

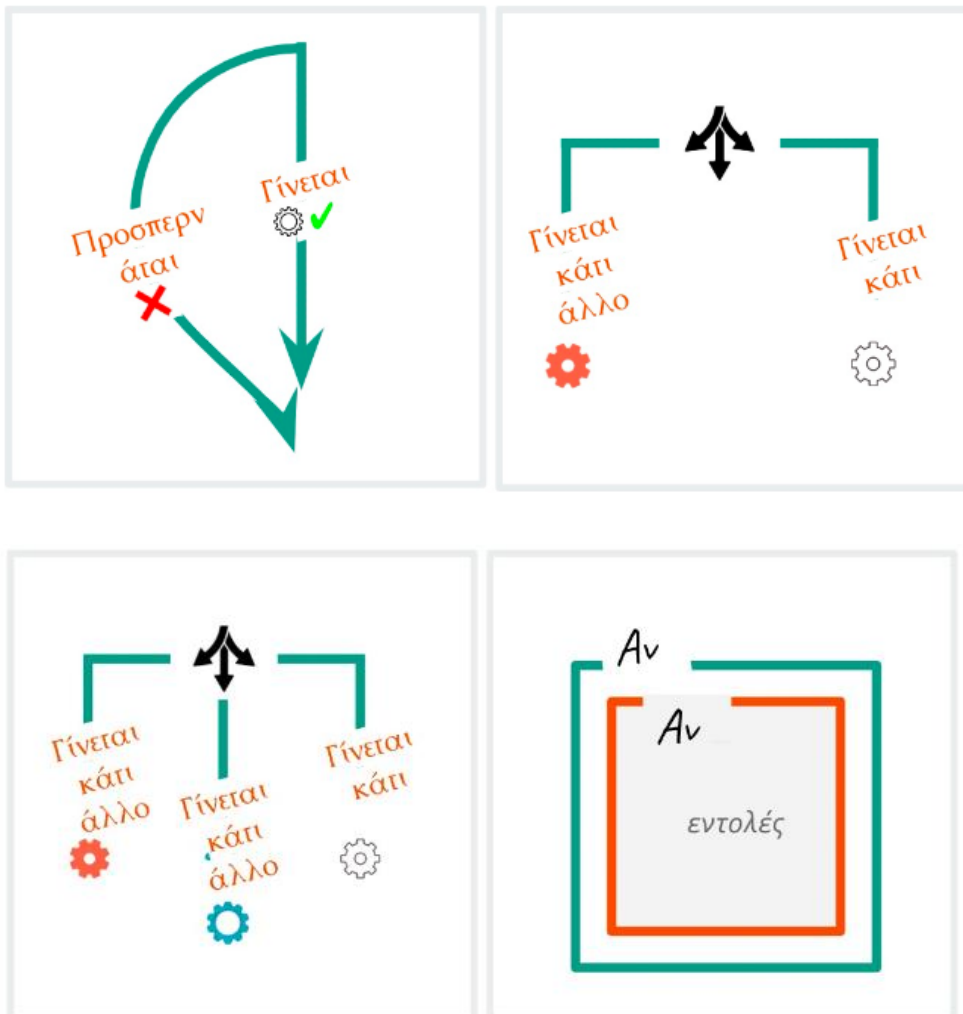
Δομή Επιλογής

Αν θέλουμε να εκτελέσουμε μία ακολουθία εντολών, εφόσον ισχύει μία συνθήκη, τότε χρησιμοποιούμε την εντολή `if`. Η εντολή `if` χρησιμοποιείται σε διάφορες μορφές, ανάλογα με τις ανάγκες του προγράμματος.

Υπάρχουν τέσσερις (4) μορφές Δομών Επιλογών

- Απλή Δομή Επιλογής
- Σύνθετη Δομή Επιλογής
- Πολλαπλή Δομή Επιλογής
- Εμφωλευμένη Δομή Επιλογής

και συνδυασμός των παραπάνω...



Γραφική αναπαράσταση των βασικών Δομών Επιλογής

Απλή Δομή Επιλογής

Δομή Επιλογής `if else`

```
Σύνταξη:  
if <συνθήκη ελέγχου>:  
    εντολές  
else:  
    εντολές
```

Απλή Δομή
Επιλογής

Δομή Επιλογής `if elif else`

```
Σύνταξη:  
if <συνθήκη ελέγχου 1>:  
    εντολές  
elif <συνθήκη ελέγχου 2>:  
    εντολές  
else:  
    εντολές
```

Σύνθετη
Δομή
Επιλογής

Παράδειγμα 1°

“Αν η μνήμη (του υπολογιστή) < 8GB
αναβάθμισε μνήμη”

Άρα, πρώτα θα ελέγξουμε αν η μνήμη μας είναι μικρότερη
(το σύμβολο “<”) από 8GB. Αν ισχύει αυτό, τότε και μόνο
τότε θα αναβαθμίσουμε τη μνήμη του υπολογιστή!

Συνήθως στις συνθήκες ακολουθούμε την εξής δομή:

```
mymemory=8  
if mymemory<8:  
    print(“Χρειάζεστε αναβάθμιση”)
```

Στην Python, οι συνθήκες εκτελούνται με τη χρήση της δομής `if` statement. Μια συνθήκη είναι μια δήλωση που αξιολογείται ως αληθής ή ψευδής και ανάλογα εκτελείται ένα τμήμα του κώδικα.

Η σύνταξη του `if` statement είναι η ακόλουθη:

```
1. if condition:
2.     # Κώδικας που θα εκτελεστεί αν η συνθήκη είναι αληθής
```

Στο παραπάνω παράδειγμα, η `condition` είναι η συνθήκη που πρέπει να αξιολογηθεί, και ο κώδικας που πρέπει να εκτελεστεί αν η συνθήκη είναι αληθής είναι ο κώδικας που βρίσκεται στο επόμενο επίπεδο μετά το `if` statement και είναι εσοχής.

Μπορείτε επίσης να χρησιμοποιήσετε το `else` statement για να εκτελέσετε κώδικα όταν η συνθήκη δεν είναι αληθής:

```
1. if condition:
2.     # Κώδικας που θα εκτελεστεί αν η συνθήκη είναι αληθής
3. else:
4.     # Κώδικας που θα εκτελεστεί αν η συνθήκη είναι ψευδής
```

Παράδειγμα 2°

Μπορούν επίσης να γίνουν πολλά ερωτήματα στη σειρά. Όπως πχ όταν παραγγέλνουμε μία πίτσα.

- Θέλετε τυρί; 🧀
- Θέλετε μπέικον; 🥓
- Θέλετεμανιτάρια; 🍄
- Θέλετε πιπεριά; 🌶️

ΟΙ ΕΠΙΛΟΓΕΣ ΠΟΥ
ΈΧΟΥΜΕ

Πατήστε το **Run** 🟩 κουμπί κάτω δεξιά στον editor ή **SHIFT + ENTER**.

y = Θέλω
n = Δεν Θέλω

y n y

```
1 if cheese == 'y':
2     print("+ Cheese")
3 if bacon == 'y':
4     print("+ Bacon")
5 if mashrooms == 'y':
6     print("+ Mashrooms")
7 if pepper == 'y':
8     print("+ Pepper")
9 print("_____")
```

+ Cheese
+ Mashrooms
+ Pepper

Αποτέλεσμα επιλογών

Σύνθετη Δομή Επιλογής

Μπορείτε επίσης να χρησιμοποιήσετε το `elif` statement για να ελέγξετε πολλαπλές συνθήκες:

```
1. if condition1:
2.     # Κώδικας που θα εκτελεστεί αν η πρώτη συνθήκη είναι αληθής
3. elif condition2:
4.     # Κώδικας που θα εκτελεστεί αν η δεύτερη συνθήκη είναι αληθής
5. else:
6.     #Κώδικας που θα εκτελεστεί αν καμία από τις παραπάνω συνθήκες δεν είναι αληθής
```

Παράδειγμα 1^ο

Μέχρι τώρα, ελέγχαμε έναν αριθμό και δίναμε μόνο μία απάντηση αν ίσχυε η συνθήκη. Αν όμως δεν ισχύει, γιατί να μη δίνουμε κάποια πληροφορία; Εδώ θα χρησιμοποιήσουμε την εντολή "else".

Ας δώσουμε την εξής εντολή:

```
print("Χρειάζεστε αναβάθμιση") if mymemory<8 else
print("Δεν χρειάζεστε αναβάθμιση")
```

Στο πιο πάνω παράδειγμα, ακολουθήσαμε διαφορετική σύνταξη. Σε μία γραμμή ζητάμε να εμφανιστεί στην οθόνη το μήνυμα **Χρειάζεστε αναβάθμιση** **ΑΝ** η τιμή της `mymemory` είναι μικρότερη από 8, **ΔΙΑΦΟΡΕΤΙΚΑ** (`else`) να εμφανιστεί το μήνυμα **Δεν χρειάζεστε αναβάθμιση**.

Παράδειγμα 2^ο

Το παρακάτω τμήμα εντολών ελέγχει αν η ηλικία που δίνεται αντιστοιχεί σε ενήλικο άνθρωπο (≥ 18 ετών) ή σε ανήλικο (<18 ετών).

Πατήστε το **Run**  κουμπί κάτω δεξιά στον editor ή **SHIFT + ENTER**.

age =

```
1 ✓ if age >= 18:
2     print("Ενήλικος/-η")
3 ✓ else:
4     print("Ανήλικος/-η")
```

Ανήλικος/-η

Παράδειγμα 3^ο

Το παρακάτω τμήμα εντολών ελέγχει αν ένας αριθμός είναι **άρτιος** ή **περιττός**.

number =

```
1 if number % 2 == 0:
2     print("Άρτιος - Διόδο πολλαπλάσιο του 2")
3 else:
4     print("Περιττός")
```

Περιττός

Παράδειγμα 4^ο

Παρακάτω είναι η συνθήκη για να ελέγξουμε ότι ένας άνθρωπος είναι ενήλικη γυναίκα. Διόδο το φύλο να είναι Γυναίκα **και** η ηλικία να είναι μεγαλύτερη ίση του 18.

Μπορείτε να «παίξετε» με τις τιμές των μεταβλητών **age** και **gender** για να δείτε πως θα συμπεριφερθεί το τμήμα προγράμματος παρακάτω.

```
1 age = 19
2 gender = "Female"
3
4 if age >= 18 and gender == "Female":
5     print("Ενήλικη Γυναίκα")
6 else:
7     print("Όχι Ενήλικη Γυναίκα")
```

Παράδειγμα 5^ο

Το παρακάτω πρόγραμμα Python υπολογίζει τον χρόνο που χρειάζεται ένα όχημα ώστε να διανύσει μία απόσταση (σε χλμ). Την απόσταση (χλμ) και στην ταχύτητα (χλμ/ώρα) την δίνει ως είσοδο ο χρήστης.

Hint - Από την φυσική γνωρίζουμε ότι:

απόσταση = ταχύτητα * χρόνος

```
1 apostasi = 280
2 taxytita = 85
3 xronos = apostasi / taxytita
4
5 wres = int(xronos)
6 lepta = (xronos - int(xronos)) * 60
7
8 print(f"Απόσταση(χλμ): {apostasi} ♦ Ταχύτητα(χλμ/ώρα): {taxytita}")
9
10 if wres < 1:
11     print(f"Λεπτά: {round(lepta,0)}")
12 else:
13     print(f"Ώρες: {round(wres,1)} και Λεπτά: {round(lepta,0)}")
```

Θα ξεκινήσουμε αρχικά με δύο μεταβλητές:

```
number1=10  
number2=20
```

Στις δύο αυτές μεταβλητές δώσαμε διαφορετικές τιμές.

Θέλουμε να συγκρίνουμε το περιεχόμενο των δύο μεταβλητών, ώστε να διαπιστώσουμε αν είναι το ίδιο.

```
if number1 == number2: print ("Equal")
```

Με την πιο πάνω εντολή, ελέγχουμε αν το περιεχόμενο τους είναι το ίδιο. Αυτό επιτυγχάνεται με το διπλό σύμβολο "=". Στο πιο πάνω παράδειγμα, η συνθήκη δεν ισχύει, έτσι δε θα εμφανιστεί κανένα μήνυμα.

Αν αυτό που μας ενδιαφέρει είναι να ελέγξουμε αν η τιμή τους είναι διαφορετική, τότε χρησιμοποιούμε την εντολή

```
if number1 != number2: print ("Equal")
```

Ελεγχοί που μπορούμε να κάνουμε:

- Μια μεταβλητή είναι μικρότερη από μια άλλη:
`number1 < number2`
- Μια μεταβλητή είναι μεγαλύτερη από μια άλλη:
`number1 > number2`
- Μια μεταβλητή είναι μικρότερη ή ίση από άλλη:
`number1 <= number2`
- Μια μεταβλητή είναι μεγαλύτερη ή ίση από άλλη:
`number1 >= number2`

Για συγκρίσεις, μπορούμε επίσης να χρησιμοποιήσουμε τα **And**, **Or** και **Not**.

```
if number1=100 and number2=100:  
print("Equal")
```

Στην πιο πάνω εντολή ελέγχουμε αν ισχύουν και οι δύο συνθήκες (και ο ένας αριθμός, και ο άλλος, ίσοι).

```
If number1=100 or number2=100: print("One hundred")
```

Στην πιο πάνω εντολή, ελέγχουμε αν έστω ο ένας από τους δύο ισούται με 100.

Μπορείτε επίσης να χρησιμοποιήσετε λογικούς τελεστές ('and', 'or' και 'not') για να συνδυάσετε πολλαπλές συνθήκες. Για παράδειγμα:

```
1. x = 5 # Αρχικοποίηση της μεταβλητής x με την τιμή 5
2. y = 10 # Αρχικοποίηση της μεταβλητής y με την τιμή 10
3.
4. # Έλεγχος δύο συνθηκών: x > 3 και y > 5
5. if x > 3 and y > 5:
6.     print("Both conditions are true") # Εκτύπωση μηνύματος "Both conditions are true" αν οι
    συνθήκες είναι αληθείς
```

Στον παραπάνω κώδικα, δημιουργούνται δύο μεταβλητές `x` και `y` με τιμές 5 και 10 αντίστοιχα. Στη συνέχεια, χρησιμοποιείται η δομή `if` για να ελεγχθούν δύο συνθήκες: αν το `x` είναι μεγαλύτερο από 3 και το `y` είναι μεγαλύτερο από 5. Αν και οι δύο συνθήκες είναι αληθείς, τότε εκτυπώνεται το μήνυμα "Both conditions are true" στην οθόνη.

Στην Python, μπορείτε να χρησιμοποιήσετε τη λέξη-κλειδί "`not`" για να αναστρέψετε την αξιολόγηση μιας συνθήκης. Αυτό σημαίνει ότι αν η συνθήκη είναι αληθής, τότε η "`not`" θα την καταστήσει ψευδή, και αν η συνθήκη είναι ψευδής, τότε η "`not`" θα την καταστήσει αληθή. Έτσι, μπορείτε να χρησιμοποιήσετε τη λέξη-κλειδί "`not`" για να αναστρέψετε το αποτέλεσμα μιας συνθήκης.

```
1. x = 5 # Αρχικοποίηση της μεταβλητής x με την τιμή 5
2.
3. # Έλεγχος της συνθήκης: αν η συνθήκη x > 10 είναι ψευδής (δηλαδή x δεν είναι μεγαλύτερο του
    10)
4. if not x > 10:
5.     print("x is not greater than 10") # Εκτύπωση μηνύματος "x is not greater than 10" αν η
    συνθήκη είναι ψευδής
```

Ο κώδικας αρχικοποιεί τη μεταβλητή `x` με την τιμή 5 και στη συνέχεια ελέγχει αν η συνθήκη `x > 10` είναι ψευδής. Αν η συνθήκη είναι ψευδής, τότε εκτυπώνεται το μήνυμα "x is not greater than 10".

Στην Python, μπορούμε να χρησιμοποιήσουμε συνθήκες που βασίζονται στην αριθμητική και τα μαθηματικά. Οι βασικές συνθήκες που υποστηρίζονται είναι οι εξής:

- Ίσον: `a == b` (Η τιμή της μεταβλητής `a` είναι ίση με την τιμή της μεταβλητής `b`)
- Διαφορετικό: `a != b` (Η τιμή της μεταβλητής `a` είναι διαφορετική από την τιμή της μεταβλητής `b`)
- Μικρότερο από: `a < b` (Η τιμή της μεταβλητής `a` είναι μικρότερη από την τιμή της μεταβλητής `b`)
- Μικρότερο ή ίσο με: `a <= b` (Η τιμή της μεταβλητής `a` είναι μικρότερη ή ίση με την τιμή της μεταβλητής `b`)
- Μεγαλύτερο από: `a > b` (Η τιμή της μεταβλητής `a` είναι μεγαλύτερη από την τιμή της μεταβλητής `b`)
- Μεγαλύτερο ή ίσο με: `a >= b` (Η τιμή της μεταβλητής `a` είναι μεγαλύτερη ή ίση με την τιμή της μεταβλητής `b`)

Αυτές οι συνθήκες μπορούν να χρησιμοποιηθούν με διάφορους τρόπους, κυρίως μέσα σε "if statements" (δηλαδή δηλώσεις if) και loops (επαναλήψεις). Αξιοποιώντας αυτές τις συνθήκες, μπορούμε να προγραμματίσουμε την εκτέλεση ενός τμήματος κώδικα μόνο όταν ισχύει μια συγκεκριμένη συνθήκη.

Ένα "if statement" στην Python αρχίζει με τη λέξη-κλειδί "if".

```
1. x = 5 # Αρχικοποίηση της μεταβλητής x με την τιμή 5
2.
3. # Έλεγχος συνθήκης: x > 3
4. if x > 3:
5.     print("x είναι μεγαλύτερο από 3") # Εκτύπωση μηνύματος "x είναι μεγαλύτερο από 3" αν η
    συνθήκη είναι αληθής
```

Στο παραπάνω παράδειγμα, ο αριθμός `x` είναι μεγαλύτερος από 3, οπότε η συνθήκη `x > 3` είναι αληθής και θα εκτυπωθεί το μήνυμα "x είναι μεγαλύτερο από 3".

Μπορείτε επίσης να συνδυάσετε συνθήκες χρησιμοποιώντας τα λογικά σύμβολα "and", "or" και "not" στην Python. Το "and" χρησιμοποιείται για να ελέγξετε αν ισχύουν και οι δύο συνθήκες, το "or" για να ελέγξετε αν τουλάχιστον μία από τις δύο συνθήκες είναι αληθής, και το "not" για να αντιστρέψετε την αλήθεια μιας συνθήκης.

Παρακάτω παρουσιάζονται παραδείγματα:

```
1. x = 5
2. y = 10
3.
4. # Έλεγχος αν και οι δύο συνθήκες είναι αληθείς
5. if x > 3 and y > 5:
6.     print("Και οι δύο συνθήκες είναι αληθείς")
7.
8. # Έλεγχος αν τουλάχιστον μία από τις δύο συνθήκες είναι αληθής
9. if x > 3 or y < 5:
10.    print("Τουλάχιστον μία από τις δύο συνθήκες είναι αληθής")
11.
12. # Αντίστροφη αξιολόγηση μιας συνθήκης
13. if not x > 10:
14.    print("Η συνθήκη δεν είναι αληθής")
```

Με τη χρήση των λογικών συμβόλων μπορείτε να κατασκευάσετε πιο σύνθετες συνθήκες που εξαρτώνται από την αξιολόγηση πολλών συνθηκών ταυτόχρονα.

Στην Python, μπορείτε να χρησιμοποιήσετε τις συνθήκες με τις εντολές `if`, `elif` και `else` για να αντιμετωπίσετε πολλές περιπτώσεις. Η εντολή `if` αξιολογεί μια συνθήκη και εκτελεί ένα τμήμα κώδικα αν η συνθήκη είναι αληθής. Η εντολή `elif` (συντομογραφία του "else if") χρησιμοποιείται για να ελέγξετε επιπλέον συνθήκες μετά την εντολή `if`. Αν καμία από τις προηγούμενες συνθήκες δεν είναι αληθής, μπορείτε να χρησιμοποιήσετε την εντολή `else` για να εκτελέσετε έναν τμήμα κώδικα που ισχύει γενικά.

Παράδειγμα χρήσης:

```
1. x = 5 # Αρχικοποίηση της μεταβλητής x με την τιμή 5
2.
3. # Έλεγχος της τιμής της μεταβλητής x
4. if x < 0:
5.     print("Το x είναι αρνητικό") # Εκτύπωση μηνύματος "Το x είναι αρνητικό" αν η τιμή του x
   είναι αρνητική
6. elif x == 0:
7.     print("Το x είναι μηδέν") # Εκτύπωση μηνύματος "Το x είναι μηδέν" αν η τιμή του x είναι
   μηδέν
8. else:
9.     print("Το x είναι θετικό") # Εκτύπωση μηνύματος "Το x είναι θετικό" αν η τιμή του x
   είναι θετική
```

Στο παραπάνω παράδειγμα, ο κώδικας αξιολογεί την τιμή της μεταβλητής `x` και εκτελεί το αντίστοιχο τμήμα κώδικα ανάλογα με τη συνθήκη που ισχύει. Αν το `x` είναι αρνητικό, εκτυπώνεται το μήνυμα "Το x είναι αρνητικό". Αν το `x` είναι μηδέν, εκτυπώνεται το μήνυμα "Το x είναι μηδέν". Αν καμία από τις προηγούμενες συνθήκες δεν ισχύει, εκτελείται το τμήμα κώδικα που βρίσκεται στην εντολή `else`, και εκτυπώνεται το μήνυμα "Το x είναι θετικό".

Παραδείγματα Βιβλίου

| Γενική Μορφή | Παραδείγματα |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>1η περίπτωση if <συνθήκη>: μπλοκ εντολών { εντολές αν η συνθήκη είναι Αληθής</p> <p>εντολές</p> <p>Λειτουργία: Αν η συνθήκη είναι Αληθής εκτελούνται οι εντολές του μπλοκ.</p> | <p>Παράδειγμα 1 # εμφάνισης της απόλυτης τιμής ενός ακέραιου αριθμού a = int(input('Δώσε ένα ακέραιο αριθμό ')) if a <= 0: a=(-1)*a print a</p> |
| <p>2η περίπτωση if <συνθήκη>: 1. μπλοκ εντολών { εντολές αν η συνθήκη είναι Αληθής</p> <p>else: 2. μπλοκ εντολών { εντολές αν η συνθήκη είναι Ψευδής</p> <p>εντολές</p> <p>Λειτουργία: Αν η συνθήκη είναι Αληθής εκτελούνται οι εντολές του 1ου μπλοκ του if, ενώ αν η συνθήκη είναι Ψευδής εκτελούνται οι εντολές του 2ου μπλοκ του else.</p> | <p>Παράδειγμα 2 #έλεγχος αν ένας αριθμός είναι άρτιος ή περιττός a=int(input('Δώσε ένα ακέραιο αριθμό ')) if a/2=0: print "Άρτιος" else: print "Περιττός"</p> |
| <p>3η περίπτωση (Σύνθετη δομή) if <συνθήκη1>: 1. μπλοκ εντολών { εντολές αν η συνθήκη1 είναι Αληθής</p> <p>elif <συνθήκη2>: 2. μπλοκ εντολών { εντολές αν η συνθήκη1 είναι Αληθής</p> <p>else: 3. μπλοκ εντολών { εντολές αν η συνθήκη1 και η συνθήκη2 είναι Ψευδής</p> <p>εντολές</p> | <p>Παράδειγμα 3 #έλεγχος αν ένας αριθμός είναι θετικός ή αρνητικός ή μηδέν a=int(input('Δώσε ένα ακέραιο αριθμό ')) if a>0: print "Θετικός" elif a<0: print "Αρνητικός" else: print "Μηδέν"</p> |

Σημείωση: Τα μπλοκ των εντολών που θα εκτελεστούν σε όλες τις περιπτώσεις της if, ορίζονται ως ένα μπλοκ με εσοχή βάζοντας τη μια εντολή κάτω από την άλλη. Αν το μπλοκ εντολών αποτελείται από μια εντολή, τότε μπορεί να γραφτεί στην ίδια γραμμή με την if ή την else. Π.χ.

```
a=input('give a number ')
if a>0: print a,'>0'
else: print a,'<0'
```

Παράδειγμα 1-4 Να υπολογιστεί ο μεγαλύτερος δύο ακέραιων αριθμών που δίνονται από το πληκτρολόγιο.

1ος τρόπος με απλή επιλογή

| | |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Αλγόριθμος MAX1 Διάβασε x, ψ $\text{μεγ} \leftarrow x$ Αν $\text{μεγ} < y$ τότε $\text{μεγ} \leftarrow y$ Τέλος_αν Τέλος MAX1 | # Ο μεγαλύτερος δύο αριθμών $x = \text{int}(\text{input}(\text{"Δώσε τον 1ο ακέραιο αριθμό "}))$ $y = \text{int}(\text{input}(\text{"Δώσε τον 2ο ακέραιο αριθμό "}))$ $\text{max} = x$ if $\text{max} < y$: $\text{max} = y$ print "Μεγαλύτερος ο ", max |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

2ος τρόπος με σύνθετη επιλογή

| | |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Αλγόριθμος MAX2 Διάβασε x, ψ Αν $x < y$ τότε $\text{μεγ} \leftarrow y$ Αλλιώς $\text{μεγ} \leftarrow x$ Τέλος_αν Τέλος MAX2 | # Ο μεγαλύτερος δύο αριθμών $x = \text{int}(\text{input}(\text{"Δώσε τον 1ο ακέραιο αριθμό "}))$ $y = \text{int}(\text{input}(\text{"Δώσε τον 2ο ακέραιο αριθμό "}))$ if $x > y$: $\text{max} = x$ else: $\text{max} = y$ print "Μεγαλύτερος ο ", max |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

Πολλαπλή Δομή Επιλογής

Ο σωστός τρόπος χρήσης του `elif` είναι:

```
1. if condition1:
2.     # Κώδικας όταν η condition1 είναι αληθής
3. elif condition2:
4.     # Κώδικας όταν η condition1 είναι ψευδής και η condition2 είναι αληθής
5. elif condition3:
6.     # Κώδικας όταν η condition1 και η condition2 είναι ψευδείς και η condition3 είναι αληθής
7. else:
8.     # Κώδικας όταν καμία από τις προηγούμενες συνθήκες δεν είναι αληθείς
```

Οι συνθήκες ελέγχονται διαδοχικά, και μόνο η πρώτη που ικανοποιείται εκτελείται. Όλες οι υπόλοιπες συνθήκες παραλείπονται. Αυτό εξασφαλίζει ότι ο κώδικας εκτελείται μόνο για την πρώτη ικανοποιημένη συνθήκη και εξοικονομεί χρόνο και πόρους εκτέλεσης.

Παρακάτω φαίνεται ένα παράδειγμα `if` statement με πολλά `elif` clauses:

```
1. x = 10 # Αρχικοποίηση της μεταβλητής x με την τιμή 10
2.
3. # Έλεγχος της τιμής της μεταβλητής x με τη χρήση συνθηκών
4. if x < 0:
5.     print("Το x είναι αρνητικό") # Εκτύπωση μηνύματος "Το x είναι αρνητικό" αν το x είναι
    αρνητικό
6. elif x == 0:
7.     print("Το x είναι μηδέν") # Εκτύπωση μηνύματος "Το x είναι μηδέν" αν το x είναι μηδέν
8. elif x > 0 and x < 10:
9.     print("Το x είναι ένας θετικός αριθμός με μία ψηφίο") # Εκτύπωση μηνύματος "Το x είναι
    ένας θετικός αριθμός με μία ψηφίο" αν το x είναι θετικός αριθμός με μία ψηφίο
10. else:
11.     print("Το x είναι ένας θετικός αριθμός μεγαλύτερος από 10") # Εκτύπωση μηνύματος "Το x
    είναι ένας θετικός αριθμός μεγαλύτερος από 10" αν το x είναι θετικός αριθμός μεγαλύτερος από
    10
```

Σε αυτό το παράδειγμα, η μεταβλητή `x` είναι ορισμένη με την τιμή 10. Ο κώδικας ελέγχει την τιμή της `x` χρησιμοποιώντας μια σειρά από συνθήκες.

- Αν η τιμή της `x` είναι αρνητική (μικρότερη του 0), τότε εκτυπώνεται το μήνυμα "Το x είναι αρνητικό".
- Αν η τιμή της `x` είναι μηδέν, τότε εκτυπώνεται το μήνυμα "Το x είναι μηδέν".
- Αν η τιμή της `x` είναι θετική (μεγαλύτερη του 0) και μικρότερη του 10, τότε εκτυπώνεται το μήνυμα "Το x είναι ένας θετικός αριθμός με μία ψηφίο".
- Σε όλες τις υπόλοιπες περιπτώσεις, δηλαδή αν η τιμή της `x` είναι θετική και μεγαλύτερη ή ίση με 10, εκτυπώνεται το μήνυμα "Το x είναι ένας θετικός αριθμός μεγαλύτερος από 10".

Είναι σημαντικό να σημειώσουμε ότι ένα `if` statement μπορεί να έχει κανένα ή περισσότερα `elif` clauses για να ελέγξει πολλαπλές συνθήκες. Ωστόσο, μπορεί να έχει μόνο ένα `else` clause, το οποίο πρέπει να τοποθετηθεί στο τέλος του statement, αν υπάρχει. Το `else` clause εκτελείται όταν καμία από οι προηγούμενες συνθήκες δεν ικανοποιείται.

Η σειρά των συνθηκών στο `if` statement είναι σημαντική. Η πρώτη αληθής συνθήκη που αντιστοιχεί εκτελείται και οι υπόλοιπες αγνοούνται. Για αυτό τον λόγο, είναι σημαντικό να τοποθετείτε τις πιο συγκεκριμένες συνθήκες προς την κορυφή του `if` statement, ενώ οι πιο γενικές συνθήκες τοποθετούνται προς το τέλος. Έτσι, δίνεται προτεραιότητα στην εκτέλεση των πιο συγκεκριμένων συνθηκών πριν από τις γενικότερες συνθήκες.

Στην Python, το "else" είναι μια λέξη-κλειδί που χρησιμοποιείται για να ορίσει ένα τμήμα κώδικα που θα εκτελεστεί όταν καμία από τις προηγούμενες συνθήκες δεν ικανοποιείται.

Μπορούμε να τοποθετήσουμε το “else” μετά από ένα “if” ή ένα “elif” block και να ορίσουμε τις εντολές που θα εκτελεστούν στην περίπτωση που η συνθήκη είναι ψευδής. Αυτή η δομή μας επιτρέπει να αντιδράσουμε σε όλες τις πιθανές περιπτώσεις ελέγχοντας την τελευταία εναλλακτική επιλογή.

Παράδειγμα:

```
1. x = 5 # Αρχικοποίηση της μεταβλητής x με την τιμή 5
2.
3. # Έλεγχος της συνθήκης x > 10
4. if x > 10:
5.     print("Ο αριθμός είναι μεγαλύτερος από 10")
6.
7. # Έλεγχος της συνθήκης x > 5
8. elif x > 5:
9.     print("Ο αριθμός είναι μεγαλύτερος από 5")
10.
11. # Περίπτωση που η συνθήκη x > 10 και η συνθήκη x > 5 δεν ισχύουν
12. else:
13.     print("Ο αριθμός είναι μικρότερος ή ίσος του 5")
```

Ο παραπάνω κώδικας εκτελεί μια σειρά από ελέγχους σε μια μεταβλητή `x` και εκτυπώνει ανάλογα με το αποτέλεσμα των ελέγχων.

Αν η τιμή της μεταβλητής `x` είναι μεγαλύτερη από 10, τότε εκτυπώνεται το μήνυμα “Ο αριθμός είναι μεγαλύτερος από 10”.

Αν η τιμή της μεταβλητής `x` δεν είναι μεγαλύτερη από 10, αλλά είναι μεγαλύτερη από 5, τότε εκτυπώνεται το μήνυμα “Ο αριθμός είναι μεγαλύτερος από 5”.

Αν και οι παραπάνω δύο συνθήκες δεν ισχύουν, τότε εκτελείται η εντολή που βρίσκεται στο μπλοκ `else` και εκτυπώνεται το μήνυμα “Ο αριθμός είναι μικρότερος ή ίσος του 5”.

Συνολικά, ο κώδικας ελέγχει την τιμή της μεταβλητής `x` και εκτελεί διαφορετικές ενέργειες ανάλογα με το αποτέλεσμα των ελέγχων.

Παρακάτω είναι ένα παράδειγμα `if` statement με πολλές συνθήκες `elif` και μια συνθήκη `else` :

```
1. x = 20 # Αρχικοποίηση της μεταβλητής x με την τιμή 20
2.
3. # Έλεγχος της τιμής της μεταβλητής x
4. if x < 0:
5.     print("Το x είναι αρνητικός") # Εκτύπωση μηνύματος "Το x είναι αρνητικός" αν η τιμή του
x είναι αρνητική
6. elif x == 0:
7.     print("Το x είναι μηδέν") # Εκτύπωση μηνύματος "Το x είναι μηδέν" αν η τιμή του x είναι
μηδέν
8. elif x > 0 and x < 10:
9.     print("Το x είναι θετικός αριθμός με μία ψηφίο") # Εκτύπωση μηνύματος "Το x είναι
θετικός αριθμός με μία ψηφίο" αν η τιμή του x είναι θετική και έχει ένα ψηφίο
10. elif x >= 10 and x < 20:
11.     print("Το x είναι θετικός αριθμός με δύο ψηφία") # Εκτύπωση μηνύματος "Το x είναι
θετικός αριθμός με δύο ψηφία" αν η τιμή του x είναι θετική και έχει δύο ψηφία
12. else:
13.     print("Το x είναι ένας θετικός αριθμός μεγαλύτερος ή ίσος του 20") # Εκτύπωση μηνύματος
"Το x είναι ένας θετικός αριθμός μεγαλύτερος ή ίσος του 20" αν η τιμή του x είναι μεγαλύτερη ή
ίση του 20
```

Ο παραπάνω κώδικας εκτελεί τις ακόλουθες ενέργειες:

1. Ορίζει τη μεταβλητή `x` με την τιμή 20.
2. Ελέγχει την τιμή της μεταβλητής `x` χρησιμοποιώντας τη δομή `if-elif-else`.
3. Αν η τιμή της μεταβλητής `x` είναι αρνητική (λιγότερη από 0), τότε εκτυπώνει το μήνυμα “Το x είναι αρνητικός”.
4. Αλλιώς, αν η τιμή της μεταβλητής `x` είναι μηδέν, τότε εκτυπώνει το μήνυμα “Το x είναι μηδέν”.
5. Αλλιώς, αν η τιμή της μεταβλητής `x` είναι θετική και έχει ένα ψηφίο (μικρότερη από 10), τότε εκτυπώνει το μήνυμα “Το x είναι θετικός αριθμός με μία ψηφίο”.
6. Αλλιώς, αν η τιμή της μεταβλητής `x` είναι θετική και έχει δύο ψηφία (μεγαλύτερη ή ίση του 10 και μικρότερη από 20), τότε εκτυπώνει το μήνυμα “Το x είναι θετικός αριθμός με δύο ψηφία”.
7. Αλλιώς, για οποιαδήποτε άλλη περίπτωση, εκτυπώνει το μήνυμα “Το x είναι ένας θετικός αριθμός μεγαλύτερος ή ίσος του 20”.

Παράδειγμα 1^ο

Το παρακάτω τμήμα εντολών ελέγχει σε ποια βαθμίδα εκπαίδευσης ανήκει η ηλικία που δίνεται:

Πατήστε το **Run**  κουμπί κάτω δεξιά στον editor ή **SHIFT + ENTER**.

age =

```
1 if age >= 4 and age < 6:
2     print("Νηπιαγωγείο")
3 elif age <= 12:
4     print("Δημοτικό")
5 elif age <= 15:
6     print("Γυμνάσιο")
7 elif age < 18:
8     print("Λύκειο")
9 else:
10    print("N/A")
```

Λύκειο

Παράδειγμα 2^ο - Δείκτης Μάζας Σώματος

Το παρακάτω τμήμα εντολών ελέγχει σε ποια βαθμίδα βάρους ανήκει ένα άτομο με βάση το ΒΑΡΟΣ και το ΥΨΟΣ που δίνει:

baros = ypsos =

```
1 bmi = baros / ypsos ** 2
2
3 if bmi < 18.5:
4     print("Υπό του Κανονικού")
5 elif bmi < 25:
6     print("Κανονικό")
7 elif bmi < 30:
8     print("Ελαφρώς υπέρβαρο")
9 elif bmi < 35:
10    print("Υπέρβαρο")
11 else:
12    print("Αρκετά υπέρβαρο")
```

Ελαφρώς υπέρβαρο

Παράδειγμα Βιβλίου

Παράδειγμα 1-5 Ο δείκτης μάζας σώματος (ΔΜΣ) είναι μία ένδειξη για το βαθμό παχυσαρκίας ενός ατόμου και υπολογίζεται από τον τύπο: $\text{βάρος}/\text{ύψος}^2$. Δεδομένων του βάρους και του ύψους ενός ατόμου, να εμφανίσετε χαρακτηρισμό που αφορά στο βαθμό παχυσαρκίας του με βάση το ΔΜΣ. Υπάρχουν τέσσερις περιπτώσεις για τον χαρακτηρισμό του ατόμου, που μπορούν να εξετασθούν με μια δομή πολλαπλής επιλογής.

| ΔΜΣ | Χαρακτηρισμός ατόμου |
|-------------|-----------------------|
| <18,5 | Ελλιποβαρές |
| 18,5 - 24,9 | Φυσιολογικό βάρος |
| 25 - 29,9 | Υπέρβαρο |
| >30 | Πάσχει από παχυσαρκία |

| Αλγόριθμος Χαρακτηρισμός Ατόμου | #Χαρακτηρισμός Ατόμου |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Γράψε 'Δώσε βάρος' Διάβασε βάρος Γράψε 'Δώσε ύψος' Διάβασε ύψος $\Delta\text{Μ}\Sigma \leftarrow \text{βάρος} / \text{ύψος}^2$ Αν $\Delta\text{Μ}\Sigma < 18.5$ τότε Γράψε 'ελλιποβαρές άτομο' Αλλιώς_αν $\Delta\text{Μ}\Sigma < 25$ τότε Γράψε 'άτομο με φυσιολογικό βάρος' Αλλιώς_αν $\Delta\text{Μ}\Sigma < 30$ τότε Γράψε 'υπέρβαρο άτομο' αλλιώς Γράψε 'άτομο που πάσχει από παχυσαρκία' Τέλος_αν Τέλος Χαρακτηρισμός Ατόμου | <pre>#Χαρακτηρισμός Ατόμου varos=input("Δώσε βάρος") yposos=input("Δώσε ύψος") dms=varos/(yposos**2) if dms<18.5: print 'ελλιποβαρές άτομο' elif dms<25: print 'άτομο με φυσιολογικό βάρος' elif dms<30: print 'υπέρβαρο άτομο' else: print 'άτομο που πάσχει από παχυσαρκία'</pre> |

Εμφωλευμένη Δομή Επιλογής

Η Εμφωλευμένη Δομή Επιλογής λειτουργεί όπως τον λογικό τελεστή **KAI**.

Δηλαδή τα παρακάτω τμήματα κώδικα είναι ισοδύναμα, δηλαδή παράγουν το ίδιο ακριβώς αποτέλεσμα.

| | |
|-------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------|
| <pre>if age >= 18: if gender == 'Female': print("Γυναίκα Ψηφοφόρος")</pre> | <pre>if age>=18 and gender=='Female': print("Γυναίκα Ψηφοφόρος")</pre> |
|-------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------|

Οι εντολές “if” μπορούν να περιλαμβάνονται μέσα σε άλλες εντολές “if”, και αυτή η τεχνική ονομάζεται “nested if statements” (εμφωλευμένες εντολές “if”). Η χρήση των φωλιασμένων εντολών “if” σας επιτρέπει να δημιουργήσετε πιο σύνθετες συνθήκες, επιτρέποντάς σας να ελέγχετε πολλές πτυχές μιας κατάστασης. Κάθε φορά που η συνθήκη στον εξωτερικό “if” είναι αληθής, η εκτέλεση μεταβιβάζεται στον εσωτερικό “if” για περαιτέρω έλεγχο. Αυτό σας επιτρέπει να κατασκευάσετε πιο πολύπλοκες συνθήκες που εξαρτώνται από πολλαπλές παραμέτρους.

Στην Python, η σύνταξη του nested if statement είναι η εξής:

```
1. if condition1:
2.     statement
3.     if condition2:
4.         statement
```

Σε αυτό το παράδειγμα, η συνθήκη του πρώτου “if” ελέγχει εάν η συνθήκη “condition1” είναι αληθής. Εάν η συνθήκη είναι αληθής, εκτελείται η εντολή “statement” και στη συνέχεια ελέγχεται η συνθήκη “condition2”. Εάν και αυτή η συνθήκη είναι αληθής, εκτελείται η δεύτερη εντολή “statement”.

Παρακάτω είναι ένα παράδειγμα χρήσης του nested if statement:

```
1. x = 10 # Αρχικοποίηση της μεταβλητής x με την τιμή 10
2. y = 5  # Αρχικοποίηση της μεταβλητής y με την τιμή 5
3.
4. # Έλεγχος αν η μεταβλητή x είναι μεγαλύτερη από 5
5. if x > 5:
6.     print("x είναι μεγαλύτερο από 5") # Εκτύπωση μηνύματος "x είναι μεγαλύτερο από 5"
7.
8.     # Έλεγχος αν η μεταβλητή y είναι μεγαλύτερη από 2
9.     if y > 2:
10.        print("y είναι μεγαλύτερο από 2") # Εκτύπωση μηνύματος "y είναι μεγαλύτερο από 2"
```

Ο κώδικας εκτελεί τις παρακάτω ενέργειες:

1. Ορίζει τις μεταβλητές `x` και `y` με τις τιμές 10 και 5 αντίστοιχα.
2. Ελέγχει αν η μεταβλητή `x` είναι μεγαλύτερη από 5.
3. Αν η προηγούμενη συνθήκη είναι αληθής, εκτελεί την εντολή `print("x είναι μεγαλύτερο από 5")`, που εκτυπώνει το μήνυμα “x είναι μεγαλύτερο από 5”.
4. Εσωτερικά στην προηγούμενη συνθήκη, ελέγχει αν η μεταβλητή `y` είναι μεγαλύτερη από 2.
5. Αν η προηγούμενη εσωτερική συνθήκη είναι αληθής, εκτελεί την εντολή `print("y είναι μεγαλύτερο από 2")`, που εκτυπώνει το μήνυμα “y είναι μεγαλύτερο από 2”.

Έτσι, αν τόσο η μεταβλητή `x` είναι μεγαλύτερη από 5 όσο και η μεταβλητή `y` είναι μεγαλύτερη από 2, τότε θα εκτυπωθούν και τα δύο μηνύματα.

Μπορείτε να χρησιμοποιήσετε όσα επίπεδα εμφωλευμένων if statements χρειάζεστε. Ωστόσο, κάποιοι προγραμματιστές πιστεύουν ότι η υπερβολική χρήση εμφωλευμένων if statements μπορεί να δυσκολεύει την κατανόηση του κώδικα. Συνήθως, όταν χρειάζεστε περίπλοκες συνθήκες, είναι καλύτερο να χρησιμοποιήσετε λογικούς τελεστές (όπως το “and”, “or”, “not”) και τοτισμένες συνθήκες.

Αντί να χρησιμοποιείτε πολλαπλά επίπεδα εμφωλευμένων if statements, μπορείτε να χρησιμοποιήσετε σύνθετες συνθήκες που συνδυάζουν πολλές συνθήκες με λογικούς τελεστές. Έτσι, μπορείτε να διατηρήσετε τον κώδικα πιο συμπαγή και ευανάγνωστο.

Για παράδειγμα, αντί να γράψετε:

```
1. if condition1:
2.     if condition2:
3.         # Κώδικας
```

Μπορείτε να γράψετε:

```
1. if condition1 and condition2:
2.     # Κώδικας
```

Αυτή η προσέγγιση μπορεί να καταστήσει τον κώδικα πιο ευανάγνωστο και ευκολότερο στη συντήρηση.

Στην Python, οι δηλώσεις "if" πρέπει να έχουν κάποιο περιεχόμενο. Ωστόσο, αν για οποιονδήποτε λόγο χρειάζεστε μια δήλωση "if" χωρίς περιεχόμενο, μπορείτε να χρησιμοποιήσετε τη λέξη-κλειδί "pass" για να αποφύγετε την πρόκληση σφάλματος. Η δήλωση "pass" είναι μια τεχνητή εντολή που δεν κάνει τίποτα και χρησιμοποιείται ως προσωρινή συμπλήρωση, επιτρέποντάς σας να αγνοήσετε τη δήλωση "if" χωρίς να προκληθεί σφάλμα.

Η δήλωση "pass" χρησιμοποιείται ως μια κενή εντολή στην Python. Αλλά περνά πάνω από την εκτέλεση και δεν προσθέτει κάποια λειτουργικότητα. Χρησιμοποιείται συχνά ως προσωρινή συντακτική κατασκευή όταν χρειάζεστε μια εντολή στον κώδικα σας, αλλά δεν θέλετε να προσθέσετε κάποιον συγκεκριμένο κώδικα ή λειτουργία σε αυτήν τη στιγμή. Ουσιαστικά, η δήλωση "pass" σας επιτρέπει να παρακάμψετε την απαίτηση ενός κώδικα σε ένα σημείο όπου η σύνταξη απαιτεί μια εντολή, αλλά δεν χρειάζεται να γίνει τίποτα.

Εδώ έχουμε ένα παράδειγμα χρήσης της δήλωσης "pass":

```
1. for i in range(5):
2.     if i == 3:
3.         pass # Παράβλεψη της εκτέλεσης του εσωτερικού μπλοκ κώδικα χρησιμοποιώντας την
           εντολή pass
4.     else:
5.         print(i) # Εκτύπωση της τιμής της μεταβλητής i αν η συνθήκη i == 3 δεν ισχύει
```

Στο παραπάνω παράδειγμα, η δήλωση "pass" χρησιμοποιείται για να αγνοηθεί η συγκεκριμένη περίπτωση (όπου i ισούται με 3) και να συνεχιστεί η εκτέλεση του κώδικα. Όταν η συνθήκη είναι αληθής, δεν εκτελείται καμία ενέργεια, αλλά ο κώδικας συνεχίζει να εκτελείται κανονικά στη συνέχεια.

Παράδειγμα 1^ο

Τρέξτε το παρακάτω τμήμα εντολών ώστε να δείτε πως συμπεριφέρονται οι Εμφωλευμένες Δομές Επιλογής.

- αν ισχύει μόνο η **1η** συνθήκη → εκτυπώνεται ένα μήνυμα
- αν ισχύουν **και οι 2** συνθήκες → εκτυπώνονται 2 μηνύματα
- αν **δεν** ισχύει **καμία** συνθήκη → δεν εκτυπώνεται κανένα μήνυμα

Πατήστε το **Run**  κουμπί κάτω δεξιά στον editor ή **SHIFT + ENTER**.

```
1 import random
2 age = random.randint(1,100)
3 gender = random.choice(['Male', 'Female', 'Other'])
4 print(f"age: {age} ◊ gender: {gender} → ")
5
6 if age >= 18:
7     print("[1] Ενήλικας")
8     if gender == 'Female':
9         print('[2] Ενήλικη Γυναίκα')
10
11 print("_____")
```

age: 47 ◊ gender: Female →

[1] Ενήλικας

[2] Ενήλικη Γυναίκα

Παράδειγμα 2^ο

Τρέξτε το παρακάτω τμήμα εντολών ώστε να δείτε πως μπορεί να χρησιμοποιηθεί Εμφωλευμένη Δομές Επιλογής για τα αποτελέσματα 2 ζαριών.

```
1 import random
2 zar1 = random.randint(1,6)
3 zar2 = random.randint(1,6)
4 print(f"zar1: {zar1} ◊ zar2: {zar2} + ")
5
6 if zar1 == zar2:
7     print("Διπλή Ζαριά")
8     if zar1 == 1:
9         print("Άσσοι")
10    if zar1 == 2:
11        print("Διπλές")
12    if zar1 == 3:
13        print("Τρείτσες")
14    if zar1 == 4:
15        print("Ντφόρτα")
16    if zar1 == 5:
17        print("Πεντάρες")
18    if zar1 == 6:
19        print("Εξάρες")
20
21 print("_____")
```

zar1: 6 ◊ zar2: 6 →

Διπλή Ζαριά

Εξάρες

Παραδείγματα Βιβλίου

Παράδειγμα 1-6 (Εμφωλευμένη Δομή επιλογής)

Στο ταχυδρομείο το κόστος αποστολής υπολογίζεται συναρτήσει του προορισμού και του βάρους της επιστολής με βάση τον πίνακα. Δεδομένων του προορισμού και του βάρους μιας επιστολής, να εμφανίσετε το ποσό που στοιχίζει η αποστολή της.

| | | Βάρος αποστολής | |
|-------------------------|-----------|-------------------------|------------------------|
| | | Μέχρι και 100 γραμμάρια | Πάνω από 100 γραμμάρια |
| Προορισμός επιστολής | Εσωτερικό | 1€ | 2€ |
| | Εξωτερικό | 2.50€ | 4€ |

1ος Τρόπος: Με εμφωλευμένη Δομή Επιλογής

| | |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>Αλγόριθμος Επιστολή1 Γράψε 1. Εσωτερικό' Γράψε '2.Εξωτερικό' Γράψε 'Επιλογή' Διάβασε προορισμό Γράψε 'Δώσε βάρος' Διάβασε βάρος Αν προορισμός = 1 τότε Αν βάρος <= 100 τότε κόστος ← 1 αλλιώς κόστος ← 2 τέλος_αν αλλιώς Αν βάρος <= 100 τότε κόστος ← 2.5 αλλιώς κόστος ← 4 τέλος_αν τέλος_αν Γράψε κόστος Τέλος Επιστολή1</p> | <p>#Επιστολή1 print '1.Εσωτερικό' print '2. Εξωτερικό' proorismos=input("Επιλογή: ") varos=input("Δώσε βάρος: ") if proorismos==1: if varos<=100: kostos=1 else: kostos=2 else: if varos<=100: kostos=2.5 else: kostos=4 print 'Κόστος μεταφοράς της επιστολής:',kostos</p> |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

2ος Τρόπος: Με σύνθετη Δομή Επιλογής

| | |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>Αλγόριθμος Επιστολή2 Γράψε 1. Εσωτερικό' Γράψε '2.Εξωτερικό' Γράψε 'Επιλογή' Διάβασε pr Γράψε 'Δώσε βάρος' Αν pr = 1 και βάρος <= 100 τότε κόστος ← 1 αλλιώς_αν pr = 1 και βάρος > 100 τότε κόστος ← 2 αλλιώς_αν pr = 2 και βάρος <= 100 τότε κόστος ← 2.5 αλλιώς κόστος ← 4 τέλος_αν Γράψε 'Κόστος=',κόστος Τέλος Επιστολή2</p> | <p>#Επιστολή2 print '1.Εσωτερικό' print '2. Εξωτερικό' pr=input("Επιλογή: ") varos=input("Δώσε βάρος: ") if pr==1 and varos<=100: kostos=1 elif pr==1 and varos>100: kostos=2 elif pr==2 and varos<=100: kostos=2.5 else: kostos=4 print 'Κόστος μεταφοράς της επιστολής:',kostos</p> |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

Αναφορές

- Ευσταθίου Χρήστος, *Αρχές Προγραμματισμού Υπολογιστώ σε γλώσσα Python*, ΕΠΑΛ Τρικάλων, 2021
- Κοφτερός Αλέξανδρος, *Η Python με Απλά Λόγια: Σύντομος οδηγός για αρχάριους*, Λευκωσία, 2023
- Γυφτάκης Γιάννης, <https://courses.giftakis.gr/>, ανακτήθηκε 08/11/2023
- <https://pliroforiki-edu.gr/>, ανακτήθηκε 08/11/2023