



Базы от данни

Релационна алгебра

Алгебра на релационните операции

- ▶ Релационната алгебра или алгебра върху релации са операции, които се прилагат върху релации.
- ▶ Релационната алгебра е била въведена от Едгар Код, като допълнение на предложението по-рано от него релационен модел.
- ▶ Релационната алгебра се състои от оператори и атомарни операнди. Операндите са релации, а операторите - операциите, които могат да бъдат извършвани върху операндите.
- ▶ Резултатът от прилагане на оператор върху релации е също релация.

Алгебра на релационните операции

- ▶ Релационната алгебра се състои от оператори и атомарни операнди.
- ▶ Операндите могат да бъдат:
 - ▶ Променливи, които стоят зад релациите
 - ▶ Константи, които са крайни релации
- ▶ В класическата релационна алгебра, всички операнди и резултати от изрази са множества. Това е така, защото самите релации се разглеждат като множества (без повтарящи се кортежи)
- ▶ В съвременните реализации на СУБД, операциите върху релации, се разглеждат като операции върху мултимножества (т.е. възможни са повтарящи се кортежи)

Алгебра на релационните операции

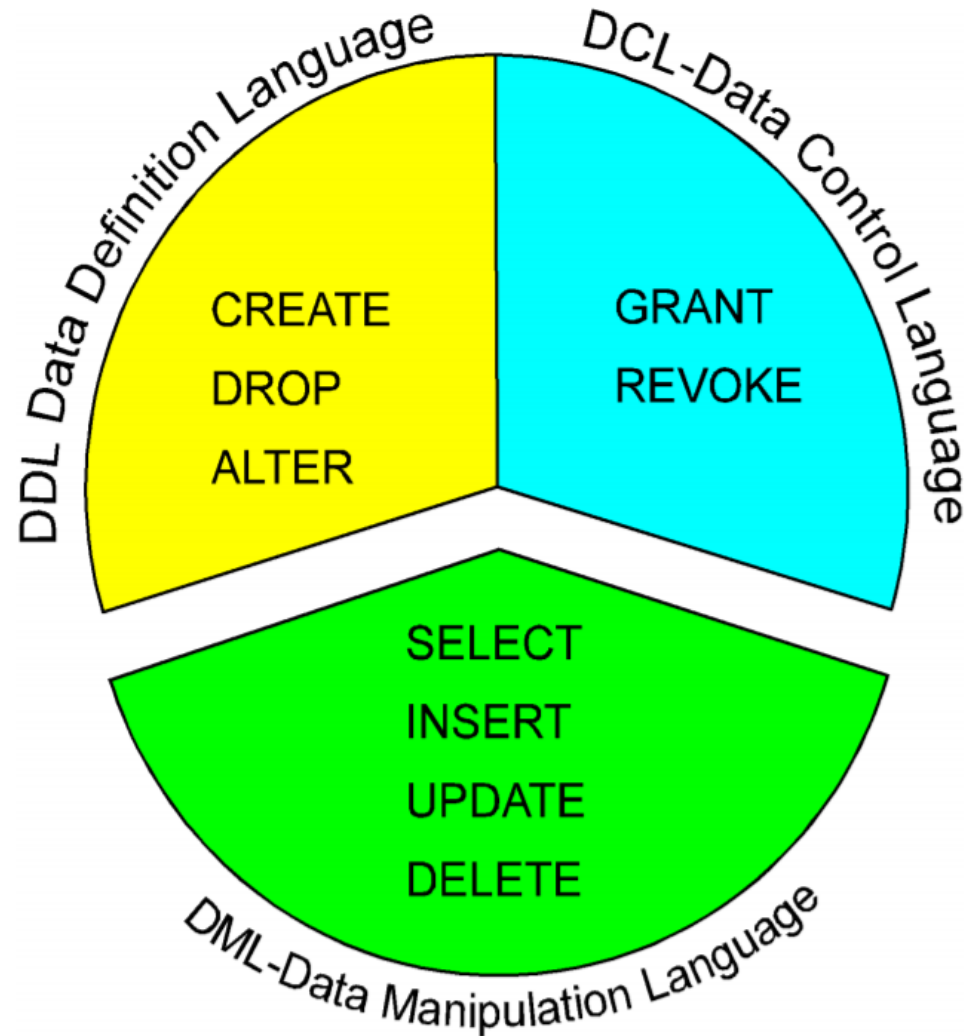
- ▶ Операциите на традиционната релационна алгебра могат да бъдат разделение в 4 групи:
 - ▶ Обикновените операции върху множества: обединение, сечение и разлика приложени върху множества
 - ▶ Операции, които махат част от релацията: селекция, проекция
 - ▶ Операции, които съединяват кортежите на две релации по всички възможни начини: декартово произведение, съединения
 - ▶ Операция преименуване, която не засяга кортежите на релацията, но променя релационната схема, например имената на атрибутите

SQL - Structured Query Language

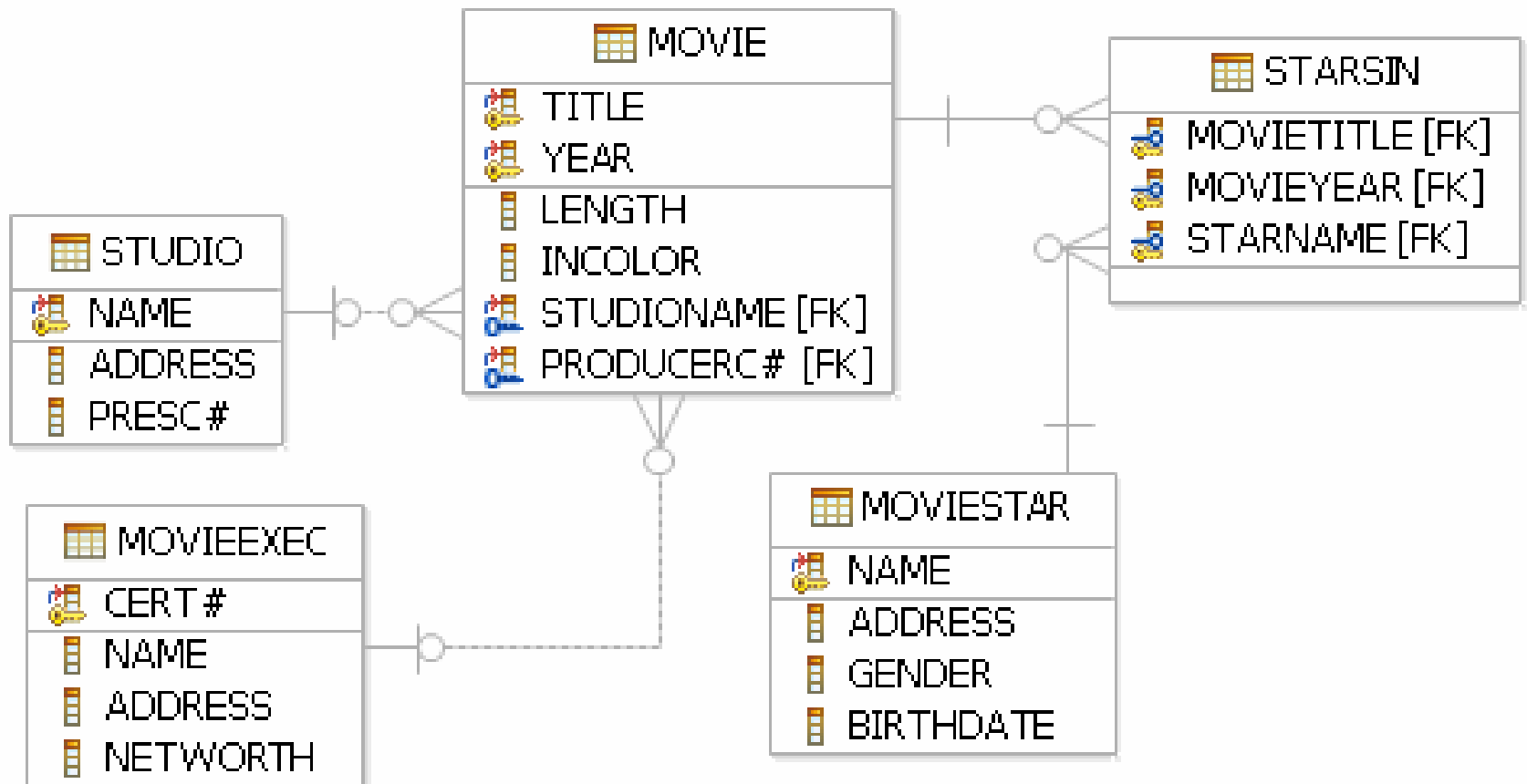
- ▶ SQL е език за заявки към релационни БД. Посредством езика могат да се създават, изтриват и променят релационни схеми в СУБД.
- ▶ SQL – реализира операциите от релационната алгебра
- ▶ SQL се базира на заявки, групирани в следните под-езици:
 - ▶ **Data Control Language (DCL)** — контролира достъпа до обектите в базата от данни.
 - ▶ **Data Definition Language (DDL)** — използва се за създаване, модифициране или изтриване на обект в базата от данни.
 - ▶ **Data Manipulation Language (DML)** — използва се за извличане, обновяване, вмъкване или изтриване на данни.



SQL



Movies Schema



StarsIn

	MOVIE TITLE	MOVIE YEAR	STAR NAME
1	Star Wars	1977	Alec Baldwin
2	Star Wars	1977	Harrison Ford
3	Star Wars	1977	Kim Basinger
4	Star Wars: Empire Strikes Back	1980	Harrison Ford
5	Terms of Endearment	1983	Jack Nicholson
6	Terms of Endearment	1983	Jane Fonda
7	The Usual Suspects	1995	Jack Nicholson
8	The Usual Suspects	1995	Sandra Bullock

MovieStar

	NAME	ADDRESS	GENDER	BIRTHDATE
1	Jane Fonda	Turner Av.	F	1977-07-07
2	Alec Baldwin	Baldwin Av.	M	1977-07-06
3	Kim Basinger	Baldwin Av.	F	1979-07-05
4	Harrison Ford	Prefect Rd.	M	1955-05-05
5	Debra Winger ...	A way	F	1978-06-05
6	Jack Nicholson ...	X path	M	1949-05-05
7	Sandra Bullock ...	X path	F	1964-07-26

Studio

	NAME	ADDRESS	PRESC#
1	Disney ...	500 S. Buena Vista Street	1
2	USA Entertainm. ...		2
3	Fox ...	10201 Pico Boulevard	3
4	Paramount ...	5555 Melrose Ave	4
5	MGM ...	MGM Boulevard	5

Movie

	TITLE	YEAR	LENGTH	INCOLOR	STUDIO	PRODUCER#
1	Pretty Woman	1990	119	Y	Disney ...	199
2	The Man Who Wasn't There	2001	116	N	USA Entertainm. ...	555
3	Logan's run	1976	NULL	Y	Fox ...	333
4	Star Wars	1977	124	Y	Fox ...	555
5	Empire Strikes Back	1980	111	Y	Fox ...	555
6	Star Trek	1979	132	Y	Paramount ...	222
7	Star Trek: Nemesis	2002	116	Y	Paramount ...	123
8	Terms of Endearment	1983	132	Y	MGM ...	123
9	The Usual Suspects	1995	106	Y	MGM ...	199
10	Gone With the Wind	1938	238	Y	MGM ...	123

Проекция

- ▶ Операторът проекция се прилага върху релация и като резултат се получава нова релация, която съдържа някои от колоните на първоначалната релация
- ▶ Проекция се бележи с $\Pi_L(R)$
- ▶ Нека релацията $R(A_1, A_2, A_3, \dots, A_n)$
- ▶ Проекцията на релацията R по атрибута A_1 се бележи с $\Pi_{A_1}(R)$ и е нова релация състояща се само от атрибута A_1 и стойностите на всички кортежи от релацията R в атрибута A_1

Проекция - Пример

R			
A1	A2	A3	A4
A	B	C	D
C	D	E	F

$\pi_{A1, A3}(R)$	
A1	A3
A	C
C	E

Проекция – Пример SQL (SELECT)

```
SELECT title, year  
FROM MOVIE;
```

	TITLE	YEAR
1	Empire Strikes Back	1980
2	Gone With the Wind	1938
3	Logan's run	1976
4	Pretty Woman	1990
5	Star Trek	1979
6	Star Trek: Nemesis	2002
7	Star Wars	1977
8	Terms of Endearment	1983
9	The Man Who Wasn't There	2001
10	The Usual Suspects	1995

Израз от PA: $\prod_{title, year} (Movie)$

Множествени операции върху релации

- ▶ Трите най-чести операции върху множества са обединение, сечение и разлика.
- ▶ Нека R и S са множества:
 - ▶ $R \cup S$ - обединение на две множества е множество, чиито елементи са или от R или от S или и от двете. Резултатното множество съдържа елементите само по веднъж, дори и да има един и същи елемент в R и в S , той се появява само веднъж в резултата.
 - ▶ $R \cap S$ - сечени на две множества е множество, чиито елементи са едновременно и от R и от S (общите елементи за двете множества)
 - ▶ R / S ($R - S$) – разлика на две множества е множество, чиито елементи са от R но не се срещата в S .
 - ▶ R / S и S / R са различни множества.

Множествени операции върху релации

- ▶ Когато прилагаме множествените операции, трябва да имам предвид следните условия, които трябва да а в сила за R и S а да могат да бъдат извършени съответните операции: обединение, сечение и разлика
- ▶ R и S трябва да имат схеми с еднакви множества от атрибути и типовете на всички атрибути трябва да са едни и същи за R и за S
- ▶ Преди да бъде приложена, която и да е от горните операции, колоните на R и S трябва да бъдат подредени така, че реда на атрибутите да бъде еднакъв и за двете релации

Множествени операции - Обединение

R	
A1	A2
A	B
C	D
E	F

S	
A1	A2
G	H
I	J

R ∪ S	
A1	A2
A	B
C	D
E	F
G	H
I	J

Множествени операции – Обединение (SQL)

```
SELECT NAME FROM MOVIESTAR  
UNION  
SELECT STARNAME FROM STARSIN
```

	NAME
1	Alec Baldwin
2	Debra Winger
3	Harrison Ford
4	Jack Nicholson
5	Jane Fonda
6	Kim Basinger
7	Sandra Bullock

Израз от PA: $\Pi_{name}(\text{MovieStar}) \cup \Pi_{starname}(\text{StarsIn})$

Множествени операции - Сечение

R	
A1	A2
A	B
C	D
E	F

S	
A1	A2
A	B
C	D
G	H
I	J

$R \cap S$	
A1	A2
A	B
C	D

Множественни операции – Сечение (SQL)

```
SELECT NAME FROM MOVIESTAR
```

```
INTERSECT
```

```
SELECT STARNAME FROM STARSIN
```

	NAME
1	Alec Baldwin
2	Harrison Ford
3	Jack Nicholson
4	Jane Fonda
5	Kim Basinger
6	Sandra Bullock

$$\Pi_{name}(\text{MovieStar}) \cap \Pi_{starname}(\text{StarsIn})$$

Множествени операции - Разлика

R	
A1	A2
A	B
C	D
E	F

S	
A1	A2
A	B
I	J

R - S	
A1	A2
C	D
E	F

Множествени операции – Разлика (SQL)

```
SELECT NAME FROM MOVIESTAR  
EXCEPT  
SELECT STARNAME FROM STARSIN
```

	NAME
1	Debra Winger

$$\Pi_{name}(\text{MovieStar}) / \Pi_{starname}(\text{StarsIn})$$

Селекция

- ▶ Операторът селекция се прилага върху релация и като резултат се получава нова релация, която съдържа някои от кортежите на първоначалната релация, според условието (критериите) записани в оператора
- ▶ Селекция се бележи с $\sigma_C(R)$, където R е релацията върху която се прилага селекцията, а C условието на което трябва да отговарят кортежите в R за да попаднат в резултатната релация
- ▶ Схемата на резултатната релация е същата като схемата на R , като реда на атрибутите в новата релация е същият като реда на атрибутите в R

Селекция - Пример

- Условието C се прилага към всеки кортеж от релацията R . Ако кортежът отговаря на условието C то той попада в резултатната релация.

► $\sigma_{A1=A}(R)$

$\sigma_{A1=A}(R)$	
A1	A2
A	B
A	D
A	J

R	
A1	A2
A	B
A	D
E	F
G	H
A	J

Селекция – Пример SQL (WHERE)

```
SELECT *  
FROM movie  
WHERE length > 120;
```

	TITLE	YEAR	LENGTH	INCOLOR	STUDIONAME	PRODUCERC#
1	Star Wars	1977	124	Y	Fox ...	555
2	Star Trek	1979	132	Y	Paramount ...	222
3	Terms of ...	1983	132	Y	MGM ...	123
4	Gone Wit...	1938	238	Y	MGM ...	123

Израз от PA: $\sigma_{length > 120}(\text{Movie})$

Декартово произведение

- ▶ Декартово произведение на две релации R и S е нова релация, която се състои от всички атрибути на релацията R и релацията S , като кортежите на новата релация са множество от двойки на които първият елемент от двойката е кои и да е кортеж от релацията R , а вторият кортеж на двойката е кои и да е кортеж от релацията S
- ▶ Декартово произведение на релациите R и S се бележи с $R \times S$
- ▶ Релационната схема за резултатната релация е обединение на релационните схеми на R и S
- ▶ Ако R и S имат атрибут с едно и също име, трябва еднозначно да обозначим кой атрибут от коя релация е

Декартово произведение

R	
A1	A2
A	B
C	D

S	
B1	B2
J	K
L	M

R x S			
A1	A2	B1	B2
A	B	J	K
A	B	L	M
C	D	J	K
C	D	L	M

Декартово произведение - SQL

```
SELECT *  
FROM MovieStar CROSS JOIN StarsIN;
```

```
SELECT *  
FROM MovieStar , StarsIN;
```

	NAME	ADDRESS	GENDER	BIRTHDATE	MOVIETITLE	MOVIEYEAR	STARNAME
1	Jane Fon...	Turner Av.	F	1977-07-07	Star Wars	1977	Alec Baldwin
2	Alec Bal...	Baldwin Av.	M	1977-07-06	Star Wars	1977	Alec Baldwin
3	Kim Basi...	Baldwin Av.	F	1979-07-05	Star Wars	1977	Alec Baldwin
4	Harrison...	Prefect Rd.	M	1955-05-05	Star Wars	1977	Alec Baldwin
5	Debra W...	A way	F	1978-06-05	Star Wars	1977	Alec Baldwin
6	Jack Nic...	X path	M	1949-05-05	Star Wars	1977	Alec Baldwin
7	Sandra B...	X path	F	1964-07-26	Star Wars	1977	Alec Baldwin
8	Jane Fon...	Turner Av.	F	1977-07-07	Star Wars	1977	Harrison Ford
9	Alec Bal...	Baldwin Av.	M	1977-07-06	Star Wars	1977	Harrison Ford
10	Kim Basi...	Baldwin Av.	F	1979-07-05	Star Wars	1977	Harrison Ford
11	Harrison...	Prefect Rd.	M	1955-05-05	Star Wars	1977	Harrison Ford
12	Debra W...	A way	F	1978-06-05	Star Wars	1977	Harrison Ford
13	Jack Nic...	X path	M	1949-05-05	Star Wars	1977	Harrison Ford
14	Sandra B...	X path	F	1964-07-26	Star Wars	1977	Harrison Ford
15	Jane Fon...	Turner Av.	F	1977-07-07	Star Wars	1977	Kim Basinger
16	Alec Bal...	Baldwin Av.	M	1977-07-06	Star Wars	1977	Kim Basinger
17	Kim Basi...	Baldwin Av.	F	1979-07-05	Star Wars	1977	Kim Basinger
18	Harrison...	Prefect Rd.	M	1955-05-05	Star Wars	1977	Kim Basinger
19	Debra W...	A way	F	1978-06-05	Star Wars	1977	Kim Basinger
20	Jack Nic...	X path	M	1949-05-05	Star Wars	1977	Kim Basinger
21	Sandra B...	X path	F	1964-07-26	Star Wars	1977	Kim Basinger

$\Pi_{name,...}(\text{MovieStar} \times \text{StarsIn})$

Естествено съединение

- ▶ Естествено съединение на две релации R и S е нова релация, която се състои от всички атрибути на релацията R и релацията S , като кортежите на новата релация са множество от двойки на който първият елемент от двойката е кортеж от релацията R , а вторият кортеж на двойката кортеж от релацията S , такива че кортежите да съвпадат по общите атрибути на релациите R и S
- ▶ Естествено съединение на релациите R и S се бележи с $R \bowtie S$
- ▶ Нека A_1, A_2, \dots, A_n са общите атрибути за двете релации R и S , тогава $R \bowtie S$ ще се състои от тези кортежи от релацията R и релацията S , които успяват да се съединят успешно по общите атрибути на двете релации (т.е. имат едни и същи стойности по общите атрибути). Резултатният кортеж се нарича съединен кортеж (joined tuple)



Естествено съединение – Пример

R		
A	B	C
1	2	3
6	7	8
9	7	8

S		
B	C	D
2	3	4
2	3	5
7	8	10

R ⋈ S			
A	B	C	D
1	2	3	4
1	2	3	5
6	7	8	10
9	7	8	10

Естествено съединение – Пример SQL

```
SELECT BIRTHDATE, GENDER  
FROM MOVIESTAR NATURAL JOIN MOVIEEXEC;
```

$$\Pi_{birthdate, gender}(\text{MovieStar} \bowtie \text{MovieExec})$$

⋈-съединение

- ▶ Ако при естественото съединени, съединяваме кортежите според общите атрибути на двете релации при θ -съединението това става според зададен критерии
- ▶ Нека R и S са две релации
- ▶ θ -съединение се бележи с $R \bowtie_C S$, където C е критерии по които ще бъдат съединени кортежите на двете релации
- ▶ θ -съединението на двете релации е също релация, която се получава:
 - ▶ Като вземем декартовото произведение на R и S
 - ▶ И след това от него вземаме само тези кортежи които отговарят на условието C
- ▶ Резултатната схема е обединение на схемите на двете релации R и S . И отново ако е необходимо преименуваме някои от атрибутите, за да е еднозначно на коя релация принадлежат

⊙-съединение – Пример

R		
A	B	C
1	2	3
6	7	8
9	7	8

S		
B	C	D
2	3	4
2	3	5
7	8	10

$R \bowtie_{A < D} S$					
A	R.B	R.C	S.B	S.C	D
1	2	3	2	3	4
1	2	3	2	3	5
1	2	3	7	8	10
6	7	8	7	8	10
9	7	8	7	8	10

⊕-съединение – Пример SQL

```
SELECT NAME, BIRTHDATE, MOVIE TITLE
FROM MovieStar JOIN StarsIN
ON MOVIEYEAR>1990;
```

	NAME	BIRTHDATE	MOVIE TITLE
1	Jane Fonda	1977-07-07	The Usual Suspects
2	Alec Baldwin	1977-07-06	The Usual Suspects
3	Kim Basinger	1979-07-05	The Usual Suspects
4	Harrison Ford	1955-05-05	The Usual Suspects
5	Debra Winger	1978-06-05	The Usual Suspects
6	Jack Nicholson	1949-05-05	The Usual Suspects
7	Sandra Bullock	1964-07-26	The Usual Suspects
8	Jane Fonda	1977-07-07	The Usual Suspects
9	Alec Baldwin	1977-07-06	The Usual Suspects
10	Kim Basinger	1979-07-05	The Usual Suspects
11	Harrison Ford	1955-05-05	The Usual Suspects
12	Debra Winger	1978-06-05	The Usual Suspects
13	Jack Nicholson	1949-05-05	The Usual Suspects
14	Sandra Bullock	1964-07-26	The Usual Suspects

$\Pi_{name, birthdate, movietitle} (MovieStar \bowtie_{movieyear>1990} StarsIn)$

Комбиниране на операции

- ▶ Релационната алгебра, както и всяка алгебра позволява да пишем изрази чрез прилагане на операнди върху релации или върху релации, които са резултата от прилагане на един или повече оператори към релации. Това се нарича комбиниране на операции.
- ▶ Например ако разгледаме релацията
`Movies(title, year, length, studioName)`
- ▶ И искаме да изведем заглавията и годините на филми направени от студио Fox, които са с дължина на филма > 100 мин.

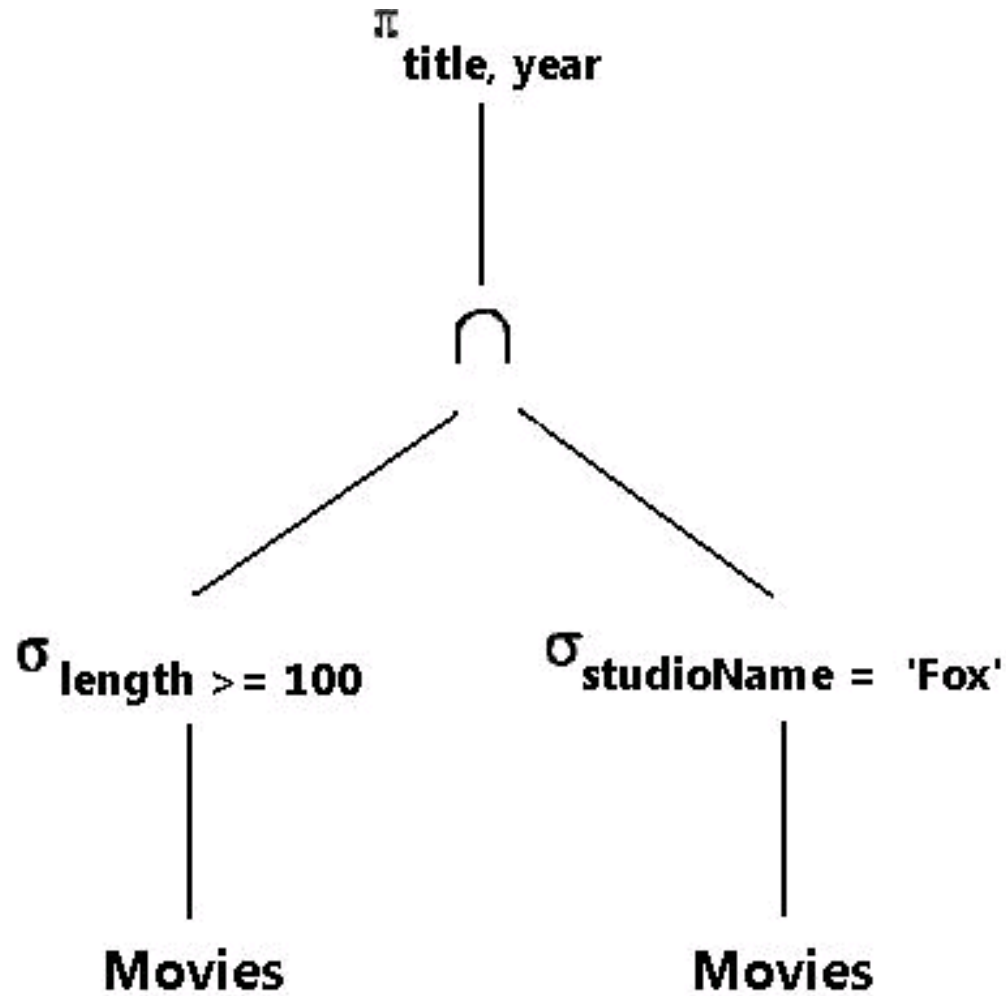
Комбиниране на операции

► Заглавията и годините на филми направени от студио Fox, които са с дължина на филма > 100 мин.

1. Избираме тези кортежи от Movies, за които $\text{length} > 100$
2. Избираме тези кортежи от Movies, за които $\text{studioName} = \text{'Fox'}$
3. Намираме общите кортежи от 1) и 2)
4. Проектираме релацията от 3) по title и year

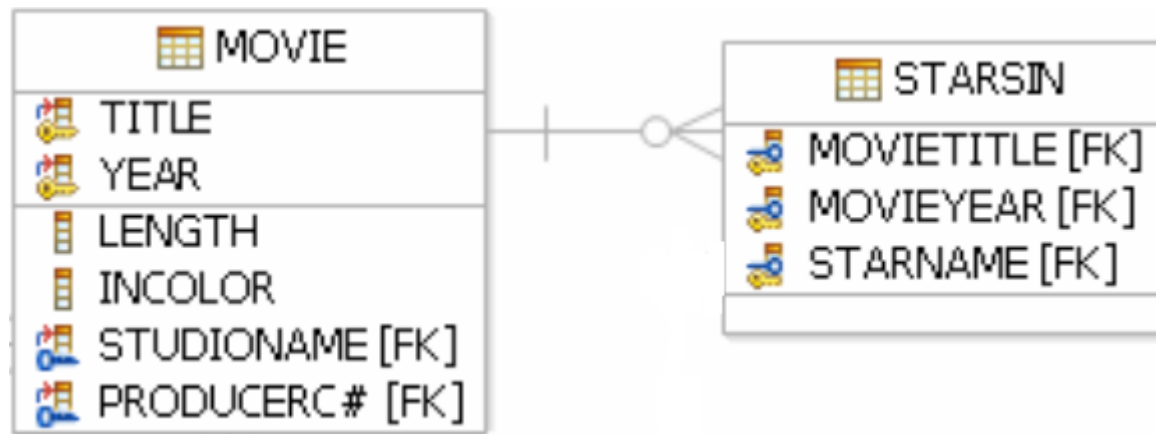
1. $\sigma_{\text{length} > 100}(\text{Movies})$
2. $\sigma_{\text{studioName} = \text{'Fox'}}(\text{Movies})$
3. $\sigma_{\text{length} > 100}(\text{Movies}) \cap \sigma_{\text{studioName} = \text{'Fox'}}(\text{Movies})$ или $\sigma_{\text{length} > 100}(\text{Movies}) \text{ and } \sigma_{\text{studioName} = \text{'Fox'}}(\text{Movies})$
4. $\Pi_{\text{title, year}}$ (Релацията получена в 3.)

Израз от релационната алгебра - дърво



Комбиниране на операции - Задача

- ▶ Да се намерят имената на актьорите, участвали в тези филми, които са с дължина на филма > 100 мин.
- ▶ Напишете израз от релационната алгебра, който да отразява горната заявка
- ▶ Начертайте дървото отговарящо на израза



Преименуване

- ▶ Операторът за преименуване се използва за преименуване на атрибути на релации или имена на релации.
- ▶ Преименуване се бележи с $\rho_{S(A_1, \dots, A_n)}(R)$, където R е релацията върху, която се прилага преименуването, а S е новата релация, получена в резултат на преименуването на R , като атрибутите на R се преименуват на A_1, A_2, \dots, A_n
- ▶ Ако искаме да преименуваме само името на релацията R записваме $\rho_S(R)$

Преименоване – Пример SQL

```
SELECT NAME AS STARNAME, BIRTHDATE, MOVIEITITLE
FROM MovieStar AS M JOIN StarsIN AS S
ON M.NAME=S.STARNAME
```

	STARNAME	BIRTHDATE	MOVIEITITLE
1	Alec Baldwin	1977-07-06	Star Wars
2	Harrison Ford	1955-05-05	Star Wars
3	Kim Basinger	1979-07-05	Star Wars
4	Harrison Ford	1955-05-05	Star Wars: Empire Strikes Back
5	Jack Nicholson	1949-05-05	Terms of Endearment
6	Jane Fonda	1977-07-07	Terms of Endearment
7	Jack Nicholson	1949-05-05	The Usual Suspects
8	Sandra Bullock	1964-07-26	The Usual Suspects

$$\Pi_{name \rightarrow starname, birthdate, movietitle}(\rho_M(\text{MovieStar}) \bowtie_{name=starname} \rho_S(\text{StarsIn}))$$

Зависими и независими операции

- ▶ Някои от операциите, които описахме могат да бъдат изразени чрез другите операции.

- ▶ Например сечението на две релации:

$$R \cap S = R - (R - S)$$

- ▶ Но има и такива, които са основни и не могат да бъдат изразени чрез другите операции (независими). Те са

- ▶ Обединение
- ▶ Разлика
- ▶ Декартово произведение
- ▶ Селекция
- ▶ Проекция
- ▶ Преименуване

Зависими операции

- ▶ Сечение

$$R \cap S = R - (R - S)$$

- ▶ Θ -съединение

$$R \bowtie_C S = \sigma_C (R \times S)$$

- ▶ Естествено съединение

$$R \bowtie S = \Pi_L \sigma_C (R \times S)$$

- ▶ където условието C е $R.A_1=S.A_1$ and $R.A_n=S.A_n$ и A_1, A_2, \dots, A_n са общите атрибути на двете релации, а L е списъкът от атрибути в R последвани от тези в S , но без атрибутите от R

Линейни нотации на алгебрични изрази

- ▶ Освен чрез дърво изразите от релационната алгебра могат да бъдат изразени и като именуваме резултатните релации и операциите върху тях.
- ▶ Нотацията е следната
 - ▶ Записваме името на релацията последвана от списък от атрибути заградени в кръгли скоби.
 - ▶ Името *Answer* се използва като име на релацията отговаряща на резултата от последната стъпка от релационния израз
 - ▶ След което записваме свързващият израз $:=$
 - ▶ В дясно стои кой и да е релационен алгебричен израз

Линейни нотации на алгебрични изрази

Movies(title, year, length, studioName)

- ▶ Заглавията и годините на филми направени от студио Fox, които са с дължина на филма > 100 мин.

1. Избираме тези кортежи от Movies, за които $\text{length} > 100$
2. Избираме тези кортежи от Movies, за които $\text{studioName} = \text{'Fox'}$
3. Намираме общите кортежи от 1) и 2)
4. Проектираме релацията от 3) по title и year

1. $R(t, y, l, s) := \sigma_{\text{length} > 100}(\text{Movies})$
2. $S(t, y, l, s) := \sigma_{\text{studioName} = \text{'Fox'}}(\text{Movies})$
3. $T(t, y, l, s) := R \cap S$
4. $\text{Answer}(\text{title}, \text{year}) := \Pi_{t, y}(T)$ или
5. $\text{Answer}(\text{title}, \text{year}) := \Pi_{t, y}(R \cap S)$

Операции върху мултимножества

- ▶ Мултимножества (bags) са множества, които позволяват повтаряне на кортежи
- ▶ Съвременните СУБД, рядко се базират на чисти множества и по тези причини позволяват операции над мултимножества, т.е. на множества, в които може да има повтарящи се кортежи.
- ▶ Защо е полезно да се поддържат мултимножествата?
- ▶ Например представете си, че правите проекция по атрибута `starname` от релацията `Starsin(title, year, starname)`, като всеки един актьор в релацията е участвал в повече от 2 филма.
- ▶ Тогава в резултатната релация ще имаме повтарящи се имена, за да получи множество СУБД, трябва да сортира и да премахне дублиращите се редове. Това води до забавяне.

Обединение, сечение и разлика

- ▶ Ако R е мултимножество, в което кортежа t се повтаря n -пъти, а S е мултимножество, в което кортежа t се повтаря m -пъти, тогава:
 - ▶ $R \cup S$ – кортежа t ще се среща $m+n$ пъти
 - ▶ $R \cap S$ – кортежа t ще се среща $\min(m, n)$ пъти
 - ▶ R / S – кортежа t ще се среща $\max(0, n-m)$ пъти
 - ▶ S / R – кортежа t ще се среща $\max(0, m-n)$ пъти
- ▶ Всички условия, които са в сила за обединение, сечение и разлика на множества са в сила и за мултимножества

Обединение, сечение и разлика - Пример

R		S	
A	B	A	B
1	2	1	2
3	4	3	4
1	2	3	4
1	2	5	6

$R \cup S$ – (1,2) се среща 4 пъти, (3,4) ще се среща 3 пъти, (5,6) ще се среща веднъж

$R \cap S$ - (1,2) се среща веднъж, (3,4) ще се среща също веднъж

$R - S$ - (1,2) се среща 2 пъти

$S - R$ - (3,4) и (5,6) ще се срещат по веднъж

Обединение мультимножества (SQL)

```
SELECT NAME FROM MOVIESTAR  
UNION ALL  
SELECT STARNAME FROM STARSIN
```

	NAME
1	Alec Baldwin
2	Debra Winger
3	Harrison Ford
4	Jack Nicholson
5	Jane Fonda
6	Kim Basinger
7	Sandra Bullock

Сечение мултимножества (SQL)

```
SELECT NAME FROM MOVIESTAR  
INTERSECT ALL  
SELECT STARNAME FROM STARSIN
```

	NAME
1	Alec Baldwin
2	Harrison Ford
3	Jack Nicholson
4	Jane Fonda
5	Kim Basinger
6	Sandra Bullock

Разлика мултимножества (SQL)

```
SELECT NAME FROM MOVIESTAR
```

```
EXCEPT ALL
```

```
SELECT STARNAME FROM STARSIN
```

	NAME
1	Debra Winger

Проекция

- ▶ Операторът проекция се прилага както при множества с тази разлика че резултатната релация, може да съдържа повтарящи се кортежи
- ▶ Например $\Pi_{A,B}(R)$ – ще се състои от 3 кортежа (1,2) и един кортеж (3,4)

R		
A	B	C
1	2	5
3	4	6
1	2	7
1	2	8

Селекция

- ▶ Операторът селекция се прилага както при множества с тази разлика , че резултатната релация, може да съдържа повтарящи се кортежи
- ▶ Например $\sigma_{c \geq 6}(R)$ – ще се състои от кортежите (1,2, 7), (1,2, 8), (3, 4, 6)

R		
A	B	C
1	2	5
3	4	6
1	2	7
1	2	8

Декартово произведение

- ▶ Всеки кортеж от едната релация се съединява с всеки кортеж от другата релация, независимо от това дали се дублират кортежите или не.
- ▶ Ако R е мултимножество, в което кортежа r се повтаря n -пъти, а S е мултимножество, в което кортежа s се повтаря m -пъти, тогава в $R \times S$ кортежа rs ще се среща $m \cdot n$ пъти

Декартово произведение

R	
A	B
1	2
1	2

S	
B	C
2	3
4	5
4	5

R x S			
A	R.B	S.B	C
1	2	2	3
1	2	2	3
1	2	4	5
1	2	4	5
1	2	4	5
1	2	4	5

Съединения

- ▶ Както при множества, с разликата че резултатната релация позволява повтарящи се кортежи.
- ▶ Пример:

R	
A	B
1	2
1	2

S	
B	C
2	3
4	5
4	5

R ⋈ S R.B < S.B			
A	R.B	S.B	C
1	2	4	5
1	2	4	5
1	2	4	5
1	2	4	5

Закони на релационната алгебра

- ▶ $(R \cup S) - T = (R - T) \cup (S - T)$ дистрибутивен закон, в сила за множества, но не и за мултимножества
- ▶ $R \cup S = S \cup R$ важи и за двете и за множества и за мултимножества

Други операции в релационната алгебра

- ▶ Оператор за премахване на дубликати (δ) – преобразува мултимножество в множество, като премахва всички повтарящи се кортежи и оставя само по едно копие от кортеж
- ▶ Групиране на кортежи (γ) – използва се за разделяне на групи на кортежите от релацията според техните стойности по един или повече атрибута.
- ▶ Агрегатни оператори – използват се от оператора за групиране, приложени върху кортежи на релацията връщат една стойност. Например $\min()$, $\max()$, $\text{avg}()$
- ▶ Оператор за сортиране (τ) – връща релацията в списък от кортежи, сортирани по един или няколко атрибута.
- ▶ Разширена проекция – дава допълнителна функционалност към проекцията, като например аритметични изрази върху проектираните колони
- ▶ Външно съединение – вариант на съединение, което ни предпазва от загуба на висящи кортежи.

Премахване на дубликати (δ)

S	
B	C
2	3
4	5
4	5

$\delta(S)$	
B	C
2	3
4	5

Премахване на дубликати (δ) - SQL

```
SELECT DISTINCT length/60 as HOURS  
FROM MOVIE;
```

	HOURS
1	1
2	2
3	3
4	NULL

$\delta \left(\pi_{\text{length}/60 \rightarrow \text{HOURS}} (\text{MOVIE}) \right)$

Агрегатни оператори

- ▶ Използват се за да агрегират стойностите в една колона на релацията
- ▶ Стандартни операции от този тип са:
 - ▶ **SUM** – сума на стойностите в колоната
 - ▶ **AVG** – средна стойност на колоната
 - ▶ **MIN** и **MAX** – минимален и максимален елемент на колоната
 - ▶ **COUNT** – връща броя на стойностите в колоната
 - ▶ Например:
 - ▶ **SUM(B)=10**
 - ▶ **MIN(C)=3**
 - ▶ **COUNT(B)=3**

S	
B	C
2	3
4	5
4	5

Агрегатни оператори - SQL

```
SELECT SUM(LENGTH) AS SUM_LENGTH  
FROM MOVIE  
WHERE STUDIOName='MGM' ;
```

	SUM_LENGTH
1	476

$$\pi_{\text{sum(LENGTH)} \rightarrow \text{sm}} \left(\pi_{\text{length}} \left(\sigma_{\text{StudioName} = \text{'MGM'}} (\text{MOVIE}) \right) \right)$$

Групиране

- ▶ Операторът за групиране се използва за да се направи група по някоя от колоните на релацията, при което кортежите се групират според стойностите им в съответната колона
- ▶ За така направената група можем да приложим агрегатните функции.
- ▶ Нека разгледаме релацията

Movies(title, year, length, studioName)

- ▶ Искаме да намерим общият брой на филми, за всяко филмово студио.
- ▶ Започваме от горната релация, групираме по студио и прилагаме агрегатната операция COUNT

Групиране (γ)

- ▶ Операторът за групиране се бележи с $\gamma_L(R)$, където L е списък състоящ се от:
 - ▶ атрибута, по които ще се групира
 - ▶ агрегатната операция за групата, стрелка и име на резултатната колона
- ▶ Релацията върната от $\gamma_L(R)$ се конструира по следният начин:
 - ▶ Разделяме кортежите на R в групи. Всяка група се състои от всички кортежи имащи една и съща стойност за атрибута по които се групира. Ако няма групиращ атрибут, цялата релация е една група
 - ▶ За всяка група връщаме един кортеж, състоящ се от атрибута по който се групира и агрегираната стойност от агрегатната операция

Групиране (Υ) - Пример

StarsIn(title, year, starname)

- ▶ Да се изведе за всеки актьор, който е участвал най-малко в 3 филми и най-ранната година на участие
 - ▶ Групираме по име на актьор
 - ▶ За всяка група намираме $\min(\text{year})$
 - ▶ За всяка група намираме $\text{count}(\text{title})$
- 1. $R(s, y, c) := \Gamma_{\text{starname}, \min(\text{year}) \rightarrow \min\text{Year}, \text{count}(\text{title}) \rightarrow c\text{Title}} (\text{StarsIn})$
- 2. $\Pi_{s, y} (\sigma_{c \geq 3}(R))$

Групиране - SQL

```
SELECT STUDIOName, COUNT(*) AS CNT
FROM MOVIE
GROUP BY STUDIOName;
```

	STUDIOName	CNT
1	Disney	1
2	Fox	3
3	MGM	3
4	Paramount	2
5	USA Entertainm	1

for
studioName, (MOVIE)
count(title) → CNT

Оператор за сортиране

- ▶ Операторът за сортиране $T_L(R)$, където L е списък от атрибути по които се сортира
- ▶ Например ако $L=A1, A2,.. A_n$, релацията R се сортира първо по $A1$, ако са равни по $A2$, ако са равни по $A3$ и т.н.
- ▶ Забележете че, резултатът от операцията сортиране е списък, от подредени кортежи според атрибута по който се сортира.
- ▶ По тази причина операторът има смисъл да се прилага като последна стъпка от релационния израз.

Сортиране - SQL

```
SELECT STUDIOName, TITLE
FROM MOVIE
ORDER BY STUDIOName;
```

	STUDIOName	TITLE
1	Disney	Pretty Woman
2	Fox	Logan's run
3	Fox	Star Wars
4	Fox	Star Wars: Empire Strikes Back
5	MGM	Terms of Endearment
6	MGM	The Usual Suspects
7	MGM	Gone With the Wind
8	Paramount	Star Trek
9	Paramount	Star Trek: Nemesis
10	USA Entertainm	The Man Who Wasn't There

π (π (MOVIE))
studioName studioName, title

Разширена проекция

- ▶ Операторът разширена проекция се прилага върху релация и като резултат се получава нова релация, която съдържа някои от колоните на първоначалната релация
- ▶ Разширената проекция се бележи по същият начин както и обикновената проекция с $\Pi_L(R)$ с тази разлика, че списъкът L може да съдържа:
 - ▶ Атрибутите на релацията R
 - ▶ Израз от вида $x \rightarrow y$, където x и y са имена на атрибути от релацията R и означава x се преименува в y
 - ▶ Израз от вида $E \rightarrow z$, където E е аритметичен израз включващ аритметични операции върху атрибутите на релацията R , като събиране, изваждане, конкатенация и други, а z е ново име за резултата от израза E .

Разширена проекция- пример

R	
B	C
2	3
4	5
4	5

► $\pi_{B, B+C \rightarrow X}(R)$

R	
B	X
2	5
4	9
4	9

Разширена проекция

- ▶ Резултатът от проекцията се получава, като разгледаме всички кортежи на релацията R подредени
- ▶ Оценяваме списъка L , като заместваем стойности от кортежите и извършваме съответните аритметични действия
- ▶ След което преименуваме, където е указано
- ▶ Проекцията може да се приложи върху релации съдържащи повтарящи се редове или да доведе до релация съдържаща повтарящи се редове

Разширена проекция - SQL

```
SELECT title, year, length, length/60  
FROM movie;
```

	TITLE	YEAR	LENGTH	4
1	Pretty Woman	1990	119	1
2	The Man Who Wasn't ...	2001	116	1
3	Logan's run	1976	NULL	NULL
4	Star Wars	1977	124	2
5	Empire Strikes Back	1980	111	1
6	Star Trek	1979	132	2
7	Star Trek: Nemesis	2002	116	1
8	Terms of Endearment	1983	132	2
9	The Usual Suspects	1995	106	1
10	Gone With the Wind	1938	238	3

$\Pi_{title, year, length, length/60} (MOVIE)$

Външно съединение

- ▶ При съединението новата релация, която се получава съдържа само тези кортежи, които са успели да се съединят.
- ▶ Ако искаме в резултатната релация да имаме и кортежите които не са успели да се съединят, трябва да използваме външно съединение
- ▶ Външното естествено съединение се бележи с $R \bowtie S$
- ▶ Където R и S са релации
- ▶ Външното естествено съединение се оценява, като първо направим естествено съединение и след това добавим висящите кортежи, т.е. тези които не са успели да се съединят.
- ▶ Добавените кортежи, трябва да се допълват със специален символ (null - \perp), за всеки атрибут който те не притежават

Външно съединение - пример

R		
A	B	C
1	2	3
4	5	6
7	8	9

S		
B	C	D
2	3	10
2	3	11
6	7	12

R ⋈ S			
A	B	C	D
1	2	3	10
1	2	3	11
4	5	6	⊥
7	8	9	⊥
⊥	6	7	12

Ляво външно съединение

- ▶ Има различни варианти на външното естествено съединение:
 - ▶ Ляво естествено външно съединение $R \bowtie_L S$
 - ▶ Дясно естествено външно съединение $R \bowtie_R S$
 - ▶ При лявото естествено външно съединение се взимат само висящите кортежи от ляво (на релацията R), като се записват **NULL** стойности за атрибутите на релацията S , за които няма съвпадение
 - ▶ При дясното естествено външно съединение се взимат само висящите кортежи от дясно (на релацията S), за които се записват **NULL** стойности за атрибутите на релацията R , за които няма съвпадение

Ляво и дясно външно съединение - пример

R		
A	B	C
1	2	3
4	5	6
7	8	9

S		
B	C	D
2	3	10
2	3	11
6	7	12

$R \bowtie_L S$			
A	B	C	D
1	2	3	10
1	2	3	11
4	5	6	\perp
7	8	9	\perp

$R \bowtie_R S$			
A	B	C	D
1	2	3	10
1	2	3	11
\perp	6	7	12

⊕-Външно съединение

- ▶ Всички естествени външни съединение имат и Θ -аналог, където първо се прави Θ -съединение и после кортежите които пропадат при съединението се добавят към крайната релация.
- ▶ Символът за Θ -външно съединение е \bowtie_c

⊕-Външно съединение - пример

R		
A	B	C
1	2	3
4	5	6
7	8	9

S		
B	C	D
2	3	10
2	3	11
6	7	12

R $\bowtie_{a>s.c}$ S					
A	R.B	R.C	S.B	S.C	D
4	5	6	2	3	10
4	5	6	2	3	11
7	8	9	2	3	10
7	8	9	2	3	11
1	2	3	⊥	⊥	⊥
⊥	⊥	⊥	6	7	12

Видове съединения (JOIN)

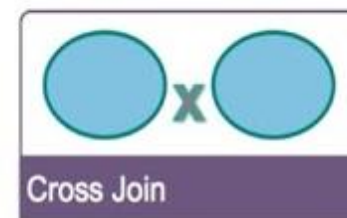
- ▶ Natural JOIN – естествено съединение

- ▶ [Inner] JOIN



- ▶ LEFT [Outer] JOIN

- ▶ RIGHT [Outer] JOIN



- ▶ FULL [Outer] JOIN

- ▶ Cross JOIN



Съединения - SQL

```
SELECT M.NAME  
FROM MOVIESTAR AS M  
LEFT JOIN STARSIN AS S  
ON M.NAME=S.STARNAME  
WHERE S.STARNAME IS NULL;
```

	NAME
1	Debra Winger

$$\pi \left(\sigma \left(\rho_M(\text{Moviestar}) \bowtie_{\substack{M.name=S.starname \\ S.starname \text{ is NULL}}} \rho_S(\text{Starsin}) \right) \right)$$

Ограничение върху релациите и RA

- ▶ Има два начина, по които може да използваме изразите на релационната алгебра за да изразим ограниченията:
 - ▶ $R = \emptyset$, т.е. няма кортежи в резултатната релация R
 - ▶ $R \subseteq S$, т.е. всеки кортеж в резултата на R трябва също така да бъде и в резултата на S . Ограничението може да се запише и като $R - S = \emptyset$, ако всеки кортеж от R го има и в S , тогава $R - S$ със сигурност е празно
 - ▶ Ограничението $R = \emptyset$, може да се запише и като $R \subseteq \emptyset$
 - ▶ Технически \emptyset не е израз на релационната алгебра, но след като има изрази които се оценяват с $R - R = \emptyset$, не би навредило да го използваме.

Ограничения по референтна цялостност

- ▶ Това ограничение, ограничава стойностите от дадена колона на дадена таблица да са подмножество на стойностите, на друга колона в друга таблица. В езика SQL това ограничение се реализира с външен ключ. Като в повечето случаи е вярно следното стойностите от $FK \subseteq PK$
- ▶ Например да разгледаме двете релации:

StarsIn(movietitle, movieyear, starname)

MovieStars(name, birthdate, gender, address)

- ▶ В случая колоната starname от таблицата StarsIn е външен ключ към колоната name на таблицата MovieStars, тази референтна цялостност в термините на релационната алгебра може да бъде изразена по следния начин:

Ограничения по референтна цялостност

$$\Pi_{\text{starname}}(\text{StarsIn}) \subseteq \Pi_{\text{name}}(\text{MovieStar})$$

- ▶ Значението е следното, стойностите на израза в ляво са подмножество на стойностите от израза в дясно. Т.е. всички актьори, които участват в филм задължително ги има в таблицата MovieStar. Обратното не е вярно, възможно е да има актьор в таблицата MovieStar, които да не е участвал в нито един филм, т.е. да го няма в таблицата StarsIn
- ▶ Горното ограничение може да бъде записано и така

$$\Pi_{\text{starname}}(\text{StarsIn}) - \Pi_{\text{name}}(\text{MovieStar}) = \emptyset$$

Други видове ограничения

- ▶ Някои ограничения по домейн могат също да бъдат изразени с релационната алгебра. Например, ако искаме да укажем че единствената легална стойност за атрибута gender е M и F, това може да бъде изразено чрез следния израз:
- ▶ $\sigma_{\text{gender} \in \{M, F\}}(\text{MovieStars}) = \emptyset$