Tartalomjegyzék

- 1 Bevezetés
- 2 Feladatok
  - 2.1 Teszt html-en
  - 2.2 Élesben a web-rõl
  - 2.3 Bónusz

Bevezetés

- Ami mindenképp kelleni fog a python file-od elejére:

```
from lxml import etree
```

- Az lxml dokumentációját megtaláljátok ezen az oldalon.

- Valószinüleg kelleni fog még, hogy a karakter kódolás jó legyen:

```
import sys
reload(sys)
sys.setdefaultencoding("http://wiki.math.bme.huiso-8859-2"http://wiki.math.bme.hu)
```

- A webrõl html oldalt a következõ függvénnyel tölthetünk le egy string-be:

```
import urllib

def get_html(uri):
    u = urllib.urlopen(uri)
    return u.read()
```

- Utána simán beírhatjuk:

```
html_str = get_html("http://wiki.math.bme.huhttp://www.google.com"http://wiki.math.bme.hu)
```

- Hogy elkezdhessünk bányászni, a következõket kell megtennünk:

```
import StringIO
parser = etree.HTMLParser()
html = etree.parse(StringIO.StringIO(html_str),parser)
```

- Ezek után használhatjuk az xpath függvényt, a következõ módon:

```
html.xpath("http://wiki.math.bme.hu/html/body/p"http://wiki.math.bme.hu)
```

ahol a /html/body/p az útvonal a html-fában.

- Az xpath függvény egy listát fog visszaadni, így vagy [0], [1] stb.-vel kérdezzétek le az elemeit, vagy egy for ciklussal menjetek végig rajta.
- Ha már lekértétek valamelyik elemét, akkor a text adattagjában van a benne tárolt szöveg, pl:

```
h.xpath("http://wiki.math.bme.hu/html/body/p"http://wiki.math.bme.hu)[1].text
```

ez lekéri a második html/body-n belüli p-nek a szövegét

- Az xpath-ban a következõket használhatjátok még:
  - /html//p - a html-en belül bármilyen útvonalon elérhetõ p-k
  - /html/body/* - a közvetlenül /html/body alatti összes csúcs
  - /*/*/ul - a 2 hosszú úton elérhetõ ul-ek
  - //a[@href='http://math.bme.hu'] - bármilyen útvonalon elérhetõ 'a'-k amiknek a href attribútuma a 'http://math.bme.hu'
  - az elõzõt tagadhatjátok is, hogy olyanokat keressen ami nem rendelkezik valamilyen attribútummal: //hr[not(@noshade)]
- Ha valamelyik csúcs attribútumának meg szeretnétek tudni az értéket, a következõképpen tudjátok ezt megtenni:

```
html.xpath("http://wiki.math.bme.hu//hr"http://wiki.math.bme.hu)[0].get('size')
```

ez lekéri az elsõ hr-nek a size attribútumát

- Ez a függvény még nagyon hasznos lehet:

```
def inner_html(xpath):
    return etree.tostring(xpath, pretty_print=True, method="http://wiki.math.bme.huhtml"http://wik
```

Ez nem csak a megadott csúcsban található szöveget írja ki, hanem az egész részfát, pl:

```
inner_html(html.xpath('/html/body/p')[1])
```

Kiírná a teljes html kódot ami a /html/body-n belül a második 'p'-ben van.

Példákat ezekre találhattok Simon András eredeti elõadásanyagában itt

Feladatok

Teszt html-en

Töltsétek le a következõt html-t: html

A következõ feladatokat mind ezen a test.html-en végezzétek el, ezt parse-olni a következõ módon tudjátok:

```
parser = etree.HTMLParser()
html = etree.parse("http://wiki.math.bme.hu./test.html"http://wiki.math.bme.hu, parser)
```

1. Írjátok ki az elsõ 'p'-nek a tartalmát!
2. Írjátok ki a második ul részfájának a teljes html-ét! (Ezt az inner_html függvénnyel tudjátok könnyen.)
3. Írjátok ki az összes li tartalmát egy for ciklussal!
4. Navigáljatok el ahhoz a hr-hez, aminek a size attribútuma 1! (Úgy tudjátok leellenõrizni, hogy jól megtaláltátok-e, ha az inner_html-el írjátok ki.)
5. Írjátok ki annak az ul-nek a tartalmát, ami 3 lépésben elérhetõ (tehát õ 4 mélységben van).
6. Írjátok ki annak a táblázatnak a summary attribútumát amelyiknek a border-e "http://wiki.math.bme.hu1"http://wiki.math.bme.hu!
7. Gyûjtsétek ki bármelyik táblázatból azokat az adatokat amiknek a class-a "http://wiki.math.bme.huallat"http://wiki.math.bme.hu!
8. Mentsétek file-ba annak a táblázatnak a részhtml-jét amelyik elérési útvonalában legalább 2 div van!

   Ehhez egy kis segítség, a második gyakorlat végén van a file kezelésrõl. Vagy csinálhatjátok úgy, hogy printelitek a részhtml-t és a shell-ben nem python file.py-t hanem python file.py >> resz.html -t írtok, ezzel amit printelne a terminálba azt helyette elmenti a resz.html-be.

Élesben a web-rõl

1. A math.bme.hu-ról (lehetséges, hogy http://math.bme.hu/html/index.html kell a get_html-nek) bányásszátok ki a legelsõ hírt (jobb fent), ezt mentsétek egy külön html-be és nézzétek meg szép-e!
2. A math.bme.hu algebra oktatói oldaláról bányásszátok ki az összes oktató nevét!
3. http://www.weatheronline.co.uk/-rõl bányásszátok ki Budapest Tmin és Tmax-át, a mai napról (a script úgy mûködjön, hogy mindig az adott napról olvassa ki a hõmérsékletet, szóval ha holnap futtatod, akkor a holnapi hõmérsékletet írja ki, azaz mindig a táblázat elsõ oszlopából kell válogatni)
4. Az index.hu-ról olvassátok ki a fõ hír linkjét és címét!

Bónusz

- Írjatok egy scriptet, ami kér egy magyar nagyvárost, és ha a weatheronline-on fent van az adott városra a hõmérséklet, akkor visszaadja a Tmin és Tmax-ot. Segítség:
  - A http://www.weatheronline.co.uk/Hungary/Budapest.htm bal oldalán a menüben ott van az összes város amit ismer.
  - A link amint láthatjátok a városnév alapján van, így a Bicskeit könnyen megkaphatod, ha a Budapestet lecseréled a linkben Bicskére.
  - Elõször csináljátok meg úgy, hogy az ékezet nélküliekre mûködjön, az ékezetesekkel szenvedjetek csak utoljára.