

Δομές Δεδομένων

Σύνολα (set)

- ◆ Τι είναι τα σύνολα (set);
- ✓ Ένα σύνολο (set) είναι μια συλλογή από μοναδικά, μη διατεταγμένα στοιχεία.
- ✓ Χρησιμοποιείται όταν θέλουμε να αποθηκεύσουμε μοναδικές τιμές και να κάνουμε γρήγορες συγκρίσεις.

📌 Δημιουργία ενός συνόλου

```
my_fruit_set = set(["apple", "orange", "banana"])  
print my_fruit_set
```

Δομές Δεδομένων

Σύνολα (set)

📌 Προσθήκη & Αφαίρεση Στοιχείων

```
my_fruit_set = set(["apple", "banana"])  
my_fruit_set.add("orange")  
my_fruit_set.remove("apple")
```

📌 Πράξεις μεταξύ Συνόλων

- ✓ Ένωση (union) – Όλα τα στοιχεία από δύο σύνολα
- ✓ Τομή (intersection) – Τα κοινά στοιχεία
- ✓ Διαφορά (difference) – Τα στοιχεία που υπάρχουν στο ένα σύνολο αλλά όχι στο άλλο

```
A = set([1, 2, 3, 4])  
B = set(3,4,5, 6)  
print A.union(B) # {1, 2, 3, 4, 5, 6}  
print A.intersection(B) # {3, 4}  
print A.difference(B) # {1, 2}
```

Δομές Δεδομένων

Σύνολα (set)

💡 Άσκηση για την Τάξη

♦ Γράψτε ένα πρόγραμμα που:

1. Ζητά από τον χρήστη να εισάγει 5 τυχαίους αριθμούς.
2. Τους αποθηκεύει σε ένα σύνολο A.
3. Δείχνει στον χρήστη όλους τους μοναδικούς αριθμούς που εισήγαγε.
4. Ζητά από τον χρήστη να εισάγει 5 νέους τυχαίους αριθμούς
5. Τους αποθηκεύει σε ένα σύνολο B.
6. Δείχνει στον χρήστη τους κοινούς αριθμούς του συνολου A και B.

Δομές Δεδομένων

Πλειάδες (tuple)

- ◆ Τι είναι οι πλειάδες (tuple);
- ✓ Μία **πλειάδα (tuple)** είναι μια τακτοποιημένη, **αμετάβλητη** συλλογή στοιχείων.
- ✓ Μοιάζει με λίστα, αλλά δεν μπορεί να αλλάξει μετά τη δημιουργία της.
- ✓ Συνήθως οι πλειάδες χρησιμοποιούνται για την αναπαράσταση μιας σειράς χαρακτηριστικών/ιδιοτήτων μιας οντότητας

📌 Δημιουργία Πλειάδας

```
my_data = "Giorgos", "Antoniou", 35  
print my_data
```

📌 Πρόσβαση σε Στοιχεία Πλειάδας

```
print my_data [0] # "Giorgos"  
print my_data [-1] # 35
```

Δομές Δεδομένων

Πλειάδες (tuple)

- 📌 Τι Μπορούμε (και δεν Μπορούμε) να Κάνουμε με Πλειάδες;
- ✓ Διαβάζουμε τιμές με ευρετήριο (index).
- ✓ Ελέγχουμε αν ένα στοιχείο υπάρχει με in:

```
print "Giorgos" in my_data # True
```

- ✓ Μετατρέπουμε πλειάδα σε λίστα για να την επεξεργαστούμε:

```
my_data_list = list(my_data)
my_data_list.append("Oikodomos")
my_data = tuple(my_data_list)
```

- ✗ Δεν μπορούμε να αλλάξουμε στοιχεία απευθείας!

```
my_fruits[0] = "Kostas" # ✗ Σφάλμα!
```

Δομές Δεδομένων

Πλειάδες (tuple)

💡 Άσκηση για την Τάξη

◆ Γράψτε ένα πρόγραμμα που:

1. Έχει μια πλειάδα με τα μαθήματα του σχολείου.
2. Ζητά από τον χρήστη ένα μάθημα και ελέγχει αν υπάρχει στην πλειάδα.
3. Εμφανίζει το μήνυμα "Το μάθημα υπάρχει!" ή "Το μάθημα δεν υπάρχει!".

Δομές Δεδομένων

Λεξικά (dict)

- ◆ Τι είναι τα Λεξικά (dictionary / dict);
- ✓ Ένα λεξικό είναι μια δομή δεδομένων που αποθηκεύει ζευγάρια κλειδιών-τιμών.
- ✓ Τα κλειδιά είναι μοναδικά και χρησιμοποιούνται για την αναζήτηση των τιμών.
- ✓ Μοιάζουν με πραγματικά λεξικά:
 - Κλειδί → Η λέξη
 - Τιμή → Ο ορισμός της λέξης

📌 Δημιουργία Λεξικού

```
student = { "name": "Giorgos", "age": 15, "grade": 18.5 }  
print student
```

Τα λεξικά χρησιμοποιούν άγκιστρα {} και κάθε στοιχείο είναι σε μορφή "κλειδί": τιμή.

Δομές Δεδομένων

Λεξικά (dict)

📌 Πρόσβαση σε Τιμές

◆ Με το κλειδί (key)

```
print student["name"]  
print student["grade"]
```

◆ Με τη get() (για αποφυγή σφαλμάτων)

```
print student.get("name")  
print student.get("grade", "unknown")  
print student.get("address", "address does not exist")
```

Δομές Δεδομένων

Λεξικά (dict)

📌 Προσθήκη & Τροποποίηση Στοιχείων

```
student["school"] = "1ο Ιμερисиό Επαλ Θήρας"  
student["grade"] = 20
```

📌 Διαγραφή Στοιχείων

```
del student["age"]
```

Δομές Δεδομένων

Λεξικά (dict)

💡 Άσκηση για το σπίτι

◆ Γράψτε ένα πρόγραμμα που διαχειρίζεται ένα λεξικό προϊόντων και τιμών σε ένα κατάστημα.

📌 Βήματα:

1 Το πρόγραμμα ξεκινά με ένα προκαθορισμένο λεξικό προϊόντων και τιμών.

2 Να ζητά διαδοχικά ο χρήστης να:

- Εμφανίσει όλα τα προϊόντα και τις τιμές.
- Προσθέσει ένα νέο προϊόν με την τιμή του.
- Αναζητήσει την τιμή ενός προϊόντος.
- Διαγράψει ένα προϊόν.
- Εμφανίσει όλα τα προϊόντα και τις τιμές.