1 Markdown to LaTeX Comprehensive Guide

1.1 Features at a Glance

- ✅ **Standard Markdown Support**: Headings, lists, tables, links, images, blockquotes, and more.
- 📝 **Text Formatting**: Bold, italic, underline, strikethrough, highlight, superscript, subscript, small caps.
- 💻 **Code Blocks**: Syntax highlighting, inline code, executable Python code with output/plots.
- 📊 **Diagrams**: Mermaid diagrams for flowcharts and graphs.
- 📋 **Task Lists**: Checkboxes for to-do and progress tracking.
- 🦶 **Footnotes**: Inline and reference footnotes.
- 🧮 **Math**: Inline and display math with LaTeX syntax.
- 🏷️ **Definition Lists**: Term-definition pairs.
- 📑 **Tables**: Pipe tables, captions, and alignment.
- ⚠️ **Alerts Containers**: Note, tip, important, warning, caution, box, and alignment containers.
- 🎨 **Custom Extensions**: Center/right alignment, keyboard shortcuts, line breaks, and more.
- 🏷️ **Metadata Variables**: JSON metadata, document variables, and title page templates.
- 🌐 **Emoji Support**: Use emojis anywhere in your markdown for expressive documents! 😀

1.2 Headings

```
# Heading 1

## Heading 2

### Heading 3

#### Heading 4

##### Heading 5

###### Heading 6
```

2 Heading 1

2.1 Heading 2

2.1.1 Heading 3

Heading 4

Heading 5

Heading 6

2.2 Text Formatting

2.2.1 Bold

```
**bold text** or __bold text__
```

bold text or **bold text**

2.2.2 Italic

```
_italic text_ or *italic text*
```

italic text or *italic text*

2.2.3 Bold + Italic

```
**_bold and italic_** or ***bold and italic***
```

bold and italic or ***bold and italic***

2.2.4 Strikethrough

```
~~strikethrough text~~
```

~~strikethrough text~~

2.2.5 Highlight

```
==highlighted text==
```

highlighted text

2.2.6 Superscript

```
^superscript^ (e.g., x^2^)
```

superscript (e.g., x^2)

2.2.7 Subscript

```
~subscript~ (e.g., H~2~O)
```

subscript (e.g., H_2O)

2.2.8 Small Caps

```
:sc[Small Caps Text]
```

Small Caps Text

2.2.9 Underline

```
:u[Underlined Text]
```

Underlined Text

2.3 Inline Code

```
`inline code`
```

`inline code`

2.4 Code Blocks

2.4.1 Fenced Code Block

````
```python
def hello():
    print("Hello, World!")
```
````

```python
def hello():
    print("Hello, World!")
```

2.4.2 Code Block Without Language

```
plain text code block
```

```
plain text code block
```

2.4.3 Terminal Block

```terminal
$ command
output result
```

```
$ command
output result
```

2.5 Links

```
[link text](https://example.com)
```

link text

```
[link text](https://example.com "title")
```

link text

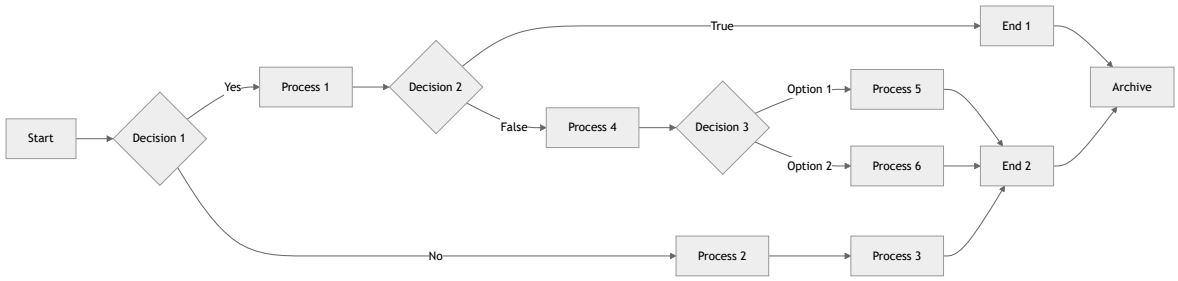
2.6 Images

```
![alt text](image.jpg)
```

```
![alt text](image.jpg "title")
```

Figure 1: title

2.7 Mermaid Diagrams

```
```mermaid
graph LR;
  A[Start] --> B{Decision 1};
  B -- Yes --> C[Process 1];
  B -- No  --> D[Process 2];
  C --> E{Decision 2};
  D --> F[Process 3];
  E -- True --> G[End 1];
  E -- False --> H[Process 4];
  H --> I{Decision 3};
  I -- Option 1 --> J[Process 5];
  I -- Option 2 --> K[Process 6];
  J --> L[End 2];
  K --> L;
  F --> L;
  G --> M[Archive];
  L --> M;
```
```



2.8 Lists

2.8.1 Unordered List

```
- Item 1
- Item 2
  - Nested item 2.1
  - Nested item 2.2
- Item 3
```

- Item 1
- Item 2
  - Nested item 2.1
  - Nested item 2.2
- Item 3

2.8.2 Ordered List

```
1. First item
2. Second item
   1. Nested item 2.1
   2. Nested item 2.2
3. Third item
```

1. First item
2. Second item
   1. Nested item 2.1
   2. Nested item 2.2
3. Third item

2.8.3 Task Lists

```
- [x] Completed task
- [ ] Incomplete task
- [/] Partially completed task
```

- ☑ Completed task
- ☐ Incomplete task
- ▣ Partially completed task

2.9 Definition Lists

```
Term 1
: Definition 1

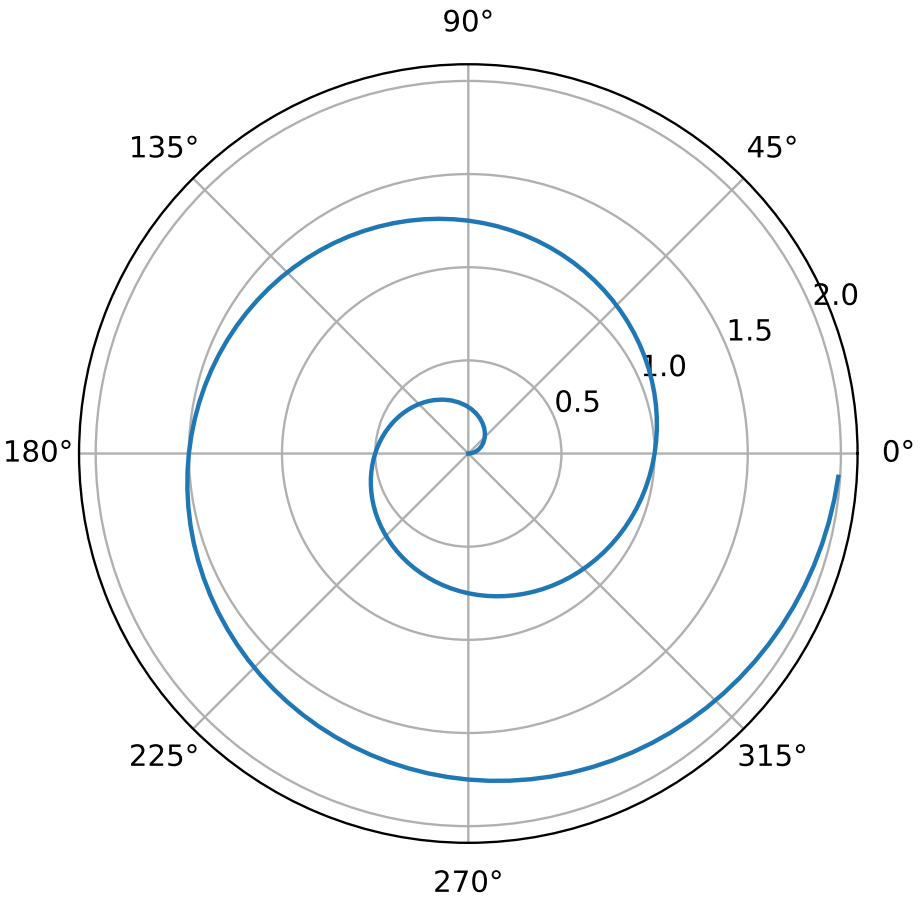
Term 2
: Definition 2a
: Definition 2b
```

Term 1
: Definition 1

Term 2
: Definition 2a
: Definition 2b

2.10 Tables

2.10.1 Pipe Tables

```
| Left | Center | Right | Default |
| :--- | :----: | ----: | ------- |
| L1   |   C1   |    R1 | Default |
| L2   |   C2   |    R2 | Default |
```

Left	Center	Right	Default
L1	C1	R1	Default
L2	C2	R2	Default

2.11 Table with Captions

```
| Header 1 | Header 2 | Header 3 |
| -------- | -------- | -------- |
| Cell 1   | Cell 2   | Cell 3   |
| Cell 4   | Cell 5   | Cell 6   |

: Sample Table with Caption
```

Header 1	Header 2	Header 3
Cell 1	Cell 2	Cell 3
Cell 4	Cell 5	Cell 6

Table 1: Sample Table with Caption

2.12 Horizontal Rule

```
---
```

```
***
```

```
---
```

2.13 Math Expressions

2.13.1 Inline Math

```
This is inline math: $E = mc^2$
```

This is inline math: $E = mc^2$

2.13.2 Display Math

```
$$
\int_{0}^{\infty} e^{-x^2} dx = \frac{\sqrt{\pi}}{2}
$$
```

$$\int_{0}^{\infty} e^{-x^2} dx = \frac{\sqrt{\pi}}{2}$$

2.14 Footnotes

2.14.1 Inline Footnote

```
Text with inline footnote^[This is the footnote content].
```

Text with inline footnote[1] .

2.14.2 Reference Footnote

```
Text with reference footnote[^1].

[^1]: This is the footnote content.
```

Text with reference footnote[2] .

2.15 Keyboard Shortcuts

2.15.1 Single Key

```
[[Ctrl]]
```

Ctrl

2.15.2 Key Combination with Plus

```
[[Ctrl] + [C]]
```

Ctrl + C

2.16 Line Break

2.16.1 Using Two Spaces

Put two spaces at the end of a line 1 to create a line break.

```
Line 1<space><space>
Line 2
```

Line 1
Line 2

2.16.2 Using Backslash

```
Line 1\
Line 2
```

Line 1
Line 2

---

[1] This is the footnote content

[2] This is the footnote content.

2.16.3 Using HTML `<br>` Tag

```
Line 1<br>
Line 2
```

Line 1
Line 2

```
Line 1<br/>
Line 2
```

Line 1
Line 2

2.16.4 Using Empty Line (Paragraph Break)

```
Line 1

Line 2
```

Line 1
Line 2

2.17 Blockquotes

2.17.1 Basic Blockquote

```
> This is a blockquote.
```

> This is a blockquote.

2.17.2 Multi-line Blockquote

```
> This is a blockquote.
> It can span multiple lines.
>
> And multiple paragraphs.
```

> This is a blockquote. It can span multiple lines.
> And multiple paragraphs.

2.17.3 Nested Blockquotes

```
> Level 1
>
> > Level 2
> >
> > > Level 3
```

> Level 1
>
> > Level 2
> >
> > > Level 3

2.18 Container Alerts

2.18.1 Note Alert

```
::: note
This is a note alert with blue styling.
:::
```

> **NOTE**
> This is a note alert with blue styling.

2.18.2 Tip Alert

```
::: tip
This is a tip alert with green styling.
:::
```

> **TIP**
> This is a tip alert with green styling.

2.18.3 Important Alert

```
::: important
This is an important alert with purple styling.
:::
```

> **IMPORTANT**
> This is an important alert with purple styling.

2.18.4 Warning Alert

```
::: warning
This is a warning alert with yellow/orange styling.
:::
```

> **WARNING**
> This is a warning alert with yellow/orange styling.

2.18.5 Caution Alert

```
::: caution
This is a caution alert with red styling.
:::
```

> **CAUTION**
> This is a caution alert with red styling.

2.19 Text Alignment Containers

2.19.1 Center Alignment

```
::: center
This text is centered
:::
```

This text is centered

2.19.2 Right Alignment

```
::: right
Right-aligned text
:::
```

Right-aligned text

2.20 Box Container

2.20.1 Basic Box

```
::: box
Text in a bordered box
:::
```

Text in a bordered box

2.21 Executable Python Code Blocks

The converter supports executing Python code blocks directly within your markdown and including their output or generated plots in the final PDF.

2.21.1 Prerequisites

For Python code execution to work, you need Python installed on your system. To generate plots with matplotlib, install the required packages:

```
python -m pip install matplotlib numpy
```

2.21.2 Available Properties

- `.execute` - Execute the code block (required)
- `.show-code` - Display the source code in the output
- `.show-output` - Display execution output/plot (default)
- `.hide-code` - Explicitly hide the source code (default)
- `.hide-output` - Hide execution output/plot

2.21.3 Examples 1

```
```python {.execute}
print("Hello, World!")
```
```

```
Hello, World!
```

2.21.4 Examples 2

```
```python {.execute}
import numpy as np
import matplotlib.pyplot as plt

r = np.arange(0, 2, 0.01)
theta = 2 * np.pi * r
fig, ax = plt.subplots(
  subplot_kw = {'projection': 'polar'}
)
ax.plot(theta, r)
ax.set_rticks([0.5, 1, 1.5, 2])
ax.grid(True)
plt.show()
```
```



2.22 Document Metadata (JSON)

Document metadata is configured in a separate JSON file:

```
{
  "title": "Document Title",
  "subtitle": "Course Name",
  "submittedto": "Professor Name",
  "university": "University Name",
  "department": "Department Name",
  "date": "January 1, 2024",
  "submittedby": [
    {
      "name": "Student Name",
      "roll": "Registration Number"
    }
  ],
  "titleTemplate": 1,
  "enableContentPage": false,
  "enablePageCredits": false,
  "moveFootnotesToEnd": false,
  "enableThatsAllPage": false,
  "tocDepth": 3,
  "variables": {}
}
```

2.22.1 Title Template Modes

The `titleTemplate` setting controls how the title page is displayed:

- `0` : No title (disabled)

- `1` : Full university title page with logo (default) - Good for assignments and reports
- `2` : Title header above content - Good for notes
- `3` : Title on separate page - Good for when the contents are enabled

2.22.2 JSON Variables

You can define variables in the JSON metadata file and use them throughout your markdown document. Variables are defined under the `"variables"` key and referenced using `{{variable_name}}` syntax.

Example JSON with variables:

```
{
  "variables": {
    "author": "abd",
    "version": "1.2.3",
    "course": "CS 101",
    "semester": "Fall 2025",
    "university_full": "University of Engineering and Technology"
  }
}
```

```
This report was written by {{author}} for {{course}} during {{semester}}.

Software version: {{version}}

Institution: {{university_full}}
```

This report was written by abd for CS 101 during Fall 2025.
Software version: 1.2.3
Institution: University of Engineering and Technology