Μάθημα 19.
Διαχείριση κειμένου με regular expressions

Νίκος Αβούρης, Πανεπιστήμιο Πατρών. hci.ece.upatras.gr/avouris

[idle]

19. Regular Expressions

Η βιβλιοθήκη re επιτρέπει τη χρήση της **γλώσσας αναγνώρισης προτύπων σε κείμενο** Regular Expressions (regex), στην Python.

Μέσω της re μπορούμε να αναζητήσουμε πρότυπα σε κείμενα, και να εκτελέσουμε πράξεις (ανάκτηση, μετατροπή, κλπ) με βάση τα πρότυπα αυτά.

[idle]

19. Regular Expressions : συναρτήσεις

re.search(pattern, string) → True/False (αναζήτηση)

re.sub(pat1,pat2, string, max=0) → τροποποιημένο string (αντικατάσταση)

re.findall(pattern, string) → λίστα ευρημάτων (εύρεση)

re.split(pattern, string) → λίστα από string (διαχωρισμός)

re.compile(pattern)→ αντικείμενο pattern* (μετάφραση προτύπου)

* Αν πρόκειται το πρότυπο να χρησιμοποιηθεί σε επαναληπτική δομή θα πρέπει να μεταφραστεί σε αντικείμενο-πρότυπο ώστε να επιταχυνθεί η επεξεργασία

[idle]

Εισαγωγή στη γλώσσα προγραμματισμού python

19. Regular Expressions : ειδικοί χαρακτήρες . ^ $ * + ? { } [ ] \ | ( )

^ Αρχή κειμένου

$ Τέλος κειμένου

. Κάθε χαρακτήρας εκτός από '\n'

re* 0 + επαναλήψεις του re

re*? 0 + επαναλήψεις re (1ο πρότυπο που συναντάει)

re+ 1 + επαναλήψεις του re

re +? 1 + επαναλήψεις (1ο πρότυπο που συναντάει)

[αβγ] 'Ενας από τη λίστα χαρακτήρων

[^αβγ] Ένας χαρακτήρας εκτός από αυτούς στη λίστα

[α-ω0-9] Ένας χαρακτήρας από πεδίο χαρακτήρων

( ) Ένδειξη έναρξης και τερματισμού περιοχής εξαγωγής προτύπου

re{2,4} 2 -4 επαναλήψεις του re

| ή

19. Regular Expressions : Κλάσεις χαρακτήρων

\w Οποιοσδήποτε αλφαριθμητικός χαρακτήρας [a-zA-Z0-9_]
\W Οτιδήποτε εκτός αλφαριθμητικών χαρακτήρων [^a-zA-Z0-9_]
\d Αριθμός, [0-9]
\D Οτιδήποτε εκτός αριθμού, [^0-9]
\s Κενό [\n\f\t\v\r]
\S Οτιδήποτε εκτός του κενού [^[\n\f\t\v\r]

\b Όριο μεταξύ αλφαριθμητικού και μη-αλφαριθμητικού (αρχή ή τέλος λέξης)

[idle]

Νίκος Αβούρης, Πανεπιστήμιο Πατρών. hci.ece.upatras.gr/avouris

19. Regular Expressions

Ειδικοί χαρακτήρες: . ^ $ * + ? { } [ ] \ | ( )

Αν θέλουμε να χρησιμοποιήσουμε έναν από τους ειδικούς χαρακτήρες ως κανονικό χαρακτήρα τότε βάζουμε πριν τον χαρακτήρα \

\. \+ \$ \^ \(κλπ

[idle]

19. Regular Expressions : Παραδείγματα

Πώς εξετάζουμε αν μια συμβολοσειρά είναι πραγματικός αριθμός;

```
^[-+]?[0-9]*\.?[0-9]+([eE][-+]?[0-9]+)?$
```

http://www.regular-expressions.info/floatingpoint.html

[idle]

19. Regular Expressions : Παραδείγματα

πώς ελέγχουμε το μήκος του προτύπου;

```
>>> s = ' 123 456 789 99999 12 34 4 '
>>> re.findall('[0-9]{2,}', s)
['123', '456', '789', '99999', '12', '34']
>>> re.findall('[0-9]{3}', s)
['123', '456', '789', '999']
>>> re.findall('[0-9]{5,}', s)
['99999']
```

http://www.regular-expressions.info/floatingpoint.html

[idle]

Εισαγωγή στη γλώσσα προγραμματισμού

19. Regular Expressions : Παραδείγματα

Φτιάξτε μια μηχανή αναζήτησης για τα παραμύθια του Ευγένιου Τριβιζά

[idle]

19. Regular Expressions : Παραδείγματα

```
import re
f = open('greece.txt', 'r', encoding = 'utf-8')
greece = f.read()
# Αντιακτάσταση re.sub()
print(re.sub(r'\bΕλλάδα\b', 'ΕΛΛΑΔΑ', greece))

# Εύρεση re.findall() των λέξεων που αρχίζουν από φωνήεν
print("\nΛέξεις από φωνήεντα")
pattern = r"\b[αάεέηήιίοόυύωώ][\w]*" # raw string r"..." χωρίς \
print(re.findall(pattern, greece, re.I))

f.close()
```

Εισαγωγή στη γλώσσα προγραμματισμού python

19. Regular Expressions : Παραδείγματα

Αποθηκεύουμε την ιστοσελίδα του Πανεπιστημίου Πατρών (upatras.gr) ως αρχείο html. Έστω upat.htm

(1) Να βρείτε αν υπάρχει email στη σελίδα.
(2) Να βρείτε το περιεχόμενο της ετικέτας <title>
(3) .. και των ετικετών <h2> της σελίδας αυτής.

[idle]

19. Regular Expressions : Παραδείγματα

```
Αναζήτηση email σε ιστοσελίδα
['rectorate@upatras.gr']

Αναζήτηση <title> σε ιστοσελίδα
['Πανεπιστήμιο Πατρών']

Αναζήτηση <h2> σε ιστοσελίδα
['Χρήσιμοι σύνδεσμοι', 'Επικοινωνία', 'Φόρμα αναζήτησης', 'Φωτογραφίες']
```

```
import re
f = open('upat.htm', 'r', encoding = 'utf-8')
html = f.read()

# Εύρεση email σε ιστοσελίδα
print("\nΑναζήτηση email σε ιστοσελίδα")
pattern = r"\b[A-Z0-9._%+-]+@[A-Z0-9.-]+\.[A-Z]{2,}\b" # http://www.regular-expressions.info/email.html
print(list(set(re.findall(pattern, html, re.I))))
print("\nΑναζήτηση <title> σε ιστοσελίδα")
pattern = r"<title\b[^>]*>(.*?)</title>" #http://www.regular-expressions.info/examples.html
print(list(set(re.findall(pattern, html, re.I))))
print("\nΑναζήτηση <h2> σε ιστοσελίδα")
pattern = r"<h2\b[^>]*>(.*?)</h2>" #http://www.regular-expressions.info/examples.html
print(list(set(re.findall(pattern, html, re.I))))
f.close()
```

[idle]

19. Regular Expressions : Ασκήσεις

Στο παράδειγμα της ενότητας 17 (αρχεία), είχαμε επεξεργαστεί το αρχείο vouna.txt χρησιμοποιώντας μεθόδους συμβολοσειρών. Να επεκτείνετε την άσκηση ώστε να περιλάβετε τη γεωγραφική περιοχή κάθε βουνού, χρησιμοποιώντας regular expressions.

http://www.regular-expressions.info/

[idle]