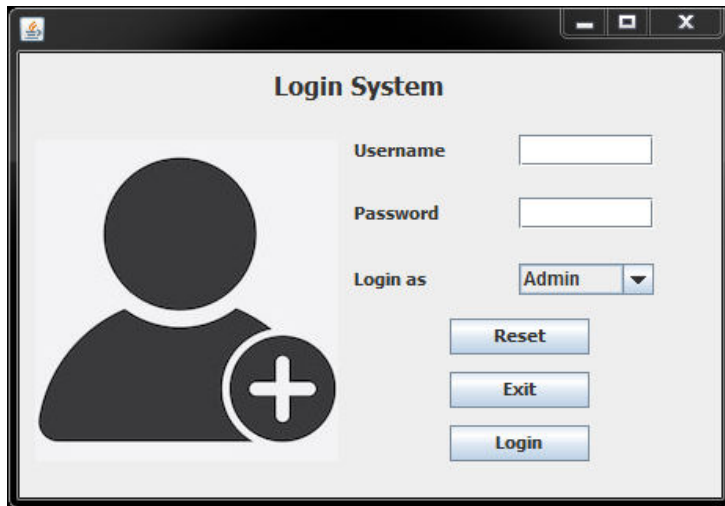


Aplicatie de feedback si clasificare a filmelor

Aplicatia va folosi o baza de date in care sunt stocate detalii despre diverse filme si actorii participanti. Fiecare utilizator va putea accesa aceasta baza de date, avand posibilitatea de a evalua filmul (pe baza unui username si a unui userID) prin acordarea unei note (cuprinsa intre 1 si 10) si de a lasa un comentariu cu opinia sa.



Totodata, aplicatia de fata foloseste mediul de dezvoltare de programe Java intitulat Eclipse prin care se doreste conectarea la baza de date (elaborata in SQL Server Management Studio 2012) propriu-zisa si executarea unor cereri fara ca utilizatorul sa aiba acces direct la baza de date. Conectarea cu baza de date se realizeaza printr-o interfata ce contine 3 campuri: Username, Password, Login as (conectarea la aplicatie se va face in calitate de admin sau client). Diferenta

dintre admin si client este faptul ca numai adminul are permisiunea de a modifica inregistrari din tabele, astfel incat in interfata pentru client vor lipsi butoanele care realizeaza aceste schimbari.

Valorile pentru aceste campuri, astfel incat sa permita conectarea aplicatiei la baza de date, se gasesc in tabela User_parola. Pe langa aceste campuri mai avem 3 butoane: Login (prin apasarea caruia se incearca conectarea la baza de date), Exit (prin care se iese din interfata) si Reset (care reseteaza valorile primelor 2 campuri: Username, Password – campului Login as ii este asociat un buton de tip JComboBox care nu poate fi resetat, fiind posibila alegerea a doua valori – Admin si Client).

Avem urmatoarele tabele:

- Actor (actor_id, actor_nume, actor_prenume, actor_sex)
- Regizor (regizor_id, regizor_nume, regizor_prenume)
- Film_actori (actor_id, film_id, rol)
- Film (film_id, film_titlu, film_an, film_durata, film_limba, film_tara, regizor_id, descriere)
- Review (review_id, film_id, comentariu, user_id, review_nota, ora_comentariu)
- User (user_id, username)

Relatii intre tabele:

Film – Regizor

Film – Review

Review – User

Film_actori – Actor

Film_actori – Film

Toate relatiile de mai sus sunt one to many.

Chei tabele:

Actor : actor_id (cheie primara)

Regizor : regizor_id (cheie primara)

Film_actori : actor_id (cheie primara) , film_id (cheie primara)

Film : film_id (cheie primara), regizor_id (cheie straina)

Review : review_id (cheie primara), user_id (cheie straina), film_id (cheie straina)

User : user_id (cheie primara)

Pentru fiecare dintre tabelele Actor, Film si Review am creat cate o clasa (de tip JFrame) care sa imi permita crearea unei interfete usor de folosit. In figura din dreapta avem interfata pentru clasa ModifActor (permite modificari in cadrul tabelii cu date despre actori). Avem astfel:

1. Un buton LoadActor care incarca in tabelul din dreapta jos inregistrarile stocate de baze de date la momentul respective
2. Un buton Save care imi permite inserarea unei inregistrari in baza de date prin completarea campurilor
3. Un buton Update care seteaza valoarea unor attribute pentru o inregistrare deja existent
4. Un buton Delete care elimina o inregistrare
5. Un buton de tip JComboBox care permite cautarea unei inregistrari dupa numele actorului in cazul de fata
6. Butoanele “Go to Film” si “Go to Review” redirectioneaza utilizatorul catre interfetele ce ofera informatii despre filme si review-uri

actor_id	actor_nume	actor_pren...	actor_sex
A101	Stewart	James	M
A102	Kerr	Deborah ...	F
A103	O'Toole	Peter	M
A104	De Niro	Robert	M
A105	Abraham ...	Murray	M
A106	Ford	Harrison	M
A107	Kidman ...	Nicole	F
A108	Baldwin	Stephen ...	M
A109	Nicholson ...	Jack	M
A110	Wahlberg ...	Mark	M
A111	Allen	Woody	M
A112	Danes	Claire	F
A113	Robbins ...	Tim	M
A114	Spacey	Kevin	M

7. Un buton Execute SQL care va afisa o interfata ce permite cautarea unor inregistrari dupa anumite criterii introduse de utilizator intr-un camp sau mai multe

film_id	film_titlu	film_an	film_durata	film_limba	film_tara	regizor_id	descriere
F901	Vertigo	1958	128	Engleza	UK	R004	drama
F902	The Innocent...	1961	100	Engleza	Suedia	R011	thriller
F903	Lawrence of A...	1962	216	Germana	Belgia	R002	comedie
F904	The Deer Hun...	1978	183	Spaniola	UK	R013	reality-show
F905	Amadeus	1984	160	Franceza	Australia	R008	horror
F906	Blade Runner...	1982	117	Engleza	SUA	R015	thriller
F907	Eyes Wide Sh...	1999	159	Portugheza	Italia	R001	telenovela
F908	The Usual Su...	1995	106	Franceza	Belgia	R013	drama
F909	Chinatown	1974	130	Chineza	China	R005	horror
F910	Boogie Night...	1997	155	Engleza	UK	R006	comedie
F911	Annie Hall	1977	93	Spaniola	Suedia	R012	telenovela
F912	Princess Mon...	1997	132	Japoneza	Japonia	R016	drama
F913	Jurassic Worl...	1994	142	Engleza	SUA	R010	horror

review_id	film_id	comentariu	user_id	review_nota	ora_comentariu
Rev201	F905	Imi place foarte m...	U9011	8.5	2018-11-10 12:56:...
Rev202	F913	Emotionant pana l...	U9004	10.0	2018-11-15 05:34:...
Rev203	F901	Mai bine faceam a...	U9002	4.0	2018-11-12 14:12:...
Rev204	F904	Nu am inteles nim...	U9012	2.5	2018-11-03 18:31:...
Rev205	F916	Nu imi convine fin...	U9010	7.0	2018-10-30 08:24:...
Rev206	F922	Asa si asa. Ce sa ...	U9012	6.5	2018-10-29 22:37:...
Rev207	F904	Merita toti banii ...	U9006	9.5	2018-11-05 10:57:...
Rev208	F916	O sa il revad ca mi...	U9004	9.0	2018-11-18 20:02:...
Rev209	F910	Efectele speciale ...	U9013	5.0	2018-10-24 15:30:...
Rev210	F923	Prea multa violent...	U9004	1.0	2018-11-22 19:01:...
Rev211	F911	Prea previzibil fina...	U9011	7.5	2018-10-29 13:24:...
Rev212	F916	A fost ok.	U9005	8.0	2018-11-21 03:48:...
Rev213	F923	Mi s-a facut pielea...	U9007	10.0	2018-11-11 16:27:...
Rev214	F919	Scenariul nu este ...	U9003	5.5	2018-11-13 10:22:...

actor_id	actor_nume	actor_prenume	actor_sex
A101	Stewart	James	M
A102	Kerr	Deborah	F
A103	O'Toole	Peter	M
A104	De Niro	Robert	M
A105	Abraham	Murray	M
A106	Ford	Harrison	M
A107	Kidman	Nicole	F
A108	Baldwin	Stephen	M
A109	Nicholson	Jack	M
A110	Wahlberg	Mark	M
A111	Allen	Woody	M
A112	Danes	Claire	F
A113	Robbins	Tim	M
A114	Spacey	Kevin	M

In stanga este o interfata pentru afisarea detaliilor despre actori pentru un utilizator de tip client, consecinta fiind ca nu avem butoanele Save, Delete sau Update.

Scrieti numele filmelor in care unul sau mai multi actori au jucat in cel putin un numar specificat de filme

Numar filme

film_titlu
Lawrence of Arabia
Blade Runner
The Usual Suspects
Donnie Darko

De asemenea, pentru executarea cererilor SQL fara ca utilizatorul sa aiba acces direct la baza de date (indifferent daca utilizatorul este admin sau client) am implementat o clasa numita `ExecuteSQLQuery` care creaza o interfata simpla. Butoanele Reset si Exit au aceeasi functionalitate ca la `LoginCheck`. In stanga este descris ce fel de rezultat returneaza executarea cererii corespunzatoare butonului Query din dreapta. Rezultatul cererii este afisat in tabelul (de tip `JTable`) de jos.

Select your query

INTEROGARI SIMPLE

Afisati filmele care au primit cel putin o anumita nota la review-uri si user-ul care a acordat nota respectiva alaturi de comentariul sau

Afisati toate filmele in care a jucat un anumit actor sau care au fost regizate de un anumit regizor

Afisati numele tuturor actorilor care au jucat in filme intr-o perioada specificata si numele filmelor

Afisati numele si prenumele regizorilor pentru care suma duratelor filmelor pe care le regizeaza este mai mare decat un anumit numar de minute

Afisati anii in care s-au regizat cel putin un film si care au primit cel putin o nota mai mare sau egala cu o anumita valoare

Scrieti numele utilizatorilor care au evaluat un anumit film

INTEROGARI COMPLEXE

Pentru un film pe care il selectati determinati numarul de actori care joaca in el

Gasiti filmele care au o medie a notelor acordate de utilizatori mai mare decat media tuturor notelor acordate pentru toate filmele

Gasiti filmele care au fost evaluate de toti in afara de un anumit utilizator pe care il specificati

Scrieti numele filmelor in care unul sau mai multi actori au jucat in cel putin un numar specificat de filme

Constrangeri de integritate

1. PRIMARY KEY

Actor : actor_id

Regizor : regizor_id

Film_actori : actor_id , film_id

Film : film_id

Review : review_id

User : user_id

2. FOREIGN KEY

Film : regizor_id (din tabela Regizor)

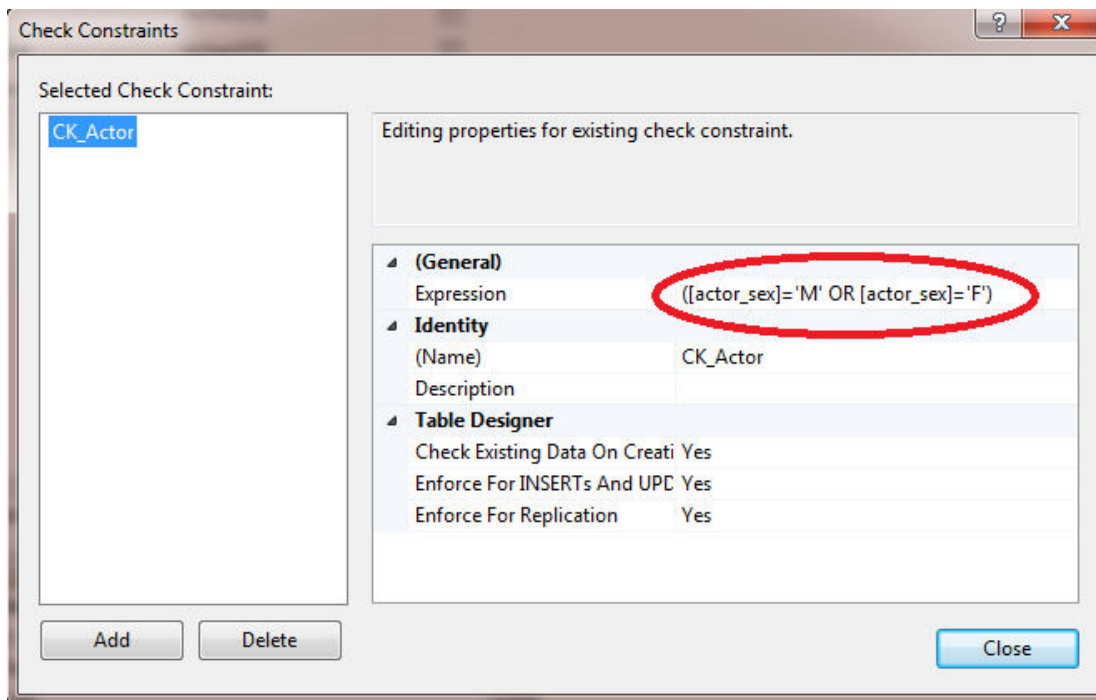
Review : user_id (din tabela User), film_id (din tabela Film)

3. NOT NULL

In toate tabelele, cel mult doua atribute au selectate optiunea Allow Nulls.

	Column Name	Data Type	Allow Nulls
🔑	actor_id	nchar(15)	<input type="checkbox"/>
	actor_nume	nchar(15)	<input type="checkbox"/>
	actor_prenume	nchar(15)	<input type="checkbox"/>
	actor_sex	nchar(1)	<input checked="" type="checkbox"/>
▶			<input type="checkbox"/>

4. CHECK



In tabela Actor, atributul actor_sex poate fi numai 'M' sau 'F'.

Interogari simple:

1) Afisati filmele care au primit cel putin o anumita nota la review-uri si user-ul care a acordat nota respectiva alaturi de comentariul sau

```
SELECT f.film_titlu, r.user_id, r.comentariu
FROM Film f INNER JOIN Review r
ON f.film_id = r.film_id
WHERE r.review_nota = ?
```

2) Afisati toate filmele in care a jucat un anumit actor sau care au fost regizate de un anumit regizor

```
SELECT f.film_titlu
FROM Film_actori fa INNER JOIN Film f
      ON f.film_id = fa.film_id
      INNER JOIN Actor a
      ON a.actor_id = fa.actor_id
WHERE a.actor_num = ? AND a.actor_prenume = ?
UNION
SELECT f.film_titlu
FROM Film f INNER JOIN Regizor r
      ON f.regizor_id = r.regizor_id
WHERE r.regizor_num = ?
```

3) Afisati numele tuturor actorilor care au jucat in filme intr-o perioada specificata si numele filmelor

```
SELECT a.actor_num, a.actor_prenume, f.film_titlu, f.film_an
FROM Film_actori fa INNER JOIN Film f
      ON f.film_id = fa.film_id
      INNER JOIN Actor a
      ON a.actor_id = fa.actor_id
WHERE f.film_an BETWEEN ? AND ?
```

4) Afisati numele si prenumele regizorilor pentru care suma duratelor filmelor pe care le regizeaza este mai mare decat un anumit numar de minute

```
SELECT r.regizor_num, r.regizor_prenume, SUM(f.film_durata) AS Durata_Filme
FROM Film f INNER JOIN Regizor r
      ON f.regizor_id = r.regizor_id
GROUP BY r.regizor_num, r.regizor_prenume
HAVING SUM(f.film_durata) > ?
ORDER BY SUM(f.film_durata) DESC
```

5) Afisati anii in care s-au regizat cel putin un film si care au primit cel putin o nota mai mare decat o anumita valoare

```
SELECT DISTINCT f.film_an
FROM Film f INNER JOIN Review r
      ON f.film_id = r.film_id
GROUP BY f.film_an, r.review_nota
HAVING COUNT(*)>=1 AND r.review_nota >=?
```

6) Scrieti numele utilizatorilor care au evaluat un anumit film

```
SELECT DISTINCT u.username
FROM Review r INNER JOIN [User] u
      ON r.user_id = u.user_id
      INNER JOIN Film f
      ON f.film_id = r.film_id
WHERE f.film_titlu = ?
```

Interogari complexe:

1) Pentru un anumit film determinati numarul de actori care joaca in el

```
SELECT f.film_titlu, f.film_id, COUNT(*) AS NrActori
FROM Film_actors fa INNER JOIN Film f
      ON f.film_id = fa.film_id
      INNER JOIN Actor a
      ON a.actor_id = fa.actor_id
WHERE f.film_id IN (SELECT fa.film_id FROM Film_actors fa WHERE a.actor_id
      IN (SELECT a.actor_id FROM Actor a WHERE EXISTS (SELECT
a.actor_id FROM Actor a WHERE a.actor_id = fa.actor_id))) AND f.film_titlu = ?
GROUP BY f.film_titlu, f.film_id
```

2) Gasiti filmele care au o medie a notelor acordate de utilizatori mai mare decat media tuturor notelor acordate pentru toate filmele

```
SELECT f.film_titlu, AVG(r.review_nota) AS MedieReview,
(SELECT AVG(r.review_nota) FROM Review r) AS MedieTotal
FROM Film f INNER JOIN Review r
      ON f.film_id = r.film_id
GROUP BY f.film_titlu
HAVING AVG(r.review_nota) > (SELECT AVG(r.review_nota) FROM Review r)
```

3) Gasiti filmele care au fost evaluate de toti in afara de un anumit utilizator pe care il specificati

```
SELECT f.film_titlu
FROM Film f
WHERE f.film_id IN
    (SELECT r.film_id
     FROM Review r
     WHERE r.user_id NOT IN
         (SELECT u.user_id
          FROM [User] u
          WHERE u.username = ?))
```

4) Scrieti numele filmelor in care unul sau mai multi actori au jucat in cel putin un anumit numar de filme

```
SELECT f.film_titlu
FROM Film f
WHERE f.film_id IN
    (SELECT fa.film_id
     FROM Film_actors fa
     WHERE fa.actor_id IN
         (SELECT a.actor_id
          FROM Actor a
          WHERE a.actor_id IN
              (SELECT fa1.actor_id
               FROM Film_actors fa1
               GROUP BY fa1.actor_id
               HAVING COUNT(fa1.actor_id)>?)))
```