

# Προγραμματισμός, σε Python

Μιχαήλ Ανάργυρος Ζαμάγιας

29 Μαΐου 2020

# Περιεχόμενα

1	Εισαγωγή	3
1.1	Πρόλογος	3
1.2	Γιατί να ασχοληθώ με τον προγραμματισμό;	3
1.3	Γιατί να μάθω Python;	4
1.4	Ποιος είναι ο σκοπός αυτού του εγχειριδίου;	4
2	Τα βασικά	5
2.1	Ξεκινώντας	6
2.1.1	Εγκατάσταση της Python	6
2.1.2	Εγκατάσταση επεξεργαστή κειμένου	6
2.2	Η γλώσσα	7
2.2.1	Διερμηνευτής	7
2.2.2	Εκτέλεση προγραμμάτων	8
2.2.3	Εύρεση κι επίλυση σφαλμάτων	8
2.2.4	Σχόλια	9
2.2.5	Μεταβλητές	10
2.2.6	Τύποι δεδομένων	10
2.2.7	Συμβολοσειρές	10
2.2.8	Λίστες	10
2.2.9	Tuples	10
2.2.10	Σύνολα	10
2.2.11	Λεξικά	10
2.2.12	Προτάσεις υπό συνθήκη	10
2.2.13	Βρόχοι	10
2.2.14	Δομοστοιχεία	10
2.2.15	Δουλεύοντας με αρχεία	10
2.2.16	Δουλεύοντας με JSON αρχεία	10
3	Προγράμματα	11

4	Βιβλιογραφία	12
4.1	Βιβλία . . . . .	13
4.2	Video . . . . .	13
4.3	Σύνδεσμοι . . . . .	13
4.4	Χρήσιμα αρχεία . . . . .	13

# Εισαγωγή

## 1.1 Πρόλογος

Άρχισα να προγραμματίζω, να γράφω Python συγκεκριμένα, στην πρώτη λυκείου ως χόμπι. Είχα χάσει το ενδιαφέρον που είχα για το hardware από το δημοτικό, ενώ παράλληλα κέρδιζε το ενδιαφέρον μου η διαδικασία του να κάνω το hardware χρήσιμο για τον χρήστη, σε αυτή την περίπτωση εμένα.

Στην δευτέρα λυκείου περίπου, είχα καταφέρει να γράψω ένα bot που έτρεχε στο terminal του υπολογιστή μου. Μπορούσε να απαντήσει όλες τις ερωτήσεις που του έκανα, δεδομένου, φυσικά, ότι τις καταλάβαινε. Δεν μπορούσε όμως να κρατήσει διάλογο. Άλλωστε ήταν ένα χαζό bot, όχι κάποια γενική τεχνητή νοημοσύνη.

Ακόμα, είχα γράψει ένα πρόγραμμα «μετεωρολόγο» για μία σχολική εκδρομή, της δευτέρας ή τρίτης λυκείου. Ήταν μεγαλύτερο πρόγραμμα σε σύγκριση με το bot, το οποίο μιλούσε με ένα API<sup>1</sup> για τα δεδομένα και τα εμφάνιζε σε μορφή ευνόητη στον άνθρωπο. Όμως, τα δελτία δεν ήταν ακριβή, συγκριτικά με άλλες υπηρεσίες, με αποτέλεσμα να σταματήσω να ασχολούμαι με αυτό.

Μερικά χρόνια μετά, θέλω να σας εξοικειώσω με τα βασικά της Python μέσω αυτού του εγχειριδίου. Δεν χρειάζονται ανεπτυγμένες γνώσεις προγραμματισμού για να γράψετε κάτι το οποίο θα σας διευκολύνει με τις δουλειές σας στον υπολογιστή, π.χ. την αυτοματοποίηση μιας διαδικασίας.

## 1.2 Γιατί να ασχοληθώ με τον προγραμματισμό;

Η γνώση του προγραμματισμού δίνει σε κάποιον την δεξιότητα να μετατρέπει ένα πρόβλημα σε κάτι που μπορεί να κατανοήσει και να λύσει, τις περισσότερες φορές, ένας υπολογιστής. Μάλιστα, θα υποστήριζα ότι είναι μια βασική δεξιότητα για κάποιον άνθρωπο στην εποχή μας,

---

<sup>1</sup>API: Application Programming Interface ή Διεπαφή Προγραμματισμού Εφαρμογών, είναι η διεπαφή των προγραμματιστικών διαδικασιών που παρέχει ένα λογισμικό προκειμένου να επιτρέπει να γίνονται προς αυτό αιτήσεις από άλλα λογισμικά ή/και ανταλλαγή δεδο

εάν εκείνος χρησιμοποιεί υπολογιστές.

«Codes are a puzzle. A game, just like any other game.»

---

— Alarn Turing

## 1.3 Γιατί να μάθω Python;

Η Python είναι μια δερμηνευόμενη, γενικού σκοπού και υψηλού επιπέδου γλώσσα προγραμματισμού. Δηλαδή:

- μπορεί να εκτελεστεί κώδικας χωρίς να αποτελεί μέρος κάποιου προγράμματος μέσα από τον διερμηνέα,
- βρίσκεται πίσω από πολλές εφαρμογές και προσφέρει λύση σε πολλά προβλήματα και
- το συντακτικό της είναι απλούστερο, ευκολότερο στην κατανόηση σε σύγκριση με άλλες γλώσσες προγραμματισμού.

## 1.4 Ποιος είναι ο σκοπός αυτού του εγχειριδίου;

Στόχος μου μέσα από αυτό το εγχειρίδιο είναι η εξοικείωση του αναγνώστη με την Python και η εισαγωγή του στον τρόπο σκέψης ενός προγραμματιστή. Το εγχειρίδιο χωρίζεται σε δύο μέρη. Στο πρώτο μέρος ο αναγνώστης μαθαίνει για τα βασικά της γλώσσας, βασικές έννοιες προγραμματισμού, είδη δεδομένων και τους τρόπους χρήσης τους. Στο δεύτερο μέρος, ο αναγνώστης θα εφαρμόσει πρακτικά όσα έμαθε στο πρώτο μέρος σε μικρά προγράμματα.

# Τα βασικά

Τα βασικά είναι βαρετά. Επιτρέψτε μου να εξηγήσω. Λόγω της φύσης της γλώσσας δεν χρειάζεστε να επιμείνετε στα βασικά. Είναι αρκετά εύκολο με μια καλή κατανόηση των βασικών «εργαλείων» της γλώσσας να χτίσετε αρκετά πράγματα. Τα βασικά «εργαλεία» σε αυτήν την περίπτωση είναι οι προτάσεις υπό συνθήκη, βρόχοι (αλλιώς επαναλήψεις), συναρτήσεις και τα δομοστοιχεία της γλώσσας. Ο πιο αποδοτικός και γρήγορος τρόπος να εξοικειωθεί ο αναγνώστης με την γλώσσα είναι να ασχοληθεί περισσότερο με το πρακτικό κομμάτι, με project, όπως π.χ. η δημιουργία εργαλείων που θα του γλιτώσει αρκετό χρόνο από συγκεκριμένες ρουτίνες στον υπολογιστή.

## 2.1 Ξεκινώντας

Η Python δεν έρχεται προεγκατεστημένη στα Windows και στις διανομές Linux συνήθως η προεγκατεστημένη έκδοση είναι η 2.7, η οποία πλέον έχει σταματήσει να λαμβάνει ενημερώσεις και υποστήριξη. Άρα, χρειάζεται μια νεότερη έκδοση της Python.

Η Python είναι μια cross-platform γλώσσα προγραμματισμού, με υποστήριξη και στα τρία κύρια λειτουργικά συστήματα (Windows, Linux και MacOS) με μερικές διαφοροποιήσεις ανά το λειτουργικό σύστημα.

Σε αυτό το εγχειρίδιο χρησιμοποιούνται οι Python 3.7.6 και conda 4.8.3 στην διανομή Linux Pop!\_OS. Τέλος, τα προγράμματα στην Python έχουν κατάληξη .py.

### 2.1.1 Εγκατάσταση της Python

Προτείνω στον αναγνώστη να εγκαταστήσει την Anaconda, η οποία είναι μια δωρεάν και ανοιχτού κώδικα διανομή της Python προσφέροντας ένα μεγάλο σύνολο εργαλείων σε ένα σημείο. Η διαδικασία εγκατάστασης παραμένει εύκολη και καθιστά μελλοντικούς πειραματισμούς πιο εύκολους λόγω του πλήθους και ποικιλίας των εργαλείων που περιέχει.

Επισκεφθείτε την [επίσημη σελίδα](#), κατεβάστε το αρχείο εγκατάστασης και ακολουθήστε τις οδηγίες εγκατάστασης για το λειτουργικό σας σύστημα:

- [Windows](#)
- [MacOS](#)
- [Linux](#)

#### Επιβεβαίωση εγκατάστασης

Σε Windows, ανοίξτε το Anaconda Prompt σε Windows ή το terminal σε MacOS και Linux. Η εκτέλεση των εντολών `python --version` και `conda --version`, πρέπει να έχει ένα παρόμοιο αποτέλεσμα με Python 3.7.6 και conda 4.8.3.

### 2.1.2 Εγκατάσταση επεξεργαστή κειμένου

Μπορούμε να γράψουμε Python σε οποιονδήποτε επεξεργαστή κειμένου, ακόμα και στο Σημειωματάριο που έρχεται με τον υπολογιστή μας. Υπάρχουν όμως άλλα προγράμματα τα οποία είναι ειδικά φτιαγμένα για προγραμματισμό, τα οποία φέρουν χαρακτηριστικά που θα κάνουν την ζωή μας πιο εύκολη. Μερικά τέτοια δωρεάν και ανοιχτού κώδικα προγράμματα είναι τα εξής: [Sublime](#), [Atom](#) και [VSCodium](#). Παρακάτω θα δείτε πως να εγκαταστήσετε το VSCodium.

Επισκεφθείτε την [επίσημη σελίδα](#) και ακολουθήστε τις οδηγίες εγκατάστασης για το λειτουργικό σας σύστημα.

### Επιβεβαίωση εγκατάστασης

Σε οποιοδήποτε λειτουργικό σύστημα, αναζητήστε «VSCodium» στην λίστα προγραμμάτων σας. Εάν μπορείτε να το εκτελέσετε από εκεί, το έχετε εγκαταστήσει επιτυχώς. Διαφορετικά ξεκινήστε την διαδικασία εγκατάστασης από την αρχή.

## 2.2 Η γλώσσα

### 2.2.1 Διερμηνευτής

Ο διερμηνευτής της Python επιτρέπει στον χρήστη να εκτελέσει διαδραστικά εντολές Python, κάτι που μπορεί να φανεί χρήσιμο για πειραματισμό ή την δοκιμή κομματιών προγράμματος. Μπορείτε να χρησιμοποιήσετε τον διερμηνευτή,

- ανοίγοντας το Anaconda Prompt στα Windows και εκτελώντας την εντολή python.
- ανοίγοντας το terminal στα Linux και εκτελώντας την εντολή python.

Αυτός είναι ο διερμηνευτής:

---

```
1 Python 3.7.6 (default, Jan 8 2020, 19:59:22)
2 [GCC 7.3.0] :: Anaconda, Inc. on linux
3 Type "help", "copyright", "credits" or "license" for more information.
4 >>>
```

---

Για να εκτυπώσετε την φράση «hello world» στον διερμηνευτή εκτελείτε την εντολή `print("hello world")`:

---

```
1 Python 3.7.6 (default, Jan 8 2020, 19:59:22)
2 [GCC 7.3.0] :: Anaconda, Inc. on linux
3 Type "help", "copyright", "credits" or "license" for more information.
4 >>> print("hello world")
5 hello world
```

---

Για να κλείσετε τον διερμηνευτή, αρκεί να εκτελέσετε `quit()`:

---

```
1 Python 3.7.6 (default, Jan 8 2020, 19:59:22)
2 [GCC 7.3.0] :: Anaconda, Inc. on linux
3 Type "help", "copyright", "credits" or "license" for more information.
4 >>> print("hello world")
5 hello world
6 >>> quit()
```

---



### 2.2.2 Εκτέλεση προγραμμάτων

Τα προγράμματα της Python εκτελούνται τις περισσότερες φορές μέσω του διερμηνευτή. Οπότε, αν έχετε το πρόγραμμα `my_first_program.py`, αρκεί να εκτελέσετε την εντολή `python my_first_program.py` στο terminal:

```
1 python hello_world.py
2 hello world
```

Το πρόγραμμα `my_first_program.py`:

```
1 print("hello world")
```

### 2.2.3 Εύρεση κι επίλυση σφαλμάτων

Καλώς ή κακώς, τα σφάλματα πάνε μαζί με τον προγραμματισμό. Δεν γίνεται να μην συναντήσετε κάποιο σφάλμα στο πρόγραμμά σας με αποτέλεσμα είτε το πρόγραμμά σας να μην εκτελείτε ή είτε να εκτελείτε αλλά να δίνει κάποια έξοδο που δεν περιμένατε. Για την εύρεση και επίλυση σφαλμάτων μπορείτε να δοκιμάσετε τα εξής:

- Όταν υπάρχει ένα σημαντικό σφάλμα σε ένα πρόγραμμα, η Python δημιουργεί ένα `traceback`, δηλαδή μια αναφορά σφαλμάτων. Η Python «ελέγχει» τον κώδικα του προγράμματος και προσπαθεί να εντοπίσει το σφάλμα. Ελέγξτε το `traceback`, μπορεί να σας δώσει μια ιδέα για το τι τυχόν πάει λάθος.
- Κάντε ένα διάλειμμα, απομακρυνθείτε για λίγο από τον υπολογιστή σας. Το συντακτικό είναι πολύ σημαντικό στον προγραμματισμό, και ένα λιγότερο ερωτηματικό, παρένθεση ή αυτάκια (`"`, `'`) θα δημιουργεί σφάλμα στο πρόγραμμά σας. Δείτε ξανά τον κώδικά σας και προσπαθήστε να εντοπίσετε το σφάλμα.
- Ξεκινήστε από την αρχή. Ενώ τις περισσότερες φορές δεν χρειάζεται να απεγκαταστήσετε κάποιο λογισμικό, μπορείτε να γράψετε από την αρχή το πρόγραμμά σας.
- Βρείτε κάποιον που γνωρίζει Python και ζητήστε του να σας βοηθήσει. Ρωτώντας τριγύρω, μπορεί και να βρείτε κάποιον που δεν περιμένατε να γνωρίζει Python.
- Ψάξτε για βοήθεια διαδικτυακά. Έχετε ήδη τον υπολογιστή μπροστά σας, πιθανότατα και σύνδεση στο διαδίκτυο, άρα δεν σας σταματάει τίποτα από το να αναζητήσετε το σφάλμα στο StackOverflow, στην Google ή στο YouTube.

## 2.2.4 Σχόλια

Ένα βασικό και χρήσιμο κομμάτι προγραμματισμού αποτελούν τα σχόλια. Τα σχόλια είναι μέρη του προγράμματος που δεν εκτελούνται και χρησιμοποιούνται από τους προγραμματιστές για να εξηγήσουν την υλοποίηση κάποιας συγκεκριμένης λογικής ή κάποιο κομμάτι του προγράμματος. Επιτρέπουν σε μια πιο αποτελεσματική και γρήγορη κατανόηση ενός προγράμματος ή κομματιού προγράμματος. Κάποιος που ενδιαφέρεται στο πρόγραμμά μπορεί να διαβάσει τα σχόλια που έχετε γράψει για τα κομμάτια του προγράμματός σας και δεν χρειάζεται να τα «εκτελεί» στο μυαλό του.

Υπάρχουν τα σχόλια μίας γραμμής με τα οποία ξεκινούν με `#` και τελειώνουν στο τέλος της γραμμής:

---

```
1 print("hello world")    # to show "hello world"
```

---

Ακόμα υπάρχουν τα σχόλια πολλαπλών γραμμών τα οποία χρησιμοποιούνται για να εξηγήσουν κάποια μεγαλύτερα κομμάτια κώδικα ή συναρτήσεις και ξεκινάνε και τελειώνουν με `"""` ή `'''`. Στην τελευταία περίπτωση ονομάζονται docstring:

---

```
1 print("hello world")    # to show "hello world"
2 """
3 This is multiline comment
4 """
```

---

2.2.5 Μεταβλητές

2.2.6 Τύποι δεδομένων

2.2.7 Συμβολοσειρές

Μορφοποίηση

Μέθοδοι

2.2.8 Λίστες

2.2.9 Tuples

2.2.10 Σύνολα

2.2.11 Λεξικά

2.2.12 Προτάσεις υπό συνθήκη

2.2.13 Βρόχοι

2.2.14 Δομοστοιχεία

2.2.15 Δουλεύοντας με αρχεία

2.2.16 Δουλεύοντας με JSON αρχεία

# Προγράμματα

# Βιβλιογραφία

Total 0 knowledge, entirely parallel with programming of any kind. Heard Python is simple, why would I want to learn it ?

---

u/Azsras\_Zuralix on r/learnpython

## 4.1 Βιβλία

- [Python Crash Course, 2nd Edition — by Eric Matthes](#)

## 4.2 Video

- [Python Crash Course](#)

## 4.3 Σύνδεσμοι

- [WordReference Dictionary](#)
- [Λεξικό της κοινής νεοελληνικής](#)
- [Βιβλιογραφία, Wikipedia](#)
- [Python, Wikipedia](#)
- [Anaconda \(Python distribution\), Wikipedia](#)
- [Anaconda Individual Edition, Anaconda | The World's Most Popular Data Science Platform](#)
- [Conditional statements, Wikipedia](#)
- [Python Cheatsheet](#)
- [VSCodium is a community-driven, freely-licensed binary distribution of Microsoft's editor VSCode](#)

## 4.4 Χρήσιμα αρχεία

- [Βιβλιογραφική ανασκόπηση, Δημοκρίτειο Πανεπιστήμιο Θράκης](#)
- [Εισαγωγή στη LaTeX για φοιτητές. \(An Introduction to Latex in Greek\)](#)
- [Python Cheat Sheet](#)