td1a_correction_session8_wikiroot

December 23, 2020

1 1A.algo - Parcours dans un graphe (wikipédia) - correction

Correction du notebook du même titre. On part d'une page, on explore les liens des pages liées à la première et on continue. On utilise le module beautifulsoup4 (web scrapping) pour parser les pages.

[1]:
```
from jyquickhelper import add_notebook_menu
add_notebook_menu()
```

[1]:
```
<IPython.core.display.HTML object>
```

Solution de Félix Revert.

1.1 Exercice 1 : lire une page web

[2]:
```
import urllib.request as ulib

def get_html(address, source="https://fr.wikipedia.org/wiki/"):
    with ulib.urlopen(source+address) as u:
        return u.read()

get_html("http://www.xavierdupre.fr", source="")[:100]
```

[2]:
```
b'<?xml version="1.0" encoding="utf-8"?>\r\n<html>\r\n<head>\r\n<link
TYPE="text/css" href="pMenu2.css" rel='
```

1.2 Exercice 2 : extraire le premier lien

Il faut écrire une fonction qui récupère le premier lien d'une page wikipedia avec BeautifulSoup

[3]:
```
def get_first_link(soup):
    for p in soup.find('div',{'id':'bodyContent'}).findAll('p'):
        for a in p.findAll('a'):
            if a and a.get('href').startswith('/wiki/') and not ":" in a.get('href'):
                return a.get('href')[6:]

from bs4 import BeautifulSoup
stru = BeautifulSoup(get_html("Python_(langage)"), "lxml")
get_first_link(stru)
```

[3]:
```
'Langage_de_programmation'
```

[4]:
```
def get_to_philosophy(initial_address, max_iterations=100, verbose=False):
    target_page = "Philosophie"
    iteration = 0
    pages_visited = []
    current_address = initial_address

    if verbose:
        print("\ninitial address: " + current_address+"\n Will you go to␣
↪"+target_page+" ?...\n")

    while iteration < max_iterations:
        current_address = get_first_link(BeautifulSoup(get_html(current_address),␣
↪"lxml"))
        if current_address is None:
            break
        if verbose:
            print(current_address)

        if current_address in pages_visited:
            print("Boucle de " + str(iteration - pages_visited.index(current_address))␣
↪+
                  " noeuds trouvée à partir de "+str(pages_visited.
↪index(current_address))+" itérations")
            return
        elif current_address.lower() == target_page.lower():
            print(str(iteration) + " itérations pour arriver à la page Philosophie")
            return
        else:
            pages_visited.extend([current_address])
            iteration += 1

    return str(max_iterations)+" itérations atteintes"

get_to_philosophy("Python_(langage)", verbose=True)
```

```
initial address: Python_(langage)
 Will you go to Philosophie ?…

Langage_de_programmation
Informatique
Sciences_exactes
Sciences_de_la_nature
Anglais
API_%CB%88
Syllabe
Latin
Langues_italiques
Langue
Syst%C3%A8me
Ensemble
Totalit%C3%A9
Concept
```

[4]: '100 itérations atteintes'

[5]: