Team Kalos WorkSheet

Problem:

Your task is to review the code in the team_ex_2.py file, discuss it as a team, and refactor the code to be more
efficient and extendable. Your review process should include:
identification of possible issues
creation of tasks
assignment of tasks
description of each issue and how it was resolved
In addition to issues/areas for improvement discovered in the codebase, the refactored solution must add the
following new functionality:
allow the user to input their own search term
if the user input search term is less than 4 characters, default to "generative artificial intelligence"
a new directory named "wiki_dl" is created to store the .txt files

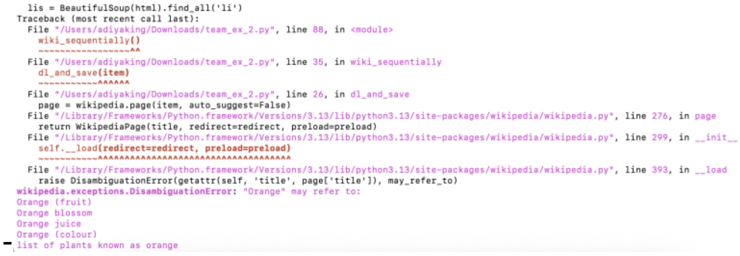
Working Space / Notes

If you would like to use the template below to assist with the problem. However you decide to approach the problem as a team, be sure to document your review process.

What are the function names?

- wiki_sequentially (loop)
- concurrent_threads (thread)
- concurrent_process (multi)

Running command:
-1st time no errors
-2nd time changed the wiki search to “orange” to confuse it



```
  lis = BeautifulSoup(html).find_all('li')
Traceback (most recent call last):
  File "/Users/adiyaking/Downloads/team_ex_2.py", line 88, in <module>
    wiki_sequentially()
    ~~~~~~~~~~~~~~~~~^^
  File "/Users/adiyaking/Downloads/team_ex_2.py", line 35, in wiki_sequentially
    dl_and_save(item)
    ~~~~~~~~~~~^^^^^^
  File "/Users/adiyaking/Downloads/team_ex_2.py", line 26, in dl_and_save
    page = wikipedia.page(item, auto_suggest=False)
  File "/Library/Frameworks/Python.framework/Versions/3.13/lib/python3.13/site-packages/wikipedia/wikipedia.py", line 276, in page
    return WikipediaPage(title, redirect=redirect, preload=preload)
  File "/Library/Frameworks/Python.framework/Versions/3.13/lib/python3.13/site-packages/wikipedia/wikipedia.py", line 299, in __init__
    self.__load(redirect=redirect, preload=preload)
    ~~~~~~~~~~~^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/Library/Frameworks/Python.framework/Versions/3.13/lib/python3.13/site-packages/wikipedia/wikipedia.py", line 393, in __load
    raise DisambiguationError(getattr(self, 'title', page['title']), may_refer_to)
wikipedia.exceptions.DisambiguationError: "Orange" may refer to:
Orange (fruit)
Orange blossom
Orange juice
Orange (colour)
list of plants known as orange
```

-

Issues

Issue 1: User Cannot Choose Concurrency Model

Issue 2: Duplicate download logic in three implementations

Issue 3: convert_to_str is not robust(1. If obj is **anything else** e.g., dict, None, custom object, the function returns **None** implicitly. 2. If some elements in list are non-string, then join function will fail)

Issue 4: The code always uses a hard-coded search term and does not ask for user input or default to “generative artificial intelligence” when input is too short.

Issue 5: No Check for Empty Search Results

Issue 6: There is no wiki_dl so every downloaded file is put in the main folder.

Tasks

task # description assignment solution

1 - Let the user choose which concurrency model to run (sequential, threads, or processes) using an input menu

2 – Create one download_and_save () helper function and have all three implementations call that function instead of repating the download one.

3 – Rewrite convert_to_str so that it always returns a string

4 - Add ask_topic_from_usr () function that asks for a search term, uses it if its at least 4 characters, and otherwise defaults to gen ai. And then pass this topic into all three concurrency functions instead of hard coded strings.

5 - After running results = wikipedia.search(term), check if results is empty. If so, print a clear message and exit gracefully.

6 - Create a wiki_dl folder and update all file-saving code to write the downloaded pages into that directory.