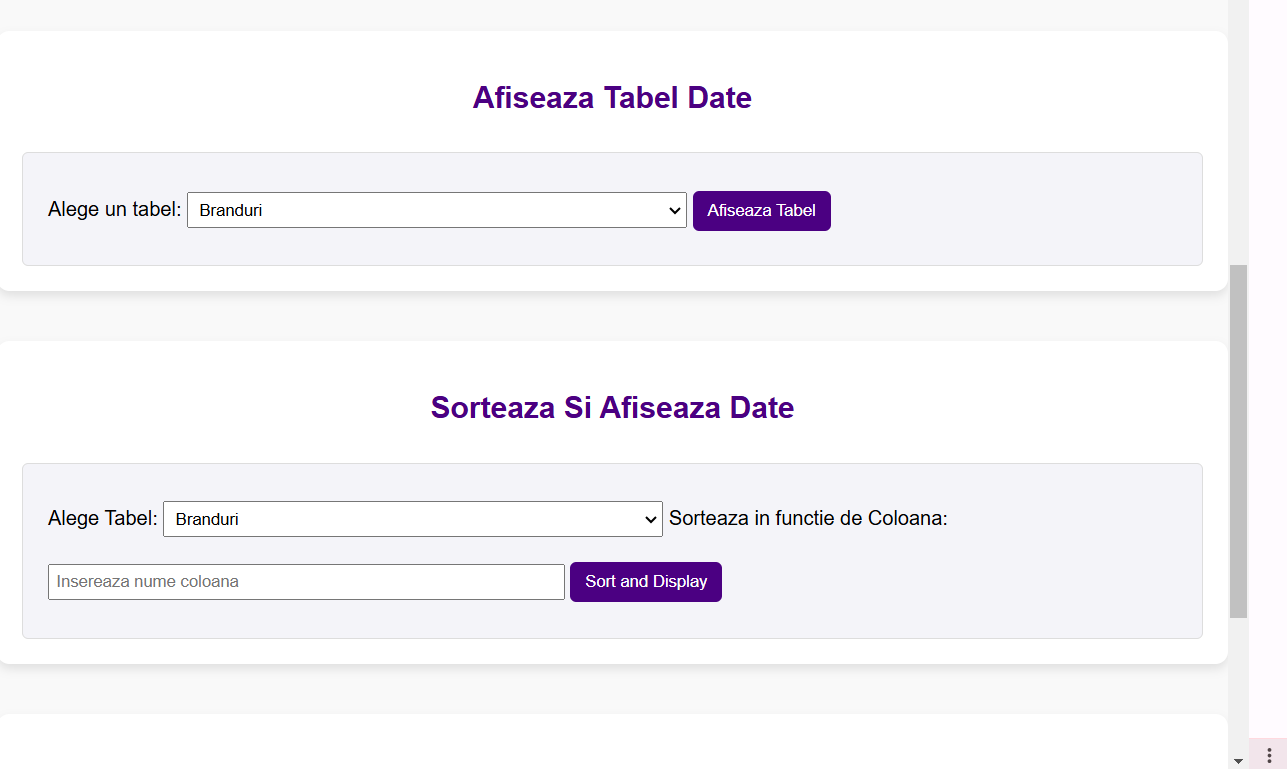
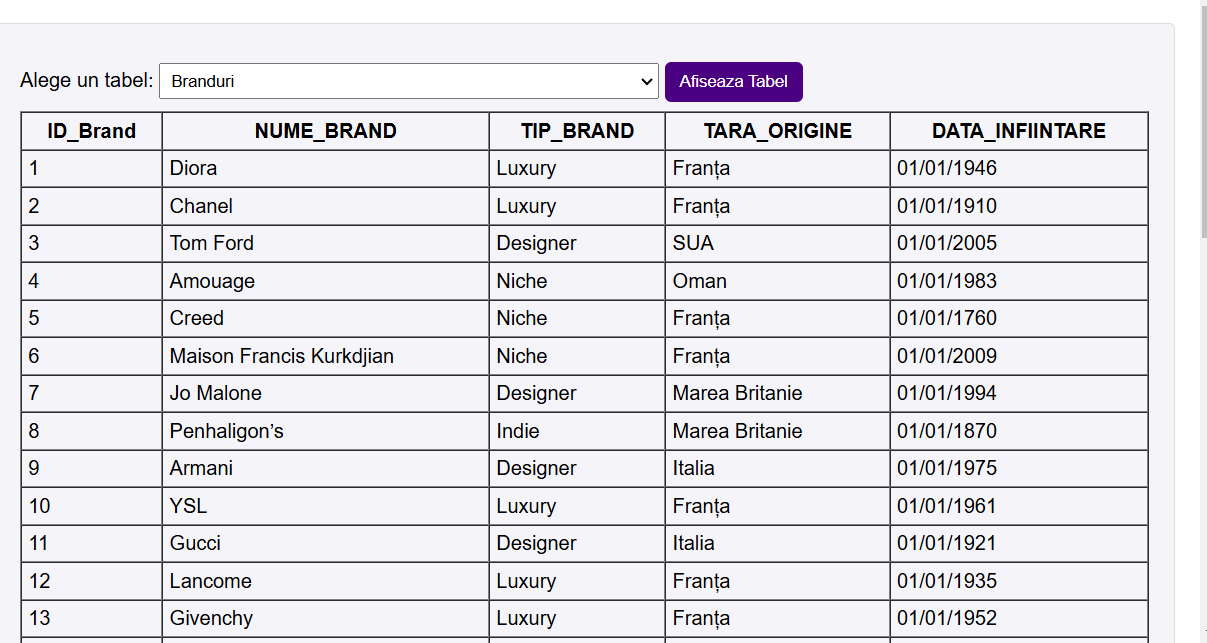
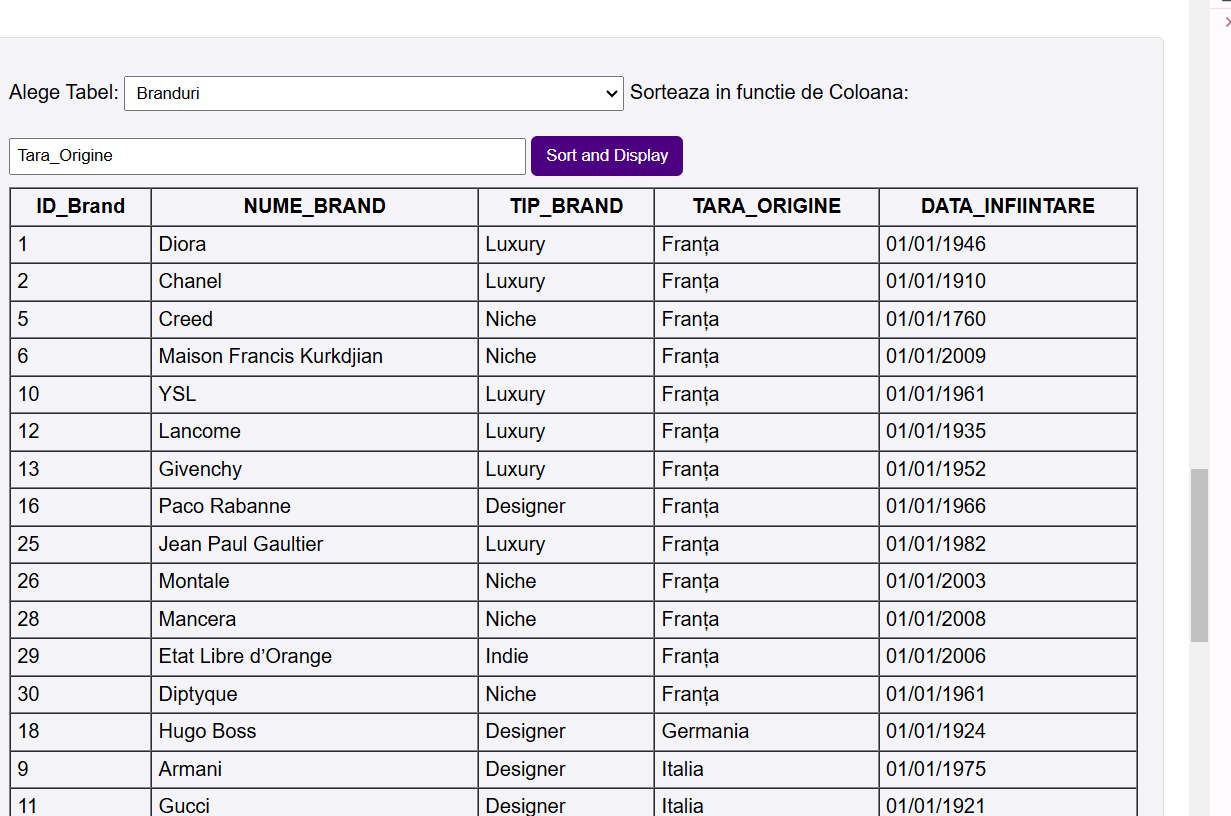
**Interfata Baza de Date Parfumuri**

* **Listare conținut cu posibilitatea de sortare**

Pentru a îndeplini cerința de sortare a conținutului din toate tabelele bazei de date, am implementat interogări SQL care permit utilizatorilor să vizualizeze datele ordonate în funcție de diverse criterii, cum ar fi numele parfumurilor, datele de lansare sau numele brandurilor. Interfața va include opțiunea de a selecta coloanele dorite pentru sortare, iar rezultatele vor fi afișate în mod dinamic pe baza opțiunilor alese de utilizatori.



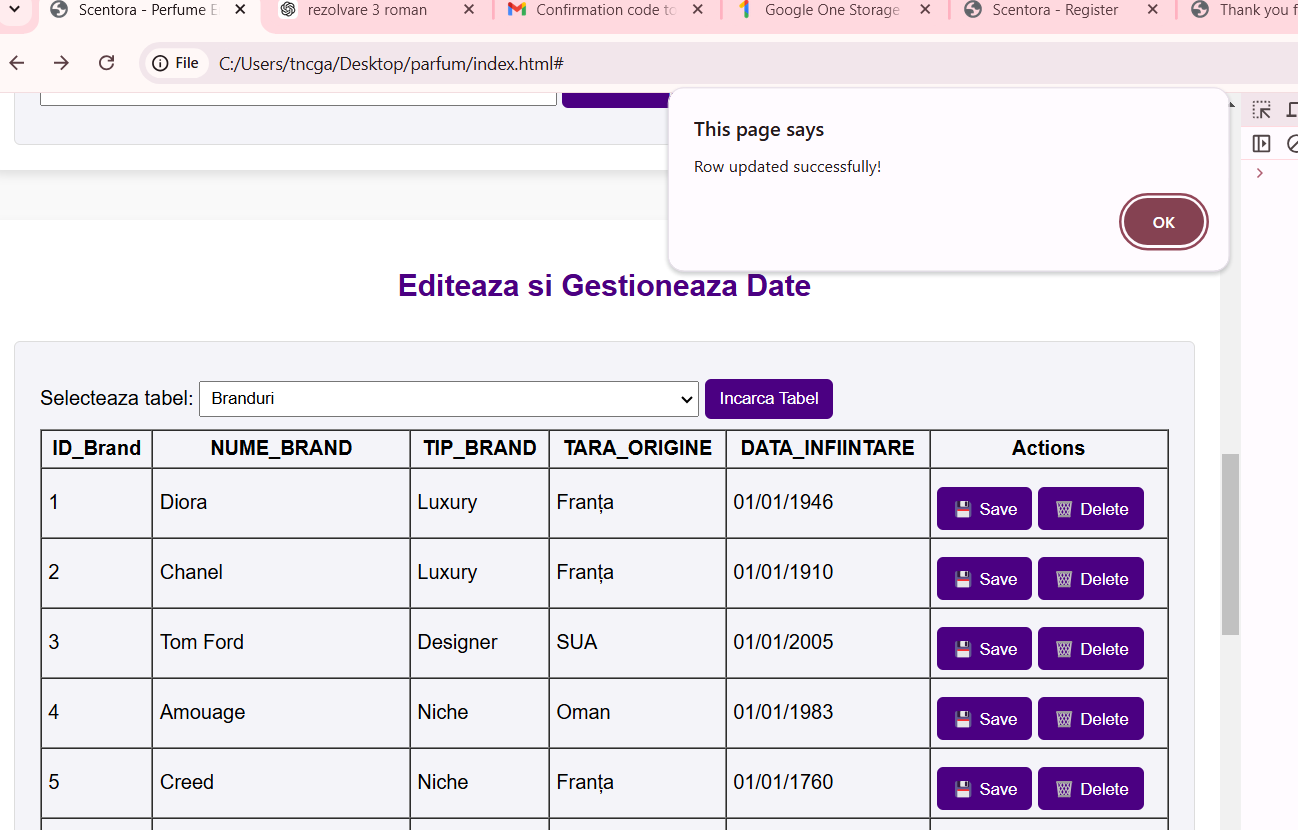




* **Modificare informații (opțiune de edit/ștergere pentru fiecare înregistrare)**

Am implementat proceduri care permit utilizatorilor să modifice informațiile din baza de date, cum ar fi adăugarea unui nou parfum sau editarea unui parfum existent (de exemplu, actualizarea prețului sau descrierii acestuia). De asemenea, s-a adăugat și opțiunea de a șterge un parfum din sistem, printr-o procedură de tip DELETE, care va șterge toate referințele și asocierile legate de acest parfum, respectând astfel constrângerea de tip ON DELETE CASCADE.



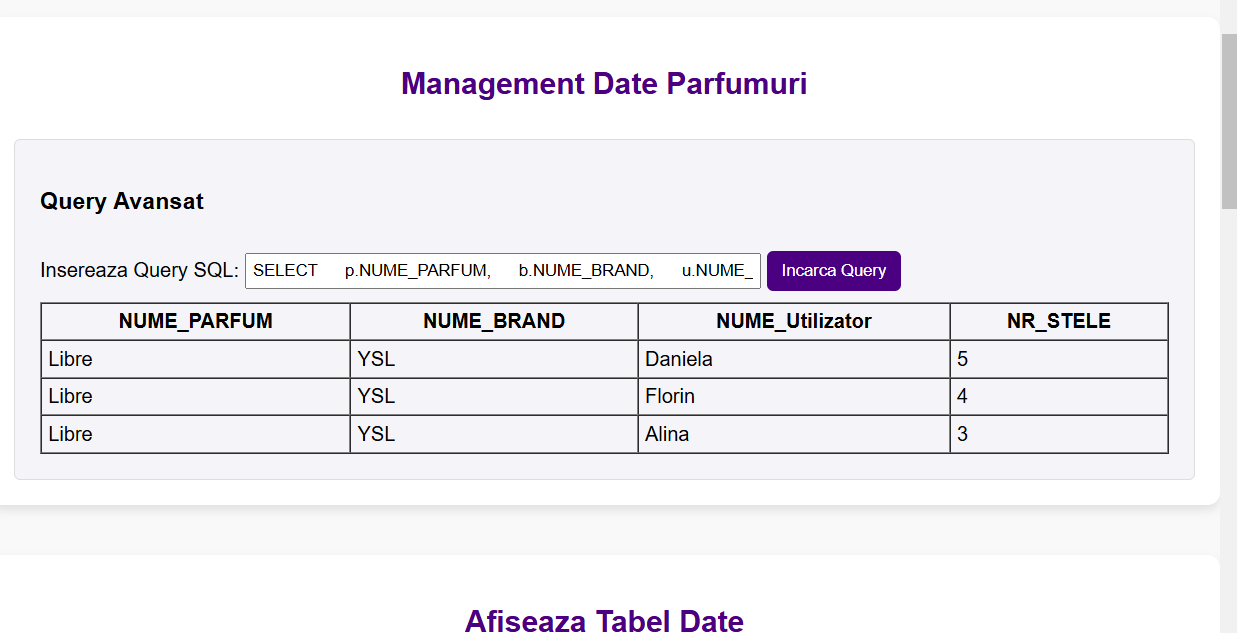


* **Afișarea rezultatelor unei cereri care extrage informații din cel puțin 3 tabele și le filtrează cu ajutorul a cel puțin 2 condiții**

Pentru a satisface această cerință, am implementat interogări SQL care unesc (join) cel puțin trei tabele din baza de date (de exemplu, Parfumuri, Branduri, Utilizatori) și filtrează rezultatele folosind condiții, precum tipul parfumului sau numele brandului. În acest mod, utilizatorul poate obține informații mai precise despre un parfum, brandul său asociat și utilizatorii care l-au evaluat.

Am pus urmatorul query:

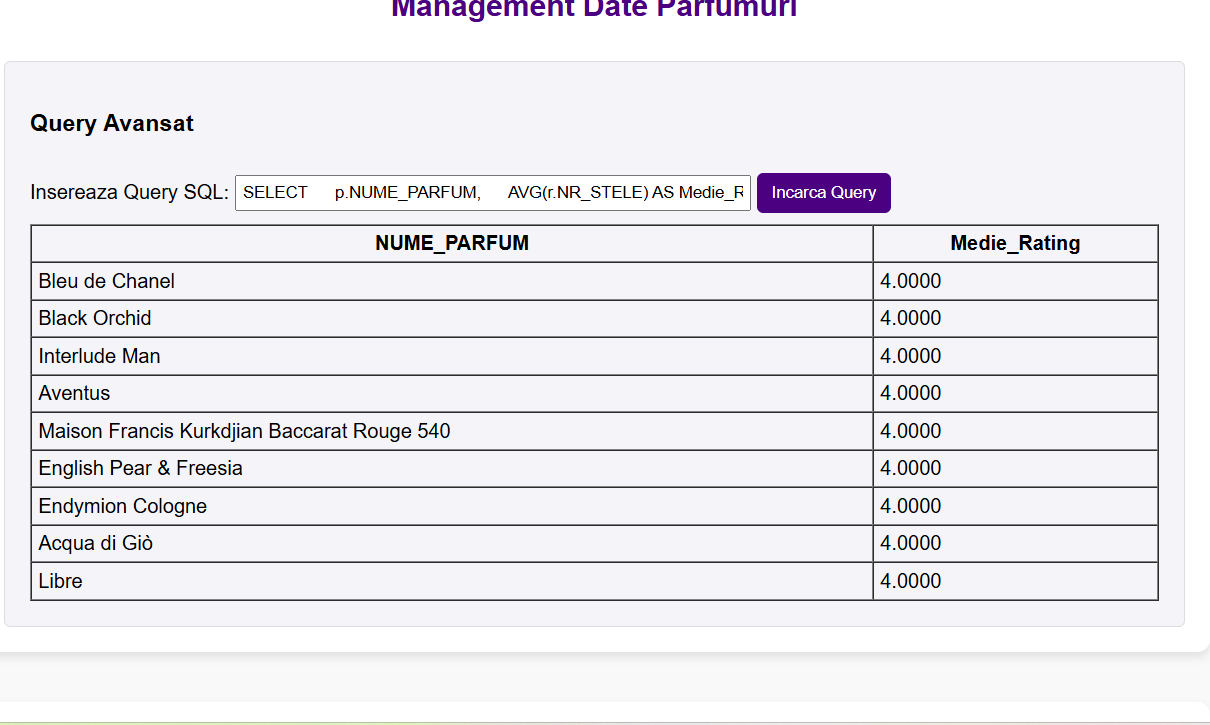
SELECT p.NUME\_PARFUM, b.NUME\_BRAND, u.NUME\_Utilizator, r.NR\_STELE FROM Parfumuri p JOIN Branduri b ON p.ID\_Brand = b.ID\_Brand JOIN Recenzii r ON p.ID\_Parfum = r.ID\_Parfum JOIN Utilizatori u ON r.ID\_Utilizator = u.ID\_Utilizator WHERE p.GEN = 'Femei' AND b.TARA\_ORIGINE = 'Franța' ORDER BY r.NR\_STELE DESC;



* **Afișarea rezultatelor unei cereri care folosește funcții grup și o clauză having**

Am implementat o interogare care utilizează funcții de agregare, cum ar fi COUNT, pentru a număra recenziile lăsate de utilizatori pentru fiecare parfum. Clauza HAVING este folosită pentru a filtra rezultatele și a afișa doar parfumurile care au primit un număr semnificativ de recenzii (de exemplu, mai mult de 3 recenzii). Aceste informații vor fi afișate pentru a oferi o perspectivă asupra celor mai apreciate parfumuri de pe platformă.

Am inserat acest query: SELECT p.NUME\_PARFUM, AVG(r.NR\_STELE) AS Medie\_Rating FROM Parfumuri p JOIN Recenzii r ON p.ID\_Parfum = r.ID\_Parfum GROUP BY p.ID\_Parfum HAVING AVG(r.NR\_STELE) >= 4;



* **Implementarea unei constrângeri de tip on delete cascade și exemplificarea din baza de date**

Am implementat constrângerea de tip ON DELETE CASCADE în relațiile dintre tabelele bazei de date. De exemplu, când un brand este șters din baza de date, toate parfumurile asociate acelui brand vor fi șterse automat, împreună cu toate asocierile relevante din tabelele de tip M:N, cum ar fi Parfumuri\_Ingrediente și Parfumuri\_Note. Acest lucru garantează integritatea datelor și evită existența unor referințe invalide în baza de date.