

Java

Laborator 6

Danciu Gabriel
Noiembrie 2018

Exemplu de scriere / citire din fisier

Obiectul pe care il vom scrie/citi din fisier:

```
class Student {  
    String nume;  
    String prenume;  
    int grupa;  
  
    public Student(String nume, String prenume, int grupa) {  
        this.nume = nume;  
        this.prenume = prenume;  
  
        this.grupa = grupa;  
    }  
    public String toString() {  
  
        return nume + ", " + prenume + ", " + grupa;  
    }  
}
```

Citirea

static ArrayList<Student> studenti = new ArrayList<Student>(); //membru al clasei ce conține funcția Citire

```
static void Citire(String filepath) {
    try {
        File f = new File(filepath);
        BufferedReader br = new BufferedReader(new FileReader(f));
        String line = br.readLine();
        //ignor prima linie (antetul)
        if (line != null) {
            line = br.readLine();
        }
        while (line != null) {
            String[] splituri = line.split(",");
            Student s = new Student(splituri[0], splituri[1].trim(), Integer.parseInt(splituri[2].trim()));
            studenti.add(s);
            line = br.readLine();
        }
    } catch (Exception ex) {
        System.out.println(ex);
    }
}
```

Scrierea

```
static void Scriere(String filepath) {
    try {
        File f = new File(filepath);
        BufferedWriter bw = new BufferedWriter(new FileWriter(f));
        try {
            studenti.add(new Student(„Ionescu", „Dan", 3));
            bw.write("nume, prenume, grupa\r\n"); //se scrie antetul
            for (Student stud : studenti) {
                bw.write(stud.toString() + "\r\n");
            }
            bw.flush();
        } catch (IOException e) {
            System.out.println(e);
        } finally {
            bw.close();
        }
    } catch (Exception ex) {
        System.out.println(ex);
    }
}
```

Exerciții

- Modificați fișierul *ManagerCursCSV.java* din proiect astfel:

```
public class ManagerCursCSV extends Manager<Curs> implements IOperatiiManagerCurs {
```

```
    File studenti, profesori, cursuri;

    public ManagerCursCSV() {
        try {
            studenti = new File("studenti.csv");
            profesori = new File("profesori.csv");
            cursuri = new File("cursuri.csv");
        } catch (Exception ex) {
            System.out.println(ex);
        }
    }
}
```

```
public ManagerCursCSV(File studenti, File profesori, File cursuri) {
    this.studenti = studenti;
    this.profesori = profesori;
    this.cursuri = cursuri;
}
```

```
/*functia va aduce din fisiere, datele despre studenti, profesori, cursuri
si va popula colectia din clasa Manager: lista*/
```

```
public void CitesteDateDinFisiere() {
    try {
        ArrayList<Student> studentlist = PopuleazaStudenti();
        ArrayList<Profesor> proflist = PopuleazaProfesori();
        lista = PopuleazaCursuri(proflist, studentlist);

    } catch (Exception ex) {
        System.out.println(ex);
    }
}....
```

Exerciții

- Scrieți cele 3 metode de populare studenți, profesori, cursuri, din exemplul de mai sus.
- Adăugați o metodă de a salva cursurile adăugate modificate pe parcurs. Metoda se va numi *ScrieCursurileInFisier*.
 - Dacă un profesor este nou adăugat, el va fi salvat în profesori.csv
 - Dacă un student este nou adăugat, el va fi salvat în studenți.csv
 - Indiciu: Id-ul unui student/profesor nou adăugat este egal cu id-ul ultimului student/profesor din listă + 1.
- Completați metodele de Add/Update/Delete ale studentului/profesorului/cursului. Aceste metode vor face parte din clasa *ManagerCursCSV*.