

Αντικειμενοστρεφής Προγραμματισμός–Python

Κ.Π. Γιαλούρης

Στόχοι του σημερινού μαθήματος

- ❑ Μέθοδοι και συναρτήσεις χειρισμού `string`
- ❑ Διαμόρφωση της εκτύπωσης

Μέθοδοι string

- .upper
- .lower
- .count
- .find
- .replace

Μέθοδοι string

□ .upper

- Επιστρέφει ένα string μετατρέποντας τα τους πεζούς ενός string σε κεφαλαίους χαρακτήρες. Το αρχικό string παραμένει αναλλοίωτο.

□ Πχ έστω:

```
met = 'Γεωπονικό Πανεπιστήμιο Αθηνών'  
print (met.upper ())  
print (met)
```

□ Η εκτέλεση θα έχει ως αποτέλεσμα

```
ΓΕΩΠΟΝΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ  
Γεωπονικό Πανεπιστήμιο Αθηνών
```

Μέθοδοι string

□ `.lower`

- Επιστρέφει ένα string μετατρέποντας τα τους κεφαλαίους χαρακτήρες ενός string σε πεζούς. Το αρχικό string παραμένει αναλλοίωτο.

□ Πχ έστω:

```
met = 'ΓΕΩΠΟΝΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ'  
print(met.lower())  
print(met)
```

□ Η εκτέλεση θα έχει ως αποτέλεσμα

```
Γεωπονικό Πανεπιστήμιο Αθηνών  
ΓΕΩΠΟΝΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ
```

Μέθοδοι string

□ `.count` (όρισμα)

- Σύνταξη : `stringA.count(stringB)`
- Η μέθοδος αυτή δέχεται ως όρισμα ένα `string` πχ `stringB` και επιστρέφει το πλήθος των επαναλήψεων του ορίσματος μέσα στο `stringA`.

□ Πχ έστω:

```
met = 'ΓΕΩΠΟΝΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ'  
print(met.count('Ο'))
```

□ Η εκτέλεση θα έχει ως αποτέλεσμα

3

Μέθοδοι string

□ `.find` (όρισμα)

- Σύνταξη1η : `stringA.find(stringB)`
- Η μέθοδος αυτή δέχεται ως όρισμα ένα `string` πχ `stringB` και επιστρέφει τη θέση του ορίσματος μέσα στο `stringA`.
- Σύνταξη2η : `stringA.find(stringB, pos)`
- Η μέθοδος αυτή δέχεται ως όρισμα ένα `string` πχ `stringB` και έναν ακέραιο αριθμό πχ **pos** και επιστρέφει τη θέση του ορίσματος μέσα στο `stringA` μετά τη θέση **pos** του `stringA`
- Εάν δεν υπάρχει το `stringB` τότε η τιμή που επιστρέφει είναι `-1`

Μέθοδοι string

- **.replace (όρισμα1 , όρισμα2)**
 - Σύνταξη1η : `stringA.find(stringB, stringC)`
 - Η μέθοδος αυτή δέχεται δύο string ορίσματα πχ `stringB` και `stringC` και αντικαθιστά το `stringB` με το `stringC`

Μέθοδοι string

□ `.split` (όρισμα)

- Η μέθοδος αυτή δέχεται ως όρισμα ένα string (συνήθως ένα χαρακτήρα) και δημιουργεί μία λίστα η οποία περιέχει ως στοιχεία τα μέρη του string τα οποία διαχωρίζονται με το όρισμα.

Μέθοδοι string - split

Π.χ έστω το πρόγραμμα

```
alfa="Ιερά Οδός 75 11855 Αθήνα"  
lexeis=alfa.split(" ")  
for lexi in lexeis:  
    print(lexi)
```

Το αποτέλεσμα της επεξεργασίας θα είναι:

Ιερά

Οδός

75

11855

Αθήνα

Συναρτήσεις

□ `len`

- Σύνταξη : `len(string)`
- Η μέθοδος αυτή δέχεται ως όρισμα ένα `string` και επιστρέφει μήκος του `string`

□ `str`

- Σύνταξη : `str(αριθμός)`
- Η μέθοδος αυτή δέχεται ως όρισμα έναν αριθμό και επιστρέφει ένα `string` με περιεχόμενο το αριθμό

Διαμόρφωση εκτύπωσης

- Ο σκοπός της διαμόρφωσης της εκτύπωσης αποτελεσμάτων είναι η καλύτερη από πλευράς παρουσίασης των αποτελεσμάτων μιας επεξεργασίας

Η συνάρτηση `format`

- Η συνάρτηση **`format`** χρησιμοποιείται προκειμένου να διαμορφώσει αριθμητικές τιμές.
- Η σύνταξη της εντολής είναι:
`format(έκφραση, string_διαμόρφωσης)`

όπου:

- **έκφραση είναι μία αριθμητική**
 - σταθερά ή
 - μεταβλητή ή
 - παράσταση που χρησιμοποιεί αριθμητικές μεταβλητές, σταθερές κλπ σε συνδυασμό με αριθμητικούς τελεστές
- **`string_διαμόρφωσης` :** Ένα `string` που δηλώνει το είδος διαμόρφωσης

String διαμόρφωσης	Λειτουργία
' .nf '	Πραγματικός αριθμός με n δεκαδικά ψηφία
' +.nf '	Πραγματικός αριθμός με n δεκαδικά ψηφία και με εμφάνιση του προσήμου
' , '	Πραγματικός ή ακέραιος αριθμός διαχωρίζοντας τις χιλιάδες με (,)
' k , .nf '	Πραγματικός αριθμός με n δεκαδικά ψηφία διαχωρίζοντας τις χιλιάδες με (,) και στοιχισμένος δεξιά σε str που έχει μήκος k
' +.nf '	Πραγματικός αριθμός με n δεκαδικά ψηφία και με εμφάνιση του προσήμου
' k , '	Ακέραιος αριθμός διαχωρίζοντας τις χιλιάδες με (,) και στοιχισμένος δεξιά σε str που έχει μήκος k
' .n% '	Μετατροπή ενός πραγματικού αριθμού σε μορφή ποσοστού με n δεκαδικά και στο τέλος το σύμβολο %
' .ne '	Πραγματικός αριθμός σε εκθετική μορφή με n δεκαδικά ψηφία

Η συνάρτηση format με %

□ Προσοχή

□ Η εντολή `format(z, '%')` όπου:

■ Το σύμβολο **%** δηλώνει ότι η τιμή του `z` θα πολλαπλασιαστεί με 100 και θα εμφανιστεί ως ποσοστό

■ Πχ οι εντολές

```
x=0.15
```

```
print (format(x, '%'), format(x, '%2'))
```

Θα έχουν ως αποτέλεσμα

```
15.000000% 15.00%
```

Αποκοπή και στρογγυλοποίηση πραγματικών αριθμών

- Η διαμόρφωση με τη συνάρτηση `format` όσον αφορά στα δεκαδικά μιας μεταβλητής έχει επίδραση μόνο στην εκτύπωση. Η μεταβλητή εξακολουθεί να έχει τα αρχικά δεκαδικά ψηφία. Στη ουσία η λειτουργία είναι τύπου αποκοπής δεκαδικών ψηφίων

Αποκοπή και στρογγυλοποίηση πραγματικών αριθμών

- Η διαμόρφωση με τη συνάρτηση `format` όσον αφορά στα δεκαδικά μιας μεταβλητής έχει επίδραση μόνο στην εκτύπωση. Η μεταβλητή εξακολουθεί να έχει τα αρχικά δεκαδικά ψηφία. Στη ουσία η λειτουργία είναι τύπου αποκοπής δεκαδικών ψηφίων.

Αποκοπή και στρογγυλοποίηση πραγματικών αριθμών

- Η διαμόρφωση αριθμητικών δεδομένων όσον αφορά στα δεκαδικά σημεία με χρήση της συνάρτησης `format` απλά δεν εμφανίζει τα δεκαδικά στοιχεία.
- Εσωτερικά στις μεταβλητές διατηρούνται τα δεκαδικά ψηφία και στο ενδεχόμενο μίας επαναληπτικής διαδικασίας που αθροίζει αυτούς τους πραγματικούς αριθμούς το άθροισμα ως αποτέλεσμα δεν συνάδει με το άθροισμα των αριθμών όπως παρουσιάζονται στην εκτύπωση.

Στρογγυλοποίηση πραγματικών αριθμών με τη συνάρτηση `round`

- Η συνάρτηση `round` χρησιμοποιείται για την στρογγυλοποίηση πραγματικών αριθμών στο επιθυμητό πλήθος δεκαδικών συντάσσεται ως εξής:
 - `round(έκφραση, πλήθος δεκαδικών)`
 - Όπου έκφραση μπορεί να είναι:
 - Πραγματικός αριθμός
 - Μεταβλητή τύπου `float`
 - Αριθμητική παράσταση

Στρογγυλοποίηση πραγματικών αριθμών με τη συνάρτηση `round`

- Έστω το παρακάτω πρόγραμμα:

```
a= 34.94396998807989
print(a)
print(format(a, '.4f'), a)
a=round(a, 4)
print(a)
```

- Το αποτέλεσμα της εκτέλεσης θα είναι:

```
34.94396998807989
34.9440 34.94396998807989
34.944
```

Στρογγυλοποίηση πραγματικών αριθμών με τη συνάρτηση `round`

□ Έστω το παρακάτω πρόγραμμα:

```
a=23.5276
b=15.3426
print(a*b, round(a*b, 2))
print(format(a*b, '.2f'))
a=round(a, 2)
b=round(b, 2)
print(a*b)
```

Το αποτέλεσμα της εκτέλεσης θα είναι:

```
360.97455576 360.97
360.97
360.9502
```

Η μέθοδος `.format`

- ❑ Η μέθοδος `.format` χρησιμοποιείται για τη μορφοποίηση μιας συμβολοσειράς (`string`) στην Python.
- ❑ Η μέθοδος αυτή δίνει στον προγραμματιστή τη δυνατότητα μορφοποίησης ενός `string` μέσω της εισαγωγής σε συγκεκριμένες θέσεις του `string` διάφορες τιμές.

Η μέθοδος `.format` (1)

- ❑ Η μορφοποίηση ενός `string` επιτυγχάνεται με την χρήση ενός ή περισσότερων πεδίων αντικατάστασης και θέσεων θέσης που ορίζονται από ένα ζεύγος αγκίστρων `{}` μέσα στον `string`.
- ❑ Οι τιμές που τιμή που θέλουμε να εισάγουμε στις αντίστοιχες θέσεις `{}` ορίζονται ως παράμετροι της μεθόδου `format`.
- ❑ **Σύνταξη:** `"{} {}" .format (value1,value2)`
- ❑ Οι παράμετροι `(value1,value2)` μπορεί να είναι σταθερές ή μεταβλητές

Η μέθοδος `.format` (2)

- Η μορφοποίηση ενός `string` επιτυγχάνεται με την χρήση ενός ή περισσότερων πεδίων αντικατάστασης και θέσεων θέσης που ορίζονται από ένα ζεύγος αγκίστρων `{}` μέσα στον `string`.
- Οι τιμές που τιμή που θέλουμε να εισάγουμε στις αντίστοιχες θέσεις `{}` ορίζονται ως παράμετροι της μεθόδου `format`.
- **Σύνταξη:** `"{} {}" .format (var1=timi1,var2=timi2)`
- Όπου (`var1`, `var2`) μπορεί να είναι μεταβλητές και `timi1`, `timi2` είναι μεταβλητές.

Λύση πρωτοβάθμιας εξίσωσης

```
print("Λύση πρωτοβάθμιας εξίσωσης  $ax+b=0$ ")
a=float(input("δώσε το a:"))
b=float(input("δώσε το b:"))
if a != 0:
    x=-b/a

    print("η λύση είναι  $x=$ ", x)
else:
    if b == 0:
        print("η εξίσωση είναι αόριστη")
    else:
        print("η εξίσωση είναι αδύνατη")
```

Λύση πρωτοβάθμιας εξίσωσης με χρήση format & round

```
print("Λύση πρωτοβάθμιας εξίσωσης ax+b=0")
minima= "Η εξίσωση είναι {eidos}"
a=float(input("δώσε το a:"))
b=float(input("δώσε το b:"))
if a != 0:
    x=-b/a
    x=round(x,3)
    print("η λύση είναι x={}".format(x))
else:
    if b == 0:
        typos="αόριστη"
        print(minima.format(eidos=typos))
    else:
        typos="αδύνατη"
        print(minima.format(eidos=typos))
```