Dokumentácia ku projektom z predmetu ISJ

Meno: Adam Bezák
Login: xbezak01

1. úkol

Našou prvou úlohou bolo napísať v Pythone alebo Ruby skript, ktorý stiahne všetky príspevky z vybraného diskusného fóra. Môj skript je implementovaný v jazyku Python Verzia 2.7.5

Použité knihovny:

BeautifulSoup4 – jednoduchý parser na vybranie potrebnych dát z HTML formátu
requests – stahovanie web stránok
re – regulárné výrazy
urlparse - práca s URL, konverzia relativnej URL na absolútnu atď..

Skript sa spúšťa vžy s jedným parametrom.

```
>> python forum.py X
```

0 < X < 1500
X – počet príspevkov ktoré chceme stiahnuť
Je plne automatizovaný a po každom kroku vypisuje, ktoré vlákno sa aktuálne sťahuje. V prvotnom prechode stránky nájde všetky linky na subfóra a v dalšom prechode z daných linkov stiahne príspevky. Pre jednoduchosť a priehladnosť (kedže moje fórum má viac než 36000 tém a 320000 príspevkov) sťahujem len posledné príspevky v daných témach. Vždy sa sťahovanie dokončí do posledného príspevku v danom subfore. Do výstupného súboru ukladám link sťahovaného vlákna, jeho názov, meno uživatela, text príspevku, dátum a ID príspevku.

Testovanie rýchlosti:

X	počet	t[s]
50	74	37s
100	171	1m27s
250	281	2m23s

fórum: http://csko.cz/forum/forum.php
súbor: forum.py
vystupny subor: output.txt

2.úkol

Druhou úlohou bolo napísať v Pythone alebo Ruby skript, ktorý stiahne všetky príspevky z daného Twitter účtu (@vossenwheels). Skript bol implementovaný v Pythone 2.7.5
Použité knihovny:

os.path – Overenie existencie priečinkov a súborov a ich prípadne vytvorenie
tweepy – Twitter API
json – knižnica na pracu s dátami vo formáte JSON
requests – sťahovanie dát z web stránok
shutil – operácie so súbormi

Skript sa spúšťa bez parametrov. Výstupný súbor je v JSON formáte. Tento formát som si pre Twitter vybral z toho dôvodu, že je to štandardizovaný formát s ktorým sa pohodlne pracuje naprieč rôznymi prostrediami. Ak by som chcel neskôr v budúcnosti pracovať s týmto JSON súborom tak ho môžem prečítať napr. Javascriptom a z neho dostať naspať list tweet objektov. Popr. Inou cestou a dekódovať ho na natívny objekt.
Na začiatku skriptu si po importovaní knižníc definujem všetky konštanty – meno tweeter účtu ktorého príspevky sťahujem, názov výstupného súboru, názov priečinku do ktorého sa ukladajú webstránky a potom všetky API klúče.
Potom volám metódu tweepy objektu – user_timeline s parametrom twitter_profile_name ktorá vráti zoznam Status objektov. Ak nie je zoznam prázdny, zistím dostupnosť priečinka určeného na ukladanie web stránok a v prípade potreby ho vytvorím.
For cyklus prejde zoznam statusov a pre každý urobí:

a) do nového zoznamu pridám tweet objekt s dátami v JSON forme
b) Ak sa nachádzajú v tweete nejaké url adresy, každú stránku na danej adrese stiahne a uloží ju na lokálny disk pod názvom, ktorý je extraktovaný zo skrátenej t.co adresy. Toto zabezpečuje jedinečnosť názvov súborov a taktiež asociovanie daného súboru s url adresou.

Po prejdení všetkých statusov sa zoserializuje list tweet objektov do JSON formátu a výsledný string sa zapíše do súboru.

Poznámka ku riadku 25-39:

V skripte som mal implementovanú aj aktualizáciu tweetov, ktorá mi fungovala avšak po testovaní pri odovzdávaní stiahne len jednu aktualizáciu a ďalšie už nie tak som ju zakomentoval.

Twitter účet: @vossenwheels
súbor: twitter.py
výstupný súbor a priečinok: tweets.json, /sites