

## Απλά Βοηθητικά Παραδείγματα Κώδικα και Προγραμμάτων Python

Χρησιμοποιήστε τα προγράμματα αυτά σαν δείγματα κώδικα, επεκτείνοντας τα και για να κατανοήσετε τις βασικές έννοιες του προγραμματισμού στη γλώσσα.

Αναζήτηση με διπλό for:

```
items = ["aaa", 111, (4, 5), 2.01]
tests = [(4, 5), 3.14]
for key in tests:      # For all keys
    for item in items: # For all items
        if item == key: # Check for match
            print(key, "was found")
            break
    else:
        print(key, "not found!")
```

Αναζήτηση με έξυπνο if:

```
items = ["aaa", 111, (4, 5), 2.01]
tests = [(4, 5), 3.14]
for key in tests:      # For all keys
    if key in items:   # Check for match
        print(key, "was found")
    else:
        print(key, "not found!")
```

Καλώντας συναρτήσεις, οι τοπικές τιμές δεν περνάνε έξω:

```
def f(a): # a is assigned to (references) passed object
    a = 99 # Changes local variable a only

b = 88
f(b)      # a and b both reference same 88 initially
print(b)  # b is not changed
```

Το παράδειγμα αυτό τυπώνει 88.

Καλώντας συναρτήσεις, οι global τιμές αλλάζουν:

```
X = 88      # Global X

def func():
    global X
    X = 99  # Changes Global X outside def

func()
print(X)   # X is changed
```

Το παράδειγμα αυτό τυπώνει 99.

Η τιμή της X εξαρτάται φυσικά από την σειρά με την οποία θα κληθούν οι συναρτήσεις:

```
X = 99
def func1():
    global X
    X = 88

def func2():
    global X
    X = 77
```

Παράδειγμα χρήσης κλάσης από το βιβλίο Think Python του Allen B. Downey:

```
"""
Code example from Think Python, by Allen B. Downey.
Available from http://thinkpython.com

Copyright 2012 Allen B. Downey.
Distributed under the GNU General Public License at
gnu.org/licenses/gpl.html.
"""

class Point(object):
    """Represents a point in 2-D space."""

def print_point(p):
    """Print a Point object in human-readable format."""
    print ('(%g, %g)' % (p.x, p.y))

class Rectangle(object):
    """Represents a rectangle.

    attributes: width, height, corner.
    """

def find_center(rect):
    """Returns a Point at the center of a Rectangle."""
    p = Point()
    p.x = rect.corner.x + rect.width/2.0
    p.y = rect.corner.y + rect.height/2.0
    return p

def grow_rectangle(rect, dwidth, dheight):
    """Modify the Rectangle by adding to its width and height.

    rect: Rectangle object.
```

```

dwidth: change in width (can be negative).
dheight: change in height (can be negative).
"""
rect.width += dwidth
rect.height += dheight

def main():
    blank = Point()
    blank.x = 3
    blank.y = 4
    print ('blank')
    print_point(blank)

    box = Rectangle()
    box.width = 100.0
    box.height = 200.0
    box.corner = Point()
    box.corner.x = 0.0
    box.corner.y = 0.0

    center = find_center(box)
    print ('center', print_point(center))

    print (box.width)
    print (box.height)
    print ('grow')
    grow_rectangle(box, 50, 100)
    print (box.width)
    print (box.height)

if __name__ == '__main__':
    main()

```