

CS Essentials

Session 4: L^AT_EX Essentials 2



Feedback forms:

Feedback form on Google

<https://goo.gl/forms/l2Nd2uNJXoQf8gT52>

First Part: Summary

What is \LaTeX ?

Is it a **word processor**?

NO! \LaTeX encourages *content* and not *design*.

You can think of it as a programming language, whose purpose is to typeset a document.

Do not confuse \TeX and \LaTeX . The latter is an extended version of the former.

How to get L^AT_EX?

Try something related to:

```
apt install texlive
```

Creating the documents

`pdflatex name.tex`

and then open it with a PDF viewer.

`\begin` and `\end`

Tags used to create environments.

What is an environment?

A way of formatting text in a given manner.

All documents have multiple environments.

Note: We always need the **document** environment as the first one.

Another Note: It is possible to create your own environments, although existing ones cover almost everything you can think of.

Title page

Modify the **preamble**:

```
\documentclass{article}

\title{Is \LaTeX simple?}
\date{\today}
%\date{1010-10-10}
\author{Possible CompSoc Member}


\begin{document}
    \maketitle
    \newpage

    Hello, world!
\end{document}
```


Page numbering

We can see the page number on the title page.

`\pagenumbering` comes into handy.

```
\documentclass{article}

\title{Is \LaTeX simple?}
\date{\today}
\author{Possible CompSoc Member}
\begin{document}
    \pagenumbering{gobble}
    \maketitle

    \newpage
    \pagenumbering{arabic}
    Hello, world!
\end{document}
```

Sectioning

- For structuring the content we use:
- `\section`
- `\subsection`
- `\paragraph`

Note: Other commands are subsubsection and subparagraph.

Table of contents

`\tableofcontents` is the only thing we need to add!

Note: You might need to compile the document twice, because the `\tableofcontents` command needs to create a new document first and then use it.

Packages

What is a package?

A way of adding more available functions.

`\usepackage{Package name}`

Note: This must be place in the *preamble*.

Packages: Example

```
%...  
\begin{equation}  
    f(x) = x^2  
\end{equation}  
%...
```

You can not turn of the automatic numbering.

Packages: Example

We use a package called **amsmath**.

```
%preamble
\usepackage{amsmath}

%...
\begin{equation*}
    f(x) = x^2
\end{equation*}
%...
```

Lists: Unordered

Using the **itemize** environment:

```
\begin{itemize}  
  \item CLI  
  \item Vim  
  \item Bash  
  \item LaTeX  
\end{itemize}
```

Lists: Unordered

```
\begin{itemize}
  \item[$-]$ To a dash
  \item[$\ast$] To an asterisk
  \item[$.$] To a dot
  \item[$\text{what}$] To do a word
\end{itemize}
```

Note: Anything can be used here, even words.

Lists: Unordered

To change every symbol at the same time:

```
\usepackage{enumitem}

\begin{itemize}[label=$.$]
  \item Wow!
  \item This
  \item is
  \item incredible!
\end{itemize}
```

Lists: Ordered

Using the **enumerate** environment:

```
\begin{enumerate}  
  \item CLI  
  \item Vim  
  \item LaTeX  
\end{enumerate}
```

Lists: Ordered

Nested lists are easy to produce:

```
\begin{enumerate}
  \item CLI
  \begin{enumerate}
    \item cd
    \item cp
    \item mv
  \end{itemize}
  \item Vim
  \item Bash
  \item LaTeX
\end{enumerate}
```

Second Part

Mathematical equations

Mathematical equations

Two ways of writing equations in \LaTeX :

Mathematical equations

Two ways of writing equations in \LaTeX :

Using $\$$, like this:

Mathematical equations

Two ways of writing equations in \LaTeX :

Using $\$$, like this:

$f(x) = 5 * x ^ 5 + x ^ 3$: gives us:

Mathematical equations

Two ways of writing equations in \LaTeX :

Using $\$$, like this:

$f(x) = 5 * x^5 + x^3$: gives us:

$$f(x) = 5 * x^5 + x^3$$

Mathematical equations: Equation

Mathematical equations: Equation

Or using **environments**, as we have already seen:

Mathematical equations: Equation

Or using **environments**, as we have already seen:

```
%...  
\begin{equation}  
    f(x) = x^2  
\end{equation}  
%...
```

Mathematical equations: Align

Another useful environment, offered by the **amsmath** package, is the **align** environment:

Mathematical equations: Align

Another useful environment, offered by the **amsmath** package, is the **align** environment:

```
\begin{align*}
2 + 2 = &4 \\
&4 - 1 = 3 \\
\end{align*}
```

Mathematical equations: Align

Another useful environment, offered by the **amsmath** package, is the **align** environment:

```
\begin{align*}
2 + 2 = &4 \\
&4 - 1 = 3 \\
\end{align*}
```

*Note: We use the * to indicate we do not want the equations to be numbered.*

Mathematical equations: Align

Mathematical equations: Align

Problem: Both equations are on the same line. We did not use `\\`.

Mathematical equations: Align

Problem: Both equations are on the same line. We did not use `\\`.

```
\begin{align*}
  2 + 2 = &4\\
  &4 - 1 = 3
\end{align*}
```

Fractions and Integrals and Square roots

Fractions and Integrals and Square roots

```
\begin{align*}
    f(a, b) &= \frac{a}{b} \\
    g(a, b, x) &= \int^a_b x \\
    h(b, x) &= \sqrt[b]{x}
\end{align*}
```

Fractions and Integrals and Square roots

```
\begin{align*}
    f(a, b) &= \frac{a}{b} \\
    g(a, b, x) &= \int^a_b x \\
    h(b, x) &= \sqrt[b]{x}
\end{align*}
```

We can also combine them:

Fractions and Integrals and Square roots

```
\begin{align*}
    f(a, b) &= \frac{a}{b} \\\
    g(a, b, x) &= \int^a_b x \\\
    h(b, x) &= \sqrt[b]{x}
\end{align*}
```

We can also combine them:

```
g(a, b, \frac{x}{y}) = \int^a_b \frac{x}{y}
```

Matrices

Matrices

```
\[  
\begin{matrix}  
1 & 2\\  
3 & 4  
\end{matrix}  
\]
```

Matrices

Matrices

But this does not produce brackets. We could try:

Matrices

But this does not produce brackets. We could try:

```
[  
\[  
\begin{matrix}  
1 & 2\\  
3 & 4  
\end{matrix}  
\]  
]
```

Matrices

Matrices

But this places the brackets in the wrong way. What we need to do is:

Matrices

But this places the brackets in the wrong way. What we need to do is:

```
\[  
\begin{bmatrix}  
1 & 2\\  
3 & 4  
\end{bmatrix}  
\]
```

Others

Others

There is too much information to choose from. Just glance over:

Others

There is too much information to choose from. Just glance over:
this link: <https://en.wikibooks.org/wiki/LaTeX/Mathematics>

Others

There is too much information to choose from. Just glance over:
this link: <https://en.wikibooks.org/wiki/LaTeX/Mathematics>

Especially over the tables at the end.

Exercise 1

Try to replicate the document!

Pictures

Pictures

Need a package, called **graphicx**.

Pictures

Need a package, called **graphicx**.

Use the **figure** environment.

Pictures

Need a package, called **graphicx**.

Use the **figure** environment.

```
\begin{figure}  
  \includegraphics[width=\linewidth]{images/compsoc.jpg}  
  \caption{CompSoc's logo}  
  \label{fig:logo}  
\end{figure}
```


Pictures

Need a package, called **graphicx**.

Use the **figure** environment.

```
\begin{figure}  
  \includegraphics[width=\linewidth]{images/compsoc.jpg}  
  \caption{CompSoc's logo}  
  \label{fig:logo}  
\end{figure}
```

Note: Each picture is indexed and tagged with consecutive numbers.

Pictures

Need a package, called **graphicx**.

Use the **figure** environment.

```
\begin{figure}  
  \includegraphics[width=\linewidth]{images/compsoc.jpg}  
  \caption{CompSoc's logo}  
  \label{fig:logo}  
\end{figure}
```

Note: Each picture is indexed and tagged with consecutive numbers.

Another Note: This is the relative path. We could have used the absolute one.

Pictures: Dimensions

Pictures: Dimensions

`\linewidth` modifies the width of the picture.

Pictures: Dimensions

`\linewidth` modifies the width of the picture.

We could also set the *height* to something different:

Pictures: Dimensions

`\linewidth` modifies the width of the picture.

We could also set the *height* to something different:

```
[width=\linewidth,height=40in]
```

Pictures: Dimensions

`\linewidth` modifies the width of the picture.

We could also set the *height* to something different:

```
[width=\linewidth,height=40in]
```

If we are not sure about the dimension and want to keep the ratio,

Pictures: Dimensions

`\linewidth` modifies the width of the picture.

We could also set the *height* to something different:

```
[width=\linewidth,height=40in]
```

If we are not sure about the dimension and want to keep the ratio, **keepaspectratio** is the option we need to add.

Pictures: Caption and Label

Pictures: Caption and Label

`\caption` adds a description to the image.

Pictures: Caption and Label

`\caption` adds a description to the image.

`\label` adds a hidden name to the image, for future references.

Pictures: Caption and Label

`\caption` adds a description to the image.

`\label` adds a hidden name to the image, for future references.

Example:

CompSoc's logo `\ref{fig:logo}` is on page `\pageref{fig:logo}`.

Pictures: Positioning

Pictures: Positioning

The **figure** environment will not place the image exactly where you insert it.

Pictures: Positioning

The **figure** environment will not place the image exactly where you insert it.

\LaTeX places figures **wherever it finds enough empty space**.

Pictures: Positioning

The **figure** environment will not place the image exactly where you insert it.

\LaTeX places figures **wherever it finds enough empty space**.

In order to correct this, use options for the environment:

Pictures: Positioning

The **figure** environment will not place the image exactly where you insert it.

\LaTeX places figures **wherever it finds enough empty space**.

In order to correct this, use options for the environment:

```
\begin{figure}[h!]
```

Pictures: Positioning

The **figure** environment will not place the image exactly where you insert it.

\LaTeX places figures **wherever it finds enough empty space**.

In order to correct this, use options for the environment:

```
\begin{figure}[h!]
```

h = here

Pictures: Positioning

The **figure** environment will not place the image exactly where you insert it.

\LaTeX places figures **wherever it finds enough empty space**.

In order to correct this, use options for the environment:

```
\begin{figure}[h!]
```

h = here

t = top of the page

Pictures: Positioning

The **figure** environment will not place the image exactly where you insert it.

\LaTeX places figures **wherever it finds enough empty space**.

In order to correct this, use options for the environment:

```
\begin{figure}[h!]
```

h = here

t = top of the page

b = bottom of the page

Pictures: Positioning

The **figure** environment will not place the image exactly where you insert it.

\LaTeX places figures **wherever it finds enough empty space**.

In order to correct this, use options for the environment:

```
\begin{figure}[h!]
```

h = here

t = top of the page

b = bottom of the page

p = on an extra page

Pictures: Positioning

The **figure** environment will not place the image exactly where you insert it.

\LaTeX places figures **wherever it finds enough empty space**.

In order to correct this, use options for the environment:

```
\begin{figure}[h!]
```

h = here

t = top of the page

b = bottom of the page

p = on an extra page

Using **!** forces the specified location.

Footnotes

Footnotes

`\footnote` command together with `\label` `\ref`

Footnotes

`\footnote` command together with `\label` `\ref`

```
\begin{document}
```

```
Course organised by CompSoc
```

```
\footnote{\label{CompSoc}Oxford University CompSoc}.\
```

```
This society, reference \ref{CompSoc}, is great!
```

```
\end{document}
```

References: First part

References: First part

```
\documentclass{article}
```

```
\usepackage[backend=biber]{biblatex}
```

```
\addbibresource{help}
```

```
\begin{document}
```

```
This journal \cite{einstein} and this  
book \cite{dirac} are nice.
```

```
\nocite{*}
```

```
\printbibliography
```

```
\end{document}
```

References: Second part

References: Second part

```
\documentclass{article}
```

```
\usepackage[backend=biber]{biblatex}
```

```
\addbibresource{help}
```

```
\begin{document}
```

```
This journal \cite{einstein} and this  
book \cite{dirac} are nice.
```

```
\nocite{*}
```

```
\printbibliography
```

```
\end{document}
```

References: Compiling part

References: Compiling part

`pdflatex name.tex`

References: Compiling part

pdflatex name.tex

biber name

References: Compiling part

pdflatex name.tex

biber name

pdflatex name.tex

References: Key things

References: Key things

The name of the *bibliography* must be the same as the name of the .tex file.

References: Key things

The name of the *bibliography* must be the same as the name of the .tex file.

References will not include unmentioned references. Use:

`\nocite{*}`

Code

Code

Need a package called **minted**.

Code

Need a package called **minted**.

You might need to install some dependencies:

```
apt install python-pygments
```

Code

Need a package called **minted**.

You might need to install some dependencies:

```
apt install python-pygments
```

and also run **pdflatex -shell-escape name.tex**

Code

Need a package called **minted**.

You might need to install some dependencies:

```
apt install python-pygments
```

and also run **pdflatex -shell-escape name.tex**

```
\usepackage[cache=false]{minted}
```

```
\begin{minted}{Python}
```

```
x = 99
```

```
if x == 99:
```

```
    #indented four spaces
```

```
    print("99 problems and writing code ain't 1.")
```

```
\end{minted}
```

Code: Notes

```
1  # Import the necessary libraries
2  import pandas as pd
3  import numpy as np
4  import matplotlib.pyplot as plt
5  import seaborn as sns
6  from sklearn.preprocessing import StandardScaler
7  from sklearn.model_selection import train_test_split
8  from sklearn.metrics import accuracy_score, confusion_matrix, roc_auc_score
9  from sklearn.svm import SVC
10 from sklearn.ensemble import RandomForestClassifier
11 from sklearn.linear_model import LogisticRegression
12 from sklearn.metrics import classification_report
13
14 # Load the dataset
15 data = pd.read_csv('data.csv')
16
17 # Check the shape of the dataset
18 print(data.shape)
19
20 # Display the first few rows of the dataset
21 print(data.head())
22
23 # Check for missing values
24 print(data.isnull().sum())
25
26 # Drop missing values
27 data = data.dropna()
28
29 # Check the distribution of the target variable
30 print(data['target'].value_counts())
31
32 # Split the data into training and testing sets
33 X = data.drop('target', axis=1)
34 y = data['target']
35 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
36
37 # Standardize the features
38 scaler = StandardScaler()
39 X_train = scaler.fit_transform(X_train)
40 X_test = scaler.transform(X_test)
41
42 # Train the Logistic Regression model
43 lr = LogisticRegression()
44 lr.fit(X_train, y_train)
45
46 # Predict the target variable for the test set
47 y_pred_lr = lr.predict(X_test)
48
49 # Calculate the accuracy of the Logistic Regression model
50 accuracy_lr = accuracy_score(y_test, y_pred_lr)
51 print('Logistic Regression Accuracy: ', accuracy_lr)
52
53 # Train the Random Forest Classifier model
54 rf = RandomForestClassifier()
55 rf.fit(X_train, y_train)
56
57 # Predict the target variable for the test set
58 y_pred_rf = rf.predict(X_test)
59
60 # Calculate the accuracy of the Random Forest Classifier model
61 accuracy_rf = accuracy_score(y_test, y_pred_rf)
62 print('Random Forest Classifier Accuracy: ', accuracy_rf)
63
64 # Train the Support Vector Machine (SVC) model
65 svm = SVC()
66 svm.fit(X_train, y_train)
67
68 # Predict the target variable for the test set
69 y_pred_svm = svm.predict(X_test)
70
71 # Calculate the accuracy of the SVC model
72 accuracy_svm = accuracy_score(y_test, y_pred_svm)
73 print('SVC Accuracy: ', accuracy_svm)
74
75 # Generate a confusion matrix for the SVC model
76 cm = confusion_matrix(y_test, y_pred_svm)
77 print('Confusion Matrix for SVC:')
78 print(cm)
79
80 # Generate a ROC curve for the SVC model
81 roc_auc = roc_auc_score(y_test, svm.predict_proba(X_test)[:, 1])
82 print('ROC AUC for SVC: ', roc_auc)
83
84 # Generate a classification report for the SVC model
85 report = classification_report(y_test, y_pred_svm)
86 print('Classification Report for SVC:')
87 print(report)
```

Code: Notes

```
\inputminted{python}{hello.py}
```

Code: Notes

```
\inputminted{python}{hello.py}
```

It would be a good idea to explore the package on your own.

Code: Notes

```
\inputminted{python}{hello.py}
```

It would be a good idea to explore the package on your own.

There are options for almost everything:

Code: Notes

```
\inputminted{python}{hello.py}
```

It would be a good idea to explore the package on your own.

There are options for almost everything:

- languages

Code: Notes

```
\inputminted{python}{hello.py}
```

It would be a good idea to explore the package on your own.

There are options for almost everything:

- languages

- colouring

Code: Notes

```
\inputminted{python}{hello.py}
```

It would be a good idea to explore the package on your own.

There are options for almost everything:

- languages

- colouring

- new lines

Exercise 2

Try to replicate the document!

Thank you!