**Developer Task: Integrating UniProt and Ensembl APIs**

Background:
UniProt and Ensembl are public APIs for biological data. This task assesses your ability to integrate APIs, handle both GET and POST requests, and process biological data in Python.

**Your Tasks**
**1. Learn about APIs**

Get familiar with the concept of APIs, how to send GET/POST requests, and how to use JSON request bodies:
- Real Python: API Integration in Python: https://realpython.com/api-integration-in-python/
- Real Python: Python Requests Library: https://realpython.com/python-requests/

**2. Explore the APIs**

- UniProt REST API information: https://www.uniprot.org/help/programmatic_access
API Docs: https://www.ebi.ac.uk/proteins/api/doc/#/proteins
- Ensembl REST API: https://rest.ensembl.org/

**Step-by-Step Instructions**
Use the UniProt API to retrieve information for the following protein accessions:
- P12345, Q8N726, O00255

**Step 0:**

Setup the project using virtual environment, prepare project structure. One can easily follow all necessary requirements for the project.
You can gain bonus points if you will use Poetry.

**Step 1: Get Protein Information**

- Write a function `get_protein_information` to fetch data from the UniProt API for a list of protein accessions.
  - Endpoint: https://www.ebi.ac.uk/proteins/api/proteins/{id}
  - Extract the following fields for each protein:
    - Protein Accession
    - Protein Name
    - Gene
    - Organism (Scientific)
    - Organism (Common)
    - Molecular Weight

- Return the results as a pandas DataFrame.

**Step 2: Get Ensembl Gene IDs**

- Write a function `get_ensembl_gene_ids` to fetch Ensembl Gene IDs for the genes retrieved in Step 1.
  - Endpoint: https://rest.ensembl.org/xrefs/symbol/{species}/{gene}
  - Documentation: https://rest.ensembl.org/
  - Replace `{species}` with the organism name (e.g., `homo_sapiens`) from `Organism (Scientific)` and `{gene}` with the gene name.
  - Add a new column, `Ensembl Gene ID`, to the DataFrame.

**Step 3: Get Gene Details**

- Write a function `get_gene_data` to fetch additional gene details using the Ensembl Gene API.
  - Endpoint: https://rest.ensembl.org/lookup/id
  - Swagger Documentation: https://rest.ensembl.org/
  - Use a list of Ensembl Gene IDs as a request body
  - Extract the following details for each Ensembl Gene ID:
    - Description
    - Seq Region Name
  - Return a new pandas DataFrame containing these details.

**Step 4: Merge Data**

- Combine the data from Steps 1, 2, and 3 into a single DataFrame.

**Step 5: Save the Data**

- Save the final DataFrame as an Excel file named `protein_gene_analysis.xlsx`.

**Additional Requirements**

- Logging: Log key steps and errors (e.g., when a request fails).
- Modular Structure: Write reusable, modular functions for each task.
- Error Handling and Exceptions: Ensure the code can handle failed requests gracefully.
- Best Practices:
  - Use clear and concise docstrings.
  - Follow PEP 8 guidelines for clean, readable code.

**Hints**

1. For species names in `Organism (Scientific)`, ensure the names are lowercase before calling the Ensembl API
2. Use the `merge` function in pandas to combine DataFrames.

**Deliverables**

1. Python script/s containing all functions and implementation.
2. The final Excel file `protein_gene_analysis.xlsx`.
3. A brief explanation of your approach in comments.
4. Upload the script and Excel file to your Github and share the link to the project..