

# Τελεστές και εκφράσεις

Οι περισσότερες εντολές που γράφετε θα περιέχουν **εκφράσεις (expressions)**.

Ένα απλό παράδειγμα μίας έκφρασης είναι  $3 + 4$ . Μία έκφραση μπορεί να διαχωριστεί σε **τελεστές (operators)** και **τελεστέους (operands)**.

Οι τελεστές είναι λειτουργίες που κάνουν κάτι και μπορούν να αναπαρασταθούν με σύμβολα όπως το  $+$  ή με ειδικές λέξεις-κλειδιά. Οι τελεστές απαιτούν κάποια δεδομένα πάνω στα οποία θα λειτουργήσουν και αυτά τα δεδομένα ονομάζονται τελεστέοι. Στη συγκεκριμένη περίπτωση, οι τελεστέοι είναι το 3 και το 4.

Τελεστής	Όνομα	Εξήγηση	Παραδείγματα
$+$	Συν	Προσθέτει δύο αντικείμενα.	Το $3 + 2$ δίνει 5. Το $'a' + 'b'$ δίνει $'ab'$ .
$-$	Μείον	Είτε δίνει έναν αρνητικό αριθμό, ή αφαιρεί έναν αριθμό από έναν άλλο.	Το $-2.5$ δίνει έναν αρνητικό αριθμό. Το $50 - 27$ δίνει 23.
$*$	Επί	Δίνει το γινόμενο δύο αριθμών ή μια συμβολοσειρά (string) επαναλαμβανόμενη τόσες φορές.	Το $3 * 3$ δίνει 9. Το $'la' * 3$ δίνει $'lalala'$ .
$**$	Δύναμη	Επιστρέφει το $x$ υψωμένο στη δύναμη $y$ .	Το $3 ** 2$ δίνει 9 (δηλαδή $3 * 3$ ).
$/$	Δια	Διαιρεί το $x$ με το $y$ .	Το $12/4$ δίνει 3
$\%$	Υπόλοιπο	Επιστρέφει το υπόλοιπο της διαίρεσης.	Το $8 \% 3$ δίνει 2. Το $-25.5 \% 2.25$ δίνει 1.5.
$<$	Μικρότερο από	Επιστρέφει το αν το $x$ είναι μικρότερο από το $y$ . Όλοι οι τελεστές σύγκρισης επιστρέφουν True (Αληθής) ή False (Ψευδής). Σημειώστε ότι τα ονόματα αυτά ξεκινούν με κεφαλαίο.	Το $5 < 3$ δίνει False και το $3 < 5$ δίνει True. Οι συγκρίσεις μπορούν να συνδυαστούν αλυσιδωτά κατά βούληση: Το $3 < 5 < 7$ δίνει True.
$!=$	Διαφορετικό	Συγκρίνει αν τα αντικείμενα ΔΕΝ είναι ίσα.	Το $x = 2; y = 3; x != y$ επιστρέφει True.
$>$	Μεγαλύτερο από	Επιστρέφει το αν το $x$ είναι μεγαλύτερο από το $y$ .	Το $5 > 3$ επιστρέφει True. Αν και οι δύο τελεστέοι είναι αριθμοί, πρώτα μετατρέπονται σε έναν κοινό τύπο. Αλλιώς, επιστρέφει πάντα False.
$<=$	Μικρότερο ή ίσο	Επιστρέφει το αν το $x$ είναι μικρότερο από ή ίσο με το $y$ .	Το $x = 3; y = 6; x <= y$ επιστρέφει True.
$>=$	Μεγαλύτερο ή ίσο	Επιστρέφει το αν το $x$ είναι μεγαλύτερο από ή ίσο με το $y$ .	Το $x = 4; y = 3; x >= 3$ επιστρέφει True.
$==$	Ίσο	Συγκρίνει αν τα αντικείμενα είναι ίσα.	Το $x = 2; y = 2; x == y$ επιστρέφει True. Το $x = 'str'; y = 'stR'; x == y$ επιστρέφει False. Το $x = 'str'; y = 'str'; x == y$ επιστρέφει True.

<b>not</b>	Λογικό ΟΧΙ	Αν το x είναι True, επιστρέφει False. Αν το x είναι False, επιστρέφει True	x = True; not x επιστρέφει False.
<b>and</b>	Λογικό ΚΑΙ	Το x and y επιστρέφει False αν το x είναι False, αλλιώς επιστρέφει υπολογίζει και επιστρέφει την τιμή του y.	x = False; y = True; x and y επιστρέφει False αφού το x είναι False. Σε αυτή την περίπτωση, η Python δε θα ελέγξει την τιμή του y αφού γνωρίζει ότι η αριστερή πλευρά της έκφρασης 'and' είναι False που υποδηλώνει ότι ολόκληρη η έκφραση θα είναι False ανεξάρτητα από τις άλλες τιμές. Αυτή η τεχνική αποκαλείται <b>short-circuit evaluation</b> .
<b>or</b>	Λογικό Ή	Αν το x είναι True, επιστρέφει True, αλλιώς υπολογίζει και επιστρέφει την τιμή του y.	Το x = True; y = False; x or y επιστρέφει True. Η Short-circuit evaluation εφαρμόζεται και εδώ.

## Προτεραιότητα πράξεων



Όταν περισσότερα από ένα σύμβολα τελεστών εμφανίζονται σε μία έκφραση, η σειρά υπολογισμού εξαρτάται από τους κανόνες προτεραιότητας.

Οι κανόνες αυτοί είναι σύμφωνοι με τη σειρά εκτέλεσης των αριθμητικών πράξεων που ξέρουμε από τα Μαθηματικά.

Οι παρενθέσεις έχουν τη μεγαλύτερη προτεραιότητα και έτσι χρησιμοποιούνται, για να «αναγκάσουν» την Python να αποτιμήσει μία έκφραση σύμφωνα με τη σειρά που θέλουμε. Εκφράσεις σε παρενθέσεις αποτιμώνται πρώτες. Επίσης, παρενθέσεις χρησιμοποιούνται, για να κάνουν τις εκφράσεις πιο αναγνώσιμες, χωρίς να αλλάξουν το τελικό αποτέλεσμα.

Η ύψωση σε δύναμη έχει την επόμενη μεγαλύτερη προτεραιότητα.

Πολλαπλασιασμός και διαίρεση έχουν την ίδια προτεραιότητα, η οποία είναι μεγαλύτερη από την πρόσθεση και την αφαίρεση που μεταξύ τους έχουν επίσης την ίδια προτεραιότητα.

- Τελεστές με την ίδια προτεραιότητα αποτιμώνται από τα αριστερά προς τα δεξιά.

```
>>> 3*(4+3)
21
>>> 3*4+3
15
>>> (3+2)**(5-3)
25
>>> 3+2**(5-3)
7
>>> 3+2**5-3
32
>>> 15/3+1
6
```

## Εφαρμογή

Δίνεται η παρακάτω έκφραση σε Python

```
(x+y)/(x**3+y**2+1)*z
```

Αν  $x=2$ ,  $y=3$  και  $z=1$  ποιο αποτέλεσμα θα εμφανιστεί στην οθόνη του υπολογιστή;

```
>>> x=2
>>> y=3
>>> z=1
>>> (x+y)/(x**3+y**2+1)*z
0
```

(θα προκύψει  $5/18 \rightarrow 0$  ακέραια διαίρεση)