INFO25 Seminar 4
Web Crawling

Today's Plan:
Today's seminar excercises will continue from last time where we extracted data from this wikipedia page: "https://simple.wikipedia.org/wiki/Abbey_Road". This time we will focuse more on how to deal with various types of links, as well as how to scrape data from multiple sources at once. Today's tasks are based on the material from *Lecture 4.*
I have uploaded the solutions from last seminar in case want to use that as inspiration or as a continuing point.

URL for the scraping tasks "https://simple.wikipedia.org/wiki/Abbey_Road".

Task 1:
Make a function that prints all links (*href attributes*) found in (a) tags on the webpage. Since you might get a KeyError you should include *Exception Handling* code for this. Example of Exception handling:

```
try:
    print(rows[1].text)
except IndexError as e:
    pass
```

Task 2:
From Task 1 you might notice that many of our links starts with *"/wiki/..."*. These are *internal/relative* wikipedia links that point to other wikipedia articles or resources. Make a function that only finds the *links that start with /wiki/.*

Task 3:
The links from task 2 are only relative URL's and can therefore not be visited right now with urlopen. We can solve this by making the links *absolute,* in other words giving them a full, well-formed URL that can be visited in a browser or through code. Improve the function from task 2 by using the *"urljoin*" library to *join the domain "https://en.wikipedia.org" with each of the links*. Make the function return a list of all the absolute urls it found.

```
from urllib.parse import urljoin
```

Task 4:
Write a function that *picks 10 random absolute links* (from task 3) and *visits them with urlopen*. Print the links you are visiting, one by one, as well as the <h1> header of each of those pages.

```
import random
```

Task 5:
Now lets put our knowledge together:
Make a function tha*t crawls all albums* that you can find on this wikipage: "https://simple.wikipedia.org/wiki/Category:The_Beatles_albums". Make the crawler *print the release date and duration of each albums.* Notice that this metadata is contained within the pages you get when you visit the album links.

Task 6:
If you have more time you can use it working on the next obligatory assignment, or alternatively you can give a go at the quizzes that are posted on the INFO215 mitt.uib.no page.