UNIVERSITATEA DIN BUCUREȘTI FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ

Instrumente și Tehnici de Bază în Informatică Proiect “XML”

XML\_PARSER

Constantin Cristiana Georgiana

# Scopul proiectului XML

Scrieți un program (script) shell care să parseze fișiere XML. Acesta trebuie să poată citi și scrie date din, respectiv, în format XML.

# Introducere despre XML

XML (eXtensible Markup Language) este un limbaj de marcare folosit pentru a structura și stoca date într-un format ușor de citit și procesat.

* 1. *Utilizări principale:*
     + Schimbul de date între aplicații ,baze de date și web services.
     + Documente și fișiere de interschimb (de exemplu, SVG, MathML).
  2. *Structura unui document XML:*

Un document XML are o structură ierarhică formată din elemente (tag-uri) care conțin date sau alte elemente. Fiecare document XML trebuie să aibă un element rădăcină care conține toate celelalte elemente.

* 1. *Caracteristici:*
     + Structură ierarhică.
     + Spre deosebire de HTML, XML permite utilizatorilor să definească propriile tag-uri, fiind astfel flexibil și adaptabil diverselor aplicații.
  2. *Parser XML*

Un **parser XML** este un program care analizează un fișier XML și îl interpretează pentru a extrage date din acesta sau pentru a-l modifica. Un parser XML convertește un document XML într-o structură de date pe care aplicațiile o pot manipula ușor.

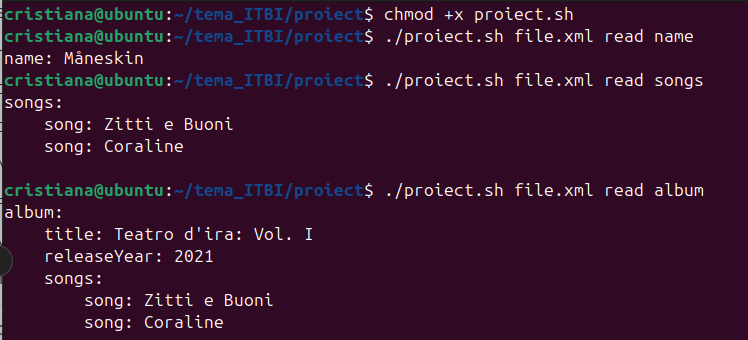
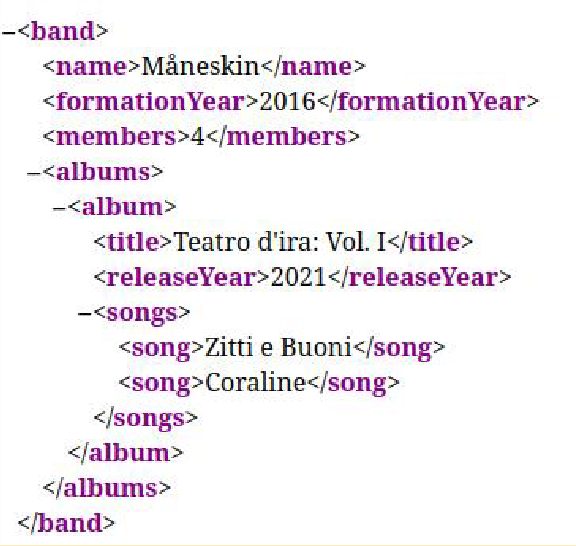
Dacă un program vrea să extragă numele unui student dintr-un fișier XML, un parser XML ar analiza documentul și ar returna valoarea conținutului din tag-ul <name>.

# Rezolvarea cerintei

* 1. Cum functioneaza read\_xml()

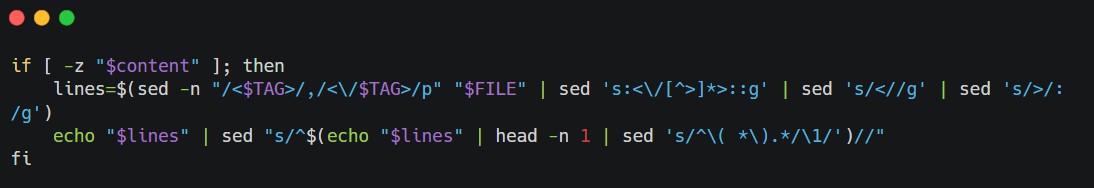
1. Argumente primite:
   * Numele fișierului XML ($FILE).
   * Numele tag-ului pe care dorim să-l citim ($TAG).
2. Fluxul funcției:
   * Verifică dacă tag-ul specificat conține doar un text simplu sau dacă este un tag mai complex (nested).
   * Pentru tag-uri simple: Extractează conținutul dintre <tag> și </tag> folosind sed si afișează valoarea găsită.
   * Pentru tag-uri complexe: Citește toate liniile dintre <tag> și </tag> folosind sed si reformatează structura pentru a afișa conținutul într-un format lizibil.

Exemplu de rulare și comportament :



Dificultăți întâmpinate la implementarea funcției read\_xml

1. Gestionarea tag-urilor inexistente
2. Afișarea lizibilă a structurii (păstrarea indentării pentru structurile adânci.
3. Gestionarea tag-urilor nested
   * Problemă: Tag-uri precum <album> conțin alte tag-uri, iar extragerea structurii complete era dificilă.
   * Soluție: Utilizarea combinată a comenzilor sed pentru extragerea liniilor și reformatarea acestora într-un format ierarhic.



1. Extragerea multiplă a valorilor pentru același tag
   * Problemă: Tag-uri multiple cu același nume (ex. <song>) trebuiau afișate complet.
   * Soluție: Folosirea sed -n pentru a itera prin toate aparițiile și afișarea fiecărei valori.



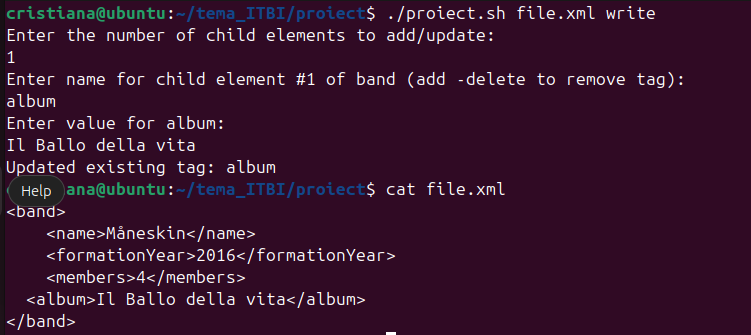
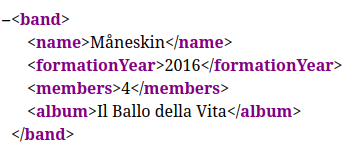
* 1. Cum functioneaza write\_xml()
* Rolul funcției

Funcția write\_xml permite modificarea sau adăugarea de noi elemente într-un

fișier XML. Aceasta gestionează atât tag-uri simple (care conțin doar o valoare), cât și tag-uri complexe (nested), adăugând copii sub un tag părinte.

1. Argumente primite:
   * Numele fișierului XML ($FILE).
   * Opțiunea write: Indică faptul că funcția va scrie sau modifica elementele din fișier.Fluxul funcției:
2. Fluxul functiei:
   * Verifică dacă tag-ul specificat conține doar un text simplu sau dacă este un tag mai complex (nested).
   * Pentru tag-uri simple: Extractează conținutul dintre <tag> și </tag> folosind sed si afișează valoarea găsită.
   * Pentru tag-uri complexe: Citește toate liniile dintre <tag> și </tag> folosind sed si reformatează structura pentru a afișa conținutul într-un format lizibil.

Exemplu de rulare și comportament :



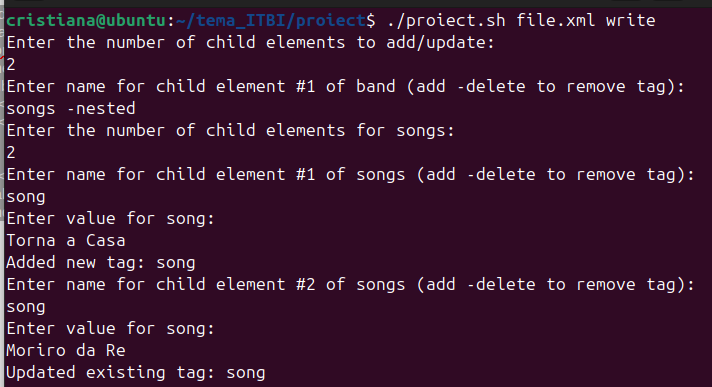
Adăugarea unei melodii într-un nou album:

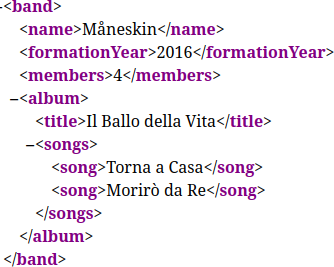
Bibliografie:

[https://www.w3schools.com/xml/xml](https://www.w3schools.com/xml/xml_examples.asp)

[\_examples.asp](https://www.w3schools.com/xml/xml_examples.asp)

ooscbsc





Dificultăți întâmpinate la implementarea funcției write\_xml:

1. Adăugarea de tag-uri nested
   * Problemă: Gestionarea tag-urilor nested (ex. <album> care conține <title> și <songs>) era complexă, necesitând o abordare recursivă pentru a adăuga copii sub un tag părinte.
   * Soluție: Funcția write\_children a fost implementată pentru a automatiza procesul de creare și gestionare a structurilor ierarhice.

Explicatie: Dacă utilizatorul specifică un tag nested (-nested), funcția verifică dacă tag-ul există deja. Dacă nu există, este creat și apoi funcția recursivă write\_children este apelată pentru a adăuga copii.

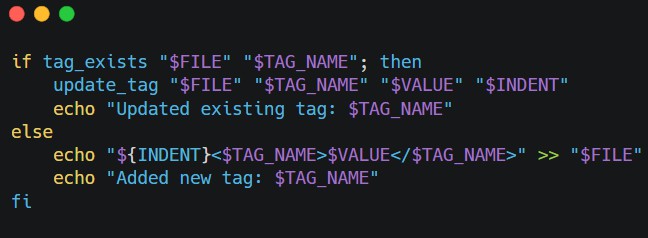
1. Menținerea indentării corecte
   * Problemă: La adăugarea de noi elemente, indentarea trebuia să fie consistentă pentru a păstra lizibilitatea fișierului XML.
   * Soluție: Utilizarea unor comenzi precise în awk și sed pentru a păstra formatul și indentarea la adăugarea sau actualizarea tag-urilor.

sed -i '/^$/d' "$FILE"

După adăugarea sau actualizarea tag-urilor, comanda sed curăță liniile goale pentru a menține un format consistent.

Indentarea este gestionată prin adăugarea spațiilor corespunzătoare în funcția write\_children: echo "${INDENT}<$TAG\_NAME>$VALUE</$TAG\_NAME>" >> "$FILE"

1. Gestionarea tag-urilor existente
   * Problemă: Dacă tag-ul specificat de utilizator exista deja, funcția trebuia să actualizeze valoarea acestuia fără a afecta alte părți din fișier.
   * Soluție: Funcția verifică existența tag-ului utilizând tag\_exists. Dacă tag-ul există, valoarea este actualizată cu ajutorul unei comenzi awk sau sed.



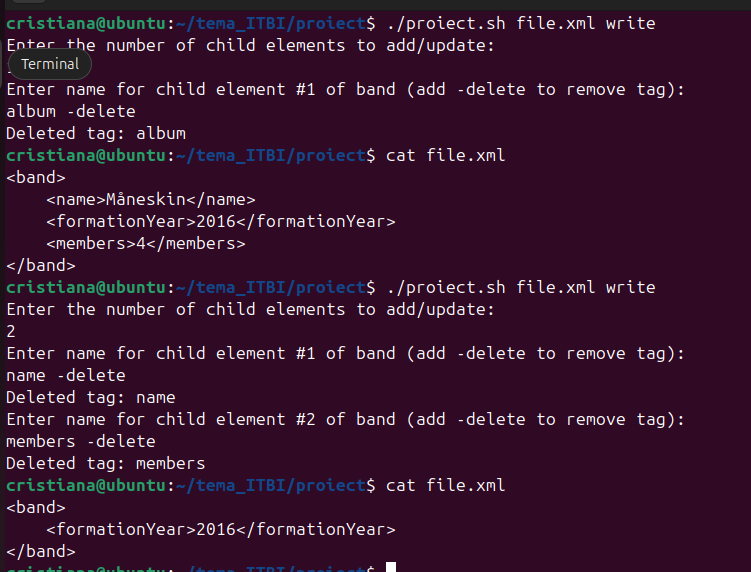
* 1. Cum functioneaza delete\_tag()

7

Funcția delete\_tag este responsabilă pentru ștergerea unui tag specific și a conținutului acestuia dintr-un fișier XML. Aceasta se asigură că ștergerea este completă și nu afectează alte părți ale fișierului.

* + - **Argumente primite**
      1. Numele fișierului XML ($FILE): Fișierul în care se efectuează ștergerea.
      2. Tag-ul de șters ($TAG): Numele tag-ului care trebuie șters, inclusiv conținutul dintre <tag> și </tag>.
    - Fluxul funcției
      1. Identificarea tag-ului:
         * Funcția folosește awk pentru a găsi toate liniile dintre <TAG> și </TAG>.
      2. Ștergerea completă:
         * Elimină atât tag-ul deschis și cel închis, cât și orice conținut dintre ele.
      3. Curățarea fișierului:
         * După ștergere, fișierul este verificat pentru a elimina liniile goale și a menține un format consistent.

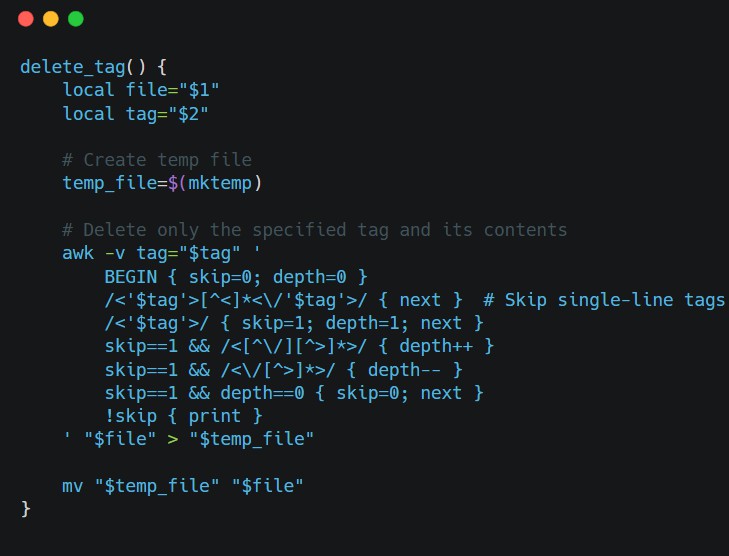
Exemplu de rulare și comportament :



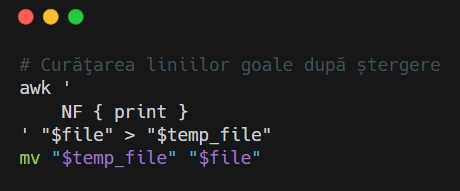


Dificultăți întâmpinate la implementarea funcției delete\_tag

1. Identificarea completă a tag-urilor nested:
   * Problemă: Tag-urile care conțineau alte tag-uri (nested) necesitau gestionare recursivă pentru a asigura ștergerea completă.
   * Soluție: Folosirea awk pentru a parcurge și șterge toate liniile dintre <TAG> și </TAG>.



1. Gestionarea liniilor goale:
   * Problemă: După ștergere, liniile goale din fișier afectau lizibilitatea.
   * Soluție: Curățarea fișierului cu comanda awk 'NF { print }'.



1. Compatibilitatea cu tag-uri similare:
   * Problemă: Ștergerea unui tag specific putea afecta alte tag-uri cu nume similare.
   * Soluție: Utilizarea expresiilor regulate în awk pentru potrivire exactă



# Încheiere

Proiectul de creare a unui script shell pentru parsarea fișierelor XML a demonstrat că este posibil să gestionăm operațiuni complexe, precum citirea, scrierea și ștergerea tag- urilor, folosind unelte standard precum awk, sed și bash. Scriptul poate manipula atât structuri simple, cât și ierarhii complexe, fiind util în diverse scenarii.

Deși s-au întâmpinat dificultăți legate de gestionarea tag-urilor nested, păstrarea indentării și compatibilitatea cu tag-uri similare, acestea au fost rezolvate prin soluții eficiente.

# Bibliografie

Documentația awk <https://www.gnu.org/software/gawk/manual/gawk.html> Documentația sed <https://www.gnu.org/software/sed/manual/sed.html> XML Standards <https://www.w3.org/XML/>

Tutorial Bash <https://tldp.org/LDP/abs/html/>

Acest proiect a fost realizat în echipă în cadrul laboratorului Instrumente si tehnici de baza in informatica .