

Σημειώσεις Τύποι Δεδομένων και Τελεστές στην Python

α) Τύποι Δεδομένων στην Python

Η **Python** υποστηρίζει διάφορους τύπους δεδομένων. Παρακάτω θα δούμε τους βασικότερους:

Τύπος	Ονομασία	Παραδείγματα
Ακέραιοι	int	1, 0, -1, 496
Πραγματικοί	float	3.14159, 0.5, 3.0
Λογικές	bool	True, False
Αλφαριθμητικά	str	"SARS-CoV-2", "Ζάννειο Γυμνάσιο"
Λίστες	list	[6, 28, 496], ["Γη", "Σελήνη", "Ήλιος"]

β) Τελεστές (Αριθμητικοί - Σχεσιακοί - Λογικοί)

Αριθμητικοί τελεστές: είναι τα σύμβολα που χρησιμοποιούμε για να κάνουμε μαθηματικές πράξεις. Ο υπολογισμός κάθε έκφραση στην οποία υπάρχουν αριθμητικοί τελεστές ακολουθεί μια αυστηρά καθορισμένη ιεραρχία πράξεων, που είναι: 1. Ύψωση σε δύναμη. 2. Πολλαπλασιασμός, διαίρεση. 3. Πρόσθεση, αφαίρεση.	Πρόσθεση	+
	Αφαίρεση	-
	Πολλαπλασιασμός	*
	Διαίρεση	/
	Πηλίκo Ακέραιας Διαίρεσης	//
	Ύψωση σε δύναμη	**
	Υπόλοιπο ακέραιας διαίρεσης	%

Ο τελεστής της πρόσθεσης «+» όταν εφαρμόζεται σε αλφαριθμητικά, εκτελεί συνένωση ενώ αυτός του πολλαπλασιασμού «*» παράγει τη συμβολοσειρά επαναλαμβανόμενη τόσες φορές, όσες είναι ο αριθμός. πχ

```
>>> "10"+"10"  
1010  
>>> 5*"10"  
1010101010
```

Οι Σχεσιακοί (ή συγκριτικοί) τελεστές χρησιμοποιούνται για τη σύγκριση εκφράσεων. Αν η σχέση ισχύει το αποτέλεσμα είναι Αληθής (True) αλλιώς Ψευδής (False). Μπορούμε να τις αντιμετωπίζουμε ως ερωτήσεις που απαντώνται με ΝΑΙ ή ΟΧΙ. <pre>>>> 496==496 True >>> 12!=13 True</pre>	Μικρότερο από	<
	Μικρότερο ή ίσο από	<=
	Μεγαλύτερο από	>
	Μεγαλύτερο ή ίσο από	>=
	Ίσο με	==
	Διάφορο από	!=

Οι Λογικοί τελεστές χρησιμοποιούνται όταν θέλουμε να σχηματίσουμε σύνθετες συνθήκες. Η σύζευξη λογικών εκφράσεων είναι Αληθής, αν και μόνον αν είναι αληθείς όλες οι εκφράσεις, ενώ η διάζευξη είναι αληθής, αν συμμετέχει σε αυτήν τουλάχιστον μια αληθής έκφραση. <pre>>>> 12<11 and 23>10 False >>> 12<11 or 23>10 True >>> not(56<=12) True</pre>	Άρνηση (ΟΧΙ)	not
	Σύζευξη (ΚΑΙ)	and
	Διάζευξη (Η)	or

📌 Μερικές Συναρτήσεις

✅ **type:** Χρησιμοποιούμε τη **type()** για να δούμε τον τύπο μιας μεταβλητής:

```
print(type(3.14)) # float
print(type("Hello")) # str
print(type([1, 2, 3])) # list
```

✅ **str:** μετατρέπει ένα αντικείμενο άλλου τύπου σε αλφαριθμητικό

```
>>> str(10)+str(3.14)
103.14
>>> str(5)+str(5)
55
```

✅ **int:** μετατρέπει ένα αντικείμενο σε ακέραιο αριθμό, εάν αυτό είναι εφικτό

```
>>> int("10")*5
50
>>> int("geia")*5
ValueError: invalid literal for int() with base 10: 'geia' at <unknown>
```

✅ **len:** επιστρέφει το πλήθος των χαρακτήρων μιας συμβολοσειράς ή αλλιώς το μήκος μιας συμβολοσειράς.

```
>>> len("python")
6
```