Εικόνα που περιέχει γραφικά, γραφιστική, στιγμιότυπο οθόνης, καρτούν

Περιγραφή που δημιουργήθηκε αυτόματα

**ΨΣ-732-ΠΔΙ ΕΚΠΑΙΔΕΥΤΙΚΗ ΤΕΧΝΟΛΟΓΙΑ**

**ΦΥΛΛΟ ΕΡΓΑΣΙΑΣ ΑΤΟΜΙΚΗΣ ΕΡΓΑΣΙΑΣ**

Θέμα:Σχεδίαση και Κατασκευή Εκπαιδευτικού Ιστοχώρου Μικρο-μαθήματος για Ανάπτυξη Ψηφιακών Ικανοτήτων Πολιτών.

Ονοματεπώνυμο: Μπούζι Αλεξάνδρα

ΑΜ: Ε21116

Διδάσκοντες καθηγητές: Σάμψων Δ.

Γκότζος Δ.

1. **Γενικές πληροφορίες Μικρο-μαθήματος**
   1. **Τίτλος Μικρο-μαθήματος :**

**Fundamentals of Programming in Python**

* 1. **Δημιουργός Μικρο-μαθήματος:**

**Bouzi Alexandra, Undergraduate in the department of Digital Systems, UniPi**

* 1. **Συνοπτική Περιγραφή Μικρο-μαθήματος:**

**This micro-course is tailored for teens and young adults who are eager to learn the basics of Python. Designed for self-learners, it enables students to start coding their own simple projects. In just 3 hours, you can begin your journey into programming!**

1. **Διάρκεια Μικρο-μαθήματος:**

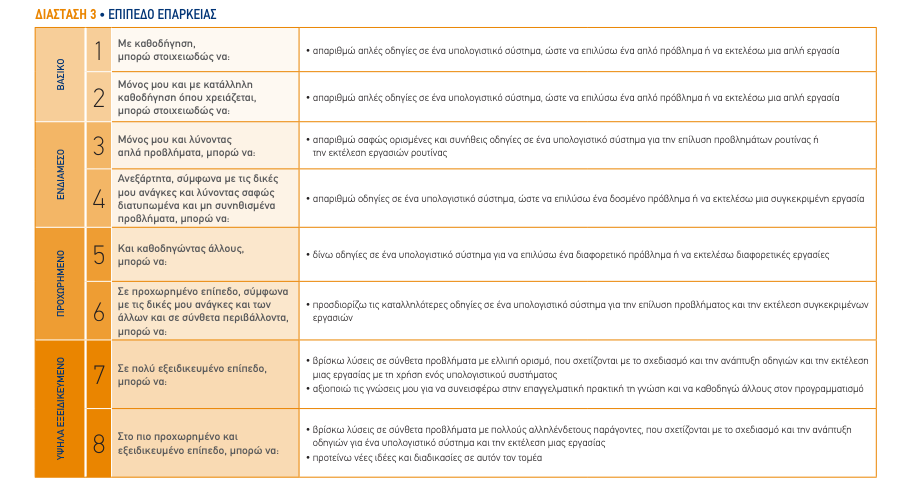
**Micro-Course duration : 3 hours, can be successfully finished in just a day .**

1. **Mαθησιακά αποτελέσματα (ΜΑ) Μικρο-μαθήματος:**

**Ψηφιακή ικανότητα «Προγραμματίζω υπολογιστικά συστήματα» - Επίπεδο Επάρκειας: ΠΡΟΧΩΡΗΜΈΝΟ 5.**

**Ο/Η εκπαιδευόμενος/η μετά την παρακολούθηση του μικρό-μαθήματος θα είναι ικανός/η να:**





**ΜΑ1.1 Κατανοεί πλήρως τις βασικές λειτουργίες και εντολές της εφαρμογής προγραμματισμού Python .**

**MA1.2 Μπορεί να χρησιμοποιήσει επιτυχώς τις δομές ελέγχου ροής για την επίλυση προβλημάτων.**

**MA1.3 Μπορεί να χρησιμοποιεί την εφαρμογή προγραμματισμού Python για την σύνθεση απλών project.**

**ΜΑ1.4 Κατανοεί σε επαρκές επίπεδο την αναγκαιότητα και την χρήση συναρτήσεων.**

**4.Αξιολόγηση**

**ΕργΑξιολογ1: [Αξιολόγηση μέσω αυτοματοποιημένων απαντήσεων] που ο εκπαιδευόμενος καλείται να απαντήσει σε μια σειρά ερωτήσεων ‘Σωστό/Λάθος’ ή επιλογής της σωστής απάντησης – αξιολογεί το ΜΑ1.1**

**ΕργΑξιολογ2: [Αυτοαξιολογήση με τη χρήση Open Response Assessment] που ο εκπαιδευόμενος βαθμολογείται ο ίδιος με τη βοήθεια ρουμπρίκας. Ο εκπαιδευόμενος καλείται να γράψει ένα απλό πρόγραμμα ,στο google collab-python πεδίο, το οποίο θα απαντάει στις ερωτήσεις που του ανατέθηκαν – αξιολογεί το ΜΑ1.2**

**ΕργΑξιολοη3: [Αξιολόγηση μέσω συνδυασμού αυτοματοποιημένων απαντήσεων και διδακτικού προσωπικού]. Ο εκπαιδευόμενος καλείται να γράψει κώδικα για δύο πιο σύνθετο προβλήματα. Στην περίπτωση που το αποτέλεσμα είναι σωστό δεν θα χρειαστεί παρέμβαση ειδικού. Αν όμως υπάρχουν σφάλματα θα παρέμβει ένας προγραμματιστής για την καθοδήγηση του εκπαιδευόμενου άμεσα – αξιολογεί το ΜΑ1.3 και ΜΑ1.4**

**5.Προαπαιτούμενες Ικανότητες Εκπαιδευομένων**

**Το μικρο-μάθημα αφορά την απόκτηση της Ψηφιακής Ικανότητας 3.4 « Προγραμματίζω υπολογιστικά συστήματα» στο επίπεδο επάρκειας ΠΡΟΧΩΡΗΜΕΝΟ 5. Ο/Η εκπαιδευόμενος/η θα πρέπει να είναι ήδη ικανός/η να:**

**Με καθοδήγηση, να μπορεί στοιχειωδώς να:**

**• απαριθμεί απλές οδηγίες σε ένα υπολογιστικό σύστημα, ώστε να επιλύσει ένα απλό πρόβλημα ή να εκτελέσει μια απλή εργασία. (3.4.1)**

**Μόνος του και με κατάλληλη καθοδήγηση όπου χρειάζεται, μπορεί στοιχειωδώς να:**

**• απαριθμεί απλές οδηγίες σε ένα υπολογιστικό σύστημα, ώστε να επιλύσει ένα απλό πρόβλημα ή να εκτελέσει μια απλή εργασία. (3.4.2)**

**Μόνος του και λύνοντας απλά προβλήματα, μπορεί να:**

**• απαριθμεί σαφώς ορισμένες και συνήθεις οδηγίες σε ένα υπολογιστικό σύστημα για την επίλυση προβλημάτων ρουτίνας ή την εκτέλεση εργασιών ρουτίνας. (3.4.3)**

**Ανεξάρτητα, σύμφωνα με τις δικές του ανάγκες και λύνοντας σαφώς**

**διατυπωμένα και μη συνηθισμένα προβλήματα, μπορεί να:**

**• απαριθμεί οδηγίες σε ένα υπολογιστικό σύστημα, ώστε να επιλύσει ένα δοσμένο πρόβλημα ή να εκτελέσει μια συγκεκριμένη εργασία. (3.4.4)**

**6. Γραφική αναπαράσταση εκπαιδευτικού σχεδιασμού Μικρο-μαθήματος**

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, διάγραμμα, σχεδίαση

Περιγραφή που δημιουργήθηκε αυτόματα

**7. Αναλυτική περιγραφή Μικρο-μαθήματος**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ΙD Δραστηριότητας** | | **Περιγραφή Εκπαιδευτικής Δραστηριότητας** | | **Ψηφιακές Τεχνολογίες** | | | **Εκτίμηση Ενδεικτικής Χρονικής Διάρκειας Εκπαιδευτικής**  **Δραστηριότητας** | | | **Σύνδεση με Μαθησιακά Αποτελέσματα** | | | |
| 1. **Γενική Επισκόπηση του Μικρό-Μαθήματος (διαθέσιμες πληροφορίες πριν την εγγραφή** | | | | | | | | | | | | | |
| **0.1** | | **Γενική Επισκόπηση μικρό-μαθήματος Μια παράγραφος** | | | **Υπερκείμενο** | | |  | | |  | | |
| **0.2** | | **Τι θα μάθετε: περιγραφή μαθησιακών αποτελεσμάτων** | | | **Υπερκείμενο,**  **hyperlinks** | | |  | | |  | | |
| **0.3** | | **Πώς θα μάθετε: Σύντομη αναφορά στους τρόπους και**  **εργαλεία μάθησης** | | | **Checklist** | | |  | | |  | | |
| **0.4** | | **Πως θα πιστοποιηθείτε: Σύντομη περιγραφή του τρόπου**  **αξιολόγησης και πιστοποίησης** | | | **Yπερκείμενο** | | |  | | |  | | |
| **0.5** | | **Σχετικά με την Εκπαιδεύτρια** | | | **Εικόνα και Υπερκείμενο** | | |  | | |  | | |
| 1. **Eισαγωγή στο μικρό-μάθημα 20’** | | | | | | | | | | | | |
| **0.1** | | **[Παρουσίαση] Εισαγωγή-Σκοπός- Λόγοι επιλογής της Python**  **Παρουσίαση του σκοπού του μαθήματος** | | | **Υπερκείμενο** | | | **5’** | | |  | |
| **0.2** | | **[Παρουσίαση] Χρήσιμες πληροφορίες πριν την έναρξη του μικρο-μαθήματος.**  **Τι είναι ο προγραμματισμός και πως δημιουργήθηκε η γλώσσα προγραμματισμού Python** | | | **Υπερκείμενο**  **Video** | | | **15’** | | |  | |
| **2: Θεματική Ενότητα (125’)**   |  |  |  |  |  | | --- | --- | --- | --- | --- | | **1.1** | **[Eπίδειξη] Βήματα εγκατάστασης εφαρμογής**  **[Παρουσίαση] Θεματικής Ενότητας για το μαθησιακά αποτέλεσμα 1**  **Έναρξη με την Python/ Eισαγωγή σε βασικές έννοιες (όπως μεταβλητή)** | **Video Presentations**  **(για download και για arithmetic και logical operators)**  **Υπερκείμενο**  **(επεξήγηση των operations, types)**  **Step by step guide (για την επισήμανση των βημάτων εγκατάστασης της εφαρμογής)** | **35’** | **ΜΑ1.1 Κατανοεί πλήρως τις βασικές λειτουργίες και εντολές της εφαρμογής προγραμματισμού Python .** | | **1.2** | **[Παρουσίαση-Επίδειξη] Control Flow structures**  **[Άσκηση] Σχετικά με την γνώση που αντλήθηκε για τα Control Flow Structures** | **Υπερκείμενο**  **Video**  **(για την παράδοση πληροφορίων σχετικά με τα if, loops)**  **Self-Assessment through Exercise (άσκηση χωρίς βαθμολόγηση)** | **35’** | **MA1.2 Μπορεί να χρησιμοποιήσει επιτυχώς τις δομές ελέγχου ροής για την επίλυση προβλημάτων.** | | **1.3** | **[Παρουσίαση-Επίδειξη]**  **Παράδοση γνώσεων σχετικά με τις λίστες και τους πίνακες.**  **[Άσκηση]**  **Μια απλή άσκηση σχετικά με την επίδειξη.** | **Video ( για απλές επιλύσεις ασκήσεων, Arrays, Lists)**  **Self-Assessment through Exercise (άσκηση χωρίς βαθμολόγηση)** | **33’** | **MA1.3 Μπορεί να χρησιμοποιεί την εφαρμογή προγραμματισμού Python για την σύνθεση απλών project.** | | **1.4** | **[Παρουσίαση-Επίδειξη] Functions**  **[Άσκηση]**  **Μια απλή άσκηση σχετικά με την ενσωμάτωση μεταβλητών στις συναρτήσεις** | **Υπερκείμενο, Video (για τα functions ,πως χρησιμοποιούνται και τις λειτουργίες τους )**  **Self-Assessment through Exercise (άσκηση χωρίς βαθμολόγηση)** | **22’** | **ΜΑ1.4 Κατανοεί σε επαρκές επίπεδο την αναγκαιότητα και την χρήση συναρτήσεων** |  1. **Αξιολόγηση και πιστοποίηση (30’)** | | | | | | | | | | | |
| **2.1** | **[Τελική Αξιολόγηση-Πιστοποίηση]** | | **Interactive Quiz Site**  **(για την αντικειμενική και**  **συνολική αξιολόγηση των μαθησιακών**  **αποτελεσμάτων του Mικρο-**  **μαθήματος).** | | | **30’** | | |  | | |

**Μέρος Β. Κατασκευή Εκπαιδευτικού Ιστοτόπου Μικρο-Μαθήματος [7.50/10.00]**

**url:** [**https://pythownpacecom.wordpress.com/**](https://pythownpacecom.wordpress.com/)

**Αναφορές**

**Το υλικό που περιλαμβάνεται στον εκπαιδευτικό ιστότοπο Mικρο-μαθήματος** [**https://pythownpacecom.wordpress.com/**](https://pythownpacecom.wordpress.com/)

**είναι πρωτότυπο εκτός από:**

**Main menu:** [**https://techvify-software.com/35-best-coding-programming-quotes/**](https://techvify-software.com/35-best-coding-programming-quotes/)

**Introduction: What is Python?** [**https://www.python.org/doc/essays/blurb/**](https://www.python.org/doc/essays/blurb/)

**Why Learn Python?** [**https://www.simplilearn.com/tutorials/python-tutorial/why-learn-python**](https://www.simplilearn.com/tutorials/python-tutorial/why-learn-python)

**Before we start.. :** <https://youtu.be/FCMxA3m_Imc>

<https://youtu.be/Kn3-8Nq2Ddo>

**Unit 1:**<https://youtu.be/hEgO047GxaQ>

<https://youtu.be/8cAEH1i_5s0>

<https://youtu.be/XfBBeyBOzF0>

<https://youtu.be/W7luvtXeQTA>

**Unit 2:**<https://youtu.be/n9TQZIiFiQs>

[**http://fizyka.umk.pl/~jacek/docs/javatutorial/java/nutsandbolts/QandE/questions\_flow.html**](http://fizyka.umk.pl/~jacek/docs/javatutorial/java/nutsandbolts/QandE/questions_flow.html)

**Unit 3:** [**https://www.youtube.com/watch?v=YmOcN1phX9Q**](https://www.youtube.com/watch?v=YmOcN1phX9Q)

[**https://www.youtube.com/watch?v=Rldzskbnjgo**](https://www.youtube.com/watch?v=Rldzskbnjgo)

[**https://www.youtube.com/watch?v=8dRjtZVYKTQ**](https://www.youtube.com/watch?v=8dRjtZVYKTQ)

[**https://www.practicepython.org/solution/2014/02/15/02-odd-or-even-solutions.html**](https://www.practicepython.org/solution/2014/02/15/02-odd-or-even-solutions.html)

**Unit 4:**<https://youtu.be/zvzjaqMBEso>

[**https://www.freelearningpoints.com/python/exercises/function/answer-1**](https://www.freelearningpoints.com/python/exercises/function/answer-1)

**Assessment-Certification:**

[**https://www.dataquest.io/blog/free-python-quiz/**](https://www.dataquest.io/blog/free-python-quiz/)

**2**

**Μέρος Β.1**

**Ολοκληρωμένη έκδοση του μαθήματος σε μορφή κειμένου**

**Tίτλος Μικρο-Μαθήματος: PythonPace- Fundamentals of Python**

**Δημιουργός Μικρο-μαθήματος : Αλεξάνδρα Μπούζι**

**«Προγραμματίζω υπολογιστικά συστήματα» - Επίπεδο Επάρκειας: ΠΡΟΧΩΡΗΜΈΝΟ 5.**

**Aρχή Σελίδας**

1. Γενική Επισκόπηση του Μικρο-μαθήματος

## 0.1 Ιmportant Information

##### Are you eager to learn programming in a quick and easy way? Discover how to get started with Python through our 3-hour micro-course «Python Basics». This course guides you through the fundamental concepts of Python programming with engaging exercises, informative videos, and detailed articles. Start your journey into the world of coding today with our user-friendly and interactive course material.

## 0.2 Description of the Learning Outcomes

###### [Unit 1](https://pythownpacecom.wordpress.com/%cf%85%cf%80%ce%b7%cf%81%ce%b5%cf%83%ce%af%ce%b5%cf%82/)

###### The self-learning student will be able to successfully download the Python app and learn about the different types of operations, how to execute the written code and the rules that variable names must follow.

###### [Unit 2](https://pythownpacecom.wordpress.com/%cf%80%ce%b5%cf%81%ce%af/)

###### Control Flow Operations ,such as if statements and loops, will be discussed with an abundance of examples.

###### [Unit](https://pythownpacecom.wordpress.com/%cf%80%ce%b5%cf%81%ce%af/) 3

###### A simple code snippet will be provided for the learner to pore over. After that, they will be able to tackle their own first simple coding project.

###### [Unit 4](https://pythownpacecom.wordpress.com/1-4-functions/)

###### The self paced student will learn about the importance and the usability of functions, for keeping the code readable to others!

## 0.3 Brief Overview of Learning Methods and Tools

###### The course includes presentations and demonstrations of the learning objectives mentioned above. Participants will then practice and self-assess through their engagement in corresponding activities. The following types of activities are utilized in this process:

###### ✓ **Video Presentations and Presentation Decks: Engaging visual content to explain key concepts and provide an overview of topics.**

###### ✓ **Video Demonstrations:** Step-by-step demonstrations to show the practical application of coding techniques.

###### ✓ **Step-by-Step Guides:** Detailed instructions to help learners follow along and implement coding tasks.

## 0.4 Brief Description of Assessment and Certification

###### The assessment in this course is conducted through a combination of hands-on coding projects and simple exercises such as true/false questions. The final certification will be provided through a variety of coding exercises.

## 0.5 About the Instructor

Εικόνα που περιέχει άτομο, ανθρώπινο πρόσωπο, ρουχισμός, φρύδι

Περιγραφή που δημιουργήθηκε αυτόματα

###### Alexandra Bouzi is currently pursuing a degree in the Digital Systems at the University of Piraeus, being in her 3rd year of undergraduate studies. Alongside her academic pursuits, she holds notable qualifications, including a proficiency degree in English and a Goethe certification in German. Her academic journey is marked by consistently high performance, demonstrating a keen understanding of the course material. With a passion for teaching and a commitment to fostering a supportive learning environment, she ensures that students receive comprehensive guidance and support throughout their self-learning educational journey.

Τέλος σελίδας

Αρχή σελίδας

# Introduction

Εικόνα που περιέχει ρουχισμός, άτομο, ανθρώπινο πρόσωπο, υπολογιστής

Περιγραφή που δημιουργήθηκε αυτόματαΕικόνα που περιέχει σκίτσο/σχέδιο, ζωγραφιά, τέχνη με γραμμές, clipart

Περιγραφή που δημιουργήθηκε αυτόματα

## 0.1Micro-course Goals

**Our Python micro-course, designed to be completed in just three hours, provides a comprehensive introduction to the fundamentals of Python programming. Learners will start with the basic concepts, including variables, data types, and control structures, which lay the groundwork for understanding how Python works. As the course progresses, participants will delve into functions, learning how to define and call them, and exploring parameters and return values to create reusable code. Additionally, the course covers essential manipulation operations, such as loops and if statements. By the end of the micro-course, learners will have gained practical experience and a solid understanding of Python, equipping them with the skills to write and understand basic scripts and to pursue more advanced Python programming on their own.**

## 0.2 What is Python?

#### Python is an interpreted, high-level programming language that supports object-oriented principles and features dynamic semantics. Its advanced data structures, combined with dynamic typing and binding, make it ideal for rapid application development and serve well as a scripting or glue language for integrating different components. Python’s syntax is straightforward and easy to grasp, enhancing readability and lowering maintenance costs. It supports modular programming with modules and packages, promoting code reuse. The Python interpreter and its extensive standard library are freely available in both source and binary forms across all major platforms, making distribution and accessibility easy and cost-free.

## 0.3 Why Python?

### 1. Career Opportunities and Salary

**Python offers numerous job opportunities and promises significant growth with substantial salary potential**.

### 3. Easy Use

**Python has a simple syntax and hence is easy to understand and learn. Thus, making it a popular pick when it comes to programming languages.**

### 2. Web Development

**Python provides a vast collection of frameworks that makes it much easier for developers to develop web applications.**

### 4. Data Science

**Python is renowned for its robustness, scalability, and extensive visualization and graphics capabilities, making it a popular choice in the field of Data Science. It supports a variety of widely-used libraries.**

**Τέλος Σελίδας**

**Αρχή Σελίδας**

# Before we start..

## What is «Programming»?

##### It is important that before we start we are fully aware of what programming is! Programming is the process of creating a set of instructions that tell a computer how to perform a task.

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, γραμματοσειρά, αριθμός

Περιγραφή που δημιουργήθηκε αυτόματα

https://youtu.be/FCMxA3m\_Imc

## How was Python created?

##### Watch this short video to find out!

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, καρτούν, Κινούμενα σχέδια

Περιγραφή που δημιουργήθηκε αυτόματα

<https://youtu.be/Kn3-8Nq2Ddo>

Τέλος σελίδας

Αρχή σελίδας

# 1.1 Python Basics

## Overview

In this Unit, you will be introduced to the fundamental concepts of Python Programming! A solid foundation for beginners to build upon will be provided.

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, λειτουργικό σύστημα, λογισμικό

Περιγραφή που δημιουργήθηκε αυτόματα

## Steps for Setting Up Python

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, γραμματοσειρά, έγγραφο

Περιγραφή που δημιουργήθηκε αυτόματα

## Python Basics

**Execute Python**: Python syntax can be executed by writing directly in the Command Line:

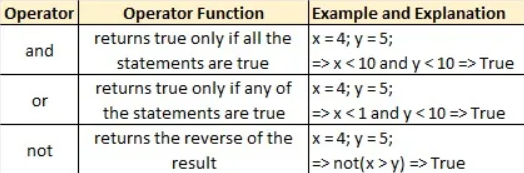
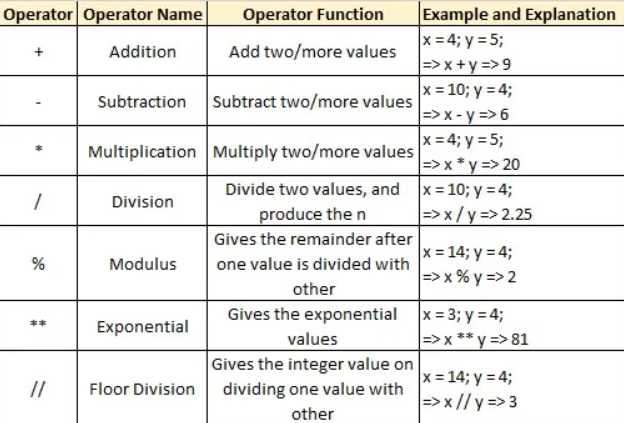
Εικόνα που περιέχει γραμματοσειρά, κείμενο, στιγμιότυπο οθόνης, γραφικά

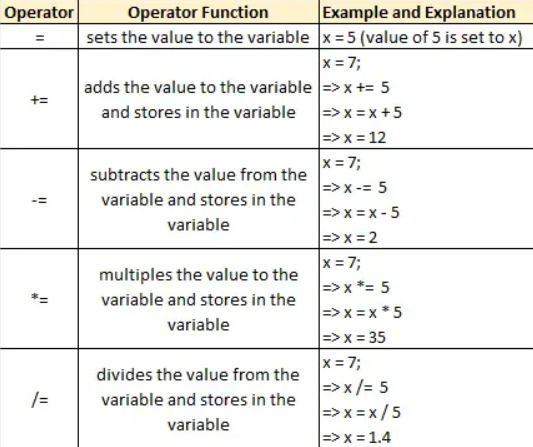
Περιγραφή που δημιουργήθηκε αυτόματα

Or by creating a python file on the server, using the **.py** file extension, and running it in the Command Line:



**Operators:** Operators are special symbols used to perform logical and arithmetic operations on values and variables. There are various types of operators in Python, including arithmetic, comparison, logical, identity, membership, bitwise, and assignment operators.





Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, γραμματοσειρά, σχεδίαση

Περιγραφή που δημιουργήθηκε αυτόματα

<https://youtu.be/XfBBeyBOzF0>

**https://youtu.be/W7luvtXeQTA**

**Python Indentation**: Indentation refers to the spaces at the beginning of a code line. Where in other programming languages the indentation in code is for readability only, the indentation in Python is very important. Python uses indentation to indicate a block of code.

Try running the block of code:

Εικόνα που περιέχει κείμενο, γραμματοσειρά, γραμμή, λευκό

Περιγραφή που δημιουργήθηκε αυτόματα

**Python Variables:**In the Python programming language, there is no specific command or statement required to declare a variable. Instead, you simply assign a value to a name, which then becomes a variable. This assignment is carried out using the equals sign (=). Once a variable has been assigned a value, you can use it throughout your Python code for further calculations or manipulations

**Example** n = 300

This can be read or interpreted as “n is assigned the value 300.” Once n is assigned a value, it can be used in statements or expressions, and its assigned value will be substituted.

Εικόνα που περιέχει κείμενο, γραμματοσειρά, στιγμιότυπο οθόνης, σχεδίαση

Περιγραφή που δημιουργήθηκε αυτόματα

You can see that an integer object is created using the built-in type() function.

Example **Input**

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, λογισμικό, σχεδίαση

Περιγραφή που δημιουργήθηκε αυτόματα

## Practice

Answer these true/false questions based on the Python Basics you got taught.

1.**True or False: In Python, the operator // performs floor division.** (Highlight the area below to reveal the answer + explanation)

Answer: **True**

Explanation: The double forward slash operator (//) in Python is used for floor division, which means it returns the largest integer less than or equal to the division result of two numbers.

2.**True or False: Variables in Python must be explicitly declared before they can be used.** (Highlight the area below to reveal the answer + explanation)

Answer: False

Explanation: Unlike some other programming languages, in Python, variables do not need to be explicitly declared before they are assigned a value. Variables are created when a value is assigned to them.

3.**True or False: The + operator is used for addition in Python**.

Answer: True

4.**True or False: The or operator in Python returns True if at least one operand is True.**

Answer: True

End of lesson 1.1

Τέλος σελίδας

Αρχή σελίδας

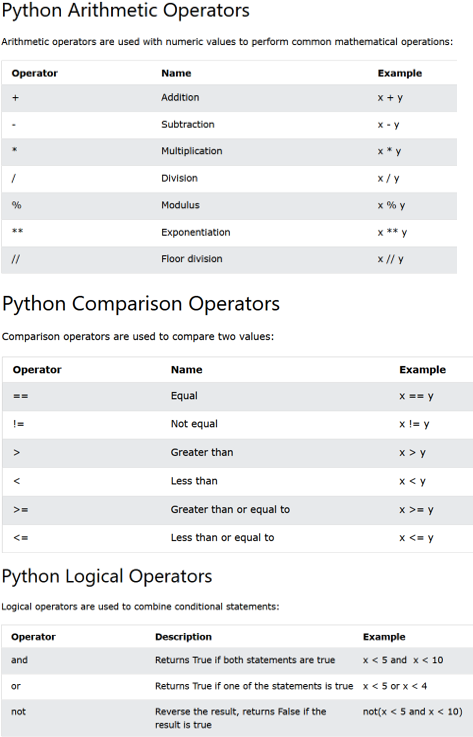
# 1.2 Control Flow Structures

### Overview:

**In our previous lesson we learned how to download the app and the very basic concepts of Python.Firstly, we became familiar with the execution of Python and the overall syntax. Furthermore, we learned about the different types of operations (such as arithmetic and logical). Lastly, we discussed how variable values are stored,changed and can be printed.**

**In this lesson we will discuss about the ways we can manipulate the flow of our project!**

## Quick Revision



Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, γραμματοσειρά, έγγραφο

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα που περιέχει κείμενο, ηλεκτρονικές συσκευές, στιγμιότυπο οθόνης, λογισμικό

Περιγραφή που δημιουργήθηκε αυτόματα

## Control Flow Structures

A program’s **control flow** is the order in which the program’s code executes. The control flow of a Python program is regulated by conditional statements, loops, and function calls. Python has three types of control structures:

**Sequential** – default mode **Selection** – used for decisions and branching **Repetition** – used for looping, i.e., repeating a piece of code multiple times.

### Sequential

**Sequential statements** are a set of statements whose execution process happens in a sequence. The problem with sequential statements is that if the logic has broken in any one of the lines, then the complete source code execution will break.

a=20

b=10

c=a-b

print("Subtraction is : ",c)

### Repetition

With **the while loop** we can execute a set of statements as long as a condition is true.

**Example**: Print i as long as i is less than 6.

i = 1

while i < 6:

print(i)

i += 1

In Python, a **for loop** is used to iterate over sequences such as [lists](https://www.programiz.com/python-programming/list), [strings](https://www.programiz.com/python-programming/string), [tuples](https://www.programiz.com/python-programming/tuple), etc.

languages = ['Swift', 'Python', 'Go']

# access elements of the list one by one

for i in languages:

print(i)

##### **Selection/Decision control statements**

**In Python, the selection statements are also known as Decision control statements or branching statements. The selection statement allows a program to test several conditions and execute instructions based on which condition is true. Some decision control statements are:**

**1**.if **2**.if-else **3**.if-elif-else  
If: It help us to run a particular code, but only when a certain condition is met or satisfied. A if only has one condition to check.  
Εικόνα που περιέχει διάγραμμα, κείμενο, γραμμή

Περιγραφή που δημιουργήθηκε αυτόματα  
2.If-else: The if-else statement evaluates the condition and will execute the body of if if the test condition is True, but if the condition is False, then the body of else is executed.  
Εικόνα που περιέχει διάγραμμα, κείμενο, γραμμή

Περιγραφή που δημιουργήθηκε αυτόματα  
  
3.Nested if: Nested if statements are an if statement inside another if statement.  
Εικόνα που περιέχει διάγραμμα, γραμμή, κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα

a = 20

b = 10

c = 15

if a > b:

if a > c:

print("a value is big")

else:

print("c value is big")

elif b > c:

print("b value is big")

else:

print("c is big")

Run

## Video explaining Flow Controls

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, γραμματοσειρά, γραφιστική

Περιγραφή που δημιουργήθηκε αυτόματα

https://youtu.be/n9TQZIiFiQs

## Exercise

**Left click and open on a new page ->** [**here**](http://fizyka.umk.pl/~jacek/docs/javatutorial/java/nutsandbolts/QandE/questions_flow.html) **<- to be redirected to the Control Flow exercise (solve ONLY the «exercises», not the «questions»).**

End of lesson 1.2

Τέλος Σελίδας

Αρχή Σελίδας

# 1.3 Create your First Project!

Εικόνα που περιέχει κείμενο, γραμματοσειρά, στιγμιότυπο οθόνης, γραφικός χαρακτήρας

Περιγραφή που δημιουργήθηκε αυτόματα

## Overview:

**In our previous lesson we learned about the different types of** [***Control Flow Structures***](https://pythownpacecom.wordpress.com/%cf%80%ce%b5%cf%81%ce%af/) **(if statements, nested if statements and loops).**

**In this lesson we will learn about arrays and create a simple coding project based on the concepts we previously learned!**

## Exercises Arrays Lists

## Watch from 00:00 to 12:35 Watch from 00:00 to 4:50 Watch from 00:00 to 3:30



## Exercise

Firstly open your preferred editor

###### The exercise comes first (with a few extras if you want the extra challenge or want to spend more time), followed by a discussion. Enjoy!

###### Ask the user for a number. Depending on whether the number is even or odd, print out an appropriate message to the user. Hint: how does an even / odd number react differently when divided by 2?

###### Extras: Ask the user for two numbers: one number to check (call it num) and one number to divide by (check). If check divides evenly into num, tell that to the user. If not, print a different appropriate message. If the number is a multiple of 4, print out a different message.

**Having trouble figuring it out?? Press the button to reveal the solution!**

[Solution](https://www.practicepython.org/solution/2014/02/15/02-odd-or-even-solutions.html)

Τέλος Σελίδας

Αρχή Σελίδας

# 1.4 Functions

## Overview

##### So far we have learnt about the Python variables, Tables, Lists, Control Flow Statements. Now it is time to learn how to code neatly using Functions!

#### What are Functions?

##### Functions in Python are reusable segments of code designed to execute a particular task. They aid in structuring code, enhancing its readability, maintainability, and re-usability. Functions can accept inputs, called parameters, and can produce outputs of any data type.

#### How do i define a Function?

##### In Python, a function is defined using the def keyword, followed by the function name, parentheses (), and a colon :. The function body is indented, and the return keyword is used to specify the output of the function.

#### Diving deeper into Functions

**Please watch from 00:00 to 7:40**

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, γραμματοσειρά

Περιγραφή που δημιουργήθηκε αυτόματα

Example of User Defined Function

### Function Exercise

##### Write a python program to add in all the numbers with using function.

##### The output is:

**Having a hard time solving it? Click the button below to reveal the solution**

[Solution](https://www.freelearningpoints.com/python/exercises/function/answer-1)

Input: numbers = (58, 5, 13, 1, 27)

Expected Answer:

The sum of number (58, 5, 13, 1, 27): 104

End of Unit 1.4

###### The exercise comes first (with a few extras if you want the extra challenge or want to spend more time), followed by a discussion. Enjoy!

###### Ask the user for a number. Depending on whether the number is even or odd, print out an appropriate message to the user. Hint: how does an even / odd number react differently when divided by 2?

###### Extras: Ask the user for two numbers: one number to check (call it num) and one number to divide by (check). If check divides evenly into num, tell that to the user. If not, print a different appropriate message. If the number is a multiple of 4, print out a different message.

**Having trouble figuring it out?? Press the button to reveal the solution!**

[Solution](https://www.practicepython.org/solution/2014/02/15/02-odd-or-even-solutions.html) https://www.practicepython.org/solution/2014/02/15/02-odd-or-even-solutions.html

End of Unit 1.3

Τέλος σελίδας

Αρχή σελίδας

# 1.4 Functions

## Overview

##### So far we have learnt about the Python variables, Tables, Lists, Control Flow Statements. Now it is time to learn how to code neatly using Functions!

#### What are Functions?

##### Functions in Python are reusable segments of code designed to execute a particular task. They aid in structuring code, enhancing its readability, maintainability, and re-usability. Functions can accept inputs, called parameters, and can produce outputs of any data type.

#### How do i define a Function?

##### In Python, a function is defined using the def keyword, followed by the function name, parentheses (), and a colon :. The function body is indented, and the return keyword is used to specify the output of the function.

#### Diving deeper into Functions

**Please watch from 00:00 to 7:40**

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, γραμματοσειρά

Περιγραφή που δημιουργήθηκε αυτόματα

Example of User Defined Function

### Function Exercise

##### Write a python program to add in all the numbers with using function.

##### The output is:

**Having a hard time solving it? Click the button below to reveal the solution**

[Solution](https://www.freelearningpoints.com/python/exercises/function/answer-1)

Input: numbers = (58, 5, 13, 1, 27)

Expected Answer:

The sum of number (58, 5, 13, 1, 27): 104

End of Unit 1.4

Τέλος σελιδας

Αρχή σελίδας

# Assessment-Certification

Εικόνα που περιέχει ζωγραφιά, τέχνη με γραμμές, σκίτσο/σχέδιο, εικονογράφηση

Περιγραφή που δημιουργήθηκε αυτόματα

## Assessment time!

##### Please press the button below in order to take the test, a passing result is >= 5 !!

[Start now](https://take.quiz-maker.com/QCVOE18S0) <https://take.quiz-maker.com/QCVOE18S0>

PYTHON CERTIFICATION ASSESSMENT

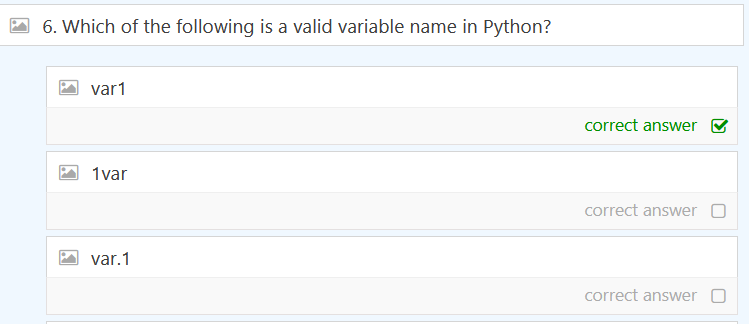
1. Python is case sensitive. T/F

2. You can run a Python script by typing python filename.py in the command line. T/F

3. The show() function can be used to display output to the screen. T/F

4. In Python, if, elif, and else are used for conditional statements. T/F

5. What will be the output of the following code? print(5 + 4) A. Error B. 54 C. 5+4 D.9

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, λογισμικό, γραμματοσειρά

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, γραμματοσειρά, αριθμός

Περιγραφή που δημιουργήθηκε αυτόματαΕικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, λογισμικό, αριθμός

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, αριθμός, λογισμικό

Περιγραφή που δημιουργήθηκε αυτόματαΕικόνα που περιέχει κείμενο, στιγμιότυπο οθόνης, λογισμικό, αριθμός

Περιγραφή που δημιουργήθηκε αυτόματα

##### Link not working? Do not worry, try this instead, a passing result is >=10 !!

[Start now](https://www.dataquest.io/blog/free-python-quiz/) [**https://www.dataquest.io/blog/free-python-quiz/**](https://www.dataquest.io/blog/free-python-quiz/)

**Τέλος Σελίδας**