستور را بعد از `\end{abstract}` بکار گیرید، در حالت `article` ، `\mathcal{L}^T\mathcal{E}X` یک صفحه را به عنوان و چکیده اختصاص می‌دهد. مگر همانند مثال بالا برای `maketitle` تعریف جدیدی ارائه کنید. این دستورات نیازمند پکیج `amsxtra` میباشند.

۲.۷ چگونه سربرگ درست کنیم

برای تهیه سربرگ پکیج زیر را فرامی‌خوانیم.

```
\usepackage{scrpage}
```

حال دو نوع دستور برای سربرگ موجود است که یکی بسیار ساده و دیگری کمی پیچیده است. مورد ساده‌تر با دستور زیر اجرا می‌شود.

```
\deftripstyle{pagename}[اندازه خطوط داخلی][اندازه خطوط خارجی]
{پایین - راست}{پایین - وسط}{پایین - چپ}{بالا - راست}{بالا - وسط}
```

این صفحه با این دستور سربرگ بندی شده است. اندازه خطوط را بر حسب pt تعیین می‌کنیم، معمولاً از 0.5pt الی 1pt مناسب‌ترین اندازه می‌باشد. البته لازم به ذکر است که این دستور مشخصات سربرگ را فقط تعیین و در خود (pagename) حفظ می‌کند و برای اجرای سربرگ باید دستور `\pagestyle{pagename}` را بکار برد. همانطور که مشخص است، اندازه خطوط دو به دو تغییر می‌کند و هرگز نمی‌توان ۳ خط در صفحه داشت. اگر بخواهیم تعداد خطوط را به دلخواه تعیین کنیم باید از دستور طولانی زیر استفاده کنیم.

```
renewpagestyle{headings}{(\textwidth,0.5pt)
```

```
{\hfill}
```

```
{\hfill}
```

```
{\hfill\hfill\pagemark}
```

```
(\textwidth,1.pt)}
```

```
{(\textwidth,0.5pt)
```

```
{\hfill}
```

```
{\hfill}
```

```
{راست \hfill \وسط \چپ}
```

```
(\textwidth,0.pt)}
```

این صفحه با دستور فوق سربرگ بندی شده است.

کلا" برای صفحه بندی و انتخاب اندازه‌های طول و عرض مقاله شما می‌توانید با فراخواندن `\usepackage{layout}` و اجرای دستور `\layout` دستورات لازم را مشاهده کنید:

حال دستورات زیر را اعمال می‌کنیم.

<code>\parskip</code>	<code>=1mm</code>	<code>\oddsidemargin</code>	<code>=1cm</code>
<code>\evensidemargin</code>	<code>=1cm</code>	<code>\textwidth</code>	<code>=14cm</code>
<code>\textheight</code>	<code>=23cm</code>	<code>\headheight</code>	<code>=25pt</code>
<code>\topmargin</code>	<code>=2cm</code>	<code>\footskip</code>	<code>=1.7cm</code>
<code>\topskip</code>	<code>=1cm</code>		

سپس با اجرا دوباره دستور `\layout` تغییرات ایجاد شده را مشاهده می‌کنیم.

۲.۸ چگونه يك مقاله را تایپ کنیم

ساختار يك مقاله بترتیب زیر است .

ساختار يك مقاله بترتیب زیر است .

`\documentclass[12pt,English]{article}` , `report` , `book` , `amsart` , `leter` or `ltxguide`

نوع مقاله و اندازه فونت را تعیین می‌کند .

`\usepackage[active]{srcltx}`

محیط فعال برای نوشتن نامه، همچنین اگر شما از ویراستار **WinEdt** استفاده می‌کنید این پکیج این امکان را برای شما فراهم می‌کند که با دو بار کلیک کردن روی متن در فایل **dvi** همان قسمت از متن در **WinEdt** باز شده و با نوار سیاه رنگی مشخص شود .

`\usepackage{amssymb,amsmath}`

برای ایجاد نمادهای ریاضی

`\usepackage{omega}`

برای فارسی نوشتن

`\usepackage{layout}`

برای صفحه بندی

`\usepackage{scrpage}`

برای ایجاد سربرگ

`\usepackage{amscd}`

برای ایجاد دیاگرام جابجایی

`\usepackage{amsxtra}`

برای ایجاد عنوان

`\usepackage{amsfonts}`

برای ایجاد فونت‌های فرکتور

`\usepackage[mathscr,mathcal]{eucal}`

برای ایجاد فونت‌های اسکریپت

`\usepackage{fontsmpl}`

برای ایجاد مثالهایی از فونت‌های

`\usepackage{longtable}`

برای ایجاد جداول بلند

`\usepackage{enumerate}`

برای ایجاد محیط متنی

`\usepackage{graphicx}`

برای فراخوانی تصویر

`\usepackage{graphics}`

برای فراخوانی تصویر و رسم نمودار

`\usepackage{color}`

برای ایجاد رنگ‌های اصلی

`\usepackage{colordvi}`

برای ایجاد رنگ‌های ترکیبی

`\usepackage{colortbl}`

برای ایجاد جداول رنگی

`\usepackage[hypertext,colorlinks,backref]{hyperref}`

برای ایجاد ارجاعات جهشی

<code>\usepackage{varioref}</code>	برای فعال کردن دستور <code>\vref{ }</code>
<code>\usepackage{url}</code>	برای فعال کردن آدرس‌های اینترنت
در این قسمت می‌توان دستورات جدید را تعریف کرد. بعنوان مثال:	
<code>\newcommand{\Cb}{\textcolor{blue}}</code>	
<code>\def\fcb[2]{\fcolorbox{#1}{#2}}</code>	
دستورات لازم برای عنوان بندی	
<code>\title{ } \author{ } \date{ }</code>	
<code>\begin{document}</code>	شروع مقاله
<code>\maketitle</code>	اجرای عنوان
از این قسمت اصل مقاله شروع می‌شود.	
<code>\begin{abstract}</code>	شروع چکیده
در این قسمت چکیده تایپ می‌شود.	
<code>\end{abstract}</code>	انتهای چکیده
در این قسمت اصل مقاله تایپ می‌شود.	
<code>\section{sec.-name}</code>	نام بخش را تایپ می‌کنید و شماره آن را خود دستور درج می‌کند.
<hr/>	
<hr/>	
<hr/>	
<code>\begin{thebibliography}</code>	شروع فهرست مراجع
<code>\bibitem{ }</code>	آدرس مرجع
<code>\bibitem{ }</code>	آدرس مرجع
<code>\bibitem{ }</code>	آدرس مرجع
<code>\end{thebibliography}</code>	انتهای فهرست مراجع
<code>\end{document}</code>	پایان مقاله

۲.۹ چگونه يك نامه را تایپ کنیم

برای تایپ يك نامه بصورت زیر عمل می‌کنیم.

```

\documentclass[12pt]{letter}

\usepackage[active]{srcltx} % SRC Specials: DVI [Inverse] Search

%-----

\signature{□}

\address{ % Return Address

□} \\

□} \\

{E-mail: }

}

%\makelabels

\begin{document}

%-----

\begin{letter}{% Addressee

□} \\

□} \\

□} \\

□} \\

\opening{Dear ,...}

متن نامه را بنویسید

\closing{Sincerely yours,}

\end{letter}

%-----

\end{document}

```

فصل ۳

امکانات ویژه در $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X} 2_{\epsilon}$

۳.۱ ارجاعات و ارجاعات جهشی یا `hyper refrens`

بهتر است قبل از پرداختن به موضوع به يك مثال توجه كنيم.
لطفاً **كنيد** كنيد.

شما می‌توانید به هر دستور شماره‌پذیر با دستور `\label{label-name}` يك شماره یا يك نام الحاق كنید و در مواقع نیاز با دستور `\ref{label-name}` به آن شماره ارجاع دهید. دستورات شماره‌پذیر دستوراتی هستند که $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ به آنها شماره‌ای اتلاق می‌کند. مانند `equation`, `align`, `caption`, `gather`, ... دستور ارجاع دیگر `\pageref{label-name}` است، که شماره صفحه هرآنچه را که به آن ارجاع شده باشد را نمایش می‌دهد. دستورات ارجاعی عبارتند از:

`\ref{label-name}` : که شماره مرجوع را نمایش می‌دهد. اگر ارجاع به کلمه‌ای یا دستوری باشد که شماره‌پذیر نیست، در اینصورت این دستور شماره اولین دستور شماره‌پذیر را نمایش خواهد داد.

`\eqref{label-name}` : مانند `ref` عمل می‌کند و با این تفاوت که شماره را داخل پرانتز قرار می‌دهد.

`\pageref{label-name}` : شماره صفحه دستور و یا کلمه ارجاع داده شده را نمایش می‌دهد.

`\vref{label-name}` : این دستور علاوه بر شماره مرجوع شماره صفحه آن را نیز نمایش می‌دهد. این دستور نیازمند فراخواندن پکیج `varioref` می‌باشد. از موارد دیگر ارجاعات، ارجاعات جهشی است که این امکان را برای شما بوجود می‌آورد که با کلیک کردن روی متن ارجاء شده به صفحه مرجوع مراجعه کنید. برای این امر باید پکیج زیر را فراخوانید.

`\usepackage{hyperref}`

این پکیج شامل عملگرهایی به شرح زیر است.

`\usepackage[hypertex, colorlinks, backref]{hyperref}`

اسامی داخل کروشه میتوانند بعضی از دستورات را فعال کنند. `colorlinks` موجب می‌شود که شماره و یا جملات ارجاعی به رنگ صورتی درآیند. موارد دیگر را در جای خود توضیح خواهیم داد. علاوه بر دستورات بالا دستورات زیر نیز در این حالت قابل اجرا خواهند بود. `\autoref{label-name}` : مانند `ref` عمل می‌کند، با این تفاوت که نام دستور شماره‌پذیر را همراه شماره آن می‌آورد.

`\hyperref` : این دستور برای ارجاع به قسمتی از یک فایل است، چه فایلی که در آن قرار دارد و یا فایل دیگر. همچنین می‌تواند یک نام و یا جمله خاصی را برای ارجاع به نمایش در آورد. شکل کلی آن بصورت زیر است.

`\hyperref{file-name}{Operator}{number}{disply-name}`

بجای Operation می‌توانید بنویسید `equation` و بجای `number` ، شماره آن مثلاً "1.1". اگر به فایل دیگری ارجاع داده نشود می‌توان نام فایل را حذف کرد. این دستور نیازمند به درج عملگر `hypertex` در فایل مرجوع می‌باشد. مانند: مقاله، بخش ۳ که تایپ شده آن بصورت فوق است.

`\href` : این دستور شکل خلاصه شده دستور قبل است. شکل کلی آن بصورت زیر است.

`\href{file:File-name#page.number, section.number or other}{disply-name}`

از دو دستور قبل برای آدرس `e-mail` یا `homepage` می‌توان استفاده کرد. ولی دستور زیر مخصوص این کار است.

`\url{Add.}` : این دستور برای استفاده از امکانات اینترنت است. شما می‌توانید آدرس e-mail یا اینترنت خود را جهت ارتباط، ارائه دهید. این دستور نیازمند پکیج `url` است.

`ftp://ftp.dante.de/tex-archive/system/win32/miktex`

`http://www.yahoo.com`

`s_hmousavi@yahoo.com`

برای دیدن چند مثال لطفاً " **کلیک** " کنید.

شما می‌توانید با دستور `\ref` و `\label` به شماره‌های مراجع در متن مقاله ارجاع دهید ولی نمی‌توانید به آنها جهش کنید.

اگر شما در مراجع خودتان بنویسید `\bibitem{a}\label{label-name}` و سپس در متن مقاله درج کنید `\ref{label-name}`، در این صورت شماره مرجع شما در متن درج خواهد شد. ولی با کلیک کردن روی شماره مرجع، به مرجع مورد نظر جهش نخواهید کرد. برای عمل جهش باید از دستور `\cite{label-name}` استفاده کنید. همچنین در قسمت مراجع هر مرجع را باید بصورت `\bibitem{label-name}` بنویسید.

حال برای اینکه در انتهای هر مرجع تعداد دفعات تکرار آن مرجع درج شود باید عملگر `backref` را در پکیج `hyperref` فعال کنید. البته مراجع را هم باید به فرم خاصی تایپ کنید. برای دیدن چند مثال لطفاً " **کلیک** " کنید.

۳.۲ چگونه يك تصویر را در متن فراخوانیم.

برای این منظور ابتدا `package` مربوطه را فرا می‌خوانیم.

```
\usepackage{graphicx}
```

سپس دستور زیر را اجرا می‌کنیم.

```
\includegraphics[width=xin, height=xin]{file-name.bmp}
```

در دستور بالا بجای `x` باید عدد درج شود. بعنوان مثال دستور زیر را در نظر بگیرید:

```
\includegraphics[width=3in, height=2in]{new.bmp}
```

اگر بخواهیم از تصاویر فشرده مانده `gif`, `pcx`, `jpg` و امثال آن استفاده کنیم باید دستور زیر را قبل از `\begin{document}` اضافه کنیم.

```
DeclareGraphicsExtensions{.pcx, .jpg, .gif}
```

```
DeclareGraphicsRule{.jpg}{.bmp}{}{}
```

و یا

دستور دیگر `figure` می باشد که که توسط پکیج `graphics` ایجاد می شود و فرم کلی آن به صورت زیر است.

```
\begin{figure}[وضعیت ]
```

```
\includegraphics{file-name}
```

```
\caption{title}\label{label-name}
```

```
\end{figure}
```

وضعیت شامل سه حالت `t`، `c` و `b` می باشد که بترتیب بالا، وسط و پایین را مشخص می کنند. البته پیش فرض آن `c` است.

این دستور به تصویر يك شماره اختصاص می دهد که با دستور `\listoffigures` می توان فهرستی از آنها تهیه کرد. اگر از `figure*` استفاده کنیم شماره حذف خواهد شد و در فهرست نیز درج نخواهد شد. ولی می توان با دستورات:

```
\addcontentsline{lof}{sectionnig}{line-to-add}
```

```
\addtocontents{lof}{line-to-add}
```

و

متن و یا شماره مورد نظر را به فهرست اضافه کرد.

برای کوچک کردن تصویر از دستور `\scalebox` استفاده می‌کنیم، بدین ترتیب که برای ۸۰٪ کردن تصویر، خط مربوطه را چنین می‌نویسیم.

```
\scalebox{.80}{\includegraphics{file-name}}
```

البته این دستور را برای هر Box دظواهی می‌توان بکار برد.

۳.۳ چگونه رنگی بنویسیم.

برای این منظور باید `package` مربوطه یعنی `color` را فراخواند.

```
\usepackage{color}
```

سپس رنگ مورد نظر را با دستور زیر بر متن مورد نظر اعمال می‌کنیم.

```
\textcolor{color-name}{text}
```

به مثالهای زیر دقت فرمایید.

<code>\textcolor{blue}{Text}</code>	Text
<code>\textcolor{green}{Text}</code>	Text
<code>\textcolor{red}{Text}</code>	Text
<code>\textcolor{magenta}{Text}</code>	Text
<code>\textcolor{yellow}{Text}</code>	Text
<code>\textcolor[gray]{0.7}{Text}</code>	Text

رنگهای بالا همراه دو رنگ سیاه و سفید رنگهای از پیش تعریف شده هستند. رنگ خاکستری یک رنگ ترکیبی است و مقدار عددی آن بین ۰ و ۱ متغیر است. اگر مقدار عددی آن را ۱ قرار دهیم به رنگ سیاه و اگر مقدار عددی آن را ۰ قرار دهیم به رنگ سفید تبدیل می‌شود.

دستورات دیگری `color` است که برای رنگی نوشتن متنهای طولانی بسیار مفید است. شکل کلی آن بصورت `{\color{color-name}... Text ...}` است.

بطور کلی رنگها بر اساس دو مدل یا سیستم رنگی بنا می‌شوند. یکی سیستم **rgb**، یعنی **Red**، **Green** و **Blue**. در این سیستم می‌توان با قرار دادن ۳ مقدار عددی بین ۰ و ۱ این رنگها را با هم تلفیق کرد و رنگهای دیگر را ایجاد نمود. شما می‌توانید بجای نام رنگ از دستور $[rgb]\{n_1, n_2, n_3\}$ که در آن n_1, n_2, n_3 سه عدد بین ۰ و ۱ هستند استفاده نمود.

```
\color[rgb]{1, 0, 0}{Text}    Text
\color[rgb]{0, 1, 0}{Text}    Text
\color[rgb]{0, 0, 1}{Text}    Text
\color[rgb]{0.58, 0.20, 0.70}{Text}    Text
```

سیستم دیگر **cmymk** است که تلفیق رنگهای **Cyan**، **Magenta**، **Yellow** و **Black** را موجب می‌شود. کاربرد این سیستم مشابه سیستم قبلی است با این تفاوت که از چهار مقدار عددی بین ۰ و ۱ در آن استفاده می‌شود.

```
\color[cmymk]{0.4, 0.8, 0.6, 0}{Text}    Text
\textcolor[cmymk]{0.6, 0.2, 0.6, 0}{Text}    Text
\textcolor[cmymk]{1, 0.5, 0, 0.2}{Text}    Text
\textcolor[cmymk]{0, 0.75, 0.6, 0.5}{Text}    Text
```

می‌توانیم رنگ مورد نظر را از قبل تعریف کنیم و در مواقع لازم از آن استفاده کنیم. این کار توسط دستور **definecolor** صورت می‌پذیرد و شکل کلی آن بصورت زیر است.

```
\definecolor{مقادیر تلفیق}{سیستم}{نام رنگ}
```

```
\definecolor{light-blue}{rgb}{0.8, 0.85, 1}
```

حال در مواقع نیاز می‌توان با دستور زیر رنگ مورد نظر را فراخواند.

```
\color{light-blue}{Text}    Text
```

بعنوان مثال دیگر $\definecolor\{myGray\}\{gray\}\{0.75\}$ می‌تواند رنگ خاکستری مورد نظر ما را با نام **myGray** ایجاد کند.

ما می‌توانیم از رنگهای ترکیبی استاندارد از پیش تعریف شده نیز استفاده کنیم، در واقع

۶۸ رنگ بندی مختلف برای این کار موجود است. این رنگ بندیها، با فراخواندن **package**

زیر قابل استفاده خواهند بود. $\usepackage\{colordvi\}$

این ۶۸ رنگ در جدول زیر فهرست شده‌اند.

مشخصات	نام رنگ	شماره
GreenYellow Approximate PANTONE 388	GreenYellow	۱
Yellow Approximate PANTONE YELLOW	Yellow	۲
Goldenrod Approximate PANTONE 109	Goldenrod	۳
Dandelion Approximate PANTONE 123	Dandelion	۴
Apricot Approximate PANTONE 1565	Apricot	۵
Peach Approximate PANTONE 164	Peach	۶
Melon Approximate PANTONE 177	Melon	۷
YellowOrange Approximate PANTONE 130	YellowGreen	۸
Orange Approximate PANTONE ORANGE-021	Orange	۹
BurntOrange Approximate PANTONE 388	BurntOrange	۱۰
Bittersweet Approximate PANTONE 167	Bittersweet	۱۱
RedOrange Approximate PANTONE 179	RedOrange	۱۲
Mahogany Approximate PANTONE 484	Mahogany	۱۳
Maroon Approximate PANTONE 201	Maroon	۱۴
BrickRed Approximate PANTONE 1805	BrickRed	۱۵
Red VERY-Approx PANTONE RED	Red	۱۶
OrangeRed No PANTONE match	OrangeRed	۱۷
RubineRed Approximate PANTONE RUBINE-RED	RubineRed	۱۸
WildStrawberry Approximate PANTONE 206	WildStrawberry	۱۹
Salmon Approximate PANTONE 183	Salmon	۲۰
CarnationPink Approximate PANTONE 218	CarnationPink	۲۱
Magenta Approximate PANTONE PROCESS-MAGENTA	Magenta	۲۲
VioletRed Approximate PANTONE 219	VioletRed	۲۳
Rhodamine Approximate PANTONE RHODAMINE-RED	Rhodamine	۲۴

مشخصات	نام رنگ	شماره
Mulberry Approximate PANTONE 241	Mulberry	٢٥
RedViolet Approximate PANTONE 234	RedViolet	٢٦
Fuchsia Approximate PANTONE 248	Fuchsia	٢٧
Lavender Approximate PANTONE 223	Lavender	٢٨
Thistle Approximate PANTONE 245	Thistle	٢٩
Orchid Approximate PANTONE 252	Orchid	٣٠
DarkOrchid No PANTONE match	DarkOrchid	٣١
Purple Approximate PANTONE PURPLE	Purple	٣٢
Plum VERY-Approx PANTONE 518	Plum	٣٣
Violet Approximate PANTONE VIOLET	Violet	٣٤
RoyalPurple Approximate PANTONE 267	RoyalPurple	٣٥
BlueViolet Approximate PANTONE 2755	BlueViolet	٣٦
Periwinkle Approximate PANTONE 2715	Periwinkle	٣٧
CadetBlue Approximate PANTONE (534+535)/2	CadetBlue	٣٨
CornflowerBlue Approximate PANTONE 292	CornflowerBlue	٣٩
MidnightBlue Approximate PANTONE 302	MidnightBlue	٤٠
NavyBlue Approximate PANTONE 293	NavyBlue	٤١
RoyalBlue No PANTONE match	RoyalBlue	٤٢
Blue Approximate PANTONE BLUE-072	Blue	٤٣
Cerulean Approximate PANTONE 3005	Cerulean	٤٤
Cyan Approximate PANTONE PROCESS-CYAN	Cyan	٤٥
ProcessBlue Approximate PANTONE PROCESS-BLUE	ProcessBlue	٤٦
SkyBlue Approximate PANTONE 2985	SkyBlue	٤٧
Turquoise Approximate PANTONE (312+313)/2	Turquoise	٤٨
TealBlue Approximate PANTONE 3145	TealBlue	٤٩

مشخصات	نام رنگ	شماره
Aquamarine Approximate PANTONE 3135	Aquamarine	۵۰
BlueGreen Approximate PANTONE 320	BlueGreen	۵۱
Emerald No PANTONE match	Emerald	۵۲
JungleGreen Approximate PANTONE 328	JungleGreen	۵۳
SeaGreen Approximate PANTONE 3268	SeaGreen	۵۴
Green VERY-Approx PANTONE GREEN	Green	۵۵
ForestGreen Approximate PANTONE 349	ForestGreen	۵۶
PineGreen Approximate PANTONE 323	PineGreen	۵۷
LimeGreen No PANTONE match	LimeGreen	۵۸
YellowGreen Approximate PANTONE 375	YellowGreen	۵۹
SpringGreen Approximate PANTONE 381	SpringGreen	۶۰
OliveGreen Approximate PANTONE 582	OliveGreen	۶۱
RawSienna Approximate PANTONE 154	RawSienna	۶۲
Sepia Approximate PANTONE 161	Sepia	۶۳
Brown Approximate PANTONE 1615	Brown	۶۴
Tan No PANTONE match	Tan	۶۵
Gray Approximate PANTONE COOL-GRAY-8	Gray	۶۶
Black Approximate PANTONE PROCESS-BLACK	Black	۶۷
White No PANTONE match	White	۶۸

مشخصه نام این رنگها بزرگ بودن حرف اول آنها است. برای استفاده از رنگهای بالا باید بصورت زیر عمل می‌کنیم.

`\Color-name{ Text}`

به مثال زیر توجه فرمایید.

`\OliveGreen{Sample Text} → Sample Text`

همچنین می‌توان رنگ زمینهٔ يك متن را نیز عوض کرد. کافیسست دستور `colorbox` را بکار گیریم.

```
\colorbox{black}{\colortext{white}{Text}} Text
```

به موارد زیر دقت کنید:

```
\colorbox{green}{\colortext{red}{Text}} Text
```

```
\colorbox{blue}{\colortext{magenta}{Text}} Text
```

```
\colorbox{red}{\colortext{blue}{Text}} Text
```

```
\colorbox{yellow}{\colortext{black}{Text}} Text
```

```
\colorbox{magenta}{\colortext{blue}{Text}} Text
```

بعنوان مثالی دیگر يك پاراگراف را رنگ می‌کنیم.

```
colorbox{yellow}{\parbox[c][1.5in][c]{5.5in}{Text}}
```

می‌توان به دُور `box` مورد نظر کادر کشید. این امر توسط دستور زیر صورت می‌پذیرد.

```
\fcolorbox{frame-color}{box-color}{Sample Text}
```

بعنوان مثال: `Sample Text`

می‌توان متن داخل يك جدول را نیز رنگی نوشت، بعنوان مثال جدول زیر:

One	Two	Tree
Text 1-1	Text 1-2	Text 1-3
Text 2-1	Text 2-2	Text 2-3

همچنین شما می‌توانید زمینهٔ يك ستون از جدولی را رنگ آمیزی کنید، برای این کار ابتدا باید `package` زیر را فراخوانیم.

```
\usepackage{colortbl}
```

حال برای راحتی، ستون رنگی را پیشاپیش تعریف می‌کنیم:

```
\newcolumnntype{G}{>{\columncolor[gray]{0.8}[5pt][7pt]}c}
```

حال در دستور جدول سطر اول را مطابق زیر می‌نویسیم.

```
begin{longtable}{||G| r |G ||}\hline \\\
```

جدول رنگهای بالا مثالی از این دستور است که دو ستون کناری آن رنگ آمیزی شده‌اند. اگر شما فایل `colordvi.sty` را در اختیار نداشته باشید می‌توانید با اضافه کردن دستورات زیر قبل از `\begin{document}` از رنگهای ترکیبی ارائه شده در جدول بالا استفاده کنید.

```
\def\background#1{\special{background #1}}
```

```
\def\subdef#1{\gdef\globalColor##1{\special{color push #1}##1\special{color pop}}}
```

```
\def\textColor#1{\subdef{cmyk #1}\special{color cmyk #1}}
```

```
\def\Color#1#2{\special{color push cmyk #1}#2\special{color pop}}
```

```
\def\newColor #1 {\expandafter\def\csname #1\endcsname##1
```

```
{\special{color push #1}##1\special{color pop}}
```

```
\expandafter\def\csname text#1\endcsname{\subdef{#1} \special{color #1}}}%
```

```
\subdef{Black}
```

۳.۴ چگونه فارسی بنویسیم

ابتدا باید پکیج آن یعنی `omega` را فراخوانیم. سپس بصورت زیر عمل کنیم.

```
\begin{arab} man Ik mtN fArsI hstm.\hfill\end{arab}
```

که پس از اجراء : `مَن يک متن فارسی هستم.`

البته `\hfill` برای این منظور بکار می‌رود که هر متن داخل `begin` و `end` بصورت وسط چین است، و با دستور فوق بصورت راست چین می‌شود.

البته باید دقت داشت که يك متن فارسی را نباید `LaTeX` کرد و برای ساختن `dvi` ابتدا با `MyComputer` به محلی که فایل خود را `Save` کردیم، می‌رویم، سپس دستور زیر را اجرا می‌کنیم.

```
Start > Run > lambda filename.tex
```

حال يك `file` با پسوند `.dvi` ساخته می‌شود، که با `کلیک` کردن روی آن متن فارسی شما قابل دیدن است.

برای مشاهده حروف معادل فارسی لطفاً "`کلیک`" کنید.

روش دیگری هم موجود است که این مشقّت ایجاد `begin` و `end` را ندارد. برای این منظور دو سطر زیر را قبل از شروع متن یا قبل از `begin{document}` اضافه می‌کنیم.

```
\fontfamily{omarb}\selectfont \pushocplist\ArabicOCP
```

```
\def\rmdefault{omarb}
```

حال هر متن نوشته شده بر اساس حروف معادل، به قلم فارسی خواهد بود. ولی مسأله اصلی این است که صفحه شما هنوز فارسی نشده، یعنی هنوز متن شما از چپ به راست نوشته می‌شود. برای رفع این مشکل دستورات زیر را اضافه می‌کنیم.

```
\pagedir TRT \bodydir TRT \pardir TRT \textdir TRT
```

حال مسأله باقی مانده نوشتن لاتین است که برای این امر سطر زیر را اضافه می‌کنیم. در واقع يك دستور ایجاد محیط لاتین را تعریف می‌کنیم.

```
\def\latin#1{\fontfamily{omlgc}\selectfont\pushocplist\BasicLatinOCP\textdir TLT#1\popocplist}
```

حال با تایپ `\latin{ABC}` خواهیم داشت `ABC`.

۳.۵ امکانات PostScript

برای استفاده از این امکانات باید فایل `dvi` را با دستور `dvips` در پای `Run` به يك فایل `ps` تبدیل کنیم. البته برای فایل‌های فارسی از دستور `odvips` استفاده می‌کنیم. یکی از امکاناتی که در `ps` هست ولی در `dvi` نیست، رنگ کردن کل صفحه است. این کار توسط دو دستور `\background{color-name}` که از پکیج `colordvi` حاصل می‌شود، و دستور `\pagecolor{color-name}` که از پکیج `graphics` حاصل می‌شود، صورت می‌پذیرد. لازم به ذکر است که باید عملگر `[dvips]` را در پکیج‌های `color`، `graphics` و `colordvi` فعال کنید. حال با اجرای دستور `\background{Green}` در فایل `ps` تمام صفحه شما به رنگ سبز در خواهد آمد.

یکی از استفاده‌های مهم `post script` رسم نمودار است. این کار کمی طولانی است وی بسیار جالب و کار آمد است. برای رسم نمودار در این سبک ابتدا دستورات لازم برای رسم را در فایلی با انشعاب `mp` مثلاً `fig.mp` می‌نویسیم. سپس با `My Computer` به محل فایل مورد نظر رفته و در پای `Run` دستور `mpost fig.mp` را اجرا می‌کنیم. این دستور يك فایل به نام `fig.1` می‌سازد. حال در فایل `tex` دستور `$$\epsfbox{fig.1}$$` را تایپ می‌کنیم. بعد از `latex` و `dvips` کردن فایل، نمودار خواسته شده در فایل `ps` ترسیم می‌شود.

در `mp` -فایل يك نمودار را چگونه تایپ کنیم؟
برای مثال به مورد زیر دقت کنید.

```
beginfig(1);
draw (20,20)-(0,0)-(0,30)-(30,0)-(0,0);
endfig;
end
```

در اینجا با مثالهایی چند این دستورات را به نمایش در می‌آوریم. خواننده عزیز با دقت در این مثالها می‌تواند به نحوه کار کردن این دستورات پی ببرد. لطفاً **کلیک** کنید. شما می‌توانید ابتدا تمام نمودارهای مورد نیاز خود را یکجا در يك فایل تایپ کنید. هر کدام را بین `beginfig` و `endfig` اما با شماره‌های متفاوت و `يك end` . سپس با اجرای

mpost فایل‌هایی با شماره‌های متفاوت ایجاد می‌شوند که شامل داده‌های نمودارهای مورد نظر هستند. حال هر کدام را که مورد نیاز باشد با دستور `\epsfbox` فرامی‌خوانید. البته برای اجرای این دستورات نیازمند فرا خواندن پکیج `epsf` که يك فایل `tex` است، می‌باشید. این کار توسط دستور `\input{epsf}` صورت می‌پذیرد.

پایان