

Laborator 8.2 – Administrarea resurselor

Obiective

După completarea acestui laborator veți dobândi următoarele cunoștințe:

- Crearea unui destructor
- Realizarea cererilor către **Garbage Collector**
- Utilizarea șabloanelor de design pentru metoda **Dispose()**

Condiții prealabile

Înainte de a realiza acest laborator trebuie să fiți familiarizați cu următoarele concepte:

- Crearea claselor și instanțierea obiectelor
- Definirea și apelarea metodelor
- Definirea și utilizarea constructorilor
- Utilizarea clasei **StreamWriter** pentru a scrie într-un fișier

➤ Exercițiul 1 – Utilizarea șablonului de design pentru **Dispose()**

În acest exercițiu veți adăuga o metodă **Dispose()** la clasa **BankAccount**. Metoda **Dispose** va itera prin toate tranzacțiile din coada de tranzacții și va salva logurile tranzacțiilor într-un fișier.

Crearea metodei **Dispose** pentru clasa **BankAccount**

- Deschideți proiectul de la exercițiul anterior.
- Adăugați modificatorul **sealed** la definirea clasei **BankAccount** și derivați din interfața **IDisposable**. O clasă marcată ca **sealed** nu poate fi moștenită. De data aceasta adăugăm acest modificator pentru a realiza o implementare cât mai simplă a metodei **Dispose**.
- Adăugați un câmp privat de tip **bool**, numit **disposed**, inițializat cu **false**.
- Adăugați o metodă publică de tip **void** numită **Dispose**.
- În metoda **Dispose** realizați următoarele operații:
 - Verificați valoarea câmpului **disposed**. Dacă este **true**, ieșiți din metodă printr-o instrucțiune **return**;
 - Dacă **disposed** este **false** veți declara o variabilă **StreamWriter** care va deschide un fișier în modul **Append** (adică va scrie datele la sfârșitul celor existente deja);

```
StreamWriter fout = File.AppendText("Tranzactii.txt");
```

- Atenție! Fișierul **"Tranzactii.txt"** îl veți crea dinainte în directorul Debug (**/bin/debug**) din proiectul vostru;
- În fișierul **"Tranzactii.txt"** veți afișa informațiile legate de cont utilizând metoda **WriteLine** (**fout.WriteLine()**):
numarul de cont, soldul, tipul contului, coada de tranzacții.



- După ce ați terminat scrierea informațiilor închideți fișierul (metoda **close**).
- Setati **disposed = true**.
- Apelați metoda **GC.SuppressFinalize(this)**.
- Adăugați un destructor clasei.
- Compilați și corectați eventualele erori.

Testarea Destructorului

- În metoda **Main()**, comentați liniile de cod existente și adăugați următoarele linii:

```
using (BankAccount account = new BankAccount ( ))
{
    account.Deposit(100);
    account.Withdraw(50);
    account.Deposit (75);
    account.Withdraw (50);
    account.Withdraw (30);
    account.Deposit (40);
    account.Deposit (200);
    account.Withdraw (250);
    account.Deposit (25);
    BankAccount.Write(account);
}
```

- Salvați și compilați programul.
- Deschideți fișierul „**Tranzactii.txt**” si verificați că informațiile au fost scrise.

