

[CSV αναπαράσταση δεδομένων](#)

[Τι είναι ένα CSV αρχείο](#)

[Διάβασμα ενός CSV αρχείου](#)

[Συνολικός Κώδικας](#)

[Γράψιμο σε ένα CSV αρχείο](#)

[Συνολικός Κώδικας](#)

[Διάβασμα από JSON/AST Συμβολοσειρών](#)

## CSV αναπαράσταση δεδομένων

### Τι είναι ένα CSV αρχείο

Το CSV σημαίνει "Τιμές διαχωρισμένες με κόμμα". Είναι η απλούστερη μορφή αποθήκευσης δεδομένων σε μορφή πίνακα.

Η τυπική δομή του αρχείου, είναι ότι η πρώτη σειρά περιέχει την κεφαλή του αρχείου, ενώ κάθε γραμμή περιέχει μία εγγραφή, όπου οι τιμές της εγγραφής χωρίζονται με κόμματα. Για παράδειγμα έστω ότι έχουμε το αρχείο "students.csv" με περιεχόμενα:

```
id, name, surname, age
el22053, Γιάννης, Παπαδόπουλος, 22
el22057, Κώστας, Ιωάννου, 25
```

Το συγκεκριμένο αρχείο αντιστοιχεί στον πίνακα

id	name	surname	age
el22053	Γιάννης	Παπαδόπουλος	22
el22057	Κώστας	Ιωάννου	25

### Διάβασμα ενός CSV αρχείου

Το διάβασμα ενός csv αρχείου σε **Python** γίνεται στα ακόλουθα βήματα:

1. Εισαγωγή της csv βιβλιοθήκης  
**import csv**

2. Άνοιγμα του csv αρχείου → Επιστρέφει ένα αρχείο  
**file = open('students.csv')**
3. Διάβασμα του csv αρχείου μέσω του αντικειμένου csv.reader  
**csvreader = csv.reader(file)**
4. Εξαγωγή των πεδίων της επικεφαλίδας: χρησιμοποιείτε την μέθοδο next η οποία θα επιστρέψει το επόμενο στοιχείο από τον csvreader (στην προκειμένη περίπτωση το πρώτο).  
**header = next(csvreader)**
5. Εξαγωγή και εκτύπωση των υπολοίπων εγγραφών. Κάθε row αρχείο έχει την μορφή ενός πίνακα όπου π.χ. η πρώτη τιμή είναι row[0] θα επιστρέψει το id του μαθητή:  
**for row in csvreader:**  
**print(row)**
6. Κλείσιμο του αρχείου: Η μέθοδος .close() χρησιμοποιείται προκειμένου να κλείσει το ανοιγμένο αρχείο. Από την στιγμή που κλείσουμε το αρχείο, δεν μπορούν να εκτελεστούν επιπλέον πράξεις σε αυτό.

## Συνολικός Κώδικας

```
import csv
file = open("Salary_Data.csv")
csvreader = csv.reader(file)
header = next(csvreader)
print(header)
for row in csvreader:
    print(row)
file.close()
```

## Γράψιμο σε ένα CSV αρχείο

Για να γράψετε δεδομένα σε ένα αρχείο CSV, ακολουθήστε τα εξής βήματα:

1. Εισαγωγή της csv βιβλιοθήκης για διάβασμα csv  
**import csv**
2. Ανοίξτε το αρχείο CSV για εγγραφή (λειτουργία w) χρησιμοποιώντας τη συνάρτηση open().  
**f = open("Salary\_Data.csv", 'w')**
3. Δημιουργήστε ένα αντικείμενο εγγραφής CSV καλώντας τη συνάρτηση writer() της μονάδας csv.  
**writer = csv.writer(f)**
4. Γράψτε δεδομένα στο αρχείο CSV καλώντας τη μέθοδο writerow() ή writerows() του αντικειμένου εγγραφής CSV.  
**writer.writerow(row)**

- Τέλος, κλείστε το αρχείο μόλις ολοκληρώσετε την εγγραφή δεδομένων σε αυτό.  
**f.close()**

## Συνολικός Κώδικας

```
import csv
f = open("Salary_Data.csv", 'w')
writer = csv.writer(f)
row = ['a', 'b', 34, 57]
writer.writerow(row)
f.close()
```

## Διάβασμα από JSON/AST Συμβολοσειρών

Θα δούμε τον τρόπο με τον οποίο μπορούμε να αναλύσουμε και να διαβάσουμε ένα JSON αρχείο μέσω python μετατρέποντας το σε ένα dictionary.

Το διάβασμα μιας JSON/AST Συμβολοσειράς γίνεται στα ακόλουθα βήματα

- Για να επεξεργαστείτε μία JSON συμβολοσειρά μπορείτε να χρησιμοποιήσετε τη βιβλιοθήκη `ast` της Python (Χρησιμοποιούμε την βιβλιοθήκη `ast` όταν έχουμε να διαβάσουμε συμβολοσειρές σε Python abstract syntax grammar που έχει παρόμοια δομή με json).

**import ast**

- Έστω ότι το αλφαριθμητικό μας περιέχει πληροφορίες για τις λέξεις κλειδιά σε μία ταινία  
**jsonString = "[{'id': 931, 'name': 'jealousy'}, {'id': 6054, 'name': 'friendship'}]"**
- Χρησιμοποιώντας την μέθοδο `literal_eval`, μπορούμε να πάρουμε τα περιεχόμενα του `jsonString` στην κατάλληλη μορφή αντικειμένου για περαιτέρω επεξεργασία. Π.χ. Η εντολή

**data = ast.literal\_eval(jsonString)**

θα επιστρέψει μία λίστα κάθε στοιχείο της οποίας είναι ένα λεξικό που αναφέρεται σε ένα αντικείμενο keyword. Π.χ

**print(data)** → εκτυπώνει [{'id': 931, 'name': 'jealousy'}, {'id': 6054, 'name': 'friendship'}]

**print(data[0])** → εκτυπώνει {'id': 931, 'name': 'jealousy'}

**print(data[0]['id'])** → εκτυπώνει 931, 'name': 'jealousy'}