

クレジット:

UTokyo Online Education Education コンピュータシステム概論 2018 小林克志

ライセンス:

利用者は、本講義資料を、教育的な目的に限ってページ単位で利用することができます。特に記載のない限り、本講義資料はページ単位でクリエイティブ・コモンズ 表示-非営利-改変禁止 ライセンスの下に提供されています。

<http://creativecommons.org/licenses/by-nc-nd/4.0/>

本講義資料内には、東京大学が第三者より許諾を得て利用している画像等や、各種ライセンスによって提供されている画像等が含まれています。個々の画像等を本講義資料から切り離して利用することはできません。個々の画像等の利用については、それぞれの権利者の定めるところに従ってください。



Markdown (マークダウン)形式

Python プログラミングの前に、jupyter-notebook でサポートされているマークアップ言語、Markdown について説明する。

なぜこのタイミングで Markdown 形式なのか？

例えば本日の課題レポートのように、一部の提出物は Markdown 形式を求めているので

マークアップ言語(Markup Language) (復習)

マークアップ言語 スーパー大辞林

文書の中にマークをつけて、文書の構造や修飾情報などを記述するための言語。SGML, HTML, XML など。

SGML 型では"マーク"として <> が用いられており、人間の可読性は犠牲にされている。利用者は難読なマークアップ形式そのものの「入力」ではなく、ブラウザを介して整形された「出力」を目にすることとなる。

厳密に言えばこの説明は、「意味マークアップ」に分類される

マークアップ言語(Markup Language) (復習)(続き)

HTML の例

以下の HTML で

1. 見出し <H1> An Example </H1> の複数行のうち、<H1>および </H1> がタグであり、このタグで囲まれた部分が要素となる。
2. 次いで、パラグラフ<p>に文章が記述されている。
3. その後、番号なしリスト、そのリスト要素はそれぞれタグで示されている。
 - A. リスト要素の最後の単語は <a>(アンカー)タグのhref属性によって、参照先の設定されたハイパーリンクとなっている。

```
<H1>
  An Example of Structure
  <br>
  In HTML
</H1>
<P>
  Here's a typical paragraph.
  <UL>
    <LI>
      Item one has an
      <A href="http://www.example.com">
        anchor
      </A>
    <LI>
      Here's item two.
  </UL>
</P>
```

この HTML の例はブラウザでは以下のように表示される:

An Example of Structure In HTML

Here's a typical paragraph.

- Item one has an anchor (<http://www.example.com>)
- Here's item two.

Markdown とは

Markdown - Wikipedia

Markdown（マークダウン）は、文書を記述するための軽量マークアップ言語のひとつである。本来はプレーンテキスト形式で手軽に書いた文書からHTMLを生成するために開発されたものである。しかし、現在ではHTMLのほかパワーポイント形式やLATEX形式のファイルへ変換するソフトウェア（コンバータ）も開発されている。各コンバータの開発者によって多様な拡張が施されるため、各種の方言が存在する。

HTML はマークアップ言語として広く普及しているが、書きやすさの問題から人間が直接 HTML を書き下すことは困難である。

実際 Web デザイナは HTML, CSS を書き下すのではなく オーサリングソフトを利用し画面を作っている。

一方で、Wiki に代表される Blog サービスでは一般人が Web オーサリングツール上でリッチな Web 画面を作成するという要求は強い。

以上の理由から、「入力」と「出力」両方の可読性ならびに整形性を最大化するマークアップ言語として Markdown が作られた。

Markdown 形式は Wiki をはじめとして広く採用されている。

講義で利用する jupyter-notebook でもセルを切り替え Markdown 形式で記述、整形することができる。

ただし、個別のシステムで独自の拡張が(現在でも)続いており、互換性には課題が残る。

jupyter-notebook セルを Markdown で記述する

1. Markdown に切り替えたいセルを選ぶ
2. command mode で m を入力、
あるいはプルダウンメニューで Code-> Markdown に切り替える
3. Markdown で記述
4. Ctl + Enter でセルを実行

```
In [ ]: <!--  
Switch cell-mode from "Code" to "Markdown"  
Then "Run" with CTL-Enter  
-->  
My first **markdown**
```

jupyter-notebook がサポートする Markdown 形式

ここでは代表的なものを紹介する。詳細は、[jupyter-notebook のサイト \(http://jupyter-notebook.readthedocs.io/en/stable/examples/Notebook/Working%20With%20Markdown%20Cells.html\)](http://jupyter-notebook.readthedocs.io/en/stable/examples/Notebook/Working%20With%20Markdown%20Cells.html) や [Markdown のサイト \(https://daringfireball.net/projects/markdown/\)](https://daringfireball.net/projects/markdown/) を参考にするのも良い。

文字飾り

太文字、斜体、取り消し線などの文字飾りを利用できます。

アスタリスクあるいはアンダーバー:

1つで*イタリック**, イタリック

2つで**太文字****, 太文字

3つで***イタリック太文字******, イタリック太文字

アスタリスクあるいはアンダーバー:

1つで*イタリック*, **イタリック** 2つで**太文字**, **太文字**

3つで***イタリック太文字***, ***イタリック太文字***

jupyter-notebook がサポートする Markdown 形式 (続き)

段落

段落の区切りは空行で分けられる。

最初の段落です。

次の段落です、直前の行は改行以外の文字が入ってはいけません。

jupyter-notebook がサポートする Markdown 形式 (続き)

改行

連続した行は、同じ行 (改行なし) とみなされる。改行したい場合は、行末に空白 2 つを入れる。空白だけだと可読性が落ちるので HTML の改行マーク `
` に代えても良い。

改行したくない場合。

2 行目

改行したい場合。

2 行目

改行したい場合 `
` の例。 `
`

2 行目

改行したくない場合。 2 行目

改行したい場合。

2 行目

改行したい場合 `
` の例。

2 行目

jupyter-notebook がサポートする Markdown 形式 (続き)

番号なしリスト (箇条書き)

- 行頭にハイフンでリストが始まります
 - 字下げでレベルを変えることができます
 - さらに字下げしてみました
 - * ハイフン以外に、*+ も使えます
 - + ほらこのとうり
- 行頭にハイフンでリストが始まります
 - 字下げでレベルを変えることができます
 - さらに字下げしてみました
 - ハイフン以外に、*+ も使えます
 - ほらこのとうり

jupyter-notebook がサポートする Markdown 形式 (続き)

番号つきリスト

1. 行頭に数字とピリオドで番号付きリストが始まります。
 1. レベルもサポートしています。
 2. 数字ではないかもしれません。
2. 字下げを戻せばリストレベルも戻ります。
 1. かならずしも連続した番号でなくてもかまいません。
 1. 全部同じ番号でも、直してくれます。
1. 行頭に数字とピリオドで番号付きリストが始まります。
 - A. レベルもサポートしています。
 - B. 数字ではないかもしれません。
2. 字下げを戻せばリストレベルも戻ります。
3. かならずしも連続した番号でなくてもかまいません。
4. 全部同じ番号でも、直してくれます。

jupyter-notebook がサポートする Markdown 形式 (続き)

引用

- > 電子メールの返信のように引用も可能です。
- >> 複数レベルにも対応しています。

- > レベルはちゃんと戻ります。

電子メールの返信のように引用も可能です。

複数レベルにも対応しています。

レベルはちゃんと戻ります。

jupyter-notebook がサポートする *Markdown* 形式 (続き)

見出し

見出しは # を使います。

```
# 見出し1
```

```
# 見出し2
```

```
## 見出し2.1
```

```
## 見出し2.2
```

見出し1

見出し2

見出し2.1

見出し2.2

jupyter-notebook がサポートする *Markdown* 形式(続き)

プログラムコード

コードは3つのバッククオート (逆引用符) `````で囲みます

```
``` print(10)
```
```

```
    print(10)
```

プログラム言語によっては、文字属性 (色) が設定できるものもあります。

```
```python print(10)
```
```

```
    print(10)
```


jupyter-notebook がサポートする Markdown 形式(続き)

表

表をこのように書くことができます。

```
This	is
a	table
```

| | |
|------|-------|
| This | is |
| a | table |

ハイパーリンク

外部参照は以下のように書きます。

[サンプル](http://www.example.com)

サンプル (<http://www.example.com>)

jupyter-notebook がサポートする Markdown 形式(続き)

LaTeX 数式

行中に LaTeX 形式の数式を入れるときはドル記号 $\$$ で囲みます。例えば、 $\$e^{i\pi}\$$ はこのように $e^{i\pi}$ になります。改行が必要な場合は、 $\$$ と繰り返します。

```
$$  
y = \int_{-\infty}^{\infty} f(x) \ dx  
$$
```

$$y = \int_{-\infty}^{\infty} f(x) dx$$

Markdown 形式まとめ

- Markup 言語の一つ
- HTML は広く普及しているが、ソースの可読性・書きやすさに難
- Blog などで簡単に使える Markup として創られた
- 実装に依存する部分が多い
- 本日の課題レポートの一つは **Markdown 形式**で提出する

課題(Markdown)

演習(Git)で作成したレポジトリを利用する。

レポジトリ内の jupyter-notebook 形式のファイル、`myclass.ipynb`を編集、リモートレポジトリにコミットする。後述するルールで学籍番号からの曜日(day)を求め、現在履修している講義の中でその曜日の最初の講義の概要を Markdown を用いて記述せよ。記述にあたっては、**4 種類**以上のマークダウン機能を利用すること。

自身の曜日に受講している講義がなければ翌日、翌々日...の講義を選ぶこと。

記述にあたっては UTAS のシラバスを参照してよいが、参照部分は Markdown の機能で明示すること。

記述量は PDF に変換した場合 A4 1 ページを目安とする。(環境によって失敗するのでPDF の変換を試す必要はない。)

曜日のルール:

| 学生証番号末尾 | 曜日 |
|---------|----|
| 0 or 5 | 月 |
| 1 or 6 | 火 |
| 2 or 7 | 水 |
| 3 or 8 | 木 |
| 4 or 9 | 金 |

その他：提出課題の末尾にこの講義（計算機システム概論）に期待する内容を追加しても良い。