

Ενότητα: Συμβολοσειρές, Δομή Επανάληψης while

Σχέδιο Δραστηριότητας: Επεξεργασία Συμβολοσειρών (strings)

Εισαγωγή

Μια λίστα στην Python είναι ουσιαστικά ένας δυναμικός πίνακας. Δηλαδή ένας πίνακας όπως στην Pascal ή στην ΓΛΩΣΣΑ με τις εξής τρεις βασικές διαφορές:

1. Δεν χρειάζεται να τον δηλώσουμε κάπου
2. Δεν έχει σταθερό μέγεθος, μπορεί να αυξάνεται και να μειώνεται κατά την εκτέλεση του προγράμματος
3. Η αρίθμηση των δεικτών ξεκινάει από το 0 όπως συμβαίνει στη C και στην Java και παρέχεται η δυνατότητα αρνητικής δεικτοδότησης.

0	1	2	3	4	5	6	7				
'Μ'	'Ο'	'Ν'	'Τ'	'Υ'	' '	'Ρ'	'Υ'	'Τ'	'Η'	'Ο'	'Ν'
-8	-7	-6	-5	-4	-3	-2	-1				

Μπορούμε να δημιουργήσουμε την παραπάνω συμβολοσειρά με διάφορους τρόπους. Δύο από αυτούς δίνονται παρακάτω:

<code>word = "MONTY PYTHON"</code>	<code>word = "MONTY" + " " + "PYTHON"</code>
------------------------------------	--

Δραστηριότητα 1

Ανοίξτε τον διερμηνευτή της Python, δώστε τις παρακάτω εντολές και παρατηρώντας τα αποτελέσματα, να περιγράψετε την λειτουργία των τελεστών +, in, not in και των συναρτήσεων len, int και str .

<pre>>>> w = "MONTY PYTHON" >>> w[0] + w[1] + w[2] + w[8] + w[9] >>> len(w) >>> 123 + "123" >>> str(123) + "123" >>> 123 + int("123")</pre>	<pre>>>> "PYTHON" in w >>> vowels = "aeiou" >>> 'e' in vowels >>> 'p' not in vowels >>> for letter in vowels: >>> print letter</pre>
---	--

Δραστηριότητα 2

- α. Να αναπτύξετε μια συνάρτηση η οποία θα δέχεται μια λέξη ή φράση και θα επιστρέφει πόσα φωνήεντα περιέχει.
- β. Να γράψετε μια συνάρτηση η οποία θα ελέγχει αν μια συμβολοσειρά αποτελεί ηλεκτρονική διεύθυνση αλληλογραφίας ελληνικού ιστότοπου, δηλαδή περιέχει το σύμβολο '@', δεν περιέχει κενά και έχει κατάληξη '.gr'.
- γ. Να αναπτύξετε μια συνάρτηση η οποία θα δέχεται μια συμβολοσειρά και θα ελέγχει αν αποτελεί πενταψήφιο ταχυδρομικό κώδικα.
- δ. Να αναπτύξετε μια συνάρτηση η οποία θα δέχεται έναν πενταψήφιο ακέραιο αριθμό και θα επιστρέφει τον κατοπτρικό του, π.χ. κατοπτρικός του 12345 είναι ο 54321.

Δραστηριότητα 3

Για την παραγωγή τυχαίων αριθμών χρησιμοποιούμε τη συνάρτηση randint(low,high) της βιβλιοθήκης random η οποία επιστρέφει ένα τυχαίο ακέραιο αριθμό στο διάστημα [low, high]. Να γράψετε ένα πρόγραμμα το οποίο θα παράγει 100 τυχαίους αριθμούς στο διάστημα [1,10] και θα τους εμφανίζει στην οθόνη.

Εφαρμογή: Βρες τη Λέξη (ή κρεμάλα)

Καλείστε να συμπληρώσετε τον κώδικα ενός προγράμματος το οποίο παίζει κρεμάλα. Ο βασικός κώδικας δίνεται στο αριστερό πλαίσιο ενώ στο δεξί δίνονται οι συναρτήσεις που χρησιμοποιούνται. Εσείς πρέπει να υλοποιήσετε όλες τις συναρτήσεις έτσι ώστε το παιχνίδι να έχει την επιθυμητή λειτουργικότητα.

Κατεβάστε το αρχείο hangman.py από την ιστοσελίδα <http://evripides.mysch.gr/python>

<pre> import random secret_word = get_secret_word() letters_found = "" tries = 6 found = False while tries > 0 and not found : letter = raw_input("Μάντεψε ένα γράμμα : ") print_word(secret_word, letters_found) if letter not in secret_word: tries = tries - 1 print_hangman(tries) else : letters_found += letter found = all_letters(secret_word, letters_found) print_word(secret_word, letters_found) print if found : print "Συγχαρητήρια, κέρδισες" else: print "Δεν πειράζει, θα ξαναπροσπαθήσεις" </pre>	<pre> def print_hangman(tries): """ Ανάλογα με το πλήθος των αποτυχημένων προσπαθειών εμφανίζει το αντίστοιχο τμήμα του hangman, όπως φαίνεται παρακάτω: _ () \ / ^\ """ def all_letters(word, alphabet): """ Επιστρέφει True αν όλα τα γράμματα της λέξης word εμφανίζονται στη λέξη alphabet. π.χ. all_letters("abracadabra", "abrcd") επιστρέφει True """ def print_word(secret, guessed): """ Εμφανίζει το τμήμα της λέξης που έχει βρει ο παίκτης μέχρι εκείνη τη στιγμή. Στη θέση των γραμμάτων που δεν έχουν βρεθεί ακόμα εμφανίζεται ο χαρακτήρας '_'. """ def get_secret_word(): """ Δημιουργεί μια λίστα από λέξεις, επιλέγει τυχαία μια λέξη και την επιστρέφει ή απλά επιστρέφει μια λέξη χωρίς τη χρήση της random """ </pre>
--	---