

SEARCH STUDENT — Ανάλυση

◆ Κώδικας

```
Console.WriteLine("Search name: ");
string search = Console.ReadLine();

var results = students.Where(s =>
s.name.ToLower().Contains(search.ToLower()));

foreach (var s in results)
{
    Console.WriteLine($"{s.name} - {s.age}");
}
```

◆ Τι κάνει συνολικά

Ο χρήστης γράφει όνομα → το πρόγραμμα ψάχνει → εμφανίζει αποτελέσματα

◆ Ανάλυση γραμμή-γραμμή

1 Είσοδος χρήστη

```
string search = Console.ReadLine();
```

☞ Ο χρήστης γράφει π.χ.:
Nik

2 Αναζήτηση

```
students.Where(...)
```

☞ Η `Where` φιλτράρει τη λίστα

3 Συνθήκη

```
s => s.name.ToLower().Contains(search.ToLower())
```

☞ Σημαίνει:
Κράτα τους μαθητές των οποίων το όνομα περιέχει το `search`

◆ Ανάλυση σημαντικών κομματιών

ToLower()

☞ Μετατρέπει σε μικρά γράμματα
Nikos → nikos

☞ Για να μην μας νοιάζει αν γράψουν:

```
nik  
NIK  
Nik
```

Contains ()

☞ Ελέγχει αν το όνομα περιέχει το κείμενο

```
"Nikos".Contains("Nik") → true
```

◆ Παράδειγμα

Αν γράψει:

```
Search name: Nik
```

Αποτέλεσμα:

```
Nikos - 20
```

◆ Συμπέρασμα

Search = φίλτρο λίστας

🗑️ □ DELETE STUDENT — Ανάλυση

◆ Κώδικας

```
Console.WriteLine("Name to delete: ");  
string name = Console.ReadLine();  
  
var student = students.FirstOrDefault(s => s.name.ToLower() ==  
name.ToLower());  
  
if (student != null)  
{  
    students.Remove(student);  
  
    string newJson = JsonSerializer.Serialize(students, new  
JsonSerializerOptions { WriteIndented = true });  
    File.WriteAllText("students.json", newJson);  
  
    Console.WriteLine("Student deleted.");  
}  
else  
{  
    Console.WriteLine("Student not found.");  
}
```

◆ Τι κάνει συνολικά

Ο χρήστης δίνει όνομα → το βρίσκει → το αφαιρεί → αποθηκεύει

◆ Ανάλυση

1 □ Είσοδος

```
string name = Console.ReadLine();
```

☞ π.χ.:
Maria

2 □ Εύρεση μαθητή

```
students.FirstOrDefault(...)
```

☞ Βρίσκει τον πρώτο μαθητή που ταιριάζει

3 □ Συνθήκη

```
s.name.ToLower() == name.ToLower()
```

☞ Σημαίνει:
βρες μαθητή με ακριβώς ίδιο όνομα

◆ Τι είναι το FirstOrDefault

☞ Επιστρέφει:

- τον μαθητή αν βρεθεί
 - null αν δεν υπάρχει
-

4 □ Έλεγχος

```
if (student != null)
```

☞ Αν βρέθηκε

5 □ Διαγραφή

```
students.Remove(student);
```

☞ Αφαιρεί τον μαθητή από τη λίστα

6 □ Αποθήκευση

```
File.WriteAllText(...)
```

☞ Ξαναγράφει το JSON

7  Αν δεν βρεθεί
Student not found

◆ Παράδειγμα

Name to delete: Maria

Αποτέλεσμα:
Student deleted.

και η Maria φεύγει από το JSON

ΣΥΝΟΛΙΚΗ ΕΙΚΟΝΑ

Search

List → Filter → Display

Delete

List → Find → Remove → Save