

SEMAINE 1

Lundi 15 mai

Matin

Présentation avec l'équipe et avec le directeur du service. Visite des locaux, découvertes des différents projets que la MSHS créer/ étudie. Passage au service informatique pour récupérer un pc portable et un écran.

Après Midi

Début de l'installation, création du cahier des charges. Recherche sur ce qu'est Nakala, comment ça fonctionne de même pour Quarto. (Nakala accessible seulement avec des autorisations et une demande d'accès rattacher à l'établissement). Nakala est une base de données de recherche permettant aux chercheurs/ professeurs de déposer leurs projets. Le service Nakala offre deux niveaux de préservation :

- Un niveau par défaut mis en pratique dès lors qu'une donnée est enregistrée dans Nakala. La donnée est décrite, contextualisée et stockée de manière sécurisée.
- Un niveau avancé qui s'inscrit dans un partenariat avec le CINES. Dans ce circuit de dépôt avancé, la préservation à long terme est assurée par le CINES. Le CINES apporte des garanties supplémentaires telles que la valeur juridique du dépôt, l'intégrité du fichier, la migration des formats pour leur lisibilité, etc.

L'équipe aimerait que je crée un prototype de génération de site web statique à partir de Nakala pour permettre aux professeurs de déposer leurs projets

Quarto est un outil permettant de coder simplement le problème et qu'il n'y a quasiment pas de vidéo ou de cours en ligne pour sinicier.

Mardi 16 mai

Matin

Reprise de la recherche sur Quarto et Nakala. Installation de l'extension Quarto sur Vscode mais beaucoup d'erreurs, je n'arrive pas à exécuter mon code en .qmd aucune possibilité de débogage car il n'y a pas d'extension de débogage pour Quarto sur Vs code. Voici le message affiché quand j'essaie d'ouvrir mon fichier de code :

Cannot GET /venv/lib/test.qmd

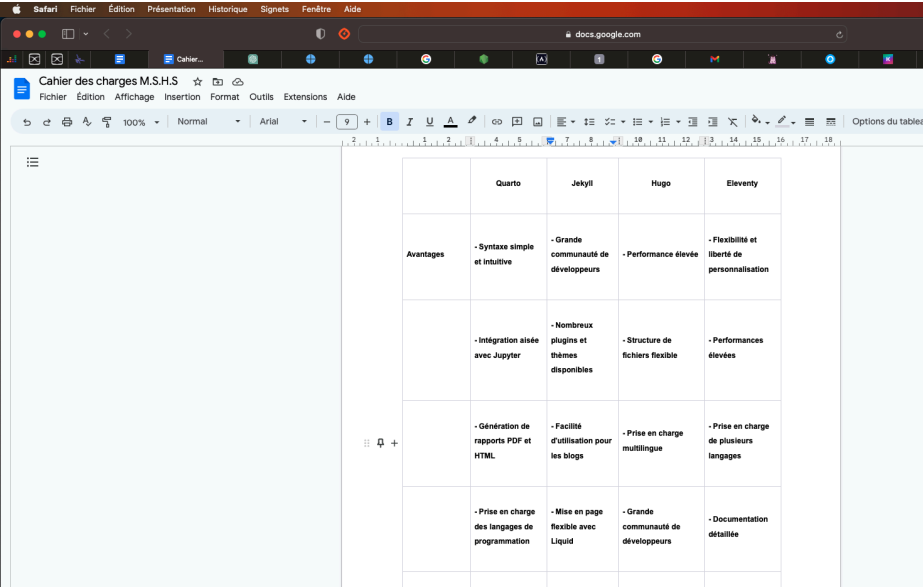
Fin de matinée

Réunion avec l'équipe pour parler des zones d'ombres sur le projet et planification du début du cahier des charges.

Solution à mon problème ci-dessus.

Après midi

Benchmarking sous forme de tableau des différents générateurs de site web statique disponible sur le marché.



	Quarto	Jekyll	Hugo	Eleventy
Avantages	- Syntaxe simple et intuitive	- Grande communauté de développeurs	- Performance élevée	- Flexibilité et liberté de personnalisation
	- Intégration aisée avec Jupyter	- Nombreux plugins et thèmes disponibles	- Structure de fichiers flexible	- Performances élevées
	- Génération de rapports PDF et HTML	- Facilité d'utilisation pour les blogs	- Prise en charge multilingue	- Prise en charge de plusieurs langages
	- Prise en charge des langages de programmation	- Mise en page flexible avec Liquid	- Grande communauté de développeurs	- Documentation détaillée

Avancement sur le cahier des charges avec mon tuteur et hypothèse de choix final pour l'outil à choisir.

Mercredi 17 mai

Matin

Modification du tableau de benchmarking car pas assez clair. Modifications du cahier des charges de quelques éléments avec mon tuteur.

Après midi

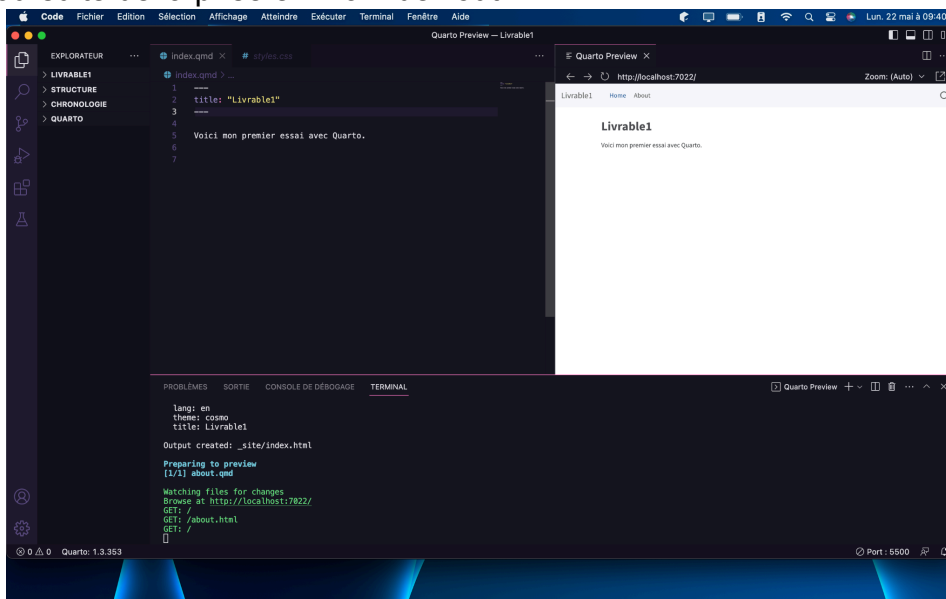
Premier contact avec l'outil Quarto. Visualisations de tuto en ligne et sur le site (quarto.org). J'ai commencé par essayer de voir ce qu'était le Markdown et comment l'outil l'utilise. j'ai ensuite observé les différents exemples de langages utilisables. Python est à mon avis le plus logique à utiliser. Pour finir j'ai observé différents exemples de projets réalisables avec Quarto. En fin d'après-midi j'ai complété le premier livrable de mon cahier des charges.

Semaine 2

Lundi 22 Mai

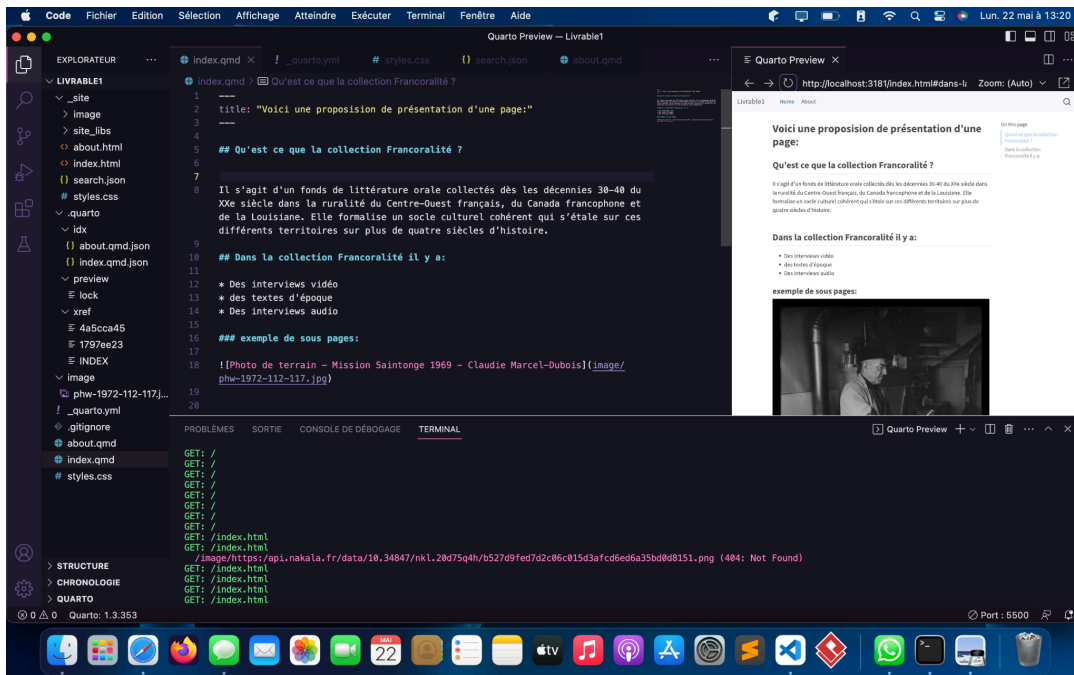
Matin

Début du livrable 2 consistant à créer une simple page en utilisant l'outil Quarto et poursuite de la prise en main de l'outil



Après midi

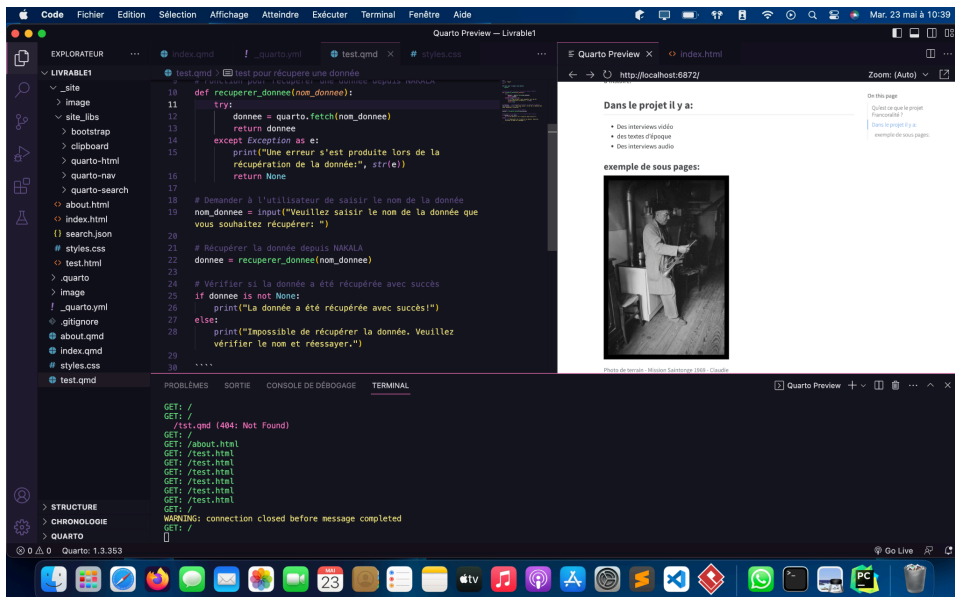
Confection d'une page dite "model" pour la suite.



Mardi 23 mai

Matin et apres midi

Conception d'une fonction python permettant de récupérer sur l'entrepôt de données nakala une image grâce à son DOI.



J'ai eu plein de soucis liés à l'installation de Python. Comme ci je ne l'avais pas installer alors que si.

Mercredi 24 mai

Matin

J'ai poursuivi dans la création de ma fonction python permettant de récupérer une image sur l'entrepôt de données. J'ai quelques soucis avec python sur ma machine. Cela m'a pris toute la matinée mais j'ai trouvé le problème: il fallait changer le chemin d'accès à python dans le bashrc.

Après-midi

Poursuite sur la fonction **download images from doi**.

En faisant une recherche d'exemple de script pour récupérer le contenu d'une page web en HTML je suis tombé sur un article de scraping avec python très intéressant qui m'a permis de bien comprendre comment je devais mettre en place ma fonction pour qu'elle soit le plus simple et efficace possible.

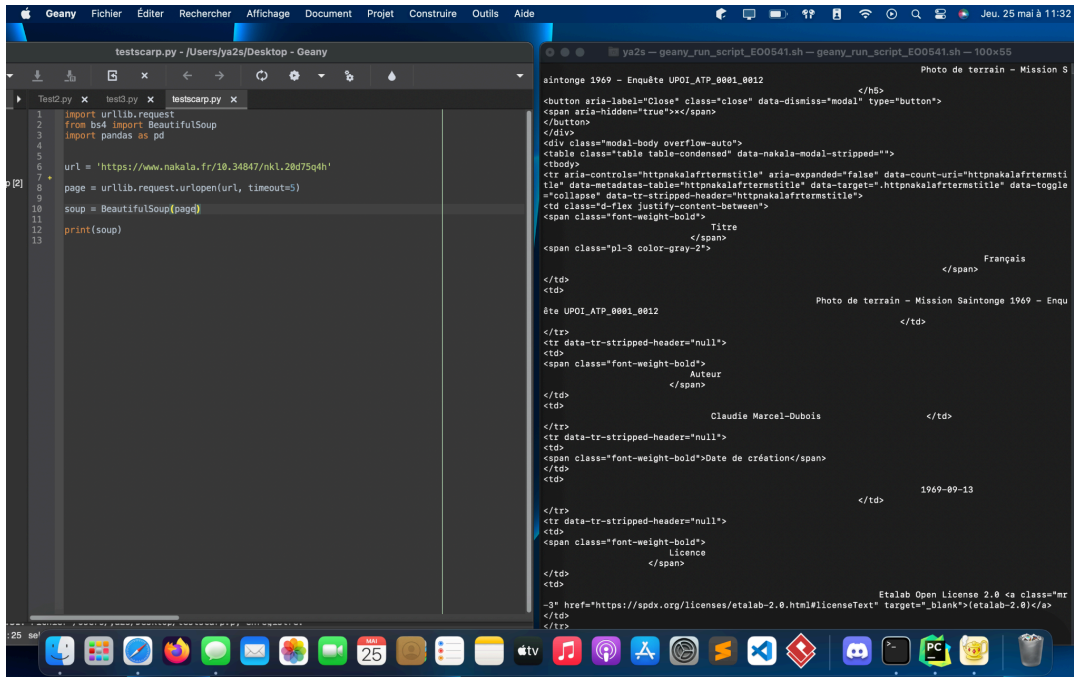
En fin d'après midi j'ai recherché les classes à récupérer dans nakala pour pouvoir les transférer dans mon site fait avec quarto. Je pense avoir récupéré les éléments les plus importants pour ne pas surcharger la page et le transfert donc j'ai noté:

- titre
- citer
- collection
- image
- dépositaire

Jeudi 25 mai

Matin

J'ai continué à regarder le scraping et à créer une page allant chercher tout ce qui est sur nakala grâce aux bibliothèques request, beautifulsoup et panda.



The screenshot shows a Geany IDE window with a Python script named 'testscarp.py' and a terminal window displaying the output of the script. The script uses 'requests' to fetch a page from 'www.nakala.fr' and 'BeautifulSoup' to parse the HTML. The terminal output shows the raw HTML of a page titled 'Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012', including a table with columns for 'Auteur' and 'Date de création'. The table contains one entry: 'Claudie Marcel-Dubois' and '1969-09-13'. The terminal also shows the license information for Etalab Open License 2.0.

```
1 import urllib.request
2 from bs4 import BeautifulSoup
3 import pandas as pd
4
5
6 url = 'https://www.nakala.fr/10_34847/nkl.28d75qdh'
7
8 page = urllib.request.urlopen(url, timeout=5)
9
10 soup = BeautifulSoup(page)
11
12 print(soup)
13
```

```
Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012
Français
Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012
Auteur
Claudie Marcel-Dubois
Date de création
1969-09-13
Licence
Etalab Open License 2.0
```

Après midi

Dans la suite de ce que j'ai fait ce matin je vais "filtrer" les éléments que je veux récupérer qui sont:

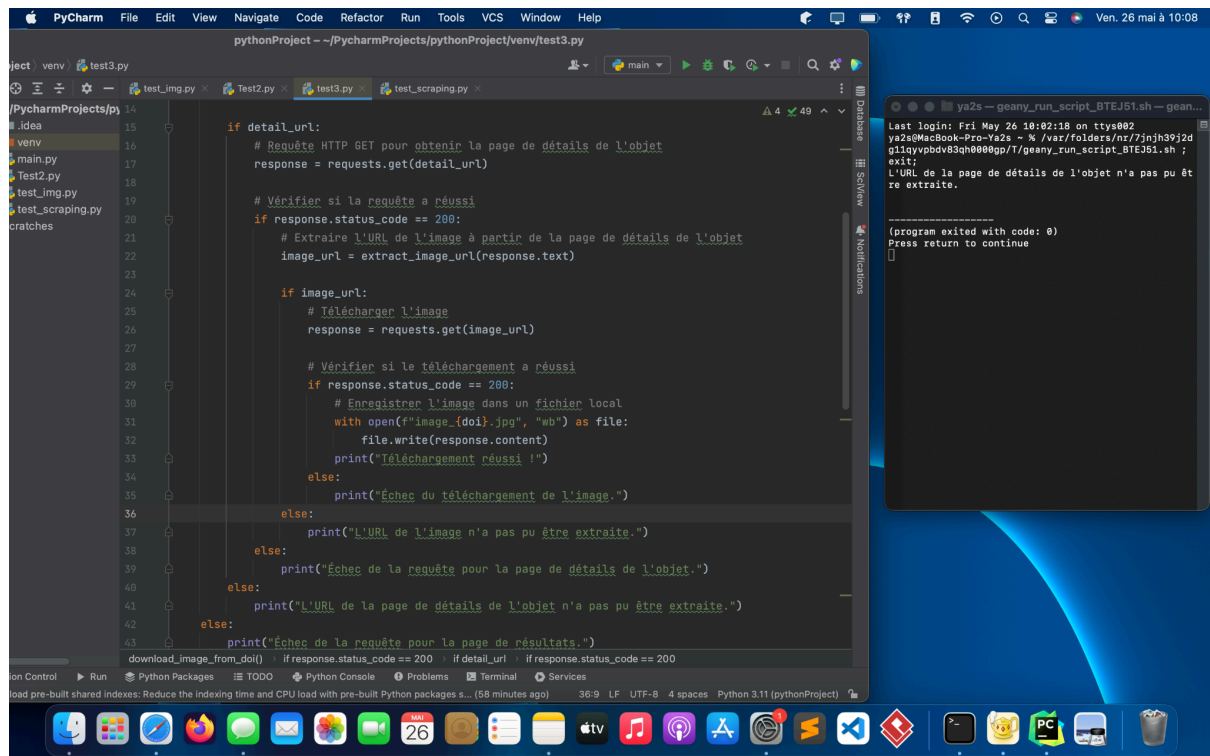
- le titre
- la description
- les images

Dans la fin d'après-midi mon tuteur m'a donné de la documentation pour l'API de NAKALA. Il m'a dit que c'était une bonne initiative de s'orienter vers le scraping mais que dans le projet il est plus logique et simple d'utiliser des scripts et des bibliothèques créés par l'équipe. Je repars donc de "zéro" sur le python alors que j'avais bien commencé.

Vendredi 26 mai

Matin

Nouvelle fonction python permettant de récupérer les informations d'une page de Nakala grâce à son url et son DOI. Je bloque car cela me sors un message d'erreur qui est : L'url de la page de détails de l'objet n' a pas pu être extraite.



```
pythonProject -- ~/PycharmProjects/pythonProject/venv/test3.py
venv test3.py
main
Python Console
Terminal
Services
download_image_from_doi() if response.status_code == 200 : if detail_url : if response.status_code == 200
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
if detail_url:
    # Requête HTTP GET pour obtenir la page de détails de l'objet
    response = requests.get(detail_url)

    # Vérifier si la requête a réussi
    if response.status_code == 200:
        # Extraire l'URL de l'image à partir de la page de détails de l'objet
        image_url = extract_image_url(response.text)

        if image_url:
            # Télécharger l'image
            response = requests.get(image_url)

            # Vérifier si le téléchargement a réussi
            if response.status_code == 200:
                # Enregistrer l'image dans un fichier local
                with open(f"image_{doi}.jpg", "wb") as file:
                    file.write(response.content)
                print("Téléchargement réussi !")
            else:
                print("Échec du téléchargement de l'image.")
        else:
            print("L'URL de l'image n'a pas pu être extraite.")
    else:
        print("Échec de la requête pour la page de détails de l'objet.")
else:
    print("L'URL de la page de détails de l'objet n'a pas pu être extraite.")
else:
    print("Échec de la requête pour la page de résultats.")
```

```
ya2s — geany_run_script_BTEJ51.sh — geany...
Last login: Fri May 26 10:02:18 on ttys002
ya2s@MacBook-Pro-Ya2s ~ % /var/folders/nz/7jnhj39j2d
g1laypbdv83qh000pp/T/geany_run_script_BTEJ51.sh ;
exit;
L'URL de la page de détails de l'objet n'a pas pu être
extraite.

(program exited with code: 0)
Press return to continue
```

En fin de matinée j'ai fait une réunion avec le reste de l'équipe pour voir où est ce que j'en étais sur le projet. Ils m'ont demandé de finir le prototype de site avec Quarto . Je l'ai déposé en début d'après-midi.


```
1 import markdown
2
3 dataIdentifier = "10.34847/nkl.2807504h"
4 targetProd = nkl.NKTargetSibaLpProd("ru", apiKey = "")
5 rs = nklID.get_datas(targetProd, dataIdentifier)
6
7 texte_markdown = ""
8
9 listecollect = rs.dictVals['collectionsIds']
10
11 for d in range(len(listecollect)):
12     texte_markdown += "# Liste collections:\n\n"
13     for f in range(len(listecollect[d])):
14         texte_markdown += "#- " + listecollect[d][f]['name'] + "\n\n"
15         texte_markdown += "#Description: " + listecollect[d][f]['description'] + "\n\n"
16         texte_markdown += "#Shot: " + listecollect[d][f]['shot'] + "\n\n"
17
18 listefich = rs.dictVals['files']
19
20 for f in range(len(listefich)):
21     f = listefich[f]
22     texte_markdown += "#- " + f['name'] + "\n\n"
23     texte_markdown += "#Description: " + f['description'] + "\n\n"
24     texte_markdown += "#Shot: " + f['shot'] + "\n\n"
25
26 listeimg = rs.dictVals
27
28 for i in range(len(listeimg)):
29     i = listeimg[i]
30     texte_markdown += "#- " + i['name'] + "\n\n"
31     texte_markdown += "#Description: " + i['description'] + "\n\n"
32     texte_markdown += "#Shot: " + i['shot'] + "\n\n"
33
34 # Convertir le texte Markdown en HTML
35 html = markdown.markdown(texte_markdown)
36
37 # Afficher le resultat
38 print(html)
```

voilà ce que cela renvoie :

```
1 <h2>Liste collections:</h2>
2
3 <ul>
4 <li>10.34847/nkl.frbady38r</li>
5 <li>10.34847/nkl.zefz26pp</li>
6 </ul>
7
8 <h2>Liste fichiers:</h2>
9
10 <ul>
11 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
12 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
13 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
14 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
15 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
16 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
17 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
18 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
19 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
20 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
21 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
22 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
23 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
24 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
25 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
26 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
27 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
28 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
29 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
30 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
31 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
32 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
33 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
34 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
35 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
36 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
37 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
38 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
39 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
40 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
41 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
42 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
43 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
44 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
45 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
46 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
47 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
48 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
49 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
50 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
51 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
52 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
53 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
54 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
55 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
56 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
57 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
58 <li>Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012</li>
59 </ul>
```

Il faut que je puisse mettre en relation mes scripts python et mon site quarto pour faire de l'auto génération de page.

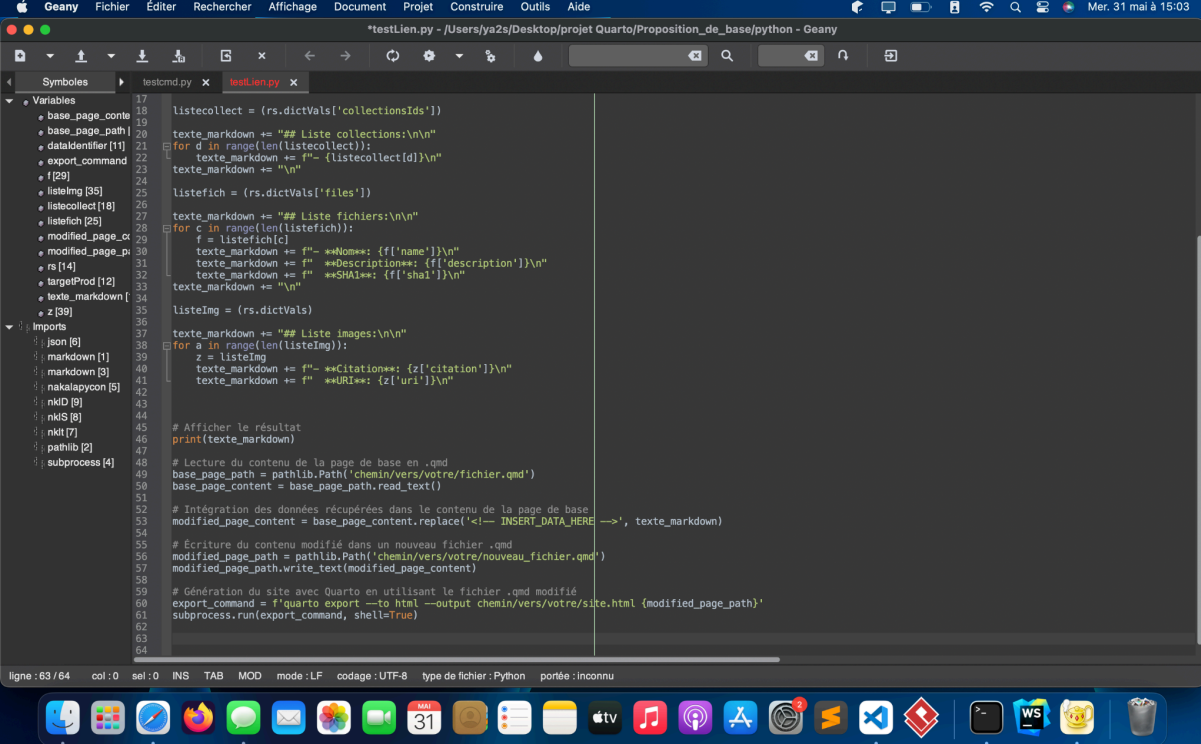
Mercredi 31 Mai

Matin

Dans la poursuite de ce que j'ai fait hier je vais faire en sorte d'automatiser le processus pour que mon script python soit directement injecté dans mon .qmd une fois transformé en markdown.

Après midi

Après avoir regardé des exemples sur internet et comment faire pour automatiser tout cela j'essaie d'optimiser mon code pour que tout cela soit fonctionnel.



```
17 listecollect = (rs.dictVals['collectionsIds'])
18
19 texte_markdown += "## Liste collections:\n\n"
20 for d in range(len(listecollect)):
21     texte_markdown += f"~ {listecollect[d]}\n"
22     texte_markdown += "\n"
23
24
25 listefich = (rs.dictVals['files'])
26
27 texte_markdown += "## Liste fichiers:\n\n"
28 for c in range(len(listefich)):
29     f = listefich[c]
30     texte_markdown += f"~ **Nom**~: {f['name']}\n"
31     texte_markdown += f"~ **Description**~: {f['description']}\n"
32     texte_markdown += f"~ **SHA1**~: {f['sha1']}\n"
33     texte_markdown += "\n"
34
35 listeImg = (rs.dictVals)
36
37 texte_markdown += "## Liste images:\n\n"
38 for a in range(len(listeImg)):
39     z = listeImg[a]
40     texte_markdown += f"~ **Citation**~: {z['citation']}\n"
41     texte_markdown += f"~ **URI**~: {z['uri']}\n"
42
43
44 # Afficher le résultat
45 print(texte_markdown)
46
47
48 # Lecture du contenu de la page de base en .qmd
49 base_page_path = pathlib.Path('chemin/vers/votre/fichier.qmd')
50 base_page_content = base_page_path.read_text()
51
52 # Intégration des données récupérées dans le contenu de la page de base
53 modified_page_content = base_page_content.replace('<!-- INSERT_DATA_HERE -->', texte_markdown)
54
55 # Ecriture du contenu modifié dans un nouveau fichier .qmd
56 modified_page_path = pathlib.Path('chemin/vers/votre/nouveau_fichier.qmd')
57 modified_page_path.write_text(modified_page_content)
58
59 # Génération du site avec Quarto en utilisant le fichier .qmd modifié
60 export_command = f'quarto export --to html --output chemin/vers/votre/site.html {modified_page_path}'
61 subprocess.run(export_command, shell=True)
62
63
64
```

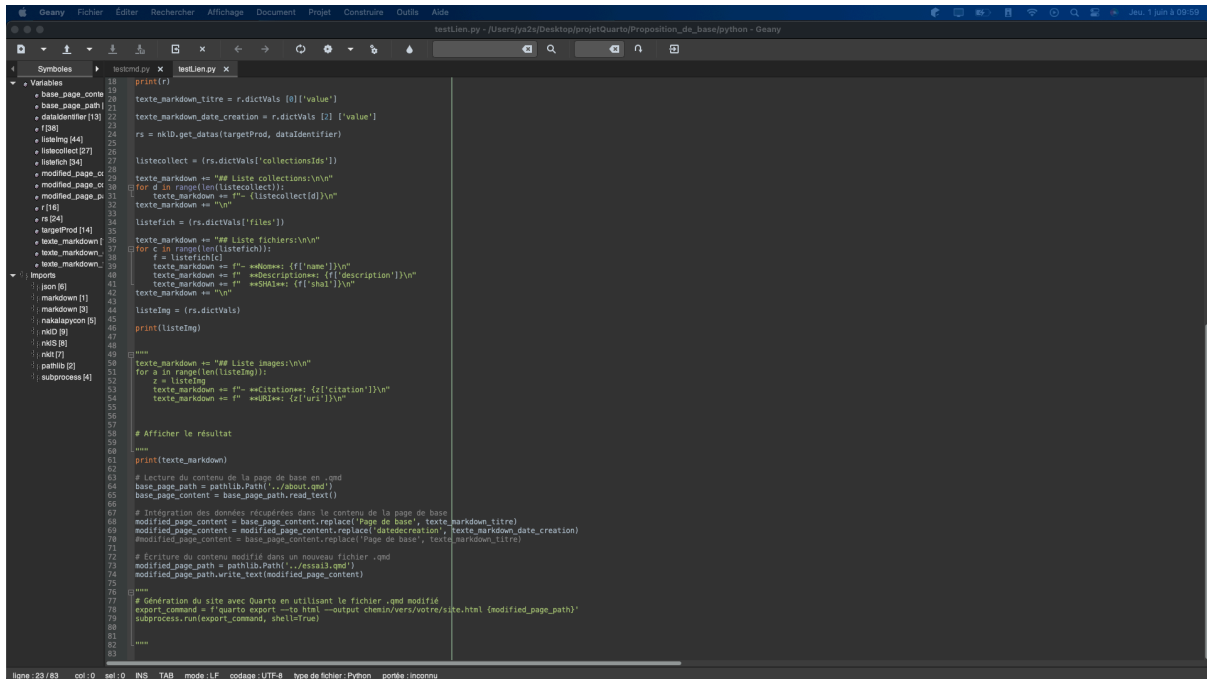
De ce que j'ai vu un utilisant la librairie pathlib et subprocess cela pourrait être fonctionnel. Je vais faire les tests nécessaires.

Dans la fin d'après midi j'ai vu avec mon tuteur et il m'a aidé à faire le remplacement et a consolidé ma base.

Jeudi 1 Juin

Matin

je reprends là où je me suis arrêté hier c'est-à-dire le processus de remplacement de la page de base.



```
18 print(r)
19
20 texte_markdown_titre = r.dictVals [0]['value']
21
22 texte_markdown_date_creation = r.dictVals [2] ['value']
23
24 rs = r.json()
25
26 listecollect = (rs.dictVals['collectionsIds'])
27
28 listefich = (rs.dictVals['files'])
29
30 texte_markdown = ""
31 for c in range(len(listecollect)):
32     texte_markdown += f"## Liste collections:\n\n"
33     for f in range(len(listefich)):
34         texte_markdown += f"  - {listefich[f]['name']}\n"
35         texte_markdown += f"    - {listecollect[c]['description']}\n"
36         texte_markdown += f"    - {listefich[f]['sha1']}\n"
37
38 listeing = (rs.dictVals)
39
40 print(listeing)
41
42 texte_markdown = ""
43 for i in range(len(listeing)):
44     texte_markdown += f"## Liste images:\n\n"
45     z = listeing[i]
46     texte_markdown += f"  - {z['citation']}\n"
47     texte_markdown += f"  - {z['url']}\n"
48
49
50 # Afficher le résultat
51
52 print(texte_markdown)
53
54 # Lecture du contenu de la page de base en .qmd
55 base_page_path = pathlib.Path("../about.qmd")
56 base_page_content = base_page_path.read_text()
57
58 # Intégration des données récupérées dans le contenu de la page de base
59 modified_page_content = base_page_content.replace('Page de base', texte_markdown_titre)
60 modified_page_content = modified_page_content.replace('datecreation', texte_markdown_date_creation)
61 modified_page_content = base_page_content.replace('Page de base', texte_markdown_titre)
62
63 # Écriture du contenu modifié dans un nouveau fichier .qmd
64 modified_page_path = pathlib.Path("../essai3.qmd")
65 modified_page_path.write_text(modified_page_content)
66
67
68
69
70
71
72 # Génération du site avec Quarto en utilisant le fichier .qmd modifié
73 export_command = f"quarto export --to html --output chemin/vers/votre/site.html {modified_page_path}"
74 subprocess.run(export_command, shell=True)
75
76
77
78
79
80
81
82
83
```

J'ai fait le code pour le remplacement de l'auteur, du DOI, de la date de création, du dépositaire.

Après-midi

Je poursuis sur ce que j'ai fait durant la matinée dans l'objectif de pouvoir intégrer les photos des pages.

Vendredi 2 Juin

Matin

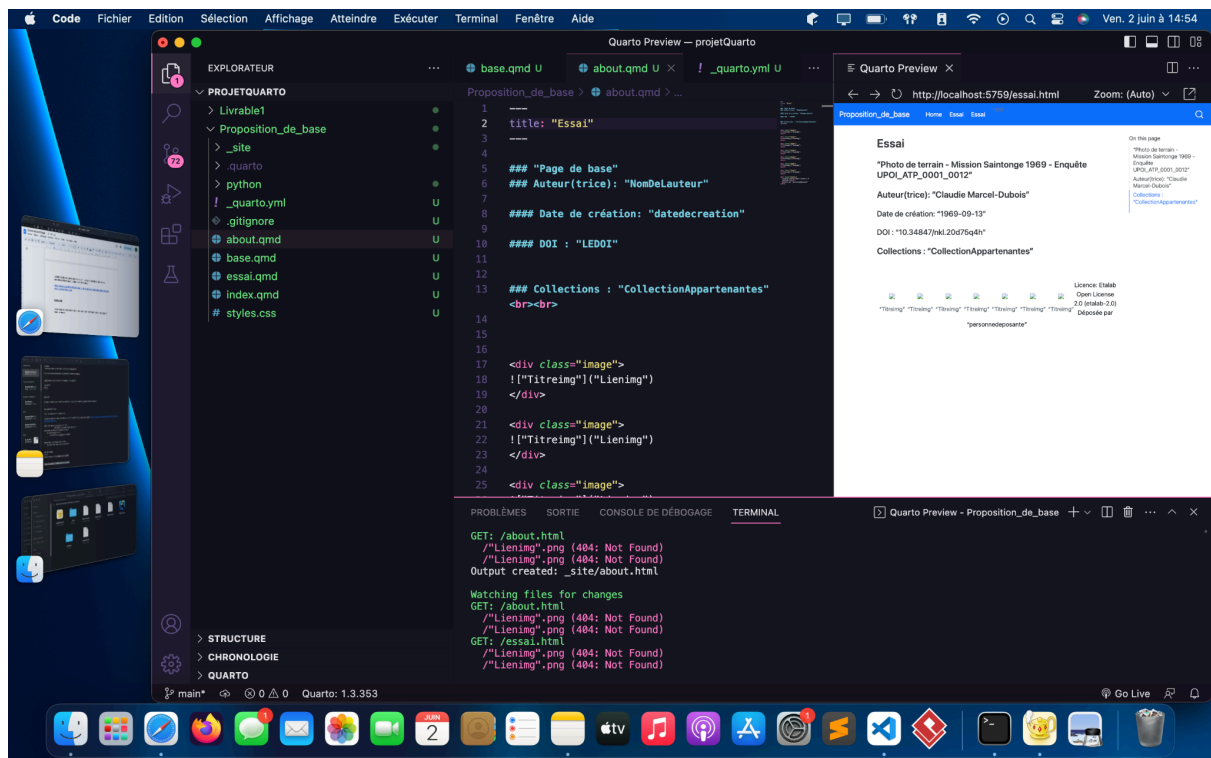
Je continue sur la récupération et l'insertion dans ma page de base. j'ai vu avec mon tuteur et il m'a expliqué comment fonctionnait l'irl d'une image sur nakala : <https://api.nakala.fr/iiif/doi/sha1/default.jpg>

Le DOI doit être récupérés et le sha1 aussi il est aussi possible de donner des paramètres d'affichage a la suite du SHA1 exemple :

https://api.nakala.fr/iiif/11280/111643de/b11ddf74a574b13ab04d790ebfd6d643db96e_aed/full/max/45/default.jpg

Après-midi

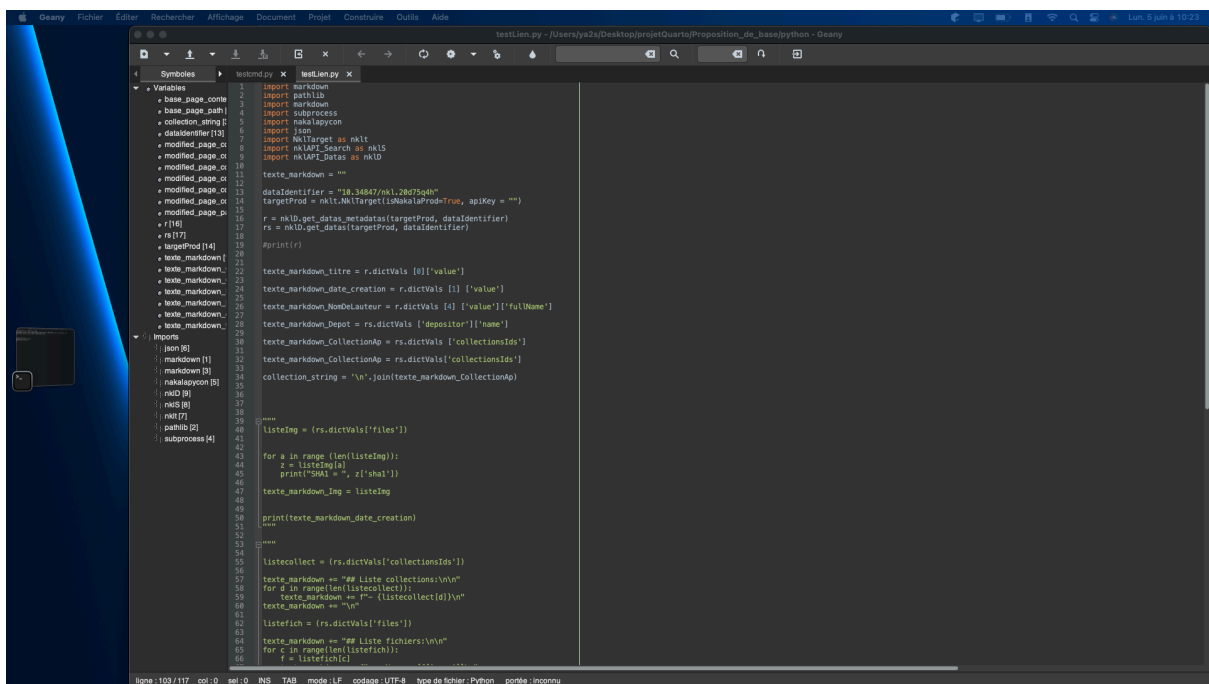
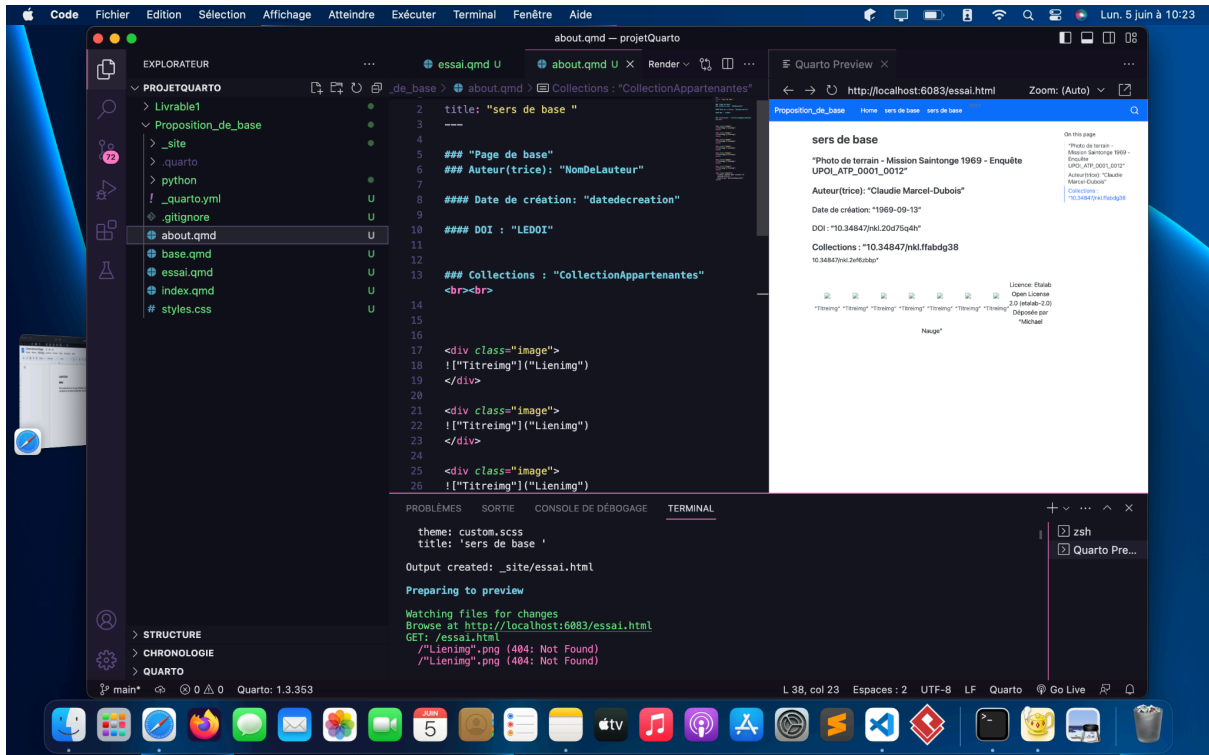
Je poursuis sur la recherche de la récupération et de l'insertion dans ma page de base. Je bloque :



Lundi 5 juin

Matin

Pour poursuivre sur ce que je faisais vendredi, j'ai repris et j'ai fait la récupération des collections appartenantes.



Je vais essayer de reprendre sur l'insertion des images.

J'ai réussi à récupérer et construire l'url de récupération d'image:

```
36
37 listefich = (rs.dictVals['files'])
38
39 for c in range(len(listefich)):
40     f = listefich[c]
41     print("https://api.nakala.fr/iiif/", dataIdentfier, "/", (f["sha1"]), "/default.jpg")
42
43
44
45 #url_img= "https://api.nakala.fr/iiif/", dataIdentfier, "/", sha1, "/default.jpg"
46
47 #print("_____ ", url_img)
48
```

Voilà ce que cela renvoie:

```
Last login: Mon Jun 5 10:46:00 on ttys011
ya2s@MacBook-Pro-Ya2s ~ % /var/folders/nr/7jnjh39j2dg11qyvbdv83qh000gp/T/geany_run_script_VJQT51.sh ; exit;
https://api.nakala.fr/datas/10.34847/nkl.20d75q4h/metadata
https://api.nakala.fr/iiif/ 10.34847/nkl.20d75q4h / b527d9fed7d2c06c015d3afcd6ed6a35bd0d8151 /default.jpg
https://api.nakala.fr/iiif/ 10.34847/nkl.20d75q4h / 384c5038e9ec76c17cbdb50b49e1e71ac0189d26 /default.jpg
https://api.nakala.fr/iiif/ 10.34847/nkl.20d75q4h / ea6eca0c44f080864f7fd57b49d0644bd143013c /default.jpg
https://api.nakala.fr/iiif/ 10.34847/nkl.20d75q4h / c4ddeb39b8e09b935c15f9c10735658a68e57cb9 /default.jpg

(program exited with code: 0)
Press return to continue
```

Après midi

Mon tuteur est venu voir comment est-ce que j'avancais. En vérifiant ce que j'avais fait, on s'est rendu compte que la façon que nous avons opté de procéder dans notre code nous limitais. En effet, en faisant de la récupération et du remplacement obligatoire le programme demande des arguments entrer en dur et donc il n'est pas automatisé. Voici comment nous avons commencé à changer le code :

```
# Lecture du contenu de la page de base en .qmd
base_page_path = pathlib.Path('../about.qmd')
base_page_content = ""
modified_page_content = ""

with open(base_page_path) as f:
    for line in f:
        if(line.find("Page de base")):
            modified_page_content += line.replace('Page de base', texte_markdown_titre)
        elif(line.find("NomDeLauteur")):
            print("a \n")
            modified_page_content += line.replace('NomDeLauteur', texte_markdown_NomDeLauteur)

        elif(line.find("datedecreation")):
            modified_page_content += line.replace('datedecreation', texte_markdown_date_creation)

        elif(line.find("LEDOI")):
            modified_page_content += line.replace('LEDOI', dataIdentfier)

        elif(line.find("CollectionAppartenantes")):
            for z in len(rs.dictVals['collectionIds']):
                modified_page_content += line.replace('CollectionAppartenantes', rs.dictVals['collectionIds'][z])

        else:
            modified_page_content += line
```

De cette façon si jamais le champs à remplacer n'est pas trouver dans la recherche il sortira le champs dit "de base"

Mardi 6 Juin

Matin

Problème sur le code, le changement de mon titre s'opère mais la suite ne fonctionne pas . Il ne rentre pas dans la suite (elif).

```
with open(base_page_path) as f:
    for line in f:
        if(line.find("Page de base")):
            modified_page_content += line.replace('Page de base', texte_markdown_titre)
        elif(line.find("NomDeLauteur")):
            print("a \n")
            modified_page_content += line.replace('NomDeLauteur', texte_markdown_NomDeLauteur)

        elif(line.find("datedecreation")):
            modified_page_content += line.replace('datedecreation', texte_markdown_date_creation)

        elif(line.find("LEDOI")):
            modified_page_content += line.replace('LEDOI', dataIdentifier)

        elif(line.find("CollectionAppartenantes")):
            for z in len(rs.dictVals['collectionIds']):
                modified_page_content += line.replace('CollectionAppartenantes', rs.dictVals['collectionIds'][z])

        else:
            modified_page_content += line
```

Voici la nouvelle version qui devrait mieux fonctionner :

```
with open(base_page_path) as f:
    for line in f:
        if "Page de base" in line:
            modified_page_content += line.replace('Page de base', texte_markdown_titre)
        elif "NomDeLauteur" in line:
            print("a \n")
            modified_page_content += line.replace('NomDeLauteur', texte_markdown_NomDeLauteur)
        elif "datedecreation" in line:
            modified_page_content += line.replace('datedecreation', texte_markdown_date_creation)
        elif "LEDOI" in line:
            modified_page_content += line.replace('LEDOI', dataIdentifier)
        elif "CollectionAppartenantes" in line:
            for z in range(len(rs.dictVals['collectionIds'])):
                modified_page_content += line.replace('CollectionAppartenantes', rs.dictVals['collectionIds'][z])
        else:
            modified_page_content += line
```

J'ai souvent des erreurs de ce type :

```
Last login: Tue Jun  6 10:59:17 on ttys001
ya2s@MacBook-Pro-Ya2s ~ % /var/folders/nr/7jnjh39j2dg11qyvpbdv83qh0000gp/T/geany_run_script_EWHN51.sh ; exit;
https://api.nakala.fr/datas/10.34847/nkl.20d75q4h/metadata
a

Traceback (most recent call last):
  File "/Users/ya2s/Desktop/projetQuarto/Proposition_de_base/python/testLien.py", line 62, in <module>
    for z in range(len(rs.dictVals['collectionIds'])):
KeyError: 'collectionIds'
```

Pourtant l'orthographe de collectionIds est respecté

Après midi

Rendez Vous médical extérieur

Mercredi 7 juin

Matin

Débogage de la fin de ma fonction pour permettre le bon fonctionnement de mon programme en réglant le soucis de collectionIds.

Soucis résolu l'erreur était dans l'orthographe collectionIds s'écrit collectionsIds. La nouvelle fonction que j'avais entrepris hier fonctionne. Je continue dans la poursuite du mon cahier des charges, la prochaine étape est de récupérer une donnée audio.

Voici tous les livrables que j'ai à réaliser sur le stage:

Livrable 1: benchmarking des outils et technologies disponibles

Livrable 2: premier pas avec l'outil (obtenir une chaîne de production minimale fonctionnelle)

Livrable 3: croquis d'une page de "base" complète (saisie manuelle d'une donnée stockée dans Nakala)

Livrable 4 A: publier une Data (de type image) en provenance de Nakala grâce à un script qui génère le fichier de base

Livrable 4 B: publier une Data (de type audio) en provenance de Nakala grâce à un script qui génère le fichier de base

Livrable 5: publier une Data complète (multi-types) dans une page (métadonnées, miniatures, liens)

Livrable 6: créer une page présentant une collection: métadonnées (éventuelles) et liste des données. Pour chaque donnée, faire une page présentant son contenu.

Livrable 7 : graphique analysant les données des collections

Pour l'instant je suis au livrable 4 A, je vais bientôt passer au 4B dans l'attente de la validation par mon tuteur.

Jeudi 8 Juin

Matin

Début de la recherche sur la récupération d'un fichier audio

Après-midi

Consultation de la documentation de quarto dans la continuité de ma recherche de ce matin.

Pour la récupération je dois utiliser mon code de base en changeant simplement le DOI (DataIdentifier) et de refaire une sélection comme fait pour le code de l'image.

Voici le code de récupération:

```
import nakalapycon
import json
import NklTarget as nklT
import nklAPI_Search as nklS
import nklAPI_Datas as nklD

dataIdentifier = "10.34847/nkl.be5b029k"
targetProd = nklT.NklTarget(isNakalaProd=True, apiKey = "")

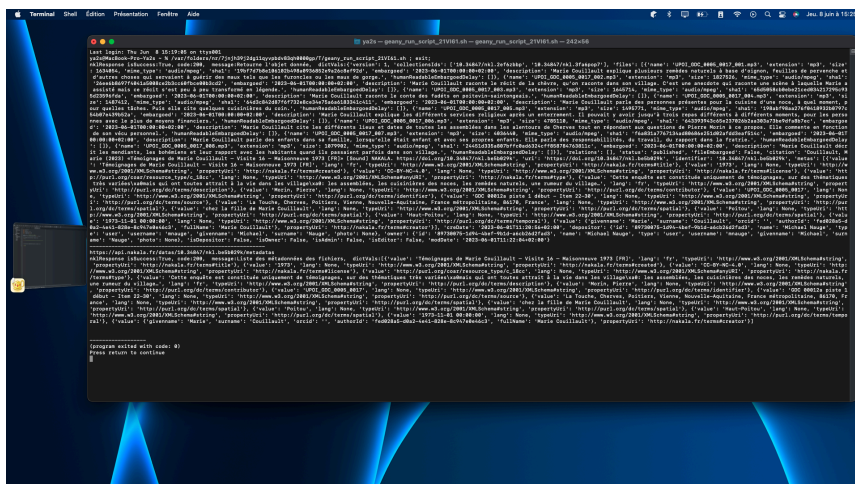
rs = nklD.get_datas(targetProd, dataIdentifier)
print(rs)

print("-----")

r = nklD.get_datas_metadatas(targetProd, dataIdentifier)

print(r)
```

Voilà ce que cela renvoie:



Amélioration du code de récupération :

```
import nakalapycon
import json
import NklTarget as nkl
import nklAPI_Search as nklS
import nklAPI_Datas as nklD

dataIdentfier = "10.34847/nkl.be5b029k"
targetProd = nkl.NklTarget(isNakalaProd=True, apiKey = "")

rs = nklD.get_datas(targetProd, dataIdentfier)
#print(rs)

listecollect = (rs.dictVals['collectionsIds'])
print("_____Liste collections:")

for d in range (len(listecollect)):
    print(listecollect[d])

    print("_____Liste fichiers")
    listefich = (rs.dictVals['files'])

    for c in range (len(listefich)):
        f = listefich[c]
        print(f['name'], " ", f['description'], " ", f["sha1"])

    print("_____")

    r = nklD.get_datas_metadatas(targetProd, dataIdentfier)
    for i in range (len (r.dictVals)):
        print(r.dictVals[i])

#print(r)
```

Voilà ce qu'il renvoie :

```
Terminal Shell Edition Présentation Fenêtre Aide
ya2s -- geany_run_script_BCPH01.sh -- geany_run_script_BCPH01.sh -- 270x72
Last login: Thu Jun 8 16:06:26 on tty001
ya2s@MacBook-Pro-Ya2s: ~ % (var/folders/nr7jnjh59j2g1qyvpbdv43qh000gq//geany_run_script_BCPH01.sh: exit:
10.34847/nkl.2ef62b0p
10.34847/nkl.19f40907

_____Liste Fichiers
UPOI_000_0007_001.mp3 Marie Couillault explique plusieurs remèdes naturels à base d'origan, feuilles de pervenche et d'autres choses qui servaient à guérir des maux tels que les furoncles ou les maux de gorge. 10f767b1861802b4f8b09368520e2
6c8e9f2d 2ee6b8697f481a580
UPOI_000_0007_002.mp3 Marie Couillault raconte le récit de la chèvre, qu'on raconte dans son village. C'est une anecdote qui raconte une scène à laquelle Marie a assisté mais ce récit s'est peu à peu transformé en légende. 66d08d88b21cedb3a21726c93d2359fda
8c23cc0c0fbc0e03c0c
UPOI_000_0007_003.mp3 Marie Couillault raconte le conte des fadaés en poitevin-saintongeais. 45d08d88b21cedb3a21726c93d2359fda
UPOI_000_0007_004.mp3 Marie Couillault parle des personnes présentes pour la cuisine d'une noce, à quel moment, pour quelles tâches. Puis elle cite quelques cuisinières du coin. 64d3c942d87f6f732e8ce3475a66183341c11
UPOI_000_0007_005.mp3 Marie Couillault explique les différents services religieux après un enterrement. Il pouvait y avoir jusqu'à trois repas différents à différents moments, pour les personnes avec le plus de moyens financiers. 198abf98aa27
d6c4c893007f5c0d76a3302a
UPOI_000_0007_006.mp3 Marie Couillault cite les différents lieux et dates de toutes les assemblées dans les alentours de Cherves tout en répondant aux questions de Pierre Morin à ce propos. Elle commente en fonction de son vécu personnel. 643
30794d0c027816e1a30e39e9f8ab7ec
UPOI_000_0007_007.mp3 Marie Couillault parle des enfants dans sa famille, lorsqu'elle était enfant et avec ses propres enfants. Elle parle des responsabilités, du travail, du rapport dans le foyer. 6e831a776734ad8086e251d82afd3ae7514c
UPOI_000_0007_008.mp3 Marie Couillault décrit les mendiants, les bonheurs et leur rapport avec les habitants quand ils passaient parfois dans son village. 2445d335a8070f0c0ad6324cf058704763811c

https://api.nakala.fr/datas/10.34847/nkl.be5b029k/metadatas
{
  "value": "Témoignage de Marie Couillault - Village 16 - Saisonneuve 1973 [FR]", "lang": "fr", "typeURI": "http://www.w3.org/2001/XMLSchema#string", "propertyURI": "http://nakala.fr/terms#title"}
{
  "value": "1973", "lang": None, "typeURI": "http://www.w3.org/2001/XMLSchema#string", "propertyURI": "http://nakala.fr/terms#created"}
{
  "value": "CC-BY-NC", "lang": None, "typeURI": "http://www.w3.org/2001/XMLSchema#string", "propertyURI": "http://nakala.fr/terms#license"}
{
  "value": "http://purl.org/coar/resource_type/c_18cc", "lang": None, "typeURI": "http://www.w3.org/2001/XMLSchema#anyURI", "propertyURI": "http://nakala.fr/terms#type"}
{
  "value": "Cet(te) élément est constitué(e) uniquement de témoignage(s) sur des évenements très variés/épisodiques qui ont toutes attiré à la vie dans les villages/les assemblées, les cuisinières des noces, les remèdes naturels, une rumeur du village.", "lang": "fr", "typeURI": "http://www.w3.org/2001/XMLSchema#string", "propertyURI": "http://purl.org/dc/terms/description"}
{
  "value": "Marie Pierre", "lang": None, "typeURI": "http://www.w3.org/2001/XMLSchema#string", "propertyURI": "http://purl.org/dc/terms/creator"}
{
  "value": "UPOI_000_0007_001", "lang": None, "typeURI": "http://www.w3.org/2001/XMLSchema#string", "propertyURI": "http://purl.org/dc/terms/identifier"}
{
  "value": "000_0002a Dicit à dicit - l'écrit 22-08", "lang": None, "typeURI": "http://www.w3.org/2001/XMLSchema#string", "propertyURI": "http://purl.org/dc/terms/source"}
{
  "value": "Le Touche, Cherves, Poitiers, Nieme, Nouvelle-Aquitaine, France métropolitaine, 86070, France", "lang": None, "typeURI": "http://www.w3.org/2001/XMLSchema#string", "propertyURI": "http://purl.org/dc/terms/spatial"}
{
  "value": "chez la fille de Marie Couillault", "lang": None, "typeURI": "http://www.w3.org/2001/XMLSchema#string", "propertyURI": "http://purl.org/dc/terms/spatial"}
{
  "value": "Poitou", "lang": None, "typeURI": "http://www.w3.org/2001/XMLSchema#string", "propertyURI": "http://purl.org/dc/terms/spatial"}
{
  "value": "Haut-Poitou", "lang": None, "typeURI": "http://www.w3.org/2001/XMLSchema#string", "propertyURI": "http://purl.org/dc/terms/spatial"}
{
  "value": "1973-11-01 00:00:00", "lang": None, "typeURI": "http://www.w3.org/2001/XMLSchema#string", "propertyURI": "http://purl.org/dc/terms/temporal"}
{
  "value": "({\"givenname\": \"Marie\", \"surname\": \"Couillault\", \"orcid\": \"\", \"authorId\": \"fed28a5-0ba2-4a41-826e-8c947e8e4c3\", \"fullName\": \"Marie Couillault\", \"propertyURI\": \"http://nakala.fr/terms#creator\"})"}
}

(program exited with code: 0)
Press return to continue
$
```

Je pense qu'il faut récupérer l'URI et l'incruster dans mon autre code.

Vendredi 9 Juin

Matin

Dans la poursuite de ce que j'ai fait hier voici le code ajouter à ma fonction ce matin pour récupérer les données audio:

```
# On peut aussi traiter des fichiers son, si on en a
elif line.find("<div audio>") != -1:
    for i in range(len(sons)):
        Output_Text += line.replace("TitreSon", sons[i]["description"]).replace("URLAudio", baseURI + dataDOI + "/" + sons[i]["SHA1"])
```

qui remplacera dans mon .qmd les champs suivants:

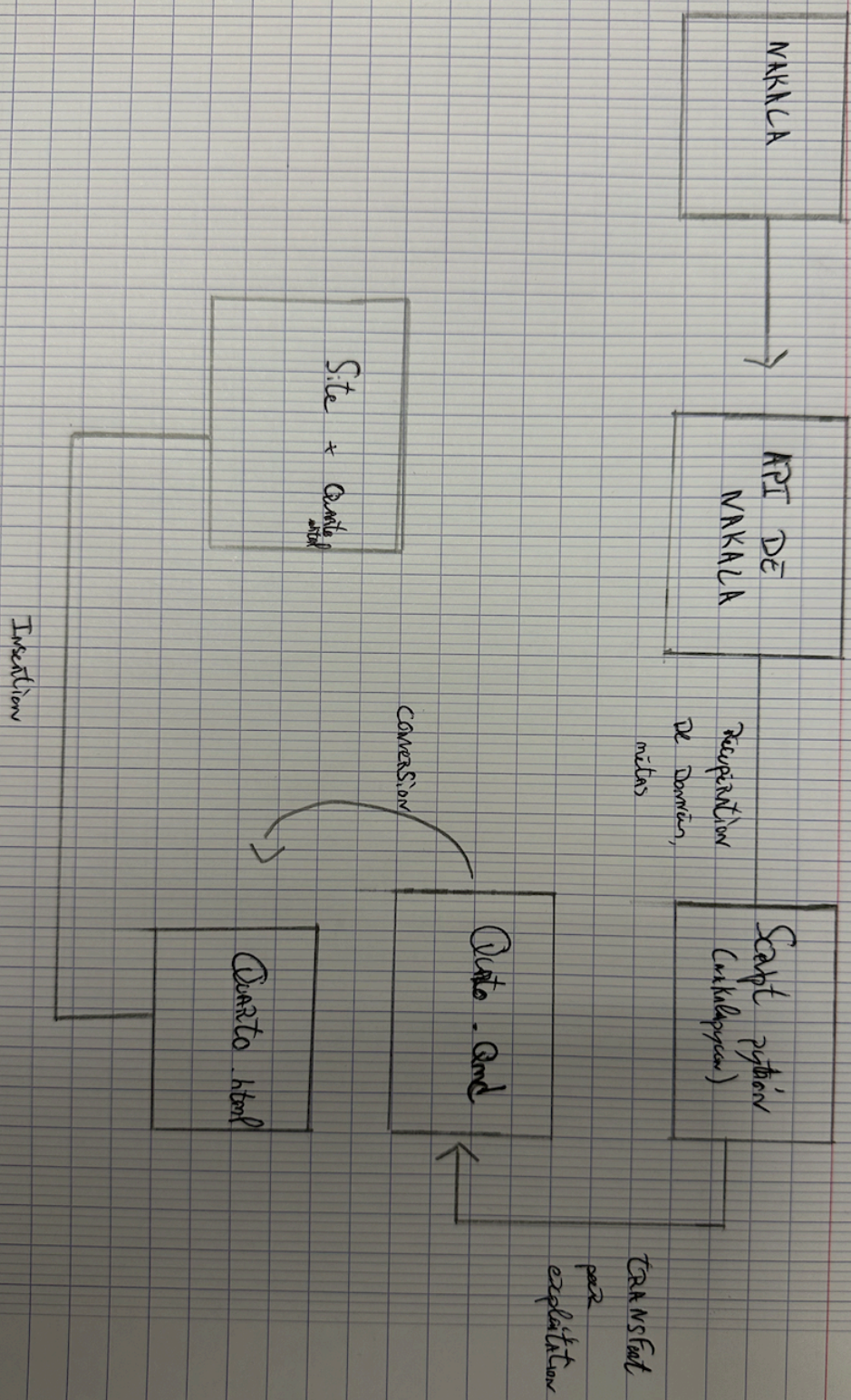
```
<div audio>![TitreSon] (URLAudio)</div>
```

Fin de matinée

Avec mon tuteur nous avons discuté des 4 semaines de stages et il m'a donné quelques conseils d'organisation. Pour bien tout saisir nous avons repris à la base, non pas car le code ne fonctionnait pas mais car il souhaité que je comprenne bien tout des réponses renvoyées par L'api de nakala.

Après midi

À la demande de mon tuteur j'ai effectué un schéma pour représenter le projet que je réalise et une explication de mon code python en algo:



Script de client:

importation de différents éléments.

création d'une requête de connexion à l'API de NAKAL.

~~Exécution~~

Initialisation des deux DOI (Image, son)
Initialisation de l'URI de base.

utilisation de la requête de connexion

Exécuter dans un if :

si la connexion a ~~pas~~ échoué

↳ message d'erreur

↳ déconnexion brutale

Affectation :

data Fils est égale au résultat de Fils dans RC. DictVals
data Mots est égale au résultat de Mots dans RC. DictVals

Initialisation de Output text.

Fonction getmots by property

à l'aide d'un paramètre nombre pour l'itérateur (prop)
et d'un champ les accès nombre pour l'itérateur (field),
ce programme va itérer ces liste à vide val.

Exécuter à l'aide d'une structure itérative (for) :

↳ recherche de les valeurs de prop

↳ si field ≠ ""

↳ recherche à l'aide du field ^{à l'aide de} \rightarrow val \rightarrow generateur

↳ affectation \rightarrow val

Structure de fichiers Abstrait en lecture

Pour chaque ligne dans F:

Soit une ligne ("LeTitre") est trouvée:

↳ remplacement de ("LeTitre") à l'aide de la fonction

Soit si une ligne ("Nom de l'auteur")

↳ remplacement de ("Nom de l'auteur") à l'aide de la fonction

↳ seule si jamais il y a plusieurs auteurs

Soit si une ligne ("Nom de l'éditeur")

↳ remplacement à l'aide de la fonction

↳ seule si jamais il y en a plusieurs

Soit si une ligne ("LeDOI")

↳ remplacement ("LeDOI") à l'aide de la fonction

Soit si une ligne ("Collection ID")

↳ remplacement grâce à la fonction

↳ seule si il y en a plusieurs

Soit si une ligne ("<div class='img'> <div>")

↳ seule si plusieurs images

↳ si dans une ligne ("image")

↳ remplacement de ("URLImage") par un autre URL avec des paramètres d'affichage.

Soit si une ligne ("<div class='video'>")

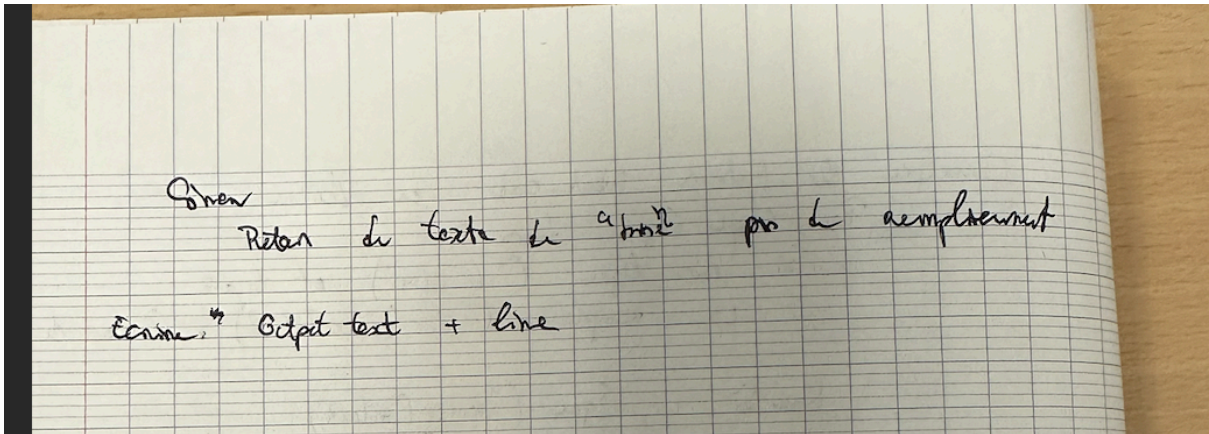
↳ seule si plusieurs vidéos

↳ si dans une ligne ("video")

↳ remplacement ("URLVideo") par le vrai URL

Soit si une ligne ("<div class='foten'>")

↳ remplacement à l'aide de la fonction



Lundi 12 Juin

Matin

Reprise sur mon code python, amélioration pour qu'il soit fonctionnel dans toutes les situations donc changement dans le code et dans le .qmd :

```
def getMetasByProperty(prop, field):
    """
    Parameters
    ---
    prop : String
    PropertyURI
    field : String
    subfield we want the value.

    Returns
    ---
    List of String.
    """
    val = []
    for i in range(len(dataMetas)):
        if dataMetas[i]["propertyUri"] == prop:
            if field != "":
                val.append(dataMetas[i]["value"][field])
            else:
                val.append(dataMetas[i]["value"])
    return val

with open("../about.qmd", "r", encoding="utf-8") as f:
    for line in f:
        if line.find("LeTitre") != -1:
            Output_Text += line.replace("LeTitre", getMetasByProperty('http://nakala.fr/terms#title', '')[0])
        elif line.find("NonDeLauteur") != -1:
            Auteurs = getMetasByProperty('http://nakala.fr/terms#creator', 'fullName')
            for i in range(len(Auteurs)):
                Output_Text += line.replace("NonDeLauteur", Auteurs[i])
        elif line.find("NonContributeur") != -1:
            Contrib = getMetasByProperty('http://purl.org/dc/terms/contributor', '')
            for i in range(len(Contrib)):
                Output_Text += line.replace("NonDuContributeur", Contrib[i])
        elif line.find("LED01") != -1:
            Output_Text += line.replace("TheDOI", dataDOI)
        elif line.find("CollectionId") != -1:
            for i in range(len(rc.dictVals["collectionsIds"])):
                Output_Text += line.replace("CollectionId", rc.dictVals["collectionsIds"][i])
        elif line.find("<div class='image'>") != -1:
            for i in range(len(dataFiles)):
                if dataFiles[i]["mime_type"].find("image") != -1:
                    Output_Text += line.replace("TitreImage", dataFiles[i]["description"]).replace("URLImage", baseURI + "iiif/" + dataDOI + "/" + dataFiles[i]["sha1"] + "/full/200,200/0") # /default.jpg
        elif line.find("<div class='audio'>") != -1:
            for i in range(len(dataFiles)):
                if dataFiles[i]["mime_type"].find("audio") != -1:
                    Output_Text += line.replace("TitreSon", dataFiles[i]["description"]).replace("URLAudio", baseURI + dataDOI + "/" + dataFiles[i]["sha1"] ) # + "." + dataFiles[i]["extension"]
        elif line.find("<div class='footer'>") != -1:
            Output_Text += line.replace("TheDeposit", rc.dictVals["depositor"]["name"]).replace("DepositDate", rc.dictVals["creDate"]).replace("TheLicence", getMetasByProperty('http://nakala.fr/terms#license', '')[0])
        else:
            Output_Text += line
```

Le code fonctionne mais tous les champs ne sont pas remplacés, je dois faire des vérifications dans l'API search .

```
1 ----
2 title: "LeTitre"
3 ----
4
5 # Metadata
6 datedecreation
7
8 ## Auteur(trice)
9 NomDeLauteur
10
11 ## Contributeur(trice)
12 NomContributeur
13
14 ### DOI : LEDOI
15
16
17 ### Appartient aux collections:
18 > CollectionId'| <br><br>--
19
20 # Files
21 <div audio>![TitreSon](URLAudio)</div>
22
23 <div class="image">
24 ![TitreImage](URLImage)
25 </div>
26
27
28 <div footer>TheDepositor TheLicence</div>
```

Après midi

Poursuite sur l'amélioration de mon code , erreur de data non récupérer régler.
Début de la recherche sur la récupération de fichier audio.

Mardi 13 juin

Maladie

Mercredi 14 juin

Maladie

Jeudi 15 juin

Matin

Reprendre sur la récupération des fichiers audio : recherche dans l'api de nakala à l'ai toujours d'un DOI.

Voilà ma fonction de récupération d'audio:

```
if line.find("<div class='audio'>") != -1:
    for i in range(len(dataFiles)):
        if dataFiles[i]["mime_type"] == "audio/mpeg":
            Output_Text += line.replace("TitreSon", dataFiles[i]["description"]).replace("URLAudio", baseURI + dataDOI + "/" + dataFiles[i]["sha1"] ) + ".mp3" + dataFiles[i]["extension"]
            #audio_element = f'![[{dataFiles[i]["description"]}][URLAudio].replace("URLAudio", f'({baseURI}{dataDOI}/{dataFiles[i]["sha1"]}')]
            #Output_Text += line.replace("TitreSon", URLAudio) + audio_element
```

Et voici ce que cela renvoie dans un terminal :

```
<div class="audio">[[On entend Hilaire Bonnin qui chante l'air, mais l'air est difficile à déterminer. Lorsque Marie Couillault se joint à lui, l'air est clair. Lorsqu'elle a fini, Hilaire et Marie font comprendre que l'air se joue deux fois complètes. On entend la voix de Jacques Bonnin (membre des Gens de Cherves).]](https://api.nakala.fr/10.34847/nkl.b7bd3k55/296af2ad54846090e38aa76a2647d8ce78841bd4/d5172eaca2df509cfd192fd355cae2b0073b78e)
</div>
<div class="audio">[[Marie Couillault chante l'air, elle précise à la fin qu'il faut le faire deux fois.]](https://api.nakala.fr/10.34847/nkl.b7bd3k55/7e03c963f0a5f4bb432a9e731a0d2377a8958ff9)
</div>
<div class="audio">[[Les informateurs discutent à propos de la manière de danser les quadrilles.]](https://api.nakala.fr/10.34847/nkl.b7bd3k55/7d32fcd830ac567e0145bd32ebbf0e9a8c8b1a)
</div>
<div class="audio">[[Suite à la demande de l'enquêtrice, Marie Couillault chante l'air de la berline, elle précise que ça se dansait avec un pas de polka avant de reprendre l'air une deuxième fois.]](https://api.nakala.fr/10.34847/nkl.b7bd3k55/ccd2a70e0ef0ac8f0ab0293c34213498e4e9e32a)
</div>
<div class="audio">[[Plusieurs témoins et des gens de Cherves dansent la danse à l'ours qu'ils chantent en même temps. Marie Couillault confirme.]](https://api.nakala.fr/10.34847/nkl.b7bd3k55/62bae0e9c3c9db6428b1f07d32ae0080880c9)
</div>
<div class="audio">[[Marie Couillault chante l'air de Jean Brisquet et avec les autres témoins, elle explique la danse. Les explications sont partielles à cause de coupe de la bande et du brouhaha par moment mais aussi car les témoins ont des souvenirs manquants.]](https://api.nakala.fr/10.34847/nkl.b7bd3k55/4681d2ef0db6f5b035abecabc7ce4ec2dbb9783)
</div>
<div class="audio">[[Marie Couillault et Simone Bonnin chantent l'air puis les témoins discutent de la danse par bribes. On devine que Simone danse.]](https://api.nakala.fr/10.34847/nkl.b7bd3k55/7dec96bf98cf801ad480c346c33c68c80c98)
</div>
<div class="audio">[[L'enregistrement commence avec Martine Roy qui essaye de reproduire un air qu'elle a entendu. Marie Couillault chante ensuite le galop du quadrille. Il n'y a que des fragments, il est difficile d'en comprendre la mélodie en entier.]](https://api.nakala.fr/10.34847/nkl.b7bd3k55/25245609b583a2f4bc4f54196702e9914c9)
</div>
<div class="audio">[[Marie Couillault chante l'air de la gigouillette au tralala, la bande contient quelques bruits parasites mais la mélodie est intelligible.]](https://api.nakala.fr/10.34847/nkl.b7bd3k55/fa5315e6095cceb9d3301a28edc860bd6598f)
</div>
<div class="audio">[[Marie Couillault et Simone Bonnin chantent l'air au tralala sur la première partie et avec des paroles sur la deuxième. A la fin, elles parlent d'une scottich qu'elles n'arrivent pas à retrouver.]](https://api.nakala.fr/10.34847/nkl.b7bd3k55/6508d092b75cb1b6f0cc79a6a819632065fac1dd)
</div>
<div class="audio">[[Marie Couillault chante l'air, apparemment des témoins la dansent en mazurka, alors elle précise à la fin qu'il s'agit d'une valse viennoise.]](https://api.nakala.fr/10.34847/nkl.b7bd3k55/d5172eaca2df509cfd192fd355cae2b0073b78e)
</div>
<div class="audio">[[Marie Couillault chante au tralala l'air d'une mazurka mais la bande coupe, il manque la fin.]](https://api.nakala.fr/10.34847/nkl.b7bd3k55/dd572a7a6c53bbc9aacebe6c9f6178302eec5d4f)
</div>
<div class="audio">[[Marie Couillault explique comment se danse la gatinelle et dans quel contexte elle était dansée.]](https://api.nakala.fr/10.34847/nkl.b7bd3k55/873af39b8dccc6ebfffd01a9b5a5c83c1b8202116)
</div>
<div class="audio">[[Marie Couillault chante l'air au tralala, on entend des danseurs en fond.]](https://api.nakala.fr/10.34847/nkl.b7bd3k55/278df0b72d8302638e0951bbcef65a8cc33bac4f)
</div>
<div class="audio">[[Marie Couillault chante la chanson. Elle ajoute à la fin que c'est plutôt les jeunes qui chantaient ça.]](https://api.nakala.fr/10.34847/nkl.b7bd3k55/fe0e73117f7a752ee956d8eb2ccb3df8e9e760)
</div>
<div class="audio">[[Marie Couillault chante l'air au tralala. Elle précise à la fin que c'est une limousine.]](https://api.nakala.fr/10.34847/nkl.b7bd3k55/bd6bd4654c6a8b0bf017732244753cc548e0c9f)
</div>
<div class="audio">[[A cause des coupures les paroles des témoins n'apportent rien de précis sur cette danse où semble-t-il on saute.]](https://api.nakala.fr/10.34847/nkl.b7bd3k55/2133003a5e6394a79b3aa13777a8144972db7)
</div>
```

et voici dans mon fichier en . qmd :

```
scottich qu'elles n'arrivent pas à retrouver.]](https://api.nakala.fr/10.34847/nkl.b7bd3k55/c500d802b75cb1b6f0cc79a6a819632065fac1dd)
</div>
<div class="audio">[[Marie Couillault chante l'air, apparemment des témoins la dansent en mazurka, alors elle précise à la fin qu'il s'agit d'une valse viennoise.]](https://api.nakala.fr/10.34847/nkl.b7bd3k55/d5172eaca2df509cfd192fd355cae2b0073b78e)
</div>
<div class="audio">[[Marie Couillault chante au tralala l'air d'une mazurka mais la bande coupe, il manque la fin.]](https://api.nakala.fr/10.34847/nkl.b7bd3k55/dd572a7a6c53bbc9aacebe6c9f6178302eec5d4f)
</div>
<div class="audio">[[Marie Couillault explique comment se danse la gatinelle et dans quel contexte elle était dansée.]](https://api.nakala.fr/10.34847/nkl.b7bd3k55/873af39b8dccc6ebfffd01a9b5a5c83c1b8202116)
</div>
```

Proposition de... Home Le titre Veillée de danse autour de Marie Couillault et Camille Bonnin - Maisonneuve 11.1971 [FR]

Roy, Martine

DOI: 10.34847/nkl.b7bd3k55

Appartient aux collections:

- 10.34847/nkl.2ef6zbbp
- 10.34847/nkl.3fa6pop7

Fichiers:

- On entend Hilaire Bonnin qui chante l'air, mais l'air est difficile à déterminer. Lorsque Marie Couillault se joint à lui, l'air est clair. Lorsqu'elle a fini, Hilaire et Marie font comprendre que l'air se joue deux fois complètes. On entend la voix de Jacques Bonnin (membre des Gens de Cherves).
- Marie Couillault chante l'air, elle précise à la fin qu'il faut le faire deux fois.
- Les informateurs discutent à propos de la manière de danser les quadrilles.
- Suite à la demande de l'enquêtrice, Marie Couillault chante l'air de la berline, elle précise que ça se dansait avec un pas de polka avant de reprendre l'air une deuxième fois.
- Plusieurs témoins et des gens de Cherves dansent la danse à l'ours qu'ils chantent en même temps. Marie Couillault confirme.
- Marie Couillault chante l'air de Jean Brisquet et avec les autres témoins, elle explique la danse. Les explications sont partielles à cause de coupe de la bande et du brouhaha par moment mais aussi car les témoins ont des souvenirs manquants.

Je pense que je dois ajouter quelque chose comme une extension pour que cela me renvoie un fichier audio avec une interface et un player .

En changeant le type de balise voila ce que cela donne :



The screenshot shows a web interface with three audio player controls, each displaying '0:00 / 0:00'. To the right, there is a section titled 'Appartient aux collections:' with two entries: '10.34847/nkl.2ef6zbbp' and '10.34847/nkl.3fa6pop7'. Below this is a section titled 'Fichiers:' with a list of files, each with a play button and a progress bar.

en revanche les audios ne sont pas accessibles.

Je repars de la balise de base car trop compliqué et cela ne fonctionne pas

Après-midi

Petit débriefing avec un collègue de la plateforme, il m'a aidé avec mes soucis de récupération d'audio et m'a donné des conseils pour mes méthodes de travail et d'organisations. Voila mon code :

```
elif line.find('<div class="audio">') != -1:
    for i in range(len(dataFiles)):
        if dataFiles[i]['mime_type'].find("audio") != -1:
            AudioBloc = ""
            AudioBloc = line.replace("TitreSon", dataFiles[i]["description"])
            AudioBloc = AudioBloc.replace("URLAudio", baseURI + "data/" + dataDOI + "/" + dataFiles[i]["sha1"])
            Output_Text += AudioBloc
```

et voici ce que cela renvoie :

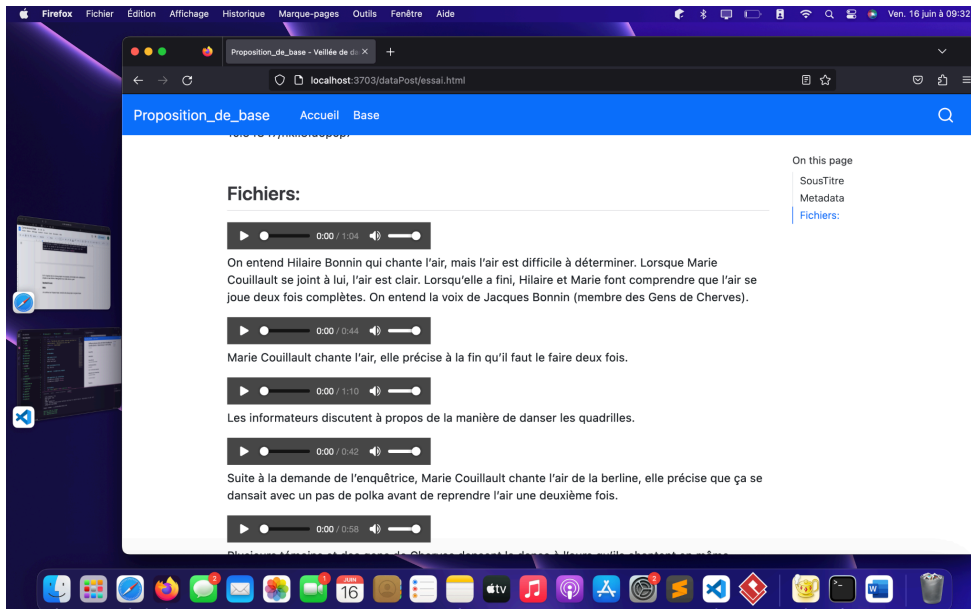
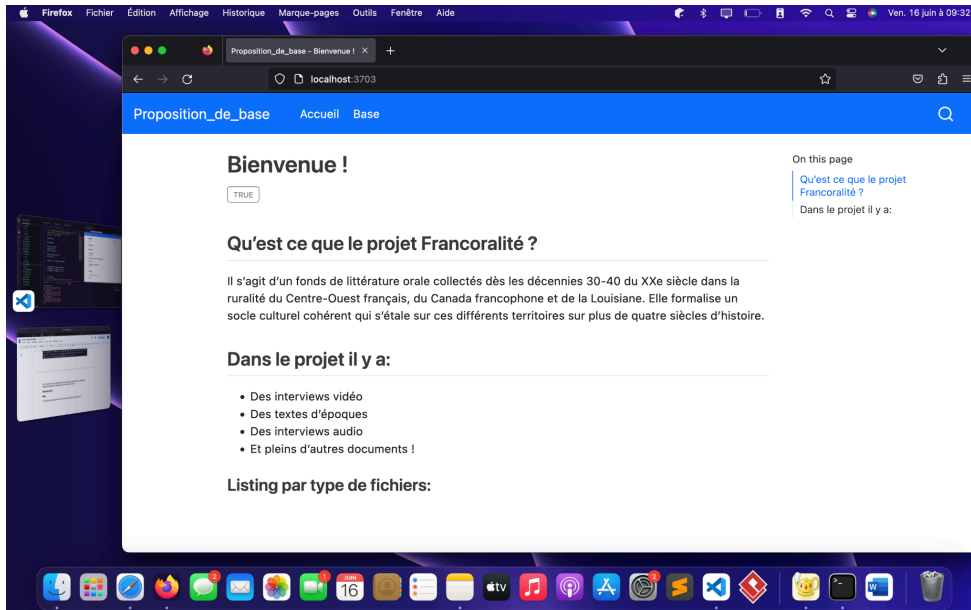
```
<div class="audio"> <audio controls> <source src="https://api.nakala.fr/data/10.34847/nkl.b7bd3k55/4881d2efd0b6f5b035abecabc7ce4eec2dbb9703" type="audio/mpeg"> <p>Marie Couillaullt chante l'air de Jean Brisquet et avec les autres témoins, elle explique la danse. Les explications sont partielles à cause de coupe de la bande et du brouhaha par moment mais aussi car les témoins ont des souvenirs manquants.</p> </audio>
```

En fin d'après midi j'ai mis au propre mes dossiers, donné des noms cohérents et simples à mes fichiers. Réorganiser mon index et mon .yaml

Vendredi 16 Juin

Matin

J'ai continué sur l'aspect visuel, rendu le site plus propre et ergonomique:



J'ai rajouter un type à la data qui peut être audio, photo, texte ou vidéo voici le code :

```
listeTypeFich = (rc.dictVals['files'])  
for c in range (len(listeTypeFich)):  
    TabType = []  
    f = listeTypeFich[c]  
    TabType = (f["mime_type"])
```

```
elif |line.find("TypeData") != -1:  
    Output_Text += line.replace("TypeData", TabType)
```

et voici ce que cela renvoie :

AUDIO/MPEG

Maintenant j'aimerais faire un listing des types de données disponibles sur mon index. De sorte que si un utilisateur veut consulter un type de données il puisse en choisissant simplement audio, vidéo, texte ou photo.

Voici ce que cela donne :

Categories

All (3)

TypeData (1)

audio/mpeg (1)

image/jpeg (1)

et le listing de page d'accueil :

Photo de terrain - Mission Saintonge 1969 - Enquête UPOI_ATP_0001_0012

IMAGE/JPEG

Claudie Marcel-Dubois



Veillée de danse autour de Marie Couillault et Camille Bonnin - Maisonneuve 11.1971 [FR]

AUDIO/MPEG

Camille Bonnin Marie Couillault

La barre de recherche réactive :

Recherche

veillée de danse

4 matching documents

- Veillée de danse autour de Marie Couillault et Camille Bonnin - Maisonneuve 11.1971 [FR]
Camille Bonnin Marie Couillault Roy, Martine 10.34847/nkl.2ef6zbbp 10.34847/nkl.3fa6pop7
Camille Bonnin Marie Couillault
Roy, Martine 10.34847/nkl.2ef6zbbp 10.34847/nkl.3fa6pop7
- Veillée de danse autour de Marie Couillault et Camille Bonnin - Maisonneuve 11.1971 [FR]
Camille Bonnin Marie Couillault Roy, Martine 10.34847/nkl.2ef6zbbp 10.34847/nkl.3fa6pop7
Camille Bonnin Marie Couillault
Roy, Martine 10.34847/nkl.2ef6zbbp 10.34847/nkl.3fa6pop7
- Veillée de danse autour de Marie Couillault et Camille Bonnin - Maisonneuve 11.1971 [FR]
Camille Bonnin Marie Couillault Roy, Martine 10.34847/nkl.2ef6zbbp 10.34847/nkl.3fa6pop7
Camille Bonnin Marie Couillault
Roy, Martine 10.34847/nkl.2ef6zbbp 10.34847/nkl.3fa6pop7
- Veillée de danse autour de Marie Couillault et Camille Bonnin - Maisonneuve 11.1971 [FR]
Metadata
Metadata Auteur(trice) Camille Bonnin Marie Couillault Contributeur(trice) Roy, Martine DOI : 10.34847/nkl.b7bd3k55 Appartient aux collections: 10.34847/nkl.2ef6zbbp 10.34847/nkl.3fa6pop7
Fichiers:
Fichiers: On entend Hilaire Bonnin qui chante l'air, mais l'air est difficile à déterminer. Lorsque Marie Couillault se joint à lui, l'air est clair. Lorsqu'elle a fini, Hilaire et Marie font comprendre que l'air se joue de

Après midi:

En fin de matinée j'ai vu mon tuteur qui m'a dit de revoir mon code pour qu'il soit exploitable sous forme de deux fonctions:

La première grâce à un DOI entrer en paramètre va faire les changements à l'aide d'un documents utilise comme template nommé "base.qmd"

```
def genDataPages(dataDOI):
    """
    Parameters
    dataDOI: String
    """
    baseURI = "https://api.nakala.fr/"

    if rc.isSuccess == True:

        Output_Text = ""
        dataFiles = rc.dictVals["files"]
        dataMetas = rc.dictVals["metas"]

        with open("../base.qmd", "r", encoding="utf-8") as f:
            for line in f:
                if line.find("LeTitre") != -1:
                    Output_Text += line.replace("LeTitre", getMetasByProperty('http://nakala.fr/terms#title', '')[0])
                    print(Output_Text, "1")

                elif line.find("TypeData") != -1:
                    ListeTypeFich = (rc.dictVals['files'])
                    for c in range(len(ListeTypeFich)):
                        TabType = []
                        f = ListeTypeFich[c]
                        TabType = (f['mime_type'])
                        Output_Text += line.replace("TypeData", TabType)
                        print(Output_Text, "2")

                elif line.find("NomDeLauteur") != -1:
                    Auteurs = getMetasByProperty('http://nakala.fr/terms#creator', 'fullName')
                    for i in range(len(Auteurs)):
                        Output_Text += line.replace("NomDeLauteur", Auteurs[i])
                        print(Output_Text, "3")

                elif line.find("NomContributeur") != -1:
                    Contrib = getMetasByProperty('http://purl.org/dc/terms/contributor', '')
                    for i in range(len(Contrib)):
                        Output_Text += line.replace("NomContributeur", "\n" + Contrib[i])
                        print(Output_Text, "4")

                elif line.find("LEDOI") != -1:
                    Output_Text += line.replace("LEDOI", dataDOI)
                    print(Output_Text, "5")

                elif line.find("CollectionId") != -1:
                    for i in range(len(rc.dictVals["collectionsIds"])):
                        Output_Text += line.replace("CollectionId", rc.dictVals["collectionsIds"][i])
                        print(Output_Text, "6")

                elif line.find('<div class="image">') != -1:
                    for i in range(len(dataFiles)):
                        if dataFiles[i]['mime_type'].find("image") != -1:
                            Output_Text += line.replace("TitreImage", dataFiles[i]["description"]).replace("URLImage", baseURI + "iiif/" + dataDOI + "/" + dataFiles[i]["sha1"] + "/full/300/0/default.jpg")
                            print(Output_Text, "7")

                elif line.find('<div class="audio">') != -1:
                    for i in range(len(dataFiles)):
                        if dataFiles[i]['mime_type'].find("audio") != -1:
                            AudioBloc = ""
                            AudioBloc = line.replace("TitreSon", dataFiles[i]["description"])
                            AudioBloc = AudioBloc.replace("URLAudio", baseURI + "data/" + dataDOI + "/" + dataFiles[i]["sha1"])
                            Output_Text += AudioBloc
                            print(Output_Text, "8")

                elif line.find('<div class="footer">') != -1:
                    Output_Text += line.replace("TheDepositor", rc.dictVals["depositor"]["name"]).replace("DepositDate", rc.dictVals["creDate"]).replace("TheLicence", getMetasByProperty('http://nakala.fr/terms#license', '')[0])
                    print(Output_Text, "9")

                else:
                    Output_Text += line

            #Ecriture du contenu modifié dans un nouveau fichier .qmd
            modified_page_path = pathlib.Path("../env_quarto/dataPost/essai2.qmd")
            modified_page_path.write_text(Output_Text)

    else:
        print("Data non récupérée")
```


La deuxième fonction va chercher une métadonnée dans L'API search de Nakala à l'aide d'un paramètre en entrée

```
def getMetasByProperty(prop, field):  
    """  
    Parameters  
    prop : String  
        PropertyURI  
    field : String  
        subfield we want the value.  
  
    Returns  
    List of String.  
    """  
    dataFiles = rc.dictVals["files"]  
    dataMetas = rc.dictVals["metas"]  
    val = []  
  
    for i in range(len(dataMetas)):  
        if dataMetas[i]["propertyUri"] == prop:  
            if field != "":  
                val.append(dataMetas[i]["value"][field])  
            else:  
                val.append(dataMetas[i]["value"])  
    return val
```

Maintenant il faut que j'effectue les changements pour que lorsque le fichier se crée le nom du fichier soit : DOI.qmd

```
NomFich = dataDOI.replace(".", "_").replace("/", "-")  
  
#Écriture du contenu modifié dans un nouveau fichier .qmd  
modified_page_path = pathlib.Path('./env_quarto/dataPost/', NomFich + '.qmd')  
modified_page_path.write_text(Output_Text)
```

remplacement des . et des / sinon cela n'est pas fonctionnel.

le dernier objectif était de faire une fonction récupérant tous les DOIS présent sur nakala et présent dans une collection certifiée voici mon code qui est non fonctionnel :

```

import NklTarget as nklT
import nklAPI_Collections as nklC
import nklAPI_Datas as nklD
import pathlib

def get_collections_datas_test(collectionIdentifier):
    nklT_prodEmpty = nklT.NklTarget(isNakalaProd=True, apiKey="")

    dataIdentifier = collectionIdentifier
    page = 1
    allDOIs = []

    while True:
        print("Récupération des données de la page", page)
        response = nklC.get_collections_datas(nklT_prodEmpty, dataIdentifier, page=page)

        if not response.isSuccess:
            print("Une erreur s'est produite lors de la récupération des données.")
            break

        data = response.dictVals['data']
        for item in data:
            doi = item.get('doi')
            if doi:
                allDOIs.append(doi)

        lastPage = response.dictVals['lastPage']
        if page >= lastPage:
            break

        page += 1

    return allDOIs

collectionIdentifier = "10.34847/nkl.ffabdg38"
DOIS = get_collections_datas_test(collectionIdentifier)

print("Liste des DOIS de la collection :")
for doi in DOIS:
    print(doi)

```

Mardi 20 juin

Soir

j'ai créé une fonction nommée `get_all_dois` qui récupère tous les dois présents dans une collection.

Voici son code :

```
def get_all_dois(collection_identifiant):
    nklT_prodEmpty = nklT.NklTarget(isNakalaProd=True, apiKey="")

    page = 1
    all_dois = []

    while True:
        print("Récupération des données de la page", page)
        response = nklC.get_collections_datas(nklT_prodEmpty, collection_identifiant, page=page)

        if not response.isSuccess:
            print("Une erreur s'est produite lors de la récupération des données.")
            break

        data = response.dictVals['data']
        for item in data:
            doi = item.get('identifiant')
            if doi:
                all_dois.append(doi)

        last_page = response.dictVals['lastPage']
        if page >= last_page:
            break

        page += 1

    return all_dois

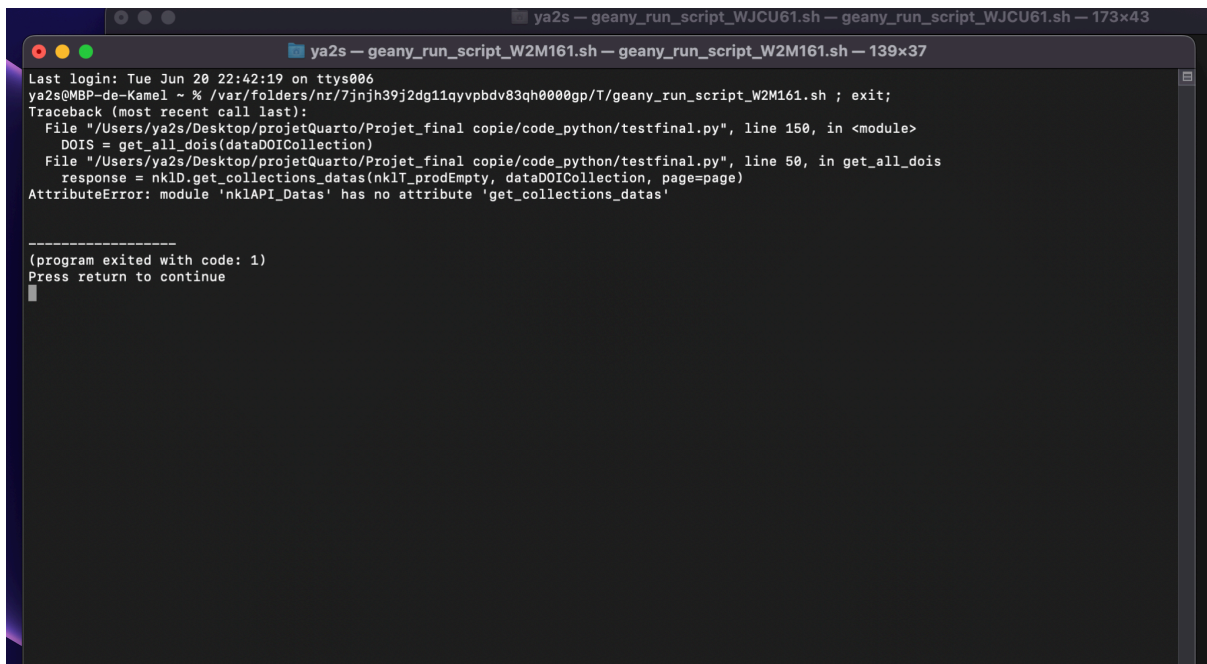
collection_identifiant = "10.34847/nkl.3769u06s"
dois = get_all_dois(collection_identifiant)

print("Liste des DOIs de la collection :")
for doi in dois:
    print(doi)
```

la fonction renvoie :

```
Liste des DOIs de la collection :
10.34847/nkl.abd8xc33
10.34847/nkl.dbcd4021
10.34847/nkl.0f4d61jr
10.34847/nkl.bcc09rhc
10.34847/nkl.0e4e5khz
10.34847/nkl.b02bb1wz
10.34847/nkl.d1f2es8d
10.34847/nkl.2c157jj0
10.34847/nkl.469d884m
10.34847/nkl.7c786rwx
10.34847/nkl.a66e8nq3
10.34847/nkl.bea118dz
10.34847/nkl.0ceb2108
10.34847/nkl.947f98pv
10.34847/nkl.b7cb6ahe
10.34847/nkl.a3647e42
10.34847/nkl.95581216
10.34847/nkl.93eb60hk
10.34847/nkl.e50fh84n
10.34847/nkl.c5ccu116
10.34847/nkl.d4430b0i
10.34847/nkl.ee0333t1
10.34847/nkl.b32f41g8
10.34847/nkl.ce8abjk1
10.34847/nkl.6e67q36t
10.34847/nkl.bcf31m97
```

Le problème que je rencontre est lorsque je la rajoute dans mon autre fichier de code avec toute mes autres fonctions et voici l'erreur que cela me mets :



```
ya2s — geany_run_script_W2M161.sh — geany_run_script_W2M161.sh — 139x37
Last login: Tue Jun 20 22:42:19 on ttys006
ya2s@MBP-de-Kamel ~ % /var/folders/nr/7jnjh39j2dg11qyvpbdv83qh0000gp/T/geany_run_script_W2M161.sh ; exit;
Traceback (most recent call last):
  File "/Users/ya2s/Desktop/projetQuarto/Projet_final copie/code_python/testfinal.py", line 150, in <module>
    DOIS = get_all_dois(dataDOICollection)
  File "/Users/ya2s/Desktop/projetQuarto/Projet_final copie/code_python/testfinal.py", line 50, in get_all_dois
    response = nk1D.get_collections_datas(nk1T_prodEmpty, dataDOICollection, page=page)
AttributeError: module 'nk1API_Datas' has no attribute 'get_collections_datas'

-----
(program exited with code: 1)
Press return to continue
```