OOP

Assignment:	Continuous Assessment 3
Project Start Date:	22/02/2021
Assignment Compiler:	mikhail+oop_dipai_sept20@cct.ie
Weighting:	50% (OOP)
Due Date:	**23:59, 07/03/2021**
Method of Submission:	A **PDF** file submission through **Moodle** **only**
Late submission:	According to the accepted **CCT** guidelines
Last document update:	21/02/2021

Assignment Description

Create a script automation in **Python** for **Moodle** that allows adding lecture material links and links to individual class recordings into the correct module sections (weeks) on each run without duplications.

Requirements

- Assume that the script resides together with folders called **wk1**, **wk2**, **wk3**, ... **wkX** and in each folder there is an **index.html** (slides) file and **wkX.pdf** file that corresponds to a lecture given in class that week.
- The sample repository with these files is located here: **https://github.com/mikhail-cct/ca3-test**
- The end-point for the links is similar to: **https://mikhail-cct.github.io/ca3-test/wk1**, **https://mikhail-cct.github.io/ca3-test/wk1.pdf**
- You would have to work with a file system (**fs**) to find out which folders and files exist beside the script file and being able to read these.
- The script also needs to be able to read all sections of the **Moodle** module (through **local_wsmanagesections** plugin) page and parse these with **BS4** in order to compare these with the weeks that exist in the file system from the previous step.
- You need to be able to compare lists and add the missing lecture notes (links to the correct end-point and the corresponding pdf file) in the correct sections (you can assume week 1 is section 1, etc).
- Your script **cannot duplicate links**, so there must be a mechanism for verification of existing materials
- For the **recordings**, you need to scrape the following page in **Python**: **https://drive.google.com/drive/folders/1pFHUrmpLv9gEJsvJYKxMdISuQuQsd_qX**
- For **recordings**, the script needs to parse the **dates** in the section title (example: 28 September - 4 October) to know which week it represents. The example of this you can find in the repository listed above.
- You need **post links in an HTML format** that goes into each section: a link to **index.html**, a link to **wkX.pdf** and any corresponding **video recording links** (there can be multiple video files for each week).
- The script needs to scrape the page with these files, utilise **RegExp** to get **Google Drive video links** and names (example: \,\"**([^\"]*\.mp4)**\"), parse the names to find out the week these belong and put the correct links in the correct sections also verifying for existing files there to avoid duplication
- Please think of **the most elegant solutions** for all of the above problems.

Markings

[80%] The logic of the **script** code and how well the program is written in the context of **OOP** principles. The code will be evaluated against the following criteria: program **correctness**, **readability** (how clean is the code), **elegance** and the quality of the submitted **report** (see the instructions above).

[10%] Possess a coherent commit history on **GitHub** (at least 8 valid and logical commits) and has your final submission code present in your **Git** repository. All code needs to be properly commented and use references, where needed. You will be marked based on how well your code is documented (code comments) and whether your **Git** history reflects your real coding progress. Outside of special circumstances, any code submitted to **GitHub** after the deadline/extension – will not be marked.

[10%] You have to record a **video screencast (10 minutes max)** showing how your app works and how exactly you decided to go about making it. You should use your **CCT Google Drive** to store it and put the link into the final **PDF** submission. Make sure you set the **correct permissions for sharing** it or otherwise it will not be possible to see the video.

Markings

- There is only **one submission** (1 **PDF** report) and it should be uploaded through this **Moodle** page:
  https://moodle.cct.ie/course/view.php?id=1710

- There must include: your full name, your student number, your GitHub link and your CCT Google Drive video link.
  If any of these are missing – you might get 0% for your work.

- Please note that the **link to test Moodle instance might change** from time to time and the most recent link will be available on CCT Moodle (page referenced above). That page also has all other relevant materials, such as an **API key**, link to the **local_wsmanagesections** plugin and a **demo module page created for each student**, etc.

Extra points

Additional marks will be awarded, if you complete the following tasks:

- You will think of a clever mechanism to find out that the module runs over **2 semesters** and each semester has a separate repository for it
- Your script can put in multiple slides and pdfs in case there are **2 or more lectures in the same week**, where an extra lecture will have a letter at the end of its identification (example: wk1s, wk7b, etc). In case there are 3 lectures, then the primary lecture would be wkX and then the rest of the wkXa and wkXb, but sort these in alphabetical order.