

Προγραμματισμός σε C++ & Python & Εφαρμογές στη Ναυπηγική & Ναυτική Μηχανολογία ΣΝΜΜ 2019

Γεώργιος Παπαλάμπρου
Επίκουρος Καθηγητής ΕΜΠ
george.papalambrou@lme.ntua.gr

Εργαστήριο Ναυτικής Μηχανολογίας (Κτίριο Λ)
Σχολή Ναυπηγών Μηχανολόγων Μηχανικών
Εθνικό Μετσοβίο Πολυτεχνείο

February 27, 2019

Περιεχόμενα

Περιεχόμενο Μαθήματος

- ▶ Εβδομάδα 1. A. Εισαγωγή. Η γλώσσα. Το περιβάλλον Linux. Command line. Python interpreter. Ιστοσελίδα μαθήματος. Βιβλιογραφία. Editors: Sublime, Spyder
B. Εισαγωγή στην γλώσσα Python. Debugging. Libraries. Hello World. juPyter (on line)
- ▶ Εβδομάδα 2. A. Data types. Loops. Control. File I/O
B. Παραδείγματα.
- ▶ Εβδομάδα 3. A. Functions. Modules
B. OOP. Classes
- ▶ Εβδομάδα 4. A. Βιβλιοθήκες NymPy, SciPy.
Errors-Exceptions
B. Παραδείγματα: Γραμμική άλγεβρα, Γραφικά
- ▶ Εβδομάδα 5. Εφαρμογή: Neural Networks. Machine Learning
- ▶ Εβδομάδα 6. Εφαρμογή: Hardware. Πλατφόρμες.
Πρωτόκολλα. Βασικό I/O

Γλώσσες προγραμματισμού

- ▶ Interpreted Programming Languages (LISP, Lua, ...)
- ▶ Functional Programming Languages (F#, Haskell, ...)
- ▶ Compiled Programming Languages
- ▶ Procedural Programming Languages (Matlab, ...)
- ▶ Scripting Programming Languages
- ▶ Markup Programming Languages (HTML, ...)
- ▶ Logic-Based Programming Languages (Prolog, ...)
- ▶ Concurrent Programming Languages
- ▶ Object-Oriented Programming Languages (SmallTalk, ...)

Γλώσσες compiled vs interpreted

- ▶ Οι γλώσσες προγραμματισμού εν γένει εμπίπτουν σε μία από τις δύο κατηγορίες: **compiled - interpreted**.
- ▶ Με compiled (μεταγλωττισμένη) γλώσσα, ο κώδικας που εισάγετε μειώνεται σε ένα σύνολο εντολών που αφορούν συγκεκριμένες μηχανές (CPU), προτού αποθηκευτούν ως εκτελέσιμο αρχείο.
- ▶ Πχ: η πρόσθεση "+" στον source code γίνεται εντολή "ADD" σε machine code.
- ▶ Με interpreted γλώσσες, ο κώδικας αποθηκεύεται στην ίδια μορφή που τον εισάγατε.
- ▶ Τα προγράμματα compiled γενικά τρέχουν ταχύτερα, επειδή τα interpreted προγράμματα μετατρέπονται σε γλώσσα μηχανής κατά τον χρόνο εκτέλεσης.
- ▶ Οι πιο γνωστές: compiled -> C/C++, (Fortran).
Interpreted -> Perl, Python

Γλώσσες compiled vs interpreted

- ▶ Ωστόσο, με μια interpreted γλώσσα, μπορείτε να κάνετε πράγματα που δεν μπορούν να γίνουν σε μια μεταγλωττισμένη γλώσσα.
- ▶ Παράδειγμα: τα interpreted προγράμματα μπορούν να τροποποιηθούν προσθέτοντας ή αλλάζοντας λειτουργίες κατά τη διάρκεια εκτέλεσης.
- ▶ Επίσης, είναι συνήθως πιο εύκολο να αναπτύξετε εφαρμογές σε ένα interpreted περιβάλλον επειδή δεν χρειάζεται να κάνετε compile την εφαρμογή σας κάθε φορά που θέλετε να δοκιμάσετε ένα μικρό τμήμα.

Γλώσσα interpreted scripting

- ▶ Γλώσσα scripting: γλώσσα προγραμματισμού του ΗΥ που χρησιμοποιείται σε αρχεία script, τα οποία ελέγχονται και εκτελούνται απευθείας από ένα περιβάλλον γραμμή-γραμμή (interpreted), συνήθως για την εκτέλεση εργασιών.
- ▶ Υπάρχουν πολλές τέτοιες γλώσσες, όπως Bash, Perl, Lua και Python που μπορούν να χρησιμοποιηθούν για την αυτοματοποίηση της εκτέλεσης εργασιών σε ΗΥ & embedded controllers, όπως η έναρξη (boot), η διαχείριση του συστήματος, η αλληλεπίδραση-διασύνδεση με ηλεκτρονικά εξαρτήματα, κλπ.

Παράδειγμα Bash Shell (*nix)

- ▶ Έλεγχος εξόδου (LED on) με κλήση του script **./bashLED** στο cmd line του Linux

```
#!/bin/bash
LED_GPIO=4          # Use a variable -- easy to change GPIO number

# An example Bash functions
function setLED
{
    # $1 is the 1st argument passed to this function
    echo $1 >> "/sys/class/gpio/gpio$LED_GPIO/value"
}

# Start of the program -- start reading from here
if [ $# -ne 1 ]; then          # if there is not exactly one argument
    echo "No command was passed. Usage is: bashLED command,"
    echo "where command is one of: setup, on, off, status and close"
    echo -e " e.g., bashLED setup, followed by bashLED on"
    exit 2                    # error that indicates invalid number of arguments
fi

echo "The LED command that was passed is: $1"
if [ "$1" == "setup" ]; then
    echo "Exporting GPIO number $1"
    echo $LED_GPIO >> "/sys/class/gpio/export"
```


Εισαγωγή

- ▶ Python: Ιδανική για τις περιπτώσεις που χρειάζονται πιο σύνθετη δομή προγράμματος και είναι πιθανό ότι αυτά θα τροποποιηθούν στο μέλλον.
- ▶ Όπως και η Lua, η Python υποστηρίζει το OOP (Object Oriented Programming) και δυναμικούς τύπους δεδομένων (dynamic typing).
- ▶ Dynamic typing σημαίνει ότι δεν ορίζεται τύπος (πχ, integer, character, string) σε μία μεταβλητή, αλλά η μεταβλητή "θυμάται" τον τύπο της.
- ▶ Πχ αν ορισθεί η $x = 5$, η μεταβλητή x συμπεριφέρεται ως integer (ακέραιος). Αν αλλάξει σε $x = \text{"test"}$, τότε η μεταβλητή x θα συμπεριφέρεται ως string.
- ▶ Statically typed γλώσσες: C/C+ , Java: Δεν το επιτρέπουν.

Η γλώσσα

- ▶ Η Python είναι μια υψηλού επιπέδου γλώσσα προγραμματισμού η οποία δημιουργήθηκε από τον Ολλανδό Guido van Rossum το 1990.
- ▶ Τύπος γλώσσας: interpreted.
- ▶ Ο κύριος στόχος της είναι η αναγνωσιμότητα του κώδικά της και η ευκολία χρήσης της και το συντακτικό της επιτρέπει στους προγραμματιστές να εκφράσουν έννοιες σε λιγότερες γραμμές κώδικα απ'ότι θα ήταν δυνατόν σε γλώσσες όπως η C++ ή η Java.
- ▶ Διακρίνεται λόγω του ότι έχει πολλές βιβλιοθήκες που διευκολύνουν ιδιαίτερα αρκετές συνηθισμένες εργασίες και για την ταχύτητα εκμάθησής της.

Βασική Πηγή: <https://el.wikipedia.org/wiki/Python>

Η γλώσσα

- ▶ Οι interpreters (διερμηνευτές ?) της Python είναι διαθέσιμοι για εγκατάσταση σε πολλά λειτουργικά συστήματα, επιτρέποντας στην Python την εκτέλεση κώδικα σε ευρεία γκάμα συστημάτων.
- ▶ Χρησιμοποιώντας εργαλεία τρίτων, όπως το Py2exe ή το Pyinstaller, ο κώδικας της Python μπορεί να μετατραπεί σε αυτόνομα εκτελέσιμα προγράμματα για μερικά από τα πιο δημοφιλή λειτουργικά συστήματα, επιτρέποντας τη διανομή του βασισμένου σε Python λογισμικού για χρήση σε αυτά τα περιβάλλοντα χωρίς να απαιτείται εγκατάσταση του interpreter της Python.

Βασική Πηγή: <https://el.wikipedia.org/wiki/Python>

Η γλώσσα

- ▶ Η Python αναπτύσσεται ως ανοιχτό λογισμικό (open source) και η διαχείρισή της γίνεται από τον μη κερδοσκοπικό οργανισμό Python Software Foundation. Βασίζεται στην γλώσσα C.
- ▶ Ο κώδικας διανέμεται με την άδεια Python Software Foundation License η οποία είναι συμβατή με την GPL.
- ▶ Το όνομα της γλώσσας προέρχεται από την ομάδα Άγγλων κωμικών Monty Python.

Βασική Πηγή: <https://el.wikipedia.org/wiki/Python>

Ο οργανισμός www.python.org

- ▶ <http://www.python.org>
- ▶ Εδώ: εκδόσεις γλώσσας, documentation, εφαρμογές, επιτυχίες...

The screenshot shows the Python.org website. At the top, there's a navigation bar with links: Python, PSF, Docs, PyPI, Jobs, and Community. Below this is the Python logo and a search bar. A secondary navigation bar contains links: About, Downloads, Documentation, Community, Success Stories, News, and Events. The main content area features a code snippet on the left and an article titled "All the Flow You'd Expect" on the right. The code snippet demonstrates a for loop that calculates the product of numbers in a list. The article discusses Python's control flow statements. At the bottom, there's a footer with four sections: "Get Started", "Download", "Docs", and "Jobs".

Python Software Foundation (US) | <https://www.python.org>

Python PSF Docs PyPI Jobs Community

python™

Donate Search GO Socialize

About Downloads Documentation Community Success Stories News Events

```
# For loop on a list
>>> numbers = [2, 4, 6, 8]
>>> product = 1
>>> for number in numbers:
...     product = product * number
...
>>> print('The product is:', product)
The product is: 384
```

All the Flow You'd Expect

Python knows the usual control flow statements that other languages speak — *if*, *for*, *while* and *range* — with some of its own twists, of course. [More control flow tools in Python 3](#)

1 2 3 4 5

Python is a programming language that lets you work quickly and integrate systems more effectively. [>>> Learn More](#)

Get Started
Whether you're new to

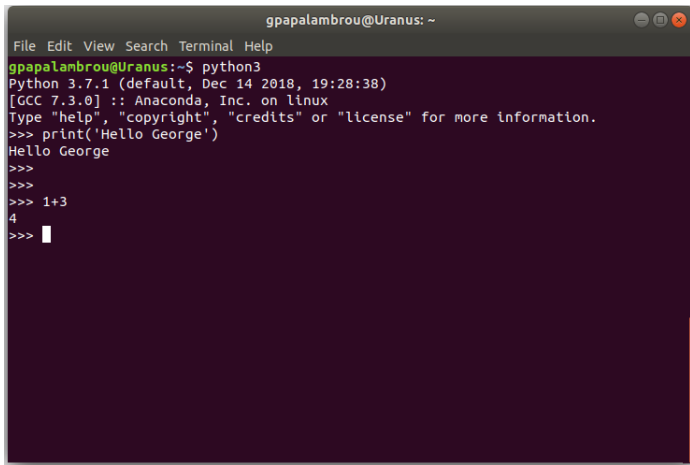
Download
Python source code and

Docs
Documentation for Python's

Jobs
Looking for work or have a

Python Interpreter

- ▶ Η γλώσσα έχει δύο εκδόσεις: 2.X και 3.X με διαφορές μεταξύ τους
- ▶ Τώρα πλέον μόνον η έκδοση 3.X (3.7.1)



```
gpapalambrou@Uranus: ~  
File Edit View Search Terminal Help  
gpapalambrou@Uranus:~$ python3  
Python 3.7.1 (default, Dec 14 2018, 19:28:38)  
[GCC 7.3.0] :: Anaconda, Inc. on linux  
Type "help", "copyright", "credits" or "license" for more information.  
>>> print('Hello George')  
Hello George  
>>>  
>>>  
>>> 1+3  
4  
>>> █
```

Επιδόσεις

- ▶ Ενδεικτική σύγκριση της γλώσσας Python με άλλες γλώσσες
- ▶ C/C++ > Java/Js ≫ Python
- ▶ Οι γλώσσες επιλέγονται με βάση την εφαρμογή

VALUE	TYPE	RPi 3 at 1.2 GHz ¹	RPi 2 at 1 GHz ²	RPi B+ at 1 GHz ³	64-BIT i7 PC ⁴
C/C++	Compiled	1.00 × (6.5s)	1.00 × (9.3s)	1.00 × (10.0s)	1.00 × (0.61s)
C++11	Compiled	1.06 × (6.9s)	0.69 × (6.4s)	0.70 × (7.03s)	0.95 × (0.58s)
Haskell	Compiled	1.16 × (7.6s)	1.17 × (10.8s)	1.07 × (10.8s)	1.15 × (0.70s)
Java ⁵	JIT	1.52 × (9.94s)	1.45 × (13.4s)	2.29 × (23.0s)	1.36 × (0.83s)
Mono C#	JIT	2.72 × (17.8s)	2.47 × (22.9s)	3.62 × (36.4s)	2.16 × (1.32s)
Cython ⁶	Compiled	2.74 × (17.9s)	2.67 × (24.8s)	2.80 × (28.0s)	1.26 × (0.77s)
Node.js ⁷	JIT	2.76 × (18.1s)	6.23 × (57.7s)	50.1 × (503s)	6.54 × (3.99s)
Lua	Interpreted	20.2 × (132s)	21.2 × (197s)	25.7 × (258s)	34.3 × (20.9s)
Cython	Compiled	64.2 × (420s)	66.6 × (618s)	163 × (1633s)	58.0 × (34.4s)
Perl	Interpreted	92.6 × (601s)	81.5 × (756s)	171 × (1716s)	82.0 × (50.0s)
Python	Interpreted	94.1 × (616s)	89.9 × (834s)	198 × (1992s)	89.7 × (54.7s)
Ruby	Interpreted	147 × (962s)	140 × (1298s)	265 × (2662s)	47.4 × (28.9s)

Figure: D. Molloy, Exploring Raspberry Pi: Interfacing to the Real World with Embedded Linux

To Linux είναι εδώ

... Linux is like air—you don't feel it or know it even exists, but you need it all the same. From your addicting Android smartphones and all the apps you love in it, to Air traffic control stations and even the Large Hadron Collider, everything runs on Linux. For those who didn't know, Linux is an open source operating system based on UNIX.

Το περιβάλλον Linux

Linux

From Wikipedia, the free encyclopedia

This article is about the family of operating systems. For the kernel, see [Linux kernel](#). For other uses, see [Linux \(disambiguation\)](#).

Linux (/ˈlɪnəks/ (listen) *LIN-əks*)^{[9][10]} is a family of [free and open-source software operating systems](#) based on the [Linux kernel](#),^[11] an [operating system kernel](#) first released on September 17, 1991 by [Linus Torvalds](#).^{[12][13][14]} Linux is typically [packaged](#) in a [Linux distribution](#) (or *distro* for short).

Distributions include the Linux kernel and supporting [system software](#) and [libraries](#), many of which are provided by the [GNU Project](#). Many Linux distributions use the word "Linux" in their name, but the [Free Software Foundation](#) uses the name *GNU/Linux* to emphasize the importance of GNU software, causing some [controversy](#).^{[15][16]}

Popular Linux distributions^{[17][18][19]} include [Debian](#), [Fedora](#), and [Ubuntu](#). Commercial distributions include [Red Hat Enterprise Linux](#) and [SUSE Linux Enterprise Server](#). Desktop Linux distributions include a [windowing system](#) such as [X11](#) or [Wayland](#), and a [desktop environment](#) such as [GNOME](#) or [KDE Plasma](#). Distributions intended for [servers](#) may omit [graphics](#) altogether, and include a [solution stack](#) such as [LAMP](#). Because Linux is freely redistributable, anyone may create a distribution for any purpose.

Linux was originally developed for [personal computers](#) based on the [Intel x86](#) architecture, but has since been [ported](#) to more [platforms](#) than any other operating system.^[20] Linux is the leading operating system on [servers](#) and other [big iron](#) systems such as [mainframe computers](#), and the only OS used on [TOP500 supercomputers](#) (since November 2017, having gradually eliminated all competitors).^{[21][22][23]} It is used by around 2.3 percent of [desktop computers](#).^{[24][25]} The [Chromebook](#), which runs the Linux kernel-based [Chrome OS](#), dominates the US K–12 education market and represents nearly 20 percent of sub-\$300 [notebook](#) sales in the US.^[26]

Linux also runs on [embedded systems](#), i.e. devices whose [operating system](#) is typically built into the [firmware](#) and is highly tailored to the system. This includes [routers](#), [automation controls](#), [televisions](#),^{[27][28]} [digital video recorders](#), [video game consoles](#), and [smartwatches](#).^[29] Many smartphones and [tablet computers](#) run [Android](#) and other Linux derivatives.^[30] Because of the dominance of Android on [smartphones](#), Linux has the [largest installed base](#) of all [general-purpose operating systems](#).^[31]

Linux is one of the most prominent examples of free and open-source [software](#) collaboration. The [source code](#) may be used, modified and distributed—commercially or non-commercially—by anyone under the terms of its respective licenses, such as the [GNU General Public License](#).

Linux



Tux the penguin, mascot of Linux^[1]

Developer

Community
[Linus Torvalds](#)

Written in

[C](#) and others

OS family

[Unix-like](#)

Working state

Current

Source model

[Open-source](#)

Initial release

September 17,
1991; 27 years
ago

Marketing target

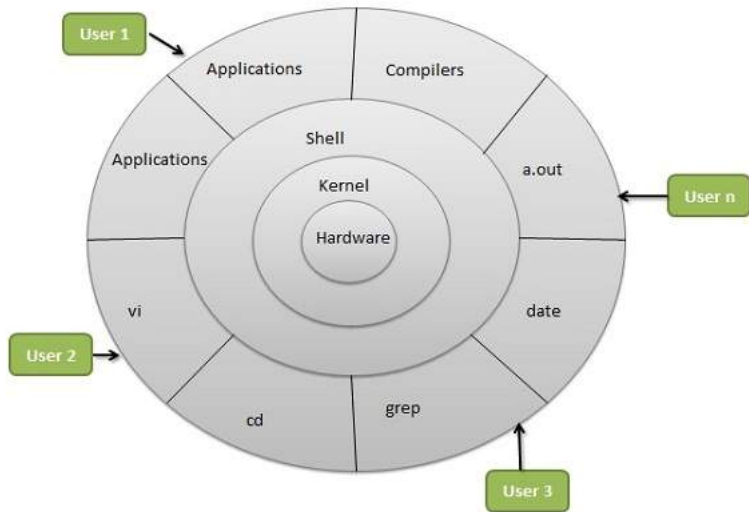
[Cloud computing](#),
[embedded devices](#),
[mainframe computers](#),
[mobile devices](#),
[personal computers](#),

Το περιβάλλον Linux - Δομή

Various layers within Linux, also showing separation between the **userland** and **kernel space**

User mode	User applications	For example, bash, LibreOffice, GIMP, Blender, 0 A.D., Mozilla Firefox, etc.				
	Low-level system components:	System daemons: <i>systemd, runit, logind, networkd, PulseAudio, ...</i>	Windowing system: <i>X11, Wayland, SurfaceFlinger (Android)</i>	Other libraries: <i>GTK+, Qt, EFL, SDL, SFML, FLTK, GNUstep, etc.</i>	Graphics: <i>Mesa, AMD Catalyst, ...</i>	
	C standard library	<i>open(), exec(), sbrk(), socket(), fopen(), calloc(), ... (up to 2000 subroutines)</i> <i>glibc</i> aims to be POSIX/SUS-compatible , <i>uClibc</i> targets embedded systems, <i>bionic</i> written for Android , etc.				
Kernel mode	Linux kernel	<i>stat, splice, dup, read, open, ioctl, write, mmap, close, exit, etc. (about 380 system calls)</i> The Linux kernel System Call Interface (SCI) aims to be POSIX/SUS-compatible				
		Process scheduling subsystem	IPC subsystem	Memory management subsystem	Virtual files subsystem	Network subsystem
		Other components: ALSA, DRI, evdev, LVM, device mapper, Linux Network Scheduler, Netfilter Linux Security Modules: SELinux, TOMOYO, AppArmor, Smack				
Hardware (CPU, main memory, data storage devices, etc.)						

Το περιβάλλον Linux - Δομή 2



Linux distributions - top 10



Το λειτουργικό σύστημα Unix

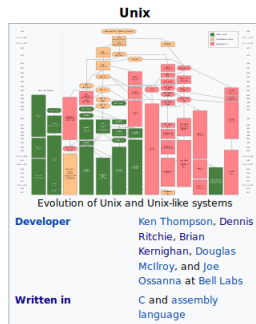
Unix

From Wikipedia, the free encyclopedia

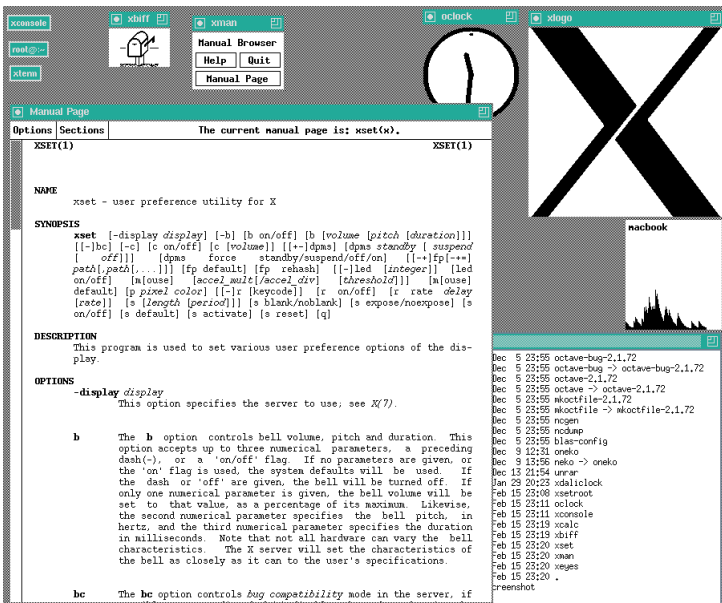
Unix (/ˈjuːnɪks/; trademarked as **UNIX**) is a family of **multitasking**, **multiuser** computer **operating systems** that derive from the original **AT&T Unix**, development starting in the 1970s at the **Bell Labs** research center by **Ken Thompson**, **Dennis Ritchie**, and others.^[3]

Initially intended for use inside the **Bell System**, AT&T **licensed** Unix to outside parties in the late 1970s, leading to a variety of both academic and commercial Unix variants from vendors including **University of California, Berkeley (BSD)**, **Microsoft (Xenix)**, **IBM (AIX)**, and **Sun Microsystems (Solaris)**. In the early 1990s, AT&T sold its rights in Unix to **Novell**, which then sold its Unix business to the **Santa Cruz Operation (SCO)** in 1995.^[4] The UNIX trademark passed to **The Open Group**, a neutral industry consortium, which allows the use of the mark for certified operating systems that comply with the **Single UNIX Specification (SUS)**. As of 2014, the Unix version with the largest installed base is **Apple's macOS**.^[citation needed]

Unix systems are characterized by a **modular design** that is sometimes called the "**Unix philosophy**". This concept entails that the operating system provides a set of simple tools that each performs a limited, well-defined function,^[5] with a unified **filesystem** (the **Unix filesystem**) as the main means of communication,^[3] and a **shell** scripting and command language (the **Unix shell**) to combine the tools to perform complex workflows. Unix distinguishes itself from its predecessors as the first **portable** operating system: almost the entire operating system is written in the **C programming language**, thus allowing Unix to reach numerous platforms.^[6]



Το λειτουργικό σύστημα Unix - X



Linux Command line

<https://help.ubuntu.com/community/UsingTheTerminal>Search

Ubuntu.comCommunitySupportPartners

Official DocumentationCommunity Help WikiContribute

ubuntu[®] documentation

Page HistoryLogin to edit

Search

UsingTheTerminal

Why use the terminal?

"Under Linux there are GUIs (graphical user interfaces), where you can point and click and drag, and hopefully get work done without first reading lots of documentation. The traditional Unix environment is a CLI (command line interface), where you type commands to tell the computer what to do. That is faster and more powerful, but requires finding out what the commands are."

~ from **man intro(1)**

This page gives an introduction to using the command-line interface terminal, from now on abbreviated to the **terminal**. There are many varieties of Linux, but almost all of them use similar commands that can be entered from the terminal.

There are also many graphical user interfaces (GUIs), but each of them works differently and there is little standardization between them. Experienced users who work with many different Linux distributions therefore find it easier to learn commands that can be used in all varieties of Ubuntu and, indeed, in other Linux distributions as well.

For the novice, commands can appear daunting:

```
sudo gobbledegook blah_blah -w -t -h --long-switch  
aWkward/ComBinationOf/mixedCase/underscores_strokes/and.dots
```

However, it is important to note that even experienced users often cut and paste commands (from a guide or manual) into the terminal; they do not memorize them.

It is important, of course, to know how to use the terminal - and anyone who can manage typing, backspacing, and cutting and pasting will be able to use the terminal (it is not more difficult than that).

Contents

1. Why use the terminal?
2. Starting a terminal
 1. In Unity
 2. In GNOME
 3. In Xfce (Xubuntu)
 4. In KDE (Kubuntu)
 5. In LXDE (Lubuntu)
3. Commands
 1. sudo: Executing Commands with Administrative Privileges
 2. File & Directory Commands
 3. Running a File Within a Directory
 4. System Information Commands
 5. Adding A New User
4. Options
5. "Man" and getting help
 1. Searching the manual pages
6. Other Useful Things
 1. Prettier Manual Pages
 2. Pasting in commands
 3. Save on typing
 4. Change the text
7. More ways to run a terminal
8. An extremely handy tool :: incremental history searching
9. How to create upsidedown and/or reverse text with

Linux Command line

```
Activities Terminal
/home/gpapalambrou/Documents/docs_processing/courses_NTUA/lesson_1
File Edit Idexif Tools LaTeX Math Wizards Bibliography Macros View Options Help
gpapalambrou@Uranus: ~
File Edit View Search Terminal Help
gpapalambrou@Uranus:~$
gpapalambrou@Uranus:~$ ls -all
total 2242056
drwxr-xr-x 83 gpapalambrou gpapalambrou 4096 0eß 26 16:19 .
drwxr-xr-x 4 root root 4096 Iav 28 17:36 ..
drwxr-xr-x 3 gpapalambrou gpapalambrou 4096 Iouλ 19 2018 .ac6
drwxr-xr-x 3 gpapalambrou gpapalambrou 4096 Iouλ 18 2018 Ac6
drwxr-xr-x 24 gpapalambrou gpapalambrou 4096 0eß 12 13:30 anaconda3
-rw-r--r-- 1 gpapalambrou gpapalambrou 0 Iav 24 20:25 a.out
-rw-rw-r-- 1 gpapalambrou gpapalambrou 142 Iav 12 11:32 .apport-ignore.xml
drwxr-xr-x 2 gpapalambrou gpapalambrou 4096 0eß 6 19:22 .arduino
drwxrwxr-x 8 gpapalambrou gpapalambrou 4096 0eß 9 08:52 Arduino
drwxrwxr-x 4 gpapalambrou gpapalambrou 4096 0eß 16 08:15 .arduino15
drwxr-xr-x 3 gpapalambrou gpapalambrou 4096 0eß 7 10:07 arduino-1.8.8-linux64
-rw----- 1 gpapalambrou gpapalambrou 31629 0eß 26 13:35 .bash_history
-rw-r--r-- 1 gpapalambrou gpapalambrou 220 0eß 27 2018 .bash_logout
-rw-r--r-- 1 gpapalambrou gpapalambrou 4373 0eß 12 13:30 .bashrc
-rw-r--r-- 1 gpapalambrou gpapalambrou 3771 0eß 12 13:30 .bashrc-anaconda3.bak
-rw-r--r-- 1 gpapalambrou gpapalambrou 273 Iav 18 13:03 bbb_take_picture.txt
drwx----- 31 gpapalambrou gpapalambrou 4096 0eß 22 17:30 .cache
drwxr-xr-x 6 gpapalambrou gpapalambrou 4096 Σen 6 18:07 CCS8.2.0.00007_linux-x64
-rw-rw-r-- 1 gpapalambrou gpapalambrou 805030053 Noe 10 18:01 CCS8.2.0.00007_linux-x64.tar.gz
drwxr-xr-x 36 gpapalambrou gpapalambrou 4096 0eß 27 09:10 .config
-rw----- 1 gpapalambrou gpapalambrou 138735616 Maλ 9 2018 core
drwx----- 2 gpapalambrou gpapalambrou 4096 Iouλ 25 2018 .cups
drwx----- 3 root root 4096 IouV 12 2018 .dbus
drwxr-xr-x 12 gpapalambrou gpapalambrou 4096 0eß 26 16:52 Desktop
drwxr-xr-x 14 gpapalambrou gpapalambrou 4096 0eß 26 20:05 Documents
drwxr-xr-x 3 gpapalambrou gpapalambrou 4096 0eß 26 15:34 Downloads
drwx----- 7 gpapalambrou gpapalambrou 4096 0eß 26 16:19 .dropbox
drwx----- 32 gpapalambrou gpapalambrou 4096 0eß 14 11:59 Dropbox
drwxr-xr-x 3 gpapalambrou gpapalambrou 4096 0eß 20 01:03 .dropbox-dist
drwxr-xr-x 4 gpapalambrou gpapalambrou 4096 Maλ 8 2018 eagle
-rw-r--r-- 1 gpapalambrou gpapalambrou 133907 Maλ 10 2018 .eaglerc
drwxr-xr-x 5 gpapalambrou gpapalambrou 4096 Iouλ 10 2018 .eaglerc
```

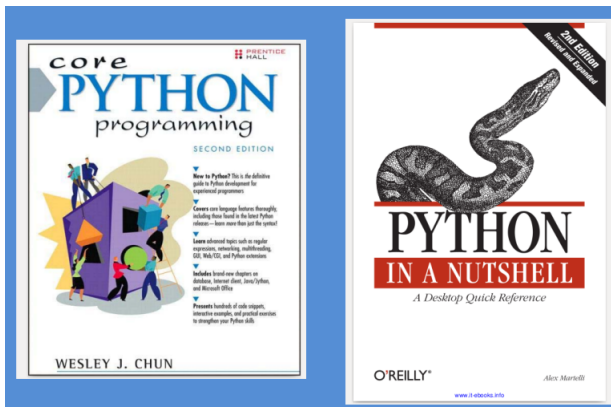

Linux - Τα βασικά

- ▶ What is the kernel?
- ▶ What is the bash / shell in Linux?
- ▶ What are file permissions in Linux?
- ▶ What is sudo?
- ▶ What are Editors in Linux?
- ▶ What is CLI(Command Line Interface) ?
- ▶ How to find help on Linux?

Βιβλιογραφία

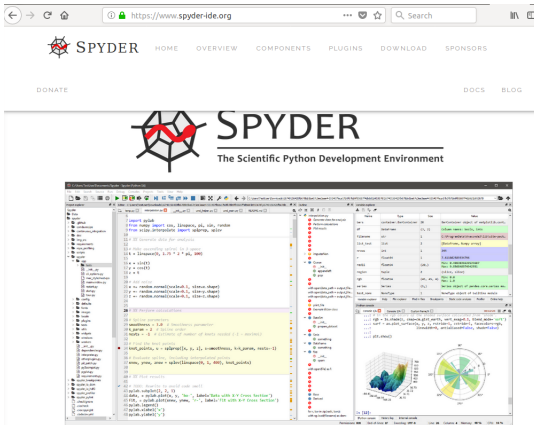
Το υλικό των παραδόσεων προέρχεται από τα εξής:

- ▶ Core Python Programming (2nd Edition), Wesley Chun
- ▶ Python in a Nutshell (2nd Edition), Alex Martelli



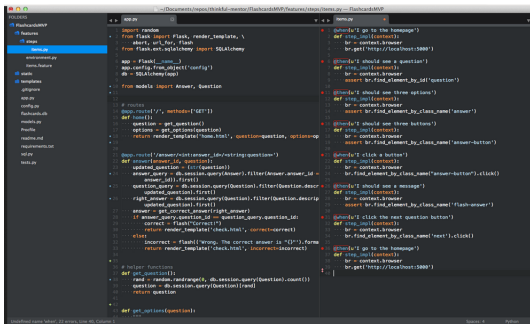
Editors

- ▶ Το περιβάλλον **Spyder** για να κατεβάσετε και να εγκαταστήσετε, <http://www.spyder-ide.org>
- ▶ Γραφικό περιβάλλον ανάπτυξης



Editors

- ▶ Το περιβάλλον **Sublime** για να κατεβάσετε ως εφαρμογή από το Linux
- ▶ Γραφικό (?) περιβάλλον ανάπτυξης



```
import random
from flask import Flask, render_template, \
    abort, url_for, flash
from flask_sqlalchemy import SQLAlchemy

app = Flask(__name__)
app.config.from_object('config')
db = SQLAlchemy(app)

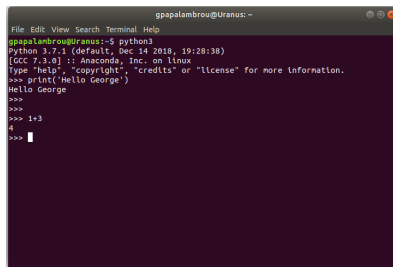
from models import Answer, Question

# Routes
@app.route('/', methods=['GET'])
def index():
    question = get_question()
    options = get_options(question)
    return render_template("home.html", question=question, options=options)

@app.route('/answer/
```

Editors

- ▶ Το περιβάλλον **Python 3** στο Command line του Ubuntu Linux
- ▶ Περιβάλλον ανάπτυξης: (Γραμμή εντολών) Command line



```
gpapalambrou@Uranus: ~  
File Edit View Search Terminal Help  
gpapalambrou@Uranus:~$ python3  
Python 3.7.1 (default, Dec 14 2018, 19:28:38)  
[GCC 7.3.0] :: Anaconda, Inc. on linux  
Type "help", "copyright", "credits" or "license" for more information.  
>>> print('Hello George')  
Hello George  
>>>  
>>>  
>>> 1+3  
4  
>>> 
```

Ιστοσελίδα μαθήματος - 1/2

- Αρχική Ιστοσελίδα ENM: <http://www.lme.ntua.gr>

You are here: Home / Academic Info / Προσφερόμενα Μαθήματα / Προγραμματισμός σε C++ & Python & Εφαρμογές στη Ναυπηγική & Ναυτική Μηχανολογία

Προγραμματισμός σε Python & Εφαρμογές στη Ναυτική Μηχανολογία

Μάθημα: Προγραμματισμός σε C/C++ & Python & Εφαρμογές στη Ναυπηγική & Ναυτική Μηχανολογία

ΕΞΑΜΗΝΟ: 8ο

ΚΩΔΙΚΟΣ: 8.XX.XX.X

ΔΙΔΑΣΚΟΝΤΕΣ: X. [C/C++] Παπαδόπουλος, Α. Γκίνης, [Python] Γ. Παπαλάμπρου

ΒΟΗΘΟΙ: [Python] Ν. Πλανιάσης

Γ. Παπαλάμπρου, Επικ. Καθηγητής ΕΜΠ

email: george.papalambrou@lme.ntua.gr

Κτίριο Λ, 1ος όροφος, γραφείο Α4, Τηλ. 210-7721793

Διαθέσιμες ώρες γραφείου: Τετάρτη 12:00-13:00, Παρασκευή 15:00-16:00.

Ώρες Μαθήματος: Τετάρτη 12:45-14:30, Παρασκευή 08:45-09:30.

Αίθουσα διδασκαλίας: 106, Κτίριο Δ.

Διάρκεια εξαμήνου 2018-2019: έναρξη 18/2/2019, λήξη 31/5/2019.

Ανακοινώσεις:

-

Περιεχόμενα μαθήματος: Εισαγωγή. Η γλώσσα. Το περιβάλλον Linux. Command line. Python compiler. Ιστοσελίδα μαθήματος. Βιβλιογραφία. Editors: Sublime, Spyder. Εισαγωγή στην γλώσσα Python. Debugging. Libraries. Πρώτο πρόγραμμα: "Hello World". Περιβάλλον jupyter (on line). Data types. Loops. Control. File I/O. Functions. Modules. OOP. Classes. Βιβλιοθήκες Numpy, SciPy. Exceptions. Παραδείγματα: Γραμμική άλγεβρα, Γραφικά. Εφαρμογή 1: Hardware. Πλατφόρμες. Πρωτόκολλα. AIN/DIN. Threads. Εφαρμογή 2: NN/ML.

* Τα περιεχόμενα του μαθήματος Python υπάρχουν και στην ιστοσελίδα GitHub: https://github.com/gpapalambrou/course_Python

Περιεχόμενα μαθήματος Python

1. Εισαγωγή [[pdf](#)], ενημέρωση: 20/2/2019

Ιστοσελίδα μαθήματος - 2/2

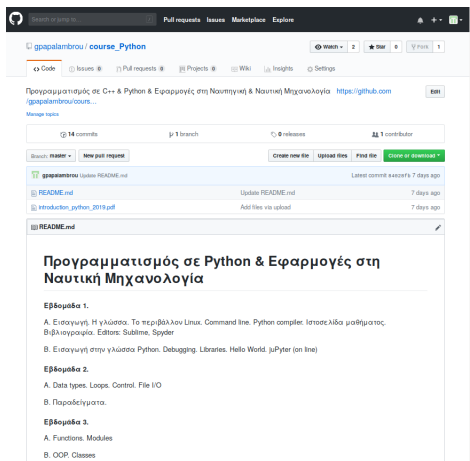
- Slides & codes: <https://github.com/gpapalambrou>

The screenshot shows the GitHub profile page for user **gpapalambrou**. The browser address bar displays <https://github.com/gpapalambrou>. The profile header includes a green and white logo, the name **gpapalambrou**, and a bio in Greek: "Προγραμματιστής σε C++ & Python & Εφαρμογές στις Ναυτιλιακές & Ναυτικές Μηχανολογίες". A green button labeled "Edit profile" is visible. The "Overview" tab is selected, showing pinned repositories: **course_Python** (described in Greek) and **SAE** (described in Greek). Below this, a "47 contributions in the last year" section features a calendar heatmap with green squares indicating activity. The "Contribution activity" section for February 2019 lists two events: "Created 9 commits in 1 repository" (for `gpapalambrou/course_Python`) and "Created 2 repositories" (`gpapalambrou/course_Python` and `gpapalambrou/baloon_experiment`).

Ιστοσελίδα μαθήματος - 2/2

► Slides & codes:

https://github.com/gpapalambrou/course_Python



The screenshot shows the GitHub interface for the repository 'gpapalambrou / course_Python'. The repository description is 'Προγραμματισμός σε C++ & Python & Εφαρμογές στη Ναυπηγική & Ναυτική Μηχανολογία'. The repository has 14 commits, 1 branch, 0 releases, and 1 contributor. The commit history shows three recent commits: 'Update README.md' (7 days ago), 'Add files via upload' (7 days ago), and 'Add files via upload' (7 days ago). The selected file is 'README.md', which contains the following content:

Προγραμματισμός σε Python & Εφαρμογές στη Ναυτική Μηχανολογία

Εβδομάδα 1.

A. Εισαγωγή. Η γλώσσα. Το περιβάλλον Linux. Command line. Python compiler. Ιστοσελίδα μαθήματος. Βιβλιογραφία. Editors: Sublime, Spyder

B. Εισαγωγή στην γλώσσα Python. Debugging. Libraries. Hello World. Jupyter (on line)

Εβδομάδα 2.

A. Data types. Loops. Control. File I/O

B. Παράδειγματα.

Εβδομάδα 3.

A. Functions. Modules

B. OOP. Classes

Επόμενα βήματα

- ▶ Εγκαταστήστε Ubuntu Linux: Full ή σε Virtual Machine
- ▶ Εγκαταστήστε κάποιον από τα Spyder, Sublime, command line (Anaconda)
- ▶ Ξεκινήστε με ένα απλό πρόγραμμα σε Python
- ▶ Μελέτη βιβλίων