**Program til at downloade pdf-filer fra links i en excel fil**

Med dette program kan du downloade store mængder af pdf-filer ved at specificere store et id, et url og evt. et alternativt url i en excel fil. Programmet producere også en excel fil med en oversig over, hvilke pdf-filer er downloadede, og hvilke ikke er.

**Repository struktur**

* **input:** Mappe til input data. Her skal Excel filen med listen af PDF’er der skal downlowdes gemmes. I det her tilfølde; ”GRI\_2017\_2020”.
* pdf\_list.csv: CSV fil der genereres af scriptet – indeholder samme information som input Excel-filen.
* **output:** Mappe til output fra python scriptet. Kommer til at indeholde en Excel-filen ”Download\_liste.xlsx”, som er en liste over alle filer specificeret i input filen, med en kolonne der viser om de er downloadede eller ej.
* Pdfs: Mappe til downloadede PDF’er.
* **src:** Mappe til Python scripts
* \_\_init\_\_.py
* download.py: Python script der downloader PDF-filer.

requirements.txt: txt fil med påkrævede python moduler for at køre scriptet.

Brugsvejledning: Brugsvejleding I word-fil i tilfæde til hvis brugeren ikke kan læse markdownen.

**Brug**

Modulerne anført i requirements.txt skal installeres, før scripts køres. Koden er skrevet til Python 3.11.1.

\_\_Download.py\_\_

Kør scriptet fra repository mappen for at downloade PDF’erne. Scriptet her 2 argumenter:

- \_-dv or --dev\_cheat: When the argument is called the code secret code will appear underneath the score display\_

- \_-c or --colour: The colours used in the game to make codes. Colours should be typed in separated by a space. Keep an eye on how you spell the colours - the way you type them in will be used as a reference for how the player is scored. You can pick as many or as few colours as you want!\_

Eksempel på kode, der kører scriptet fra terminalen:

```

python src/Master\_script.py -dv -t nr\_turns -c colour\_1 colour\_2 colour\_3

```

Uden argumenter:

```

python src/download.py

```

## Brug af terminalen på windows computer

## Sådan checker du hvor mange cores din computer arbejder med

**Designovervejelser**

* Programmet skal kunne bruges af folk uden meget teknisk viden eller erfaring.   
  Scriptet kan køres uden argumenter.   
  Har valgt at downloade lister over pdf-filer fra en .xlsx fil, i stedet for en .csv eller pickle fil, som ville have været hurtigere.

Programmet er ”chatty” – der bliver printet beskeder om, hvad der foregår i terminalen – og der er en form for improviseret progress bar.

Det meste dokumnetation er på dansk

ID’er der gentager sig fjernes.

* Det skal gå hurtigt!   
  Listen over pdf-filer bliver konverteret til en csv-fil før den loades ind til programmet – det går hurtigere.

Når filen loades ind, er det kun de tre nødvendige kolonner der bliver loaded.

Jeg har forsøgt at undgå forloops – når jeg kan her jeg erstattet dem med list-comprehensions.

Rækker med hverken 1. eller 2. url er droppet tidligt.

Hvis det 1. og 2. url er det samme, bliver det andet ikke testet

Hvis der er et 2. url, men ikke et 1. url, bliver det 2. prøvet by default.

Filer med samme BRnummer som filer der allerede er dawnloaded vil ikke blive downloaded igen.

Jeg har forsøgt at undgå if-statements

Hvis man fra uri - responsen kan se at det ikke er en pdf-fil, sorteres den fra.

Har forsøgt kun at importere de mest nødvendige functioner fra moduler.

Har forsøgt at undgå globale variabler

* Den skal kunne downloade en lang liste af PDF-filer fra en liste
* Den skal kunne bruges igen til andre projekter, af andre folk

Der er et argument der lader en skifte navn på den fil der skal læses. Antal cores kan også justeres.

* Gennemgå en liste af PDF-filer og downloade dem via en URL
* Bruge en backup URL hvis filen ikke kan downloades via den første
* Navngive de gemte filer efter en ID-kolonne (BRnum)
* Lave en liste over hvilke rapporter der er downloadede, og hvilke der ikke er
* Tjekke hvilke filer der allerede er downloadede, og springe dem over
* Programmet skal være robust nok til, at man kan køre det flere gange med en liste der kun er en smule ændret.
* Programmet skal kunne køres via command-linjen.

**Hvad kunne gøres bedre?**

* Programmet er stadig langsomt. (flere timer)
* Hvis man kører programmet igen med nogle flere filer i excel arket, vil programmet stadig forsøge at downloade de filer, som den fejlede med at downloade før, og som slap forbi filtrering.
* Progress baren viser ikke rigtigt hvor lang tid man kan forvente, at programmet kører – den er der mere for at vise at der stadig bliver gjort noget.
* Kunne måske have fundet en hurtigere måde at gemmme PDF’er på.
* Her måske ikke prioriteret min tid ideelt – har måske skåret nogle milisekunder af.
* Kunne have sørget for, at det ville være nok at skrive filnavnet uden .xlsx når man vælger fil.
* Hvis kunden ikke er teknisk kyndig kunne det være svært at finde ud af hvor mange cores en computer har.
* Endte med alligevel at have to argumenter.
* Min kode kunne se pænere ud?