

LaTeX Dersleri

Zafer Acar

2022-01-21

İçindekiler

1 Genel	1
1.1 LaTeX Nedir?	1
1.2 Önemli Yapılar	2
1.2.1 Komutlar	2
1.2.2 Paketler	2
1.2.3 Ortamlar	3
1.2.4 Gruplar	3
1.2.5 Boşluklar	3
1.2.6 Özel amaçlı karakterler	4
1.3 Kurulum	4
1.3.1 GNU/Linux	5
1.3.2 Mac OS	5
1.3.3 Windows	5
1.3.4 LaTeX Editörleri	6
1.3.5 Çevrimiçi Editörler	6
1.4 Tipik Bir Belge Yazımı	6
1.5 Belge Sınıfları ve Seçenekleri	7
1.6 Türkçe Dil Ayarları ve Çoklu Dil Kullanımı	9
1.7 Heceleme	10
1.8 Belgeye Başlık Oluşturma	10
1.9 Bölümleme ve İçindekiler Tablosu	11
1.10 Kitap Projesi Başlatma	14
1.11 Varsayılan Sayfa Düzenini Değiştirme	16
1.12 Satır ve Sayfa Kesme	17

Şekil Listesi

1.1	Kaynak dosyanın olduđu dizinin düzenlenmesi	15
-----	---	----

Tablo Listesi

Önsöz

1 Genel

Bu bölümde LaTeX kullanımıyla ilgili genel bilgilerden bahsedeceğiz.

1.1 LaTeX Nedir?

Önce TeX’le başlayalım. TeX, 1978’den itibaren Donald Knuth¹ tarafından belgelerin bilgisayarda dizilmesi için geliştirdiği bir dizgi sistemidir. LaTeX ise TeX’in kullanımını kolaylaştırmak için 1984 yılında Leslie Lamport² tarafından tasarlanmış bir makro pakettir.

LaTeX, genelde WYSIWYG³ editörleriyle karşılaştırılır. WYSIWYG, Microsoft Word, Libreoffice Writer gibi kelime işlemcilere ya da Adobe Indesign gibi programlara verilen genel bir isimdir. Hepsinin ortak özelliği, girdi ile çıktının aynı anda ve birlikte görünmesidir.

Bir metnin genel görünümü ve okunabilirliği, metnin nasıl hizalandığından ve kesildiğinden büyük ölçüde etkilenir. LaTeX, tüm paragraf için hizalamayı ve kesmeleri optimize eden son derece gelişmiş TeX algoritmalarını kullanır. Kelime işlemciler ve diğer programlar, satır başına çalıştıkları için oldukça yetersiz kalırlar. Bu, diğer şeylerin yanı sıra düzensiz aralıklara ve birçok kısa çizgiye sebep olur. Sonuçları görmemiz için Microsoft Word 2008 (Mac), Adobe InDesign CS4 ve LaTeX’le dizilmiş bir metni şuradan⁴ inceleyebilirsiniz.

Sonuç, LaTeX’in diğer programların her ikisinden de üstün olduğunu açıkça gösterir: iki kat daha az tireleme kullanır ve yine de sözcük aralığındaki varyasyon, Word veya InDesign’dan belirgin şekilde daha azdır. LaTeX’te çok büyük sözcük aralığı içeren satırlar oluşmaz.

LaTeX’de girdi ve çıktı ekranı farklıdır ve çıktıyı görmek için girdinin derleme işleminden geçmesi gerekir. Ayrıca birçok şey için WYSIWYG editörlerinde olmayan yapılar vardır. Şimdi, bu yapıların ne oldukları ve ne işe yaradıklarını açıklayalım.

¹<https://www-cs-faculty.stanford.edu/~knuth/>

²<http://www.lamport.org/>

³WYSIWYG, İngilizce’de “What You See Is What You Get” teriminin baş harflerinden oluşan bir bilgisayar terimidir. Türkçesi *Ne Görüyorsan Onu Alırsın* demek olup ekranda görülene çok benzer bir çıktı alınacağı ortamları tanımlar.

⁴<http://www.rtznet.nl/zink/comparison.pdf>

1.2 Önemli Yapılar

1.2.1 Komutlar

LaTeX komutları bir geribölü (\backslash) işaretiyle başlar ve ya sadece harflerden ya da bir tane harf olmayan karakterden oluşurlar. Komut yazıldıktan sonra ya boşluk, ya bir sayı ya da harf olmayan bir karakter gelebilir.

Çoğu komut, zorunlu değişken alır. Bu zorunlu değişken komut adından sonra çengel parantezler içine yazılır. Zorunlu değişken alan komutlar, zorunlu olmayan (isteğe bağlı) değişkenler de alabilir, bunlar da komut adından sonra gelen köşeli parantezler içine yazılırlar. Eğer değişkenler birden fazlaysa aralarına virgül koyularak ayrılır.

```
1 \:  
2 \LaTeX  
3 \item[...]  
4 \emph{...}  
5 \documentclass[...]{...}  
6 \subfloat[...][...]{...}  
7 \raisebox{...}[...][...]{...}  
8 \multicolumn{...}{...}{...}  
9 {\bfseries ...}
```

Fikir vermesi açısından yukarıda dokuz adet komut örneği verilmiştir. Birinci komut bir tane harf olmayan karakterden oluşan bir komuttur. İkincisi, değişkeni olmayan bir komuttur. Bazı harflerin büyük bazılarının da küçük olması komutların büyük-küçük harfe duyarlı olduğunu gösterir. Dokuzuncu komut ise bildirim şeklinde verilmiştir.

1.2.2 Paketler

LaTeX’de bazı özelliklerin (renkli yazmak, şekil eklemek vb.) kullanılabilmesi için kaynak dosyaya bazı paketlerin eklenmesi gerekir. Bu, `\usepackage` komutuyla yapılır. Bu komutun zorunlu değişkenine paket adı, zorunlu olmayan kısmına ise paket seçenekleri yazılır:

```
\usepackage[<seçenekler>]{<paket adı>}
```

Bu komutla paketin kaynak dosyaya eklenmesi TeX dağıtımıyla sisteminize kurulmuş olan paketin belgeye çağrılarak işe koşulması demektir.

1.2.3 Ortamlar

LaTeX'de ortamlar önemli bir yer tutar. Örneğin `document` bir ortamdır. Ortamları birden fazla öğeye uygulanan komutlar olarak düşünebiliriz.

Bir ortam `\begin` komutuyla başlayıp `\end` komutuyla biter. Her iki komutun zorunlu değişkeni ortamın adıdır:

```
\begin{<ortam adı>}
...
\end{<ortam adı>}
```

1.2.4 Gruplar

Gruplar, ortam benzeri yapılardır. Grup `\begingroup` komutuyla başlar ve `\endgroup` komutuyla biter. Grubun içinde kullanılan bir bildirim sadece gruba uygulanır.

1.2.5 Boşluklar

LaTeX'de belgenizin metnini oluştururken ister klavyedeki Space, ister Tab tuşu ile boşluk bırakın, bu boşluklar LaTeX tarafından bir karakter boşluk olarak algılanır. Arka arkaya çok sayıda boşluk bırakılsa da LaTeX bunu tek bir boşluk olarak algılar.

Bütün bir satırın boş bırakılması LaTeX tarafından paragraf başı olarak algılanır. Arka arkaya boş bırakılan çok sayıda boş satır LaTeX tarafından tek bir boş satır yani paragraf başı olarak algılanır.

İster bir boşluk, isterseniz de çok sayıda boşluk bırakın.
İkisi de bir boşluk gibi işlem görür.

Boş bir satır yeni paragraf demektir, burada olduğu gibi.

Çıktı⁵

Komutlardan sonra gelen boşlukları LaTeX dikkate almaz. Komuttan sonra gerçekten bir boşluk bırakmak için, ya `{ }` ve ardından boşluk girilir ya da komut adından sonra özel bir boşluk komutu kullanılır.

```
\LaTeX boşluk yok.\\
\LaTeX{} boşluk var.\\
\LaTeX\ boşluk komutuyla boşluk.
```

⁵<https://github.com/acarzfr/latex-dersleri/blob/main/examples/ex1.pdf>

1.2.6 Özel amaçlı karakterler

Aşağıdaki karakterlerin herbiri LaTeX'de özel bir amaç için kullanılır. Dolayısıyla bu karakterleri doğrudan kullanmak istenmeyen sonuçlara yol açabilir.

\$ % & { } ~ ^ _ \

Bu karakterleri çıktıda elde etmek isterseniz, sondaki hariç, başına bir geribölü koymanız gerekir. Sondaki için, yani bir geribölü sembolü elde etmek içinse `\textbackslashash` komutunu kullanabilirsiniz. Eğer `\\` komutunu vererseniz yeni bir satır başlatmış olursunuz.

Bu karakterlerden örneğin yüzde (%) karakteri kaynak dosyanızda yorum ya da açıklama yazmaya yarar. Bu sembolden sonra yazılanları LaTeX dikkate almaz ve çıktıda görünmez.

Diğer karakterlerden örneğin (\$) nin matematik kipini açma ve kapatmaya yarar. (&) karakteri tablo ve benzeri yapılarda dikey hizalama yapmak için veya sütun ayracı olarak kullanılır. Çengelli parantezlerden zaten yeterince bahsettik. (#) karakteri yeni komutlar tanımlamakta kullanılır. Tilda (~) ise genişlemeyen bir boşluk yaratmak için kullanılır. (^) ve (_) karakterleri de matematikte üst ve alt indis yazmak için kullanılır. Her birinin kullanımlarından yeri geldiğinde tekrar bahsedeceğiz.

1.3 Kurulum

LaTeX'i kurmak için ilk olarak bir TeX dağıtımı edinmeniz gerekir. Dağıtımlar, dizgi sistemini ve LaTeX'de belge oluşturabilmek için gereken paketleri içerir.

İkinci ihtiyaç duyacağınız şey bir LaTeX editörüdür. Edindiğiniz TeX dağıtımları genelde bir LaTeX editörüyle birlikte gelir. Tabi editör kişisel bir tercihtir ve bir LaTeX editörü yerine basit bir metin editörü kullanabilirsiniz. Ancak farklı işletim sistemleri için birçok iyi LaTeX editörü vardır ve bunların kod vurgulama, otomatik tamamlama, otomatik belge oluşturma gibi LaTeX'e özgü işlevleri vardır. Dolayısıyla LaTeX'de yeniseniz bir editör kullanmanızı tavsiye ederiz.

⁶<https://github.com/acarzfr/latex-dersleri/blob/main/examples/ex2.pdf>

1.3.1 GNU/Linux

Linux sistemlere MiKTeX⁷ ya da TeX Live⁸ kurulabilir. MiKTeX'in indirme sayfasında Ubuntu, Mint, Debian, Fedora, CentOS ve openSUSE gibi Linux dağıtımlarında nasıl kurulacağı anlatılmıştır. TeX Live ise tüm popüler Linux dağıtımlarının depolarında mevcut olup, paket yöneticisi ya da komut satırı yardımıyla kurulabilir. Örneğin Ubuntu, Debian, Mint, Pardus gibi .deb uzantılı paketlerin kullanıldığı dağıtımlarda

```
sudo apt-get install texlive-base
```

komutuyla temel kurulum,

```
sudo apt-get install texlive-full
```

komutuyla da tam kurulum yapılır.

1.3.2 Mac OS

Mac OS kullanıcıları için iki seçenek mevcuttur: MiKTeX⁹ ya da MacTeX¹⁰. MiKTeX kurulumu için .dmg uzantılı, MacTeX içinse .pkg uzantılı dosya indirilir ve standart kurulum yapılır.

1.3.3 Windows

Windows için aşağıdaki dağıtımlardan birini kurabilirsiniz.

- MiKTeX¹¹
- TeX Live¹²
- proTeXt¹³

MiKTeX veya TeX Live dağıtımını kurarsanız sisteminize TeXworks¹⁴ editörü de kurulur. proTeXt dağıtımını MiKTeX tabanlı bir dağıtım olup, tüm paketleri içerir ve beraberinde TeXstudio¹⁵ editörüyle gelir.

⁷<https://miktex.org/download>

⁸<http://www.tug.org/texlive/>

⁹<https://miktex.org/download>

¹⁰<http://www.tug.org/mactex/>

¹¹<https://miktex.org/download>

¹²<http://www.tug.org/texlive/>

¹³<https://tug.org/protext/>

¹⁴<https://www.tug.org/texworks/>

¹⁵<https://texstudio.org/>

1 Genel

1.3.4 LaTeX Editörleri

Hangi editörü kullanacağınıza birkaç deneme yaptıktan sonra karar verebilirsiniz. Burada¹⁶ en çok beğenilen editörler listelenmiş.

Her LaTeX editöründe olan özelliklerin (otomatik kod tamamlama vb.) yanı sıra kullanıcı dostu arayüzü, yüzde yüze yakın Türkçe desteği, ücretsiz oluşu ve her üç sistemde de çalışabilmesinden dolayı TeXstudio¹⁷’yu tavsiye ediyoruz. Karar sizin.

1.3.5 Çevrimiçi Editörler

LaTeX’i hiçbir kurulum yapmadan çevrimiçi de kullanabilirsiniz. Aşağıda üç tanesi listelenmiştir.

- Overleaf¹⁸
- Papeeria¹⁹
- LaTeX Base²⁰

En popüler olanı Overleaf olup, sayfasında beğenebileceğiniz binlerce şablon²¹ ve LaTeX kullanımına yönelik anlatımlar²² bulunur.

1.4 Tipik Bir Belge Yazımı

LaTeX’in varsayılan dosya uzantısı `.tex`’tir. Bu basit bir metin dosyası olup, LaTeX editörleriyle oluşturulup düzenlenebileceği gibi basit bir metin editörüyle de düzenlenebilir.

Bir belge hazırlamaya başlamak için verilecek ilk komut

```
\documentclass[...]{...}
```

olup, çengelli parantezler arasına oluşturmak istediğiniz belgenin sınıfı yazılır. Köşeli parantezlerin içine de isteğe bağlı bazı değişkenler yazılabilir. Eğer bu kısım boş bırakılırsa LaTeX varsayılan değerleri alacaktır. Bu komutun ardından sırasıyla `\begin{document}` ve `\end{document}` komutları verilerek belge ortamı oluşturulur. `\end{document}` komutuyla LaTeX’e belgenin bittiği söylenmiş olur ve LaTeX bu komuttan sonra girilenleri dikkate almaz.

¹⁶<https://beebom.com/best-latex-editors/>

¹⁷<https://texstudio.org/>

¹⁸<https://www.overleaf.com/>

¹⁹<https://papeeria.com/>

²⁰<https://latexbase.com/>

²¹<https://www.overleaf.com/latex/templates>

²²<https://www.overleaf.com/learn>

`\documentclass` komutuyla `\begin{document}` komutu arasına *sahanlık* denir. Sahanlık, belgenin ayarlarının yapıldığı kısımdır ve bu kısım çıktıda görünmez. `\begin{document}` ile `\end{document}` arasına da *gövde* denir. İçerik burada oluşturulur.

Aşağıda asgari bir LaTeX kaynak dosyası gösterilmiştir. `\documentclass` komutunun değişkeni olan `article`, belgenin makale olacağını belirtir.

```
\documentclass{article}
\begin{document}
İşte ilk belgem.
\end{document}
```

Bu noktadan sonra örnek kaynak dosyayı LaTeX editörünüzünde oluşturup önceden oluşturduğunuz bir dizine kaydedin. Kaydederken dosya adında boşluk ve Türkçe karakter kullanmayın. Örneğin kaynak dosyanız `belge1.tex` olsun.

İkinci aşama kaynak dosyanın derlenmesidir. Derleme işlemi için LaTeX editörlerinde genelde araç çubuğunda oklar bulunur. Oka tıklandığında dosya derlenir ve sonuç, çıktı ekranında görünür.

Eğer metin editörü kullanıyorsanız derlemeyi uçbirimde (terminal, konsol,...) yapmanız gerekir. Derleme için uçbirim kaynak dosyanın olduğu dizinde açılıp

```
pdflatex belge1
```

komutu verilmelidir.

Derleme işleminden sonra kaynak dosyanızın olduğu dizinde `belge1.tex` ve `belge1.pdf` dosyalarının yanında yine `belge1` ile başlayan farklı uzantılara sahip dosyalar olacaktır. Bu dosyaların ne olduklarına ilerleyen yazılarda değinilecektir ancak dileyen okur (Oetiker et al., 2006, s. 13-14)'e bakabilir.

1.5 Belge Sınıfları ve Seçenekleri

Bölüm 1.4'de `\documentclass` komutunun zorunlu değişkeninin belge sınıfı olduğunu ve köşeli parantezler içine de seçeneklerin yazılacağından bahsetmiştik. Bu yazıda bunların neler olabileceklerinden bahsedelim.

Başka sınıflar olmakla birlikte LaTeX'de varsayılan olarak kullanılan beş belge sınıfı vardır.

<code>article</code>	Makale
<code>report</code>	Makaleden daha hacimli belgeler için kullanılır. Rapor, tez gibi

1 Genel

book	Kitap
letter	Mektup
beamer	Sunu

Bu beş sınıftan article, report ve book için kullanılabilecek seçenekler aşağıdadır.

10pt, 11pt, 12pt	Belge ana yazı büyüklüğü.
a4paper, a5paper, letterpaper,...	Kağıt boyutu.
fleqn	Formülleri ortada yazmak yerine, sola bitişik yazar.
leqno	Formül numaralarını sağ yerine sol tarafa koyar.
titlepage, notitlepage	Belge başlığını attıktan sonra yeni bir sayfa açıp açmayacağını belirtir.
onecolumn, twocolumn	Belgenin tek sütun veya çift sütun dizileceğini belirtir.
twoside, oneside	Belgenin kağıdın hep tek tarafına mı yoksa iki tarafına mı basılacağını belirtir.
landscape	Belgeyi enine tutulmuş kağıda basılmak üzere hazırlar.
openright, openany	Belgede bölümleri hep sağ sayfalardan veya ilk gelen boş sayfada başlatır.
draft, final	Belgeyi sırasıyla <i>taslak</i> ve <i>son</i> şeklinde hazırlar. draft seçilirse, sağ

Bu seçeneklerin her birinin kullanılabilirliği belge sınıfına göre farklılık gösterir. Aşağıdaki tabloda hangi seçeneğin hangi sınıf için varsayılan olduğu ve kullanılabilir olup olmadığı gösterilmiştir.

Seçenek	book	report	article
10pt	1	1	1
letterpaper	1	1	1
oneside	1/2	1	1
twoside	1	1/2	1/2
openany	1/2	1	0
openright	1	1/2	0
titlepage	1	1	1/2
final	1	1	1

1: varsayılan 1/2: kullanılabilir 0:kullanılamaz

Örneğin belgeye

```
\documentclass[a4paper,12pt]{article}
```

komutuyla başlarsak LaTeX'e kağıt boyutu A4, ana yazı büyüklüğü 12 punto olan bir makale yazacağımızı bildirmiş oluruz.

Başka bir örnek

```
\documentclass[a5paper,11pt,twocolumn]{book}
```

olsun. Bu örnekte kağıt boyutu A5, ana yazı büyüklüğü 11 punto olan bir kitap yazacağımızı ve kitabın iki sütun olarak dizilmesini söyledik.

1.6 Türkçe Dil Ayarları ve Çoklu Dil Kullanımı

LaTeX’de Türkçe belgeler oluşturmak için öncelikle sahanlığa

```
\usepackage[T1]{fontenc}
\usepackage[turkish]{babel}
```

komutlarının verilmesi gerekir.

T1 seçenekli fontenc paketi yazıtipi kodlamasıyla ilgili bir paket olup, hecelemenin doğru şekilde yapılmasını sağlar. Bir çok Avrupa dilinde de T1 seçeneğiyle kullanılır. turkish seçenekli babel paketi de Chapter, Table, Contents,... gibi isimlerin Türkçeleşmesi (Bölüm, Tablo, İçindekiler,...) içindir.

Yakın zamana kadar ö, ş, ç,... gibi Türkçe karakterlerin kullanılabilmesi için sahanlığa `\usepackage[utf8]{inputenc}` ya da `\usepackage[latin5]{inputenc}` komutlarından birinin verilmesi gerekiyordu. Bu paket (inputenc) girdi kodlamasını yöneten bir pakettir. Son güncellemelerle birlikte bu paketin kullanılma zorunluluğu ortadan kalkmıştır.

Aşağıda Türkçe asgari bir LaTeX kaynak dosyası örneği verilmiştir.

```
\documentclass{article}
\usepackage[T1]{fontenc}
\usepackage[turkish]{babel}
\begin{document}
İşte Türkçe ilk belgem.
\end{document}
```

Türkçe dışında ikinci bir dil kullanmak isterseniz, örneğin İngilizce, babel paketinin seçeneğini

```
\usepackage[english,turkish]{babel}
```

şeklinde değiştirmeniz gerekir. Burada etkin olan dil Türkçedir. İngilizceyi etkin hale getirmek için `\selectlanguage{english}` komutu kullanılır. Tekrar Türkçeye geçmek için de benzer şekilde `\selectlanguage{turkish}` komutu kullanılır.

1 Genel

Bir kelime ya da cümle gibi kısa metinler kullanılacaksa `\foreignlanguage` komutu kullanılabilir:

```
\foreignlanguage{<dil>}{<metin>}
```

Uzun metinler içinse diğer bir seçenek `otherlanguage` ortamıdır.

```
\begin{otherlanguage}{<dil>}  
...  
\end{otherlanguage}
```

Bu ortamın isimleri değiştirmeyen, örneğin, dil seçeneği İngilizce olmasına rağmen belgeye bir tablo eklediğinizde “Table” yerine yine “Tablo” adını yazan yıldızlı sürümü de (`otherlanguage*`) vardır.

1.7 Heceleme

Bazen tüm bu ayarlamalara rağmen LaTeX bazı kelimeleri doğru heceleleyemeyebilir. Böyle durumlarda hecelemeyi elle yapmak gerekir. Yanlış hecelenen kelimenin bölünebileceği yerler `\-` komutuyla gösterilir:

```
He\ -ce\ -le\ -me
```

Bu sadece ilgili kelimenin tireyle ayrıldığı yerde doğru hecelenmesini sağlar. Aynı kelime belgenin başka bir yerinde yine yalnız hecelenebilir. Bunun yerine `\begin{document}` komutundan sonra `\hyphenation` komutuyla hece yerleri tire (-) işaretiyle gösterilmiş olan kelime listesi oluşturulursa belgenin tamamına bu kural uygulanmış olur. Örneğin

```
\hyphenation{He-ce-le-me FORTRAN}
```

komutuyla “Heceleme” kelimesinin nereden bölüneceği, “FORTRAN”, “Fortran” ya da “fortran” kelimelerinin bölünmeyeceği LaTeX’e söylenmiş olur.

1.8 Belgeye Başlık Oluşturma

LaTeX’de belgeye başlık oluşturmak için `\title` komutu kullanılır. Yazar adı `\author` komutuyla girilir. Birden fazla yazar varsa yazar adları arasına `\and` komutu girilir.

İsteğe bağlı olarak tarih için `\date` komutu kullanılır. Eğer `\date` komutu kullanılmazsa LaTeX belgenizi derlediğiniz günün tarihini basar. Tarihin basılmasını istemiyorsanız, bu komutu tarih yazılmadan `\date{}` şeklinde kullanmanız gerekir.

Son olarak, başlığın belgenize yazılması için `\begin{document}` komutundan sonra başlığı oluşturmak istediğiniz yere `\maketitle` komutunu girersiniz. Belge başlığını attıktan sonra yeni bir sayfanın açılıp açılmayacağı Bölüm 1.5’de belirttiğimiz gibi belgenin sınıfına bağlı olarak belirlenir.

Ayrıca `\title`, `\author` ve `\date` komutları `\thanks` komutunu içerebilir. Bu komutun değişkeni bir e-posta adresi, iş adresi veya bir teşekkür metni olabilir.

```
\documentclass[a4paper,12pt]{article}
\usepackage[T1]{fontenc}
\usepackage[turkish]{babel}
\title{Belge Başlığı}
\author{Yazar 1\thanks{A Üniversitesi} \and Yazar 2\thanks{B Üniversitesi}}
\date{XX.XX.XXXX}
\begin{document}
\maketitle

İçerik...
\end{document}
```

1.9 Bölümleme ve İçindekiler Tablosu

LaTeX’de belgenizi bölümlere ayırmak için 7 seviye bulunmaktadır.

Komut	Seviye	Açıklama
<code>\part{...}</code>	-1(book ve report) o (article)	letter hariç
<code>\chapter{...}</code>	0	sadece book ve report
<code>\section{...}</code>	1	letter hariç
<code>\subsection{...}</code>	2	letter hariç
<code>\subsubsection{...}</code>	3	letter hariç
<code>\paragraph{...}</code>	4	letter hariç
<code>\subparagraph{...}</code>	5	letter hariç

Türkçe dil paketi ekli belgelerde `\part` komutu “Kısım”, `\chapter` komutu “Bölüm” olarak yazılır. Kısımlar I, II, III,... şeklinde bölümler ise 1, 2, 3,... şeklinde numaralandırılır. `\section` komutu book ve report sınıflarında `\chapter` komutunu takip ederek 1.1, 1.2,... diğer sınıflarda 1, 2, 3,... şeklinde numaralandırılır. `\subsection` komutu da `\section` komutunu takip ederek numaralandırılır.

1 Genel

Türkçe dil paketi ekli belgelerde `\part` komutu “Kısım”, `\chapter` komutu “Bölüm” olarak yazılır. Kısımlar I, II, III,... şeklinde bölümler ise 1, 2, 3,... şeklinde numaralandırılır. `\section` komutu book ve report sınıflarında `\chapter` komutunu takip ederek 1.1, 1.2,... diğer sınıflarda 1, 2, 3,... şeklinde numaralandırılır. `\subsection` komutu da `\section` komutunu takip ederek numaralandırılır.

İçindekiler tablosu için LaTeX’e `\tableofcontents` komutu verilir. Bu komutun yazıldığı yerde İçindekiler tablosu oluşturulur. İçindekiler tablosunun doğru dizilmesi için kaynak dosyanızı en az iki kere derlemeniz gerekir.

LaTeX’de article sınıfında 4 ve 5’inci seviye başlıklara, book ve report sınıflarında ise bunlara ek 3’üncü seviye başlıklara numara verilmez ve numara verilmeyen başlıklar İçindekiler tablosuna yazılmaz. Bu seviyelerdeki başlıklara numara verilmesini ve İçindekiler tablosuna yazılması için iki adet `\setcounter` komutu

```
\setcounter{secnumdepth}{<seviye>}
\setcounter{tocdepth}{<seviye>}
```

şeklinde kullanılır. Birinci komuttaki `<seviye>` değişkeninde kaçınıcı seviyeye kadar olan başlıkların numaralandırılacağını, ikinci komuttaki `<seviye>` değişkeninde de kaçınıcı seviyeye kadar olan başlıkların İçindekiler tablosuna yazılacağını sayıyla belirtirsiniz. Örneğin book ve report sınıflarında

```
\setcounter{secnumdepth}{3}
\setcounter{tocdepth}{3}
```

komutlarıyla `\subsubsection` komutuna kadar olan başlıklara hem numara verir hem de İçindekiler tablosuna yazdırırsınız. Komutların çalışması için ya sahanlıkta ya da `\tableofcontents` komutundan önce verilmelidir.

Uzun başlıkların İçindekiler tablosunda daha kısa yazılması istenirse bölüm komutlarının zorunlu olmayan değişkenine başlıkların kısa şekli yazılır:

```
\section[Kısa Başlık]{Uzuuuuuuuuuuuuuuuun Başlık}
```

Bölüm komutlarının birde yıldızlı sürümleri vardır:

```
\part*{...}
\chapter*{...}
\section*{...}
\subsection*{...}
\subsubsection*{...}
\paragraph*{...}
\subparagraph*{...}
```

Komutlar bu şekilde verildiğinde başlığa numara verilmez ve İçindekiler tablosuna yazılmaz.

İçindekiler tablosunu LaTeX otomatik oluştursa da elle eklemeler yapılabilir, hatta Kaynakça gibi özel sayfalarda bu eklemeler gereklidir. Bunun için `\addcontentsline` komutu kullanılır.

```
\addcontentsline{toc}{<giriş formatı>}{<giriş metni>}
```

Burada `toc`, bilginin yazılacağı İçindekiler tablosunun dosya uzantısıdır. Bütünleşik olarak oluşturduğunuz kaynakçanın İçindekiler tablosuna yazılması için `\begin{thebibliography}` komutunun peşine `book` ve `report` sınıflarında

```
\addcontentsline{toc}{chapter}{Kaynakça}
```

`article` sınıfında ise

```
\addcontentsline{toc}{section}{Kaynaklar}
```

komutunun verilmesi gerekir.

“Kaynakça” ya da “Kaynaklar” isimleri yerine farklı isimler kullanılabilir elbette. Ancak `thebibliography` ortamının oluşturulduğu yerlerde LaTeX bu isimleri yazdıracağından tutarlı olması açısından bu isimler önerilmiştir.

```
\documentclass[a4paper,12pt]{article}
\usepackage[T1]{fontenc}
\usepackage[turkish]{babel}
\title{\LaTeX'de Bölümlendirme ve İçindekiler Tablosu Oluşturma}
\author{\TeX dizgi}
\begin{document}
\maketitle
\tableofcontents
\section{Birinci Seviye Başlık}
İçerik...
\subsection{İkinci Seviye Başlık}
İçerik...
\subsubsection{Üçüncü Seviye Başlık}
İçerik...
\section[Kısa Başlık]{Uzunuuuuuuun Başlık}
İçerik...
\section*{Numarasız Başlık}
\addcontentsline{toc}{section}{Numarasız Başlık}
İçerik...
\end{document}
```

1.10 Kitap Projesi Başlatma

LaTeX’de kitap yazmaya başlamak için belge sınıfı `book` seçilir. Bunun dışında kitapların *Baş*, *Gövde* ve *Son* kısımları olur. Bu kısımların nerede başlayıp nerede bittikleri aşağıdaki komutlarla LaTeX’e bildirilir:

- `\frontmatter` (baş) komutu `\begin{document}` komutundan hemen sonra verilir. Bu komut, baş taraftaki İçindekiler, Önsöz gibi kısımların sayfa numaralandırmasını Roma rakamıyla yapar. Ayrıca bu kısımda bölüm komutları (*) işareti olmadan verildiğinde (örneğin `\chapter{Önsöz}`) bunlara numara verilmez ancak İçindekiler tablosuna yazılırlar.
- `\mainmatter` (gövde) komutu kitabın ilk bölüm başlığından hemen önce verilmelidir. Buradan itibaren sayfa numaralandırmasını yeniden başlatıp Arap rakamlarına geçer.
- `\appendix` (ekler) komutu kitabınızın eklerindeki bölümleri harflerle numaralandırır (Ek A, Ek B, ...).
- `\backmatter` (son) komutu kitabınızda her şey bittikten sonra verilir fakat bilinen belge sınıflarında görünürde hiçbir etkisi yoktur.

Kitap gibi büyük hacimli belgelerle çalışırken kaynak dosyanızı parçalara ayırmak gerekebilir. LaTeX bunun için size iki komutla yardımcı olur: `\input` ve `\include`. İkisi arasındaki fark `\include` komutuyla eklediğiniz metin yeni bir sayfadan başlayarak dizilir.

Bu komutların zorunlu değişkeni eklemek istediğiniz dosyanın adıdır. Örneğin kaynak dosyanızla aynı dizinde yer alan `dosya1.tex` dosyasını eklemek için

```
\include{dosya1}
```

komutunu kullanırsınız. Eğer dosya uzantısı `.tex` değilse (örneğin `.txt` olsun) o zaman dosya adını uzantısıyla yazmanız gerekir:

```
\include{dosya1.txt}
```

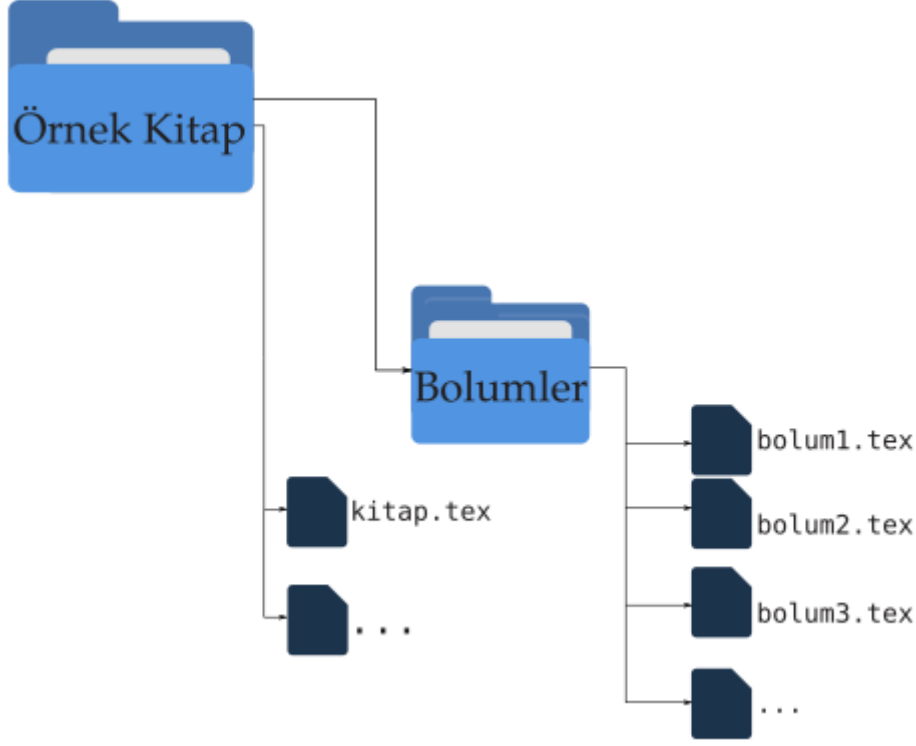
Ayrıca, hangi dosyaların eklenebileceğini LaTeX’e bildiren bir komut vardır: `\includeonly`. Bu komut, sadece sahanlığa yazılabilir. Komutun zorunlu değişkeninde eklenebilecek dosyalar aralarına virgül koyularak (ve boşluk bırakılmadan) listelenir:

```
\includeonly{dosya1,dosya2,dosya3,...}
```

Böyle bir liste oluşturulduktan sonra bu listede olmayan bir dosya artık `\include` komutuyla kaynak dosyaya eklenemez.

`\input` komutu sahanlıkta da kullanılabilir. Örneğin, sahanlığınızı tek bir dosyaya yazıp, bu dosyayı bu komutla sahanlığa ekleyebilirsiniz.

Daha düzenli çalışmak adına kaynak dosyanızın olduğu dizini de düzenleyebilirsiniz.



Şekil 1.1: Kaynak dosyanın olduğu dizinin düzenlenmesi

Bu şekilde bir düzenleme yaptığınızda `\input` ya da `\include` komutlarıyla dosya eklemek istediğinizde dosyanın bulunduğu dizini de göstermeniz gerekir.

Burada kaynak dosya `kitap.tex`'dir. Bu kaynak dosyaya `bolum1.tex` dosyasını eklemek istediğinizde komutu

```
\input{Bolumler/bolum1}
```

şeklinde verirsiniz. Bu sayede kaynak dosyanızın olduğu dizinde (Örnek Kitap) sadece `kitap` ile başlayan dosyalar olur. Diğer dosyalar alt dizinde (Bolumler) yer alır.

Dikkat edilirse “Örnek Kitap” dışında, “Bolumler” alt dizini ve tüm dosya adları Türkçe karakter ya da boşluk içermez.

```
\documentclass[a4paper,12pt]{book}
\usepackage[T1]{fontenc}
```

1 Genel

```
\usepackage[turkish]{babel}
\title{Örnek Kitap}
\author{\TeX dizgi}
\begin{document}
\frontmatter
\maketitle
\tableofcontents
\input{Bolumler/onsoz}
\mainmatter
\input{Bolumler/bolum1}
\input{Bolumler/bolum2}
\appendix
\input{Bolumler/ek1}
\backmatter
\end{document}
```

1.11 Varsayılan Sayfa Düzenini Değiştirme

LaTeX’de varsayılan kağıt boyutunun letterpaper olduğunu Bölün 1.5’de ifade etmiştik. Ayrıca aynı yazıda başka bir kağıt boyutunun nasıl seçileceğine de yer vermiştik. Şimdi ise hem sayfamızın kenar boşluklarının nasıl ayarlanacağından hem de ön tanımlı olmayan, tamamen keyfi bir sayfa boyutunun nasıl belirleneceğinden bahsedelim.

Bu tür sayfa düzenlemeleri için LaTeX’de geometry²³ paketi kullanılır. Öncelikle paketi

```
\usepackage{geometry}
```

komutuyla sahanlığa ekleyin. Ardından paket seçeneklerinde aşağıdaki tanımlamalarla sayfanın düzenini değiştirebilirsiniz:

Tanım	Değer
top	üst boşluk
bottom	alt boşluk
left	sol boşluk
right	sağ boşluk
paperwidth	sayfa genişliği
paperheight	sayfa yüksekliği

²³<http://ftp.cc.uoc.gr/mirrors/CTAN/macros/latex/contrib/geometry/geometry.pdf>

Örneğin sahanlıkta

```
\usepackage[paperwidth=175mm,paperheight=255mm,top=2cm,bottom=2cm,left=2.5cm,right=2.5cm]{geometry}
```

komutunu verdiğinizde boyutu 175×255 mm, üst ve alt boşluğu 2 cm, sol ve sağ boşluğu 2.5 cm olan bir sayfa düzeni oluşturursunuz. Dilerseniz `paperwidth` ve `paperheight` tanımlamalarını yerine, örneğin `a4paper` yazarak sadece kenar boşluklarla ilgili tanımlamaları yapabilirsiniz.

LaTeX'de milimetre (mm) ve santimetre (cm) dışında inç (in), punto (pt), em ve ex gibi ölçü birimleri de vardır. Bunlara ileride değinilecektir. Ayrıca yine ileride daha ayrıntılı sayfa düzeni oluşturmaktan da bahsedeceğiz. Bu aşamada bu kadarı yeterli olacaktır.

1.12 Satır ve Sayfa Kesme

LaTeX, kelimeler arası boşlukları otomatik ayarlayarak satırları iki yana yaslayarak dizer. Bir satırı kesip yeni bir satıra geçmek için `\` veya `\newline` komutları kullanılır.

Birinci komut `*` şeklinde verildiğinde satırdan sonra sayfa kesilmesini önler.

Benzer şeyi `\linebreak` komutu da yapar. Fakat bu komut ile satır kesilirse LaTeX kalan yarım satırı iki yana yaslar. `\nolinebreak` komutu ise satırın kesilmesini önler.

```
Yaslan güzelim yasla \linebreak göğsüne beni.
```

Birçok kelimeyi birlikte aynı satırda tutmak gerekirse `\mbox` komutu kullanılır:

```
\mbox{<metin>}
```

Buradaki `<metin>` içindeki kelimeler her durumda birleşik kalırlar. Benzer şeyi `\fbox` komutu da metin etrafına çizgi çizerek yapar.

Sayfayı kesip yeni bir sayfaya geçmek için `\newpage` ya da `\pagebreak` komutları kullanılır. `\nopagebreak` komutu sayfa kesilmesini önler. `\newpage` ile `\pagebreak` komutları arasında da `\newline` ile `\linebreak` komutlarındakine benzer bir fark vardır.

Kaynakça

Oetiker, T., Partl, H., Hyna, I., and Schlegl, E. (2006). *İnce bir LaTeX2e Elkitabı veya, 116 dakikada LaTeX2e*. <http://ftp.ntua.gr/mirror/ctan/info/lshort/turkish/lshort-tr.pdf>.
Türkçesi: Bekir Karaoğlu.