情報学概論 2 第 4 回（5/15)

```
(Introduction to Informatics 2)

 ^                ^^           ^
```

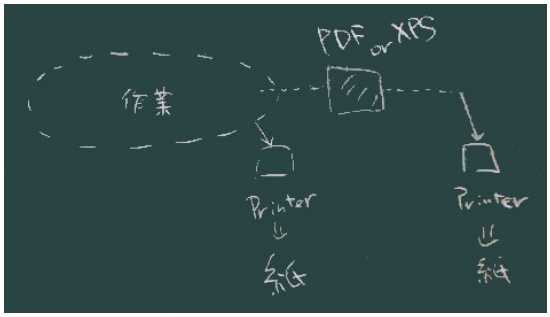
ファイル 種類 拡張子

1 PDF ファイル

a. PDF の概要と歴史

- PostScript 言語からスタートした（こういう解説的なページもある）。Adobe 社（もとは DTP 関連の会社、今はデザイン全般を広く扱う）
  - PostScript は、逆ポーランド記法のプログラミング言語（でもある）の代表格

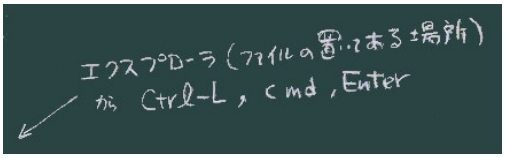
  - 現在の PDF にもその痕跡は残っているようだ。
  - 対抗するコンソーシアム主導の規格として XPS がある。

- PostScript （XPS でも）は、右図のように、コンピュータから直接プリンタ出力を行う代わりに、一旦この形式のファイルを作っておいて、そのファイルを（場合によってはネットや USB メモリ等で携行・転送して）別の場所で印刷する際にも使われる。
  - この形式だと、プリンタや PC 環境を問わずほぼ同質の印刷出力が得られることがメリット。

b. PDF を作る

- Typora の中から
  - （ファイルメニュー）エクスポート
  - （内部で pandoc を（うまく）使っているようだ。）
  - Pandoc を直接使う
  - 前々回資料も参照

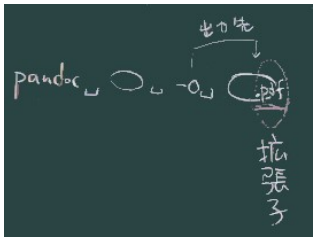
1. コマンドプロンプト起動 

- （何度かお伝えしているが）ソースファイルを置いた場所で開くこと。

2. pandoc コマンドを動かしてみる

`pandoc` ソースファイル名

3. 出力ファイル（と形式）を指定する



コマンドの文法

- リダイレクト記号 > で出力先指定できることは伝えたが、
- -o オプション で指定することで出力先の形式も同時に指定できるのでこちらがオススメ。
- コマンドプロンプト等の CUI を使う際の前提知識として、「空白区切り」つまり、（英語等の文章と同じように）空白が単語と単語の間の境目を示す記号として （暗黙の合意事項として）使われていることを認識しておくこと。
    - もしファイル名の中に空白文字（などの特殊な文字）を含んでいれば、 ファイル名をダブルクォート " " で区切れば大丈夫。
    - 名前に空白文字を含むファイル（ダミーでも）を作ってみて、そのファイルをエクスプローラからコマンドプロンプトにドラッグ＆ドロップしてみると システムがどう対処しているか少し理解できる。

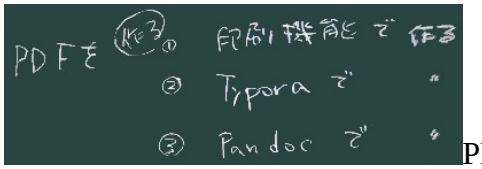
4. 変換コマンド

- Pandoc は PDF 生成部分には外部エンジンを借用するため、その外部エンジンを用意しておき、必要に応じて `--pdf-engine=...` オプションで指定することになる。

    - 現在の pandoc では platex がデフォルトになっているが、手軽でない、日本語表示でうまく動作しない等の問題があるようだ。
    - windows では wkhtmltopdf が手軽そうだ（まずこれをインストールしておこう）

      `pandoc` ソースファイル名 `-o` 出力先.pdf `--pdf-engine=wkhtmltopdf`

練習 1



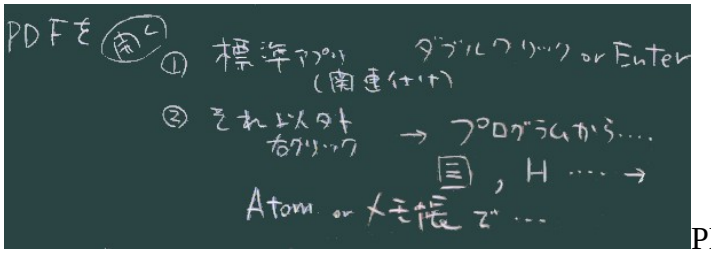
PDF 作成

- 右図のように ３種類の方法で PDF を作成してみて下さい（あとで比較もしてみますの

でそれぞれ別々のファイル名で）。

c. PDF を開く（プレビュー(?)）

- PDF を表示できるアプリケーションは、Adobe 社による Acrobat Reader / Adobe Reader 以外にも多数あり、
- Web ブラウザもその機能を内蔵（または拡張機能として装備）していることが多い。
- また、PDF を直接編集できるアプリケーションもある。
- （右下図）エクスプローラなどの画面上で当該ファイルをダブルクリック（または Enter　などの操作）で、　　標準の（自分の PC 上で 拡張子 .pdf に関連づけられた）アプリケーションが起動する。



PDF を開く操作

- 起動するアプリケーションを切り替えることもできる。方法としては（昨年の授業でも扱ったが）
    - コンテキストメニューから「プログラムから開く」で選択する、か、
    - あらかじめアプリケーションを開いておいて、「開く」メニューでファイルを見つける、か、
    - アプリケーション（の、開いた窓、或いはアイコン）の上に、ファイルをドラッグ＆ドロップする

  といった方法が使えるようにしておくこと。

d. PDF（の中）を眺めてみる

練習２

- 練習１で作成した PDF ファイルを、テキストエディタで開いてみる。

課題

PDF について 実習

1. 作る（練習１）
    - markdown から（できれば pandoc で）変換し、
    - どうやって変換したか報告下さい。
2. エディタで眺めてみる（練習２）
    - どのエディタを選び、
    - どんな方法で開いたか。
    - 開いて眺めてみてわかったこと

   について報告して下さい。

以下は来週扱います

（ここをクリックすると表示が切り替わります）

e. 拡張子が違ったら（または、ついていなければ）

- 上記の知識があれば推測可能
- アプリケーションでの推測
    - ポピュラーなのは WinExChange。
- 推測してくれるサイト
    - たとえばここ ファイル形式診断ツール
    - こういったサイトの信頼性は不明なので、重要なデータには使用しない方がいい）
- 拡張子のつけはずし の方法
- 関連づけされたアプリケーション、とそれ以外のアプリケーション（について おさらい）

2 DOCX ファイル

a. 作成

- typora のメニューから
- pandoc で

b. DOCX の実態

- （PDF と同様に）眺めてみる
    - よくわからないね
    - 実は…

c. テンプレートの活用（pandoc の小技）

1. テンプレートを作成

    ```
    pandoc --print-default-data-file reference.docx > reference.docx
    ```

2. word（など）でそのスタイルを編集

    ```
    reference.docx
    ```

    - GUI 的に開いても勿論かまわないが、ここから開くほうが手っ取り早いだろう
3. テンプレートを使用

    ```
    pandoc ソースファイル -o 出力先.docx --reference-doc reference.docx
    ```

    - オプション -reference-doc は、バージョンによる変化があったらしい。
    - 所定の場所に置いてデフォルトの（DOCX 生成時の）テンプレートにすることもできる。

3. 課題

- Typora で（エクスポート先として）扱える、または、 pandoc が（ソースまたは出力先として）扱えるファイル形式（多数ある）のうち、 目についてものを５つ（以上）選び、どんなファイル形式なのかを調べて、 まとめを作成してみてください。