

Firmă de înregistrări

Disciplină: Proiect Programare

Nume studenti: Grădinariu Flavius, Fărtan Nicoleta

Specializare: Informatică aplicată, anul 1, grupa 2, subgrupa 2

Data prezentării: 06/11/2022

Rezumat

Cerința acestui proiect este crearea și manipularea unei baze de date a unei firme de înregistrări. Programul realizat permite, cu ușurință, exact acest lucru. Acesta crează o bază de date ce stochează date personale a muzicienilor (CNP, Nume și Prenume, Adresa, Numărul de telefon, Numele instrumentului folosit de muzician, și Codul albumului creat de acesta), date despre albumul său (Codul albumului, Titlul său, Data apariției, Detalii despre drepturile de autor, Tipul formatului) și date despre instrumentele folosite (Numarul de identificare, nume, cheia muzicală). Manipularea datelor constă în crearea bazei de date, adăugarea de noi date, căutarea informațiilor dorite, modificarea datelor, ștergerea datelor și afișarea acestora în terminal pentru a nu fi nevoie să deschidem după fiecare schimbare, fișierul/fișierele în sine.

Resurse software

Acest proiect a fost realizat cu ajutorul aplicației Visual Studio Code, versiunea 1.68, și a trei fișiere de tip “txt”.

Descrierea algoritmului

- În crearea programului am utilizat limbajul C++ .
- Acesta se folosește de 3 fișiere în care sunt stocate și extrase date .
- Pentru secțiunea de meniu am utilizat un `do...while()` ce conține o serie de funcții pentru afișarea instrucțiunilor, o serie de funcții `switch()`, și funcții ce execută operațiile de ștergere, adăugare, etc.
- Majoritatea funcțiilor sunt de tip `void`.
- Executarea operațiilor se realizează prin intermediul unei clase `album` care mostesteneste alte doua clase `muzician` și `instrument` unde sunt stocate datele într-un set/vector pentru o prelucrare mai usoara a acestora.
- Operația de căutare, ștergere și modificare a datelor se realizează cu ajutorul unor comenzi de căutare :
 - `CNP`: "cnp-ul căutat"
 - `NUME_MZICIAN`: "numele muzicianului căutat"
 - `ADRESA`: "adresa muzicianului căutată"
 - `NR_TELEFON`: "numar de telefon cautat"
 - `NUME_INSTRUMENT`: "numele instrumentului cautat"
 - `NR_INSTRUMENT`: "numărul de indentificare al instrumentului cautat"
 - `CHEIE_MUZICALA`: "cheia muzicala a instrumentului căutată"
 - `COD_ALBUM`: "codul albumului căutat "
 - `NUME_ALBUM`: "titlul albumului căutat"
 - `DATA`: "data albumului cautat"

- COPYRIGHT: "numele deținătorului albumului cautat"
- FORMAT: "formatul albumului cautat"

Implementare

- Operația de creare este realizată astfel: se împarte în două tipuri: input mare și mic. Input-ul mic reprezintă preluarea datelor de la tastatură prin supraîncărcarea operatorului istream (">>"). Apoi datele sunt introduse în fisierele text cu ajutorul unei clase. Input-ul mare reprezintă preluarea datelor dintr-un fișier "input.txt" pentru a nu mai fi necesară introducerea manuală a datelor. Datele sunt preluate din "input.txt" cu funcția `input_file.open(".....",ios::in)` și mutate în cele 3 fisiere cu ajutorul unui vector.
- Operația de append funcționează la fel ca cea de creare doar că cele 3 fisierele sunt deschise cu `file.open(".....", ios::app)`, ce adaugă noile date introduse la sfârșitul listei. Un fel de push-back.
- Operația de căutare: se deschid cele trei fișiere cu `file.open(".....", ios::in)`, apoi sunt preluate datele din fișiere cu trei `getline`-uri și salvate în clase care mai apoi sunt introduse într-un set. Apoi iau o variabilă pe care o despart în alte două. Acele alte două sunt două string-uri după care se caută o clasă. După căutare se afișează doar acele clase care îndeplinesc condiția.
- Operația de ștergere funcționează cam la fel cu cea de căutare însă după ce se caută clasele dorite va trebui confirmat dacă clasele găsite sunt și cele căutate, apoi se redeschid fisierele și se rescriu fără acele clase găsite, astfel fiind șterse.

- Operația de modificare, la fel ca și la ștergere, mă folosesc de comenzi să caut clasele în care vreau să schimb ceva apoi rescriu fișierul cu datele modificate.

Interfata grafica si meniu

Pentru partea de meniu si afisari in consola am utilizat libraria: `#include <windows.h>` .

Meniul are doua parti, una de titlu care e incadrat intr-un box si aliniat la mijloc si partea de optiuni aranjata la fel ca si titlul.

Dupa boxul cu optiuni apare cerinta de introducere a optiunii dorite.

Se poate observa o parte a meniului in imaginea de mai jos:

```
*****
*                                     *
*          FIRMA DE INREGISTRARI      *
*                                     *
*****

+-----+
| Instruțiuni generale (1)           |
| Baza de date a muzicienilor (2)    |
| Baza de data a albumelor (3)       |
| Baza de date a instrumentelor (4)  |
| Iesire (0)                         |
+-----+

Introduceti cifra corespunzatoare instructiunii dorite: |
```

Pentru culori am setat o functie de `SetColor()` care are ia un set de numere de la 0 la 14 reprezentate de diferite culori. Cateva exemple sunt: `SetColor(11)` este culoarea albastru deschis, `SetColor(5)` este culoarea magenta si asa mai departe.

Meniul se deschide cu partea de Firma de înregistrari care are 4 submeniuri, aceste submeniuri se pot observa in imaginea de mai sus. Fiecare submeniu are optiunile sale.

De asemenea si functiile de afisari au culori setate .

Concluzii si directii viitoare

Programul realizat de noi este unul optim si eficient in administrarea unei baze de date a unei firme de inregistrari muzicale.

Ca orice program, aceasta este versiunea initiala ,iar in timp mai poate fi imbunatatit si chiar adaugate alte functionalitati in functie de cerintele utilizatorilor.