

Kom igång med Latex

Introduktion

Latex är ett typsättningssystem som kan användas för att skriva dokument och göra presentationer ("Powerpoints"). Den största skillnaden mot att skriva i Word eller något annat så kallat *What You See Is What You Get*-program, är att Latex är ett märkspråk där all formatering anges med taggar. Till en början kan det kännas lite ovant, men man vänjer sig snabbt. Nedan anges koden för det här stycket, vilket nu även innefattar **fetstil** och några matematiska symboler ($E = mc^2$).

```
\section*{Introduktion}
Latex är ett typsättningssystem som kan användas för att skriva
dokument och göra presentationer ('Powerpoints'). Den största
skillnaden mot att skriva i Word eller något annat så kallat
\emph{What You See Is What You Get}-program, är att Latex är ett
märkspråk där all formatering anges med taggar. Till en början
kan det kännas lite ovant, men man vänjer sig snabbt. Nedan
anges koden för det här stycket, vilket nu även innefattar
\textbf{fetstil} och några matematiska symboler ( $E=mc^2$ ).
```

Det är förstås individuellt vad man gillar och ogillar med Latex. Motståndarna brukar klaga på att deras kollegor föredrar Word, att vissa tidskrifter inte accepterar Latexfiler och att det är jobbigt när man har råkat skriva en parentes för mycket och dokumentet därför inte kan kompileras. Jag brukar lyfta fram följande fem fördelar.

- De flesta av oss har inte förmågan att göra lika snygga dokument i Word som i Latex. Dessutom ska man inte underskatta signaleffekten av vilka program man använder.
- Latex lämpar sig särskilt väl för de som använder många matematiska tecken, vill inkludera automatiskt genererade regressionstabeller eller har andra särskilda behov. Ändå föredrar jag att skriva alla texter som är längre än två sidor i Latex.
- Det är riktigt att Latex inte är så förlåtande mot att man gör fel, vilket ibland kan vara frustrerande. Samtidigt är det en av fördelarna,

eftersom det inte görs en massa gissningar om vad man egentligen avser med det man skriver. Inte heller klistras det in en massa osynlig och oönskad formatering så fort man hämtar innehåll från ett annat dokument.

- Vissa störs av att de inte ser hur texten kommer att se ut när den är klar, men själv gillar jag att jag kan fokusera på innehållet och texten utan att lägga tid på formatering eller blockeras av att texten jag skriver ger fula sidbrytningar.
- När både innehåll och formatering bestäms av koden man skriver blir det också möjligt att skapa dokument med hjälp av ett statistikprogram eller scriptspråk. Latex är därför överlägset Word om man exempelvis vill anpassa ett dokument efter innehållet i en databas. Samma egenskaper gör det också smidigt att med några enkla funktioner i Excel generera snygga tabeller eller annan typ av repetitiv kod.

Det här är avsett som en introduktion till Latex för personer i min omgivning som i alla fall till en början vill använda samma program och paket som jag gör. Det finns många Latexguider på internet och de flesta är skrivna av personer med betydligt större kunskaper än vad jag själv besitter. Jämfört med dessa är den här introduktionen förhållandevis simpel och anpassad för personer som arbetar i Windows eller Mac OS och kanske inte är så vana vid programmering och kompilering.

Installation

Latex är inte ett program som man laddar ner och installerar. Det du behöver är en TeX-distribution och en texteditor. Det finns många alternativ, men jag rekommenderar att du först installerar MiKTeX och därefter installerar texteditorn WinEdt. Förhoppningsvis hittar WinEdt då din MiKTeX-installation, vilket gör det enklare att komma igång. MiKTeX är gratis, medan WinEdt bjuder på irriterande popups efter 31 dagar. Om du då vill fortsätta använda programmet kostar en licens \$40 för studenter och \$60 för universitetsanställda. Vill du använda WinEdt på din arbetsdator kan du prata med Peter Knutar om licens.

När WinEdt har hittat MiKTeX-installationen brukar jag bara ändra en sak i konfigurationen. När man kompilarar en TeX-fil till pdf skapar Latex ett stort antal andra filer, vilket gör att mapparna fylls av en massa filer som vi inte är så intresserade av. För att dessa filer ska hamna i en subfolder gör jag följande två ändringar.

- Öppna *Execution modes* under *Options* och klicka på fliken *TeX Options*. Ändra *TeX Auxiliary Output Folder* till $\%P\backslash\text{texaux}$ och ta bort kryssset som anger att att $\%P$ används när mappen inte existerar.

- Öppna Backup.ini följt av AutoSave.ini (filerna finns under *Backup*, *Auto Saving*, *File Status* i *Options Interface*) och redigera en rad per fil så att de lyder:

```
BACKUP_FILENAME="%p\texaux\%n%t.bak"  
AUTO_SAVE_FILENAME="%p\texaux\%n%t.sav"
```

När du har installerat allt är det dags att testa om du kan skapa och kompilera ett enkelt dokument. Skapa en fil som har filändelsen .tex och följande innehåll. Tryck på knappen PDFTexify i WinEdt, säg ja när du får två frågor om att installera nya paket och håll tummarna för att du lyckas skapa ditt första kompilerade Latexdokument. I så fall har du faktiskt kommit förbi de svåra stegen!

```
\documentclass{article}  
\usepackage[T1]{fontenc}  
\usepackage[swedish]{babel}  
\title{En början på något stort}  
\author{Ditt namn}  
\begin{document}  
\maketitle  
Hello world!  
\end{document}
```

Dokumentstruktur

Allt som står ovanför `\begin{document}` kallas för preamble. Här anges vilken typ av dokument du skriver, vilka paket som ska läsas in samt andra inställningar som du vill kunna använda i resten av dokumentet. När du anger ett paket för första gången kommer MiKTeX att fråga dig om du vill installera paketet och gör det åt dig om du säger ja. Ovanför såg du ett exempel på hur en preamble kan se ut, men antalet paket du använder kommer snabbt att växa. Jag rekommenderar därför att du skapar ett dokument som heter default.sty och som innehåller alla paket du använder relativt ofta. Allra bäst är om du kan lagra det på din Dropbox eller någon annanstans som du når oavsett varifrån du arbetar. Då kan du i alla dina dokument använda en kort och enkel preamble. Jag brukar utgå från följande två rader. Skicka gärna ett mail om du vill ha en kopia av min default.sty.

```
\documentclass[11pt,a4paper,english]{article}  
\usepackage{D:/Dropbox/res/latex/default}
```

Efter `\begin{document}` kommer allt innehåll. Det struktureras i sections och subsections, vilket är samma sak som rubriker och underrubriker. Rubriken ovan skrevs som `\section*{Dokumentstruktur}`, där asterisken står

för att rubriken inte ska numreras. Om man använder dokumentklasserna book eller report kan man även ange chapters, vilket är en högre rubriknivå än sections.

Hitta mer information

Det finns tonvis med Latexinformation på internet. I början hade jag störst nytta av Wikiboken på <http://en.wikibooks.org/wiki/LaTeX>. Jag föreslår att du börjar med att läsa där. Nu googlar jag fram svar på de flesta av mina frågor. De bästa träffarna kommer i regel från tex.stackexchange.com.

Några vanliga tecken

Det mesta du vill veta kan du hitta i Wikiboken, men vissa saker tänker man kanske inte på förrän någon annan påpekar dem. Latexs hantering av några specialtecken är en sådan sak. En stor skillnad mot att skriva i Word, är att Latex inte gissar att du menar något annat än vad du skriver. Word ändrar glatt raka citationstecken (") till sneda (”) och bindestreck (-) till tankstreck (–), men i Latex måste du lära dig att ange rätt tecken själv. Eftersom några tecken används som avgränsare, måste dessa anges med ett omvänt snedstreck (\). Nedan följer en tabell med några vanliga tecken och hur du skriver dem i Latex.

Tecken	Utseende	Latexkod
Bindestreck	-	-
Minustecken	—	<code>-\$-\$</code>
Kort tankstreck	-	<code>--</code>
Långt tankstreck	—	<code>---</code>
Enkelt citationstecken	,	<code>'</code>
Dubbelt citationstecken	”	<code>''</code>
Och-tecken	&	<code>\&</code>
Procenttecken	%	<code>\%</code>
Dollartecken	\$	<code>\\$</code>

Referenshantering

Det finns två stora paket som används för att formatera referenser och skapa referenslistor: natbib och biblatex. Av dessa rekommenderas i regel biblatex, eftersom natbib börjar bli till åren och inte längre uppdateras.

Det är dock endast det senare som fortfarande utvecklas och