



ΠΑΝΕΠΙΣΤΗΜΙΟ
ΠΑΤΡΩΝ
UNIVERSITY OF PATRAS

Πολυτεχνική Σχολή

Τμήμα Ηλεκτρολόγων Μηχανικών και Τεχνολογίας Υπολογιστών

Μάθημα: ECE_Y106 Εισαγωγή στους Υπολογιστές

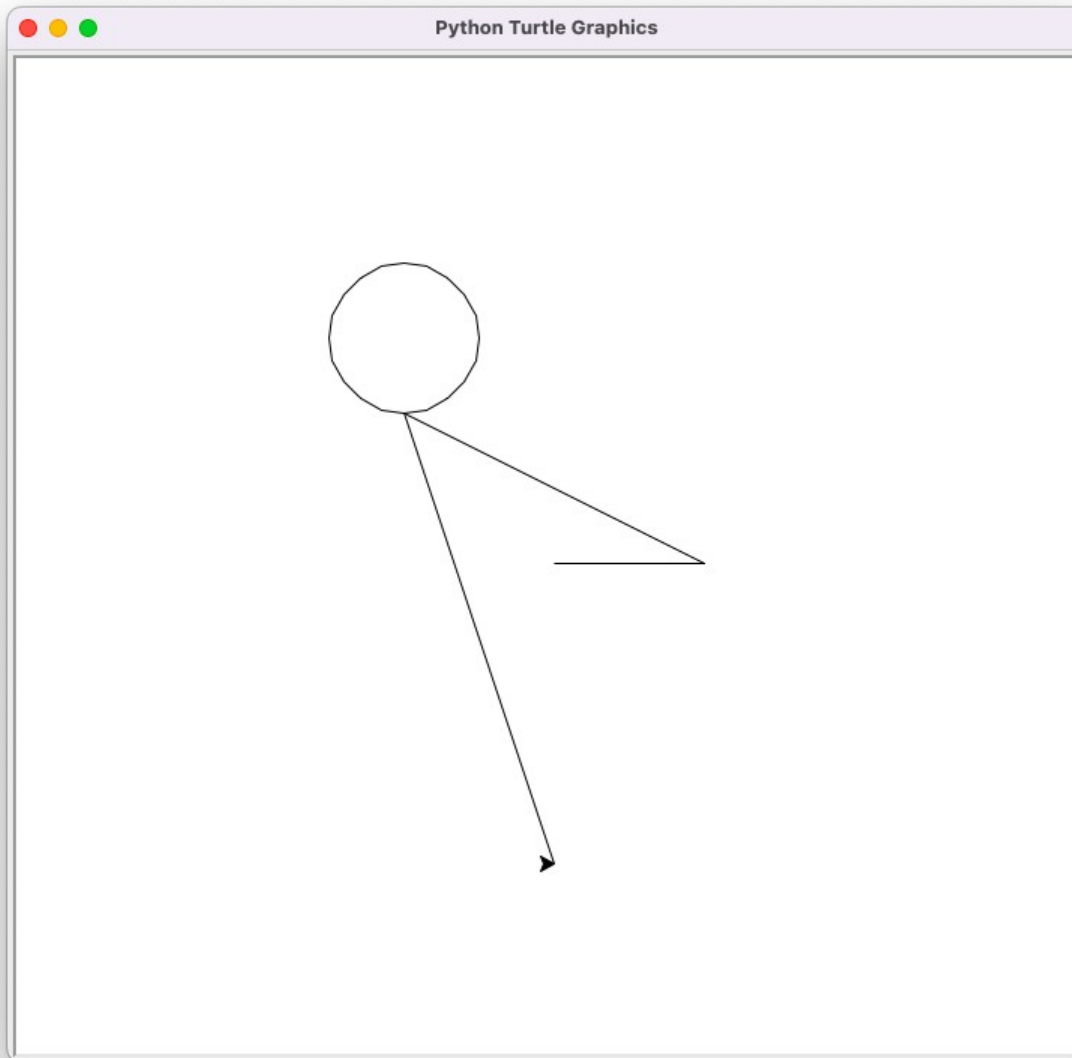
ECE_Y106: Εισαγωγή στους Υπολογιστές

ΔΙΑΛΕΞΗ #4

Γραφικά, Συναρτήσεις, Λεξικά, Σύνολα

Αν. Καθ. Κυριάκος Σγάρμπας

turtle graphics



- `reset()`
- `goto()`
- `up()`, `down()`
- `left()`, `right()`
- `forward()`
- `color()`
- `speed()`
- `bye()`

Συναρτήσεις

- εμφάνιση, namespaces, τοπικές και καθολικές μεταβλητές
- αναδρομικές
- ορίσματα/παράμετροι (προαιρετικές, προεπιλεγμένες, ονομαστικές)

```
emveleia.py
def f1():
    a = 10
    print("This is f1. a =",a)

def f2():
    a = 20
    print("This is f2. a =",a)

a = 5
print("This is main. a =",a)
f1()
f2()
print("This is main. a =",a)
```

Ln: 8 Col: 0

```
emveleia.py
def f1():
    a = 10
    print("This is f1. a =",a)

def f2():
    #a = 20
    print("This is f2. a =",a)

a = 5
print("This is main. a =",a)
f1()
f2()
print("This is main. a =",a)
```

Ln: 6 Col: 5

```
emveleia.py
def f1():
    a = 10
    a = a+1
    print("This is f1. a =",a)

def f2():
    #a = 20
    a = a+1
    print("This is f2. a =",a)

a = 5
print("This is main. a =",a)
f1()
f2()
print("This is main. a =",a)
```

Ln: 11 Col: 0

```
emveleia.py
def f1(a):
    a = 10
    a = a+1
    print("This is f1. a =",a)

def f2(a):
    a = a+1
    print("This is f2. a =",a)

a = 5
print("This is main. a =",a)
f1(a)
f2(a)
print("This is main. a =",a)
```

Ln: 14 Col: 27

Αναδρομικές Συναρτήσεις

```
anadromi.py
def paragontikol(n):
    p = 1
    for i in range(1,n+1): p = p*i
    return p

def paragontiko2(n):
    if n==0: return 1
    else: return paragontiko2(n-1)*n

def fibonacci1(n):
    if n<3: return 1
    a = b = 1
    for i in range(n-2):
        c = a+b
        a,b = b,c
    return c

def fibonacci2(n):
    if n<3: return 1
    else: return fibonacci2(n-1)+fibonacci2(n-2)
```

Ln: 21 Col: 0

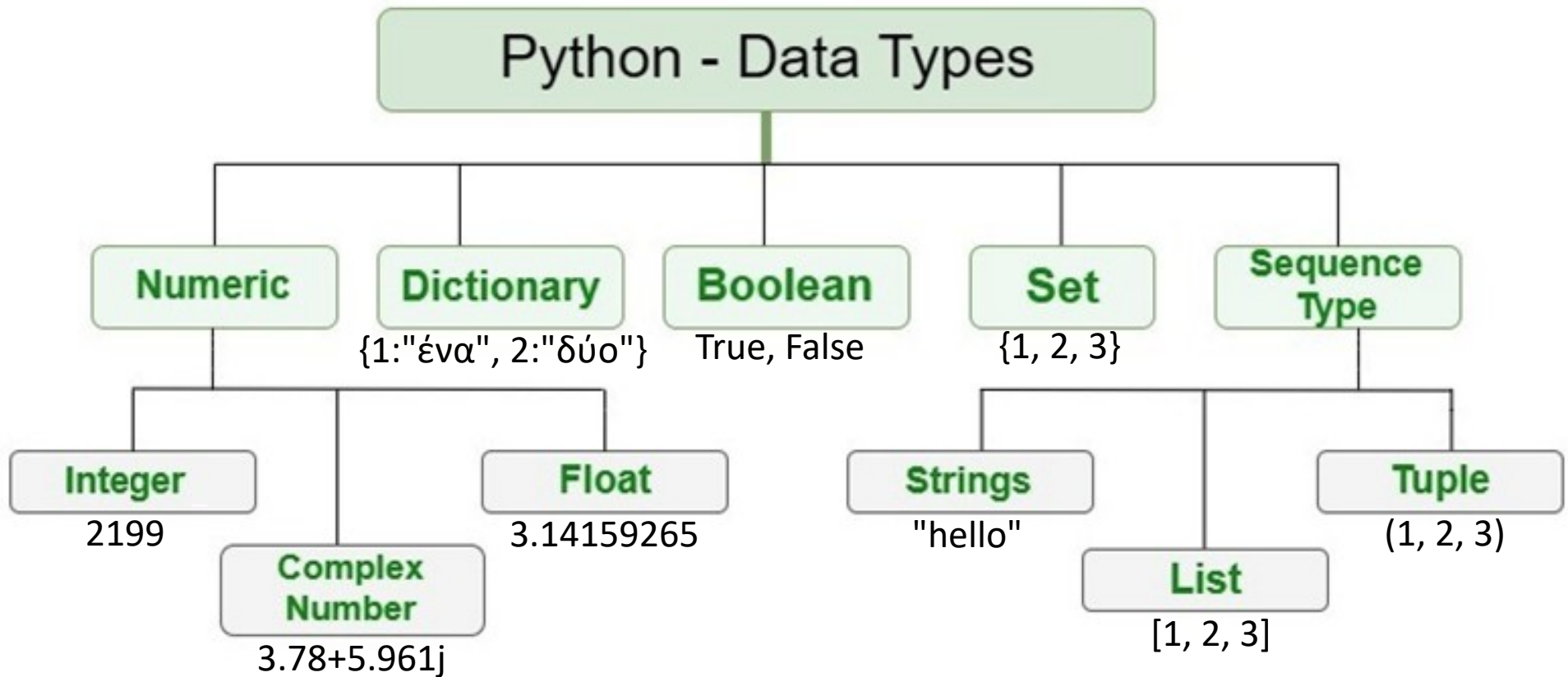
Ορίσματα

- `def func(a1, a2, a3, ...):`
- `def func(a1=v1, a2=v2, a3=v3, ...):`
- `def func(*arg):`

```
>>> def sub(a, b=0):  
...     return a-b  
...  
>>> sub(5, 3)  
2  
>>> sub(5)  
5  
>>> sub(b=3, a=5)  
2  
>>>
```

```
>>> def add(*x):  
...     s = 0  
...     for n in x: s += n  
...     return s  
...  
>>> add(1, 2, 3)  
6  
>>> add(1)  
1  
>>> add(1, 2, 3, 4, 5)  
15  
>>>
```

Τύποι Δεδομένων



Dictionaries

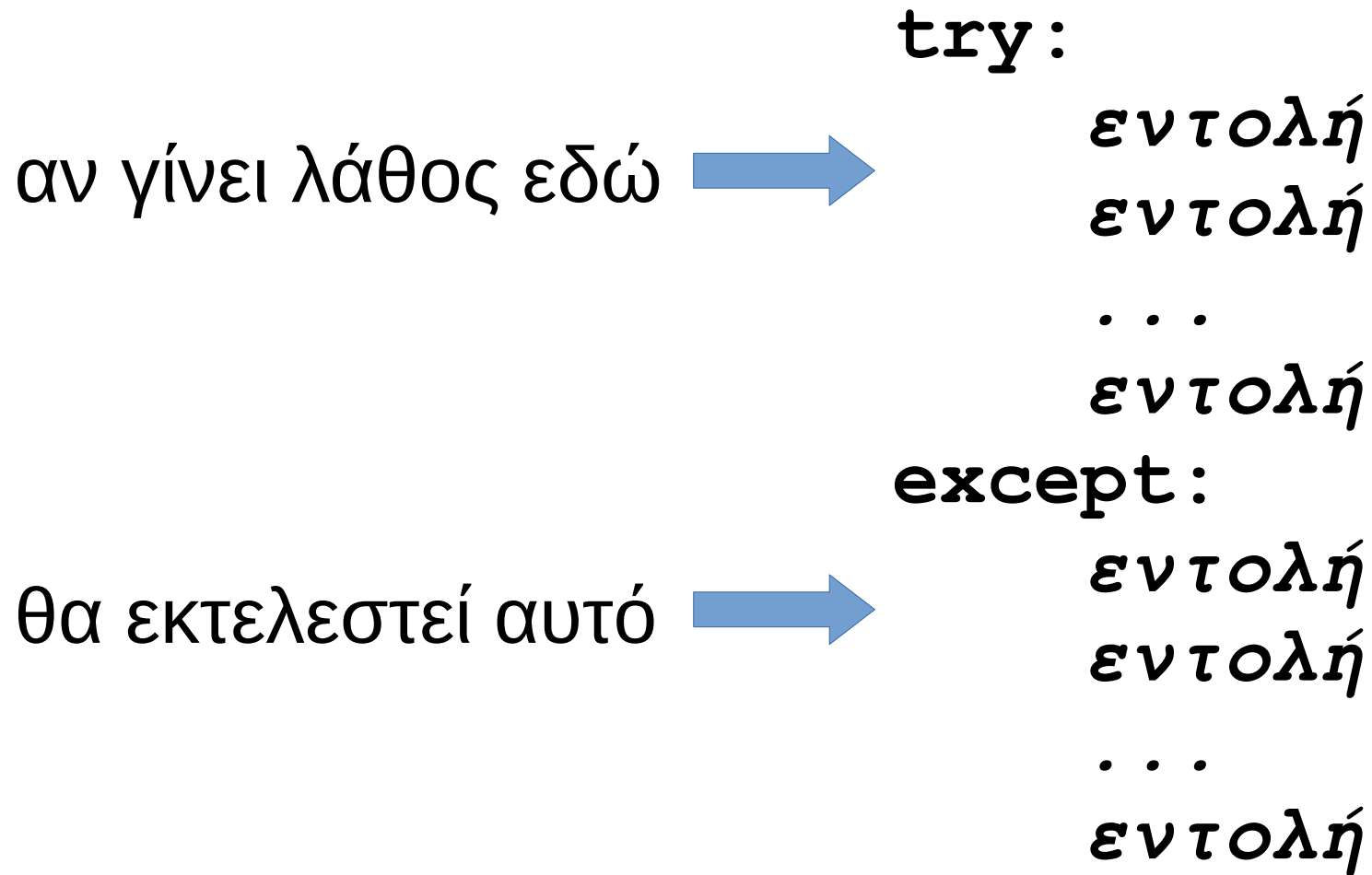
(Λεξικά/Ευρετήρια)

```
>>> d = {"john":4523, "bill":1095, "mary":5782}
>>> d
{'john': 4523, 'bill': 1095, 'mary': 5782}
>>> d["bill"]
1095
>>> d["bill"] = 1111
>>> d
{'john': 4523, 'bill': 1111, 'mary': 5782}
>>> d["paul"] = 2222
>>> d
{'john': 4523, 'bill': 1111, 'mary': 5782, 'paul': 2222}
>>> del d["bill"]
>>> d
{'john': 4523, 'mary': 5782, 'paul': 2222}
>>> d.keys()
dict_keys(['john', 'mary', 'paul'])
>>> d.values()
dict_values([4523, 5782, 2222])
>>> d.items()
dict_items([('john', 4523), ('mary', 5782), ('paul', 2222)])
>>> |
```


Sets (Σύνολα)

```
>>> s1 = {2, 1, 2, 3, 1, 2, 3, 2, 1}
>>> s1
{1, 2, 3}
>>> s2 = {2, 3, 4, 5}
>>> s2
{2, 3, 4, 5}
>>> s1.union(s2)
{1, 2, 3, 4, 5}
>>> s1.intersection(s2)
{2, 3}
>>> s2.union(s1)
{1, 2, 3, 4, 5}
>>> s2.intersection(s1)
{2, 3}
>>> s1.difference(s2)
{1}
>>> s2.difference(s1)
{4, 5}
>>> |
```

try - except



Από που διαβάζουμε

- Αγγελιδάκης, έως σελ. 103
- Μανής, έως σελ. 150
- Swaroop, έως σελ. 65
- Περάκης, έως σελ. 74
- Python Tutorial, έως ενότητα 6

Φροντιστηριακή Άσκηση #4