Assignment 2
Web Crawling and Extracting Information-Part1

Computing Lab-II
11th Jan 2023

This assignment is on crawling web pages and extracting the required information by creating suitable grammar rules.

Task 1 (Crawling FIFA world cup 2022 Wikipedia website→ https://en.wikipedia.org/wiki/2022_FIFA_World_Cup)

1. This page contains all the essential information related to the FIFA worlds cup 2022, like teams and player details, venue, match results etc.
2. Write a python code that reads a URL & saves the page in a file in HTML format. [15 minutes] [2 - 2.30]

==========

Teaching 1.5: Small example of Stadium(Tutorial). [2.30 - 3.00 PM].

==========

Task 2: Run the code and get the data for the stadium. [3.15]

==========

Task 2.5: Do it here,

a. All the teams participated in the tournament. [4.30]

==========

Task 3 (Creating grammar and parsing the files)

2. Create grammar that can be used to extract the following fields:
    a. All the teams participated in the tournament.
    b. Venue details like name and capacity.
    c. Match details
        i. Group stage
        ii. Knockout stage

    Here for a given group and a specific match below details:
        1. Stadium detail
        2. Attendance
        3. Goal scorer(if any)
        4. Referee

    Also given a group name:
        1. Teams advanced for knockouts.
        2. Given a team name, the number of goals forwarded & conceded.

    For knockout stages:
        1. Show the fixtures
        2. Given two countries:
            a. Results
            b. Scorer
            c. Stadium
            d. Attendance
            e. Referee

d. Show all the awards
e. Given a team name:
    1. Show its current squad.
    2. Show its last & upcoming five matches.
    3. Given a player name from the current squad.
        a. Show his DoB
        b. Playing position
        c. Current club
        d. Past clubs
        e. International appearance
        f. Goal count

2. You can ignore other fields except the above.

3. You need to design a menu-driven program to resolve user queries. We leave the design of the menu up to you, but keep in mind that your menu should be easy to use and address all queries. The user should also be able to go back to the previous menu.

4. Write python code using PLY to extract the above fields.
Your program should show all the possible query fields a user can ask for (from the above list items).

5. Your program should also save the result in a log file in the following format.
<Field_requested> <tab> <Field_value>

6. You must think correctly about what kind of errors can come in the process and try to handle them. Note that you cannot use the “Beautiful Soup” python package for this assignment. Use the PLY package in python.

PLY ref: https://www.dabeaz.com/ply/

7. You can write a readme file to provide any particular instructions related to program execution steps, input format, or anything that you might think is useful for the evaluator while evaluating the assignment.

Deliverables:

1. Codes for task1 and task2, readme file if any.
2. Save this in a folder named in the format: <Roll No.>_CL2_A2. Compress this folder to zip format, creating a compressed file <Roll No.>_CL2_A2.zip. Upload this compressed file to moodle. Example: If your roll no. is 22CS60R05, the folder should be 22CS60R05_CL2_A2, and the compressed file should be 22CS60R05_CL2_A2.zip.
3. Not adhering to these instructions can incur a penalty.

Evaluation Scheme

Task1: 5 marks
Task2: 70 marks (all the fields grammar + correct output)
Error handling: 10 marks
Coding Style/Design: 15 marks

Important Instructions

1. Plagiarism Rule: If your code matches (more than 60%) with another student's code, all those whose codes match will be awarded zero marks without any evaluation. Therefore, it is your responsibility to ensure that neither you copy anyone's code nor anyone can copy yours.
2. Code error: If your code doesn't run or gives an error while running, you will be awarded zero marks.