Contents

1 Basic Report Writing

1.1 Subsection

1.1.1 Subsubsection

With some text in it.

1.1.2 Another subsubsection

Hello world. Here's some text in the first paragraph of this subsubsection.

Here's another paragraph in the same subsubsection.

2 Math

2.1 Some Random Equation

$$y = \sqrt[3]{\frac{5x}{x^2}}$$

This is an inline math equation: $y = x^2 + 2x + 4$ ...in the paragraph following Some Random Equation.

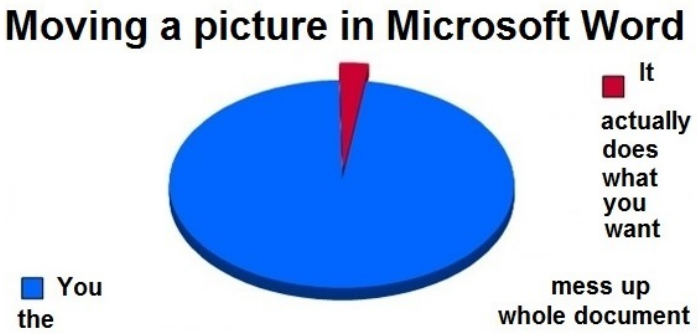
3 Figures and Images

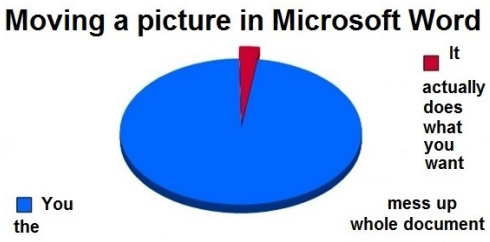
3.1 Basic Figure

For more examples and settings, click this link.

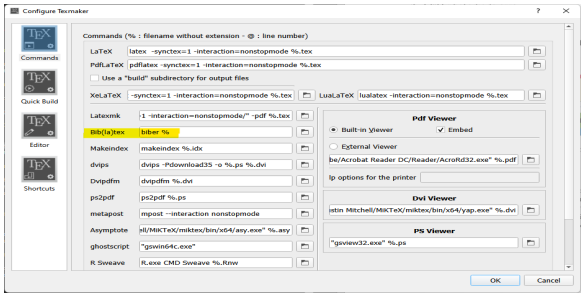
3.2 Side-by-Side Figures

Figure 1: A funny LaTeX meme





(a) Caption for 'a' [**Maxim**]



(b) Caption for 'b'

Figure 2: Theory of xyz

4 Lists

4.1 Non-numbered List

- First item
- Second item
  - First second item
  - Second second item
  - ! Last item with custom exclamation point marker

4.2 Numbered Lists

1. Numbered
2. list of
3. things
   (a) first thing
   (b) second thing

! exclaimed item

NOTE this item

→ custom arrow bullet in list

4. list cont.

5 Tables

Table 1: a table

head1	head2	head3
cell4	cell5	cell6
cell7 W/kg	cell8 N	cell9 seconds

6 Code

```
import pandas as pd
from tools.loading import print_loading_bar


data_source_str = "wordle-solver/data/words_3000"  # Designates data to process

# Open list of words
fh = open(data_source_str + "_raw.txt")
LENGTH = len(fh.readlines())  # get number of words
fh = open(data_source_str + "_raw.txt")  # reopen file handle

# Load letter frequency lookup table
freqs = pd.read_csv("wordle-solver/data/lookup_char-freqs.csv", index_col=0)

# Initialize data storage
words = pd.DataFrame()

WORD_LENGTH = 5  # Word length setting (Wordle is currently a 5-letter game)

# Process list of words
for i, word in enumerate(fh):

    print_loading_bar(i, LENGTH, title="Preparing word data: ", size=100, no_newline=True)

    word = word.strip().lower()

    if len(word) == WORD_LENGTH:
        words.loc[i, 'word'] = word
        score = 0
        for j in range(WORD_LENGTH):
            col = 'char'+str(j)
            words.loc[i, col] = word[j]
            score = freqs.loc[word[j], col] + score
        words.loc[i, 'score'] = score

words = words.sort_values('score', ascending=False)
words.to_csv(data_source_str + "_clean.csv")
```

For more code examples and parameters, click this link.

7 Citations

I'm citing something, here. [1]

References

[1] Albert Einstein. “Zur Elektrodynamik bewegter Körper. (German) [On the electrodynamics of moving bodies]”. In: *Annalen der Physik* 322.10 (1905), pp. 891–921. DOI: http://dx.doi.org/10.1002/andp.19053221004.