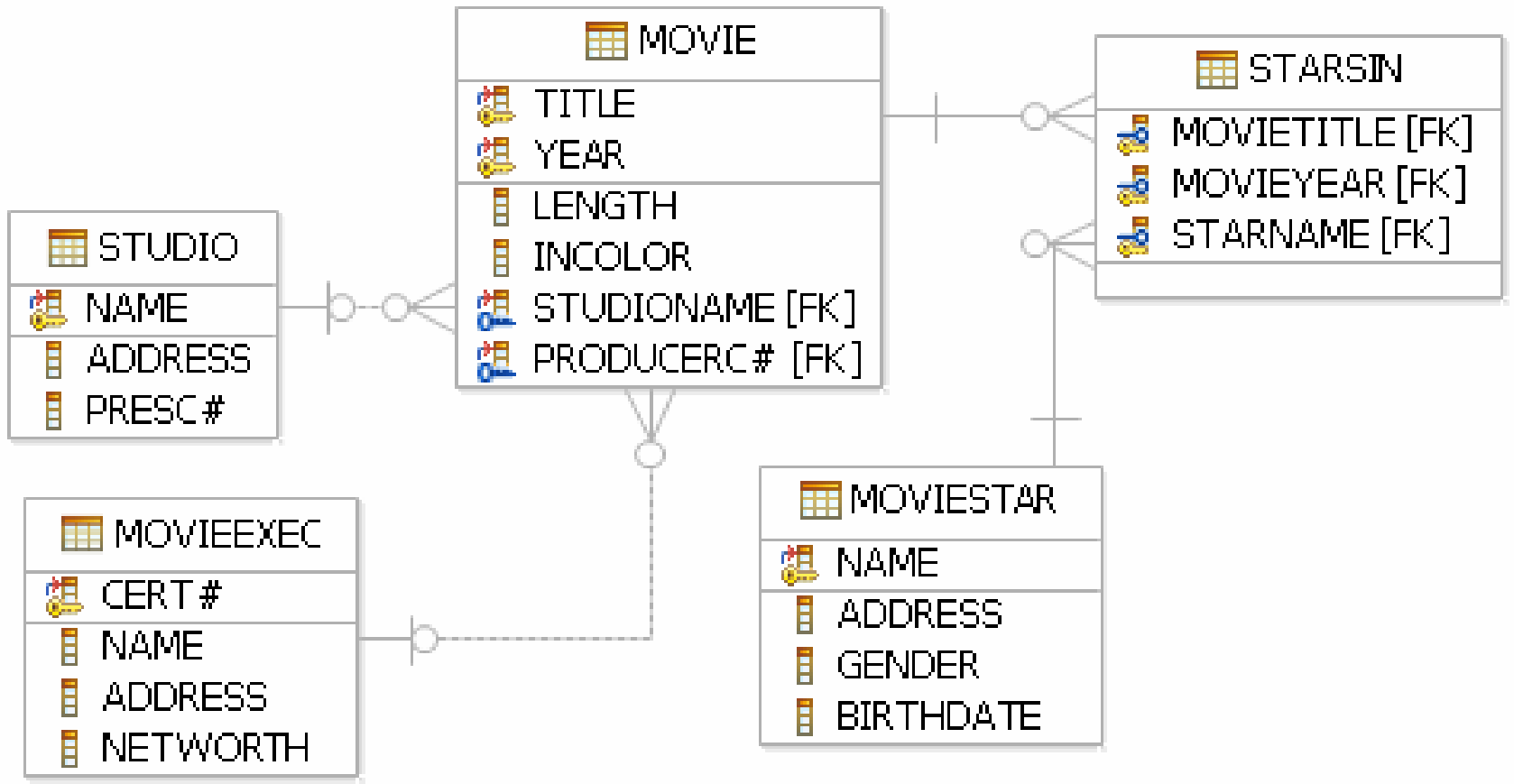
Извличане на данни

SELECT

Movies Schema



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	STUDIONAME	PRODUCERC#
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

StarsIn

	MOVIETITLE	MOVIEYEAR	STARNAME
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

SELECT

```
SELECT {* |[DISTINCT] column |
expression [alias],..}
FROM table
WHERE <condition>
GROUP BY <column>
HAVING <condition>
ORDER BY <column>;
```

SELECT

```
SELECT *
FROM MOVIE;
```

	TITLE	YEAR	LENGTH	INCOLOR	STUDIONAME	PRODUCERC#
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

SELECT Проекция

```
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

SELECT Проекция

```
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

Псевдоними

```
SELECT title, year, length, length/60 as hours
FROM MOVIE;
```

	TITLE	YEAR	LENGTH	HOURS
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

Премахване на дубликати

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

	HOURS
1	1
2	2
3	3
4	NULL

WHERE (ограничаване на връщаните редове)

```
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

WHERE - Сравнение с низове

```
SELECT TITLE, YEAR
FROM MOVIE
WHERE STUDIONAME = 'Fox';
```

	TITLE	YEAR
1	Logan's run	1976
2	Star Wars	1977
3	Star Wars: Empire Strikes Back	1980

WHERE - Сравнение с дати

```
SELECT NAME
FROM MOVIESTAR
WHERE BIRTHDATE = '1977-07-07';
```

	NAME
1	Jane Fonda

WHERE - Сравнение с NULL

```
SELECT TITLE, YEAR
FROM MOVIE
WHERE LENGTH IS NULL;
```

	TITLE	YEAR
1	Logan's run	1976

WHERE – Оператор BETWEEN

```
SELECT TITLE, YEAR
FROM MOVIE
WHERE LENGTH BETWEEN 100 AND 120;
```

	TITLE	YEAR
1	Pretty Woman	1990
2	The Man Who Wasn't There	2001
3	Star Wars: Empire Strikes Back	1980
4	Star Trek: Nemesis	2002
5	The Usual Suspects	1995

WHERE – Оператор IN

```
SELECT TITLE, YEAR
FROM MOVIE
WHERE LENGTH IN (111, 124);
```

	TITLE	YEAR
1	Star Wars	1977
2	Star Wars: Empire Strikes Back	1980

WHERE – Оператор LIKE - % и _

```
SELECT TITLE, YEAR
FROM MOVIE
WHERE TITLE LIKE 'St_r%';
```

	TITLE	YEAR
1	Star Trek	1979
2	Star Trek: Nemesis	2002
3	Star Wars	1977
4	Star Wars: Empire Strikes Back	1980

WHERE – Оператори AND, OR, NOT

```
SELECT TITLE, YEAR
FROM MOVIE
WHERE LENGTH NOT IN (111, 124)
AND   TITLE LIKE 'S%'
OR    TITLE LIKE '%Wars';
```

	TITLE	YEAR
1	Star Wars	1977
2	Star Trek	1979
3	Star Trek: Nemesis	2002

GROUP BY

```
SELECT STUDIONAME, SUM(length) AS SUM_LEN
FROM MOVIE
GROUP BY STUDIONAME;
```

	STUDIONAME	SUM_LEN
1	Disney	119
2	Fox	235
3	MGM	476
4	Paramount	248
5	USA Entertainm	116

HAVING

Когато искаме да ограничиме кои редовете да попаднат в групата използваме клузата HAVING

```
SELECT STUDIONAME, SUM(length) AS SUM_LEN
FROM MOVIE
GROUP BY STUDIONAME
HAVING SUM(length)>200;
```

	STUDIONAME	SUM_LEN
1	Fox	235
2	MGM	476
3	Paramount	248

Правила при групиране

- Можем да групираме по един или няколко атрибута или по израз
- Когато групираме в SELECT може да стоят само атрибутите, по които групираме и/или агрегатни функции (AVG, SUM и др.)
- Не можем, да имаме агрегатна функция в WHERE клаузата. В WHERE клаузата може да използваме само скаларни функции, като LENGTH, DAY и т.н.
- Агрегатни функции може да изплзваме само в SELECT и HAVING клаузата

ORDER BY

```
SELECT STUDIONAME, SUM(length) AS SUM_LEN
FROM MOVIE
GROUP BY STUDIONAME
HAVING SUM(length)>200
ORDER BY STUDIONAME DESC;
```

	STUDIONAME	SUM_LEN
1	Paramount	248
2	MGM	476
3	Fox	235

Ред на изпълнение на SELECT

```
5) SELECT {* | column | expression
[alias],..}
1) FROM table
2) WHERE <condition>
3) GROUP BY <column>
4) HAVING <condition>
6) ORDER BY <column>;
```

Скаларни функции

- Имат един или повече аргументи и връщат само една стойност
- Прилагат се върху колони или изрази върху колни
- Действат върху всеки ред от таблицата (релацията)
- За всеки ред от таблицата връщат една стойност
- Могат да бъдат влагани една в друга

Скаларни функции

- Функции за обработка на низове (Character Functions)
- Функции за обработка на числа (Numeric Functions)
- Функции за обработка на дати (Date Functions)
- Функции за преобразуване (Conversion Functions)
- Общи функции (General Functions)
  - NVL
  - NULLIF
  - COALESCE
  - DECODE

Функции за обработка на низове

- LOWER, UPPER, LENGTH
- SUBSTR, REPLACE, TRIM, CONCAT

```
SELECT UPPER(TITLE) AS TITLE,
       LENGTH(TITLE) AS LEN_TITLE
FROM MOVIE
WHERE SUBSTR(STUDIONAME,1,3)='MGM';
```

	TITLE	LEN_TITLE
1	TERMS OF ENDEARMENT	19
2	THE USUAL SUSPECTS	18
3	GONE WITH THE WIND	18

Функции за обработка на числа

- ROUND, DECIMAL

```
SELECT LENGTH/60 AS LEN_INT,
       ROUND(LENGTH/60.0) AS LEN_RND,
       DECIMAL(LENGTH/60.0, 9, 2) AS LEN_DEC
FROM MOVIE
WHERE SUBSTR(STUDIONAME,1,3)='MGM';
```

	LEN_INT	LEN_RND	LEN_DEC
1	2	2.0000000000000000000	2.20
2	1	2.0000000000000000000	1.76
3	3	4.0000000000000000000	3.96

Функции за обработка на дати

- CURRENT_DATE,YEAR

```
SELECT NAME, YEAR(CURRENT_DATE) - YEAR(BIRTHDATE) AS AGE
FROM MOVIESTAR
WHERE YEAR(CURRENT_DATE) - YEAR(BIRTHDATE)<=41;
```

	NAME	AGE
1	Jane Fonda	41
2	Alec Baldwin	41
3	Kim Basinger	39
4	Debra Winger	40

Общи функции

- COALESCE (*expr1*, *expr2*, *expr3*, . . .)

```
SELECT LENGTH, COALESCE(LENGTH,0)
FROM MOVIE
WHERE LENGTH(TITLE)<12
ORDER BY LENGTH;
```

	LENGTH	2
1	124	124
2	132	132
3	NULL	0

Агрегатни функции

- Прилагат се върху колони и действат за множество от редове от таблицата (релацията)
- За множество от редове от таблицата връщат една стойност
- Могат да се влагат, до две функции
- Типове, функции за групиране:
  - `MIN([ DISTINCT | ALL ] EXPR), MAX([ DISTINCT | ALL ] EXPR)`
  - `COUNT([ DISTINCT | ALL ] N)`
  - `SUM([ DISTINCT | ALL ] N), AVG([ DISTINCT | ALL ] N)`
  - `STDDEV([ DISTINCT | ALL ] N), VARIANCE([ DISTINCT | ALL ] N)`
- Всички агрегатни функции игнорират NULL стойностите

AVG и SUM

- AVG и SUM могат да бъдат използвани за данни от числен тип

```
SELECT SUM(LENGTH) AS SUM_LEN,
       AVG(LENGTH) AS AVG_LEN
FROM MOVIE
WHERE UPPER(TITLE) LIKE '%STAR%';
```

	SUM_LEN	AVG_LEN
1	483	120

MIN и MAX

- MIN и MAX могат да бъдат използвани за данни от числен тип, от тип дата и низове

```
SELECT MIN(LENGTH) AS MIN_LEN,
       MAX(LENGTH) AS MAX_LEN
FROM MOVIE
WHERE UPPER(TITLE) LIKE '%STAR%';
```

	MIN_LEN	MAX_LEN
1	111	132

COUNT

```
SELECT COUNT(*) AS ALL_ROWS,
       COUNT(DISTINCT STUDIONAME)  AS STUDIOS,
       COUNT(LENGTH) AS CNT_NOT_NULL_VALUES
FROM MOVIE;
```

	ALL_ROWS	STUDIOS	CNT_NOT_NULL_VALUES
1	10	5	9

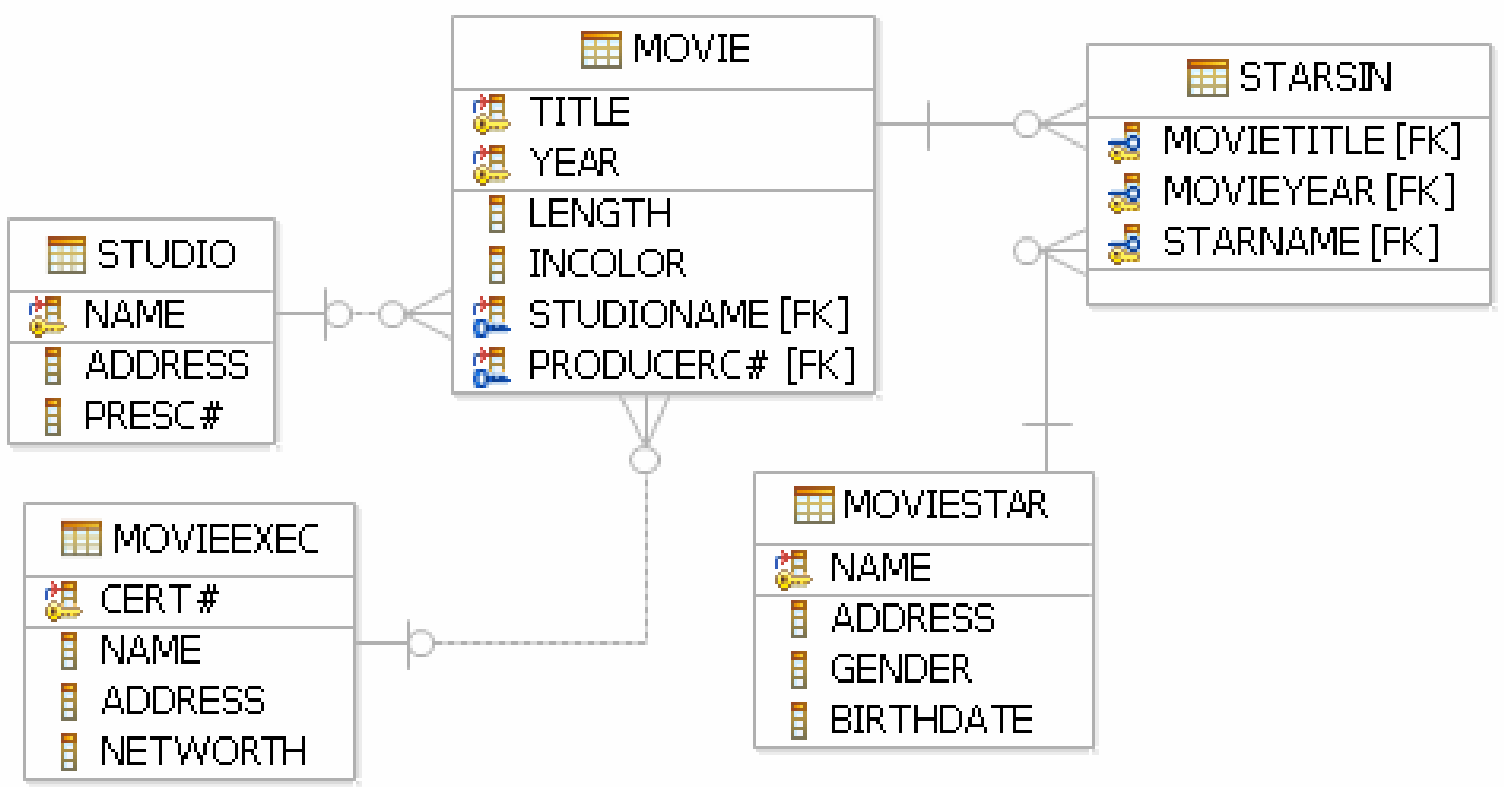
Агрегатни функции и групиране

```
SELECT SUM(LENGTH) AS SUM_LEN,
       AVG(LENGTH) AS AVG_LEN
FROM MOVIE
GROUP BY STUDIONAME
HAVING SUM(LENGTH) > 300;
```

	SUM_LEN	AVG_LEN
1	476	158

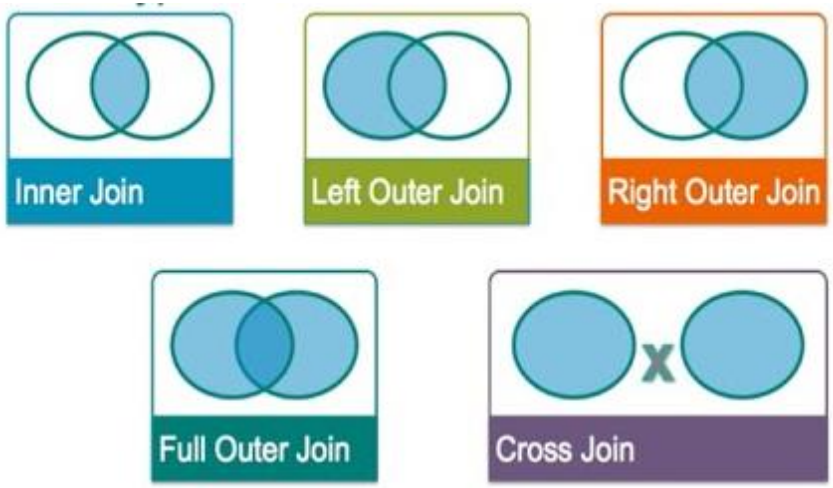
Връзка между релации (таблици)

- Първичен ключ (PRIMARY KEY)
- Външен ключ (FOREIGN KEY)



Видове съединения (JOIN)

- Natural JOIN – естествено съединение
- [Inner] JOIN
- LEFT [Outer] JOIN
- RIGTH [Outer] JOIN
- FULL [Outer] JOIN
- Cross JOIN



Natural JOIN

Съединението става по колоните от двете таблици с едни и същи имена

(MOVIE.NAME = MOVIESTAR.NAME)

```
SELECT *
FROM MOVIE NATURAL JOIN MOVIESTAR
WHERE STUDIONAME='MGM';
```

JOIN

```
SELECT STUDIONAME, TITLE, STARNAME
FROM MOVIE JOIN STARSIN
ON TITLE = MOVIETITLE
AND YEAR = MOVIEYEAR
WHERE LENGTH > 130
ORDER BY title;
```

	STUDIONAME	TITLE	STARNAME
1	MGM	Terms of Endearment	Jack Nicholson
2	MGM	Terms of Endearment	Jane Fonda

LEFT JOIN

```
SELECT STUDIONAME, TITLE, STARNAME
FROM MOVIE LEFT JOIN STARSIN
ON TITLE = MOVIETITLE
AND YEAR = MOVIEYEAR
WHERE LENGTH > 130
ORDER BY title;
```

	STUDIONAME	TITLE	STARNAME
1	MGM	Gone With the Wind	NULL
2	Paramount	Star Trek	NULL
3	MGM	Terms of Endearment	Jack Nicholson
4	MGM	Terms of Endearment	Jane Fonda

RIGHT JOIN

```
SELECT STUDIONAME, TITLE, STARNAME
FROM MOVIE RIGHT JOIN STARSIN
ON TITLE = MOVIETITLE
AND YEAR = MOVIEYEAR
WHERE LENGTH > 130
ORDER BY title;
```

	STUDIONAME	TITLE	STARNAME
1	MGM	Terms of Endearment	Jack Nicholson
2	MGM	Terms of Endearment	Jane Fonda

FULL JOIN

```
SELECT STUDIONAME, TITLE, STARNAME
FROM MOVIE FULL JOIN STARSIN
ON TITLE = MOVIETITLE
AND YEAR = MOVIEYEAR
WHERE LENGTH > 130
ORDER BY title;
```

	STUDIONAME	TITLE	STARNAME
1	MGM	Gone With the Wind	NULL
2	Paramount	Star Trek	NULL
3	MGM	Terms of Endearment	Jack Nicholson
4	MGM	Terms of Endearment	Jane Fonda

SELF JOIN

```
SELECT DISTINCT M1.TITLE, M1.YEAR
FROM MOVIE M1 JOIN MOVIE M2
ON M1.STUDIONAME=M2.STUDIONAME
WHERE M1.YEAR < M2.YEAR
AND   UPPER(M2.TITLE) LIKE '%WAR%';
```

	TITLE	YEAR
1	Logan's run	1976
2	Star Wars	1977

Multiple JOINS

```
SELECT STARNAME, BIRTHDATE, STUDIONAME
FROM MOVIE JOIN STARSIN
ON TITLE = MOVIETITLE
AND YEAR = MOVIEYEAR
JOIN MOVIESTAR ON NAME=STARNAME
WHERE LENGTH > 130
ORDER BY title;
```

	STARNAME	BIRTHDATE	STUDIONAME
1	Jane Fonda	1977-07-07	MGM
2	Jack Nicholson	1949-05-05	MGM

Cross Join (Декартово произведение)

```
SELECT STUDIONAME, TITLE, STARNAME
FROM MOVIE , STARSIN;
```

Декартово произвдение на двете релации включва и комбинации от кортежи, дори и те да не са логически свързани (верни)!

	STUDIONAME	TITLE	STARNAME
1	Disney	Pretty Woman	Alec Baldwin
2	USA Entertainm ...	The Man Who Wasn't There	Alec Baldwin
3	Fox	Logan's run	Alec Baldwin
4	Fox	Star Wars	Alec Baldwin
5	Fox	Star Wars: Empire Strikes ...	Alec Baldwin
6	Paramount	Star Trek	Alec Baldwin
7	Paramount	Star Trek: Nemesis	Alec Baldwin
8	MGM	Terms of Endearment	Alec Baldwin
9	MGM	The Usual Suspects	Alec Baldwin
10	MGM	Gone With the Wind	Alec Baldwin
11	Disney	Pretty Woman	Harrison Ford
12	USA Entertainm ...	The Man Who Wasn't There	Harrison Ford
13	Fox	Logan's run	Harrison Ford
14	Fox	Star Wars	Harrison Ford
15	Fox	Star Wars: Empire Strikes ...	Harrison Ford

Подзаявки

- Заявка, която е част от друга заявка се нарича подзаявка.
- Подзаявките се ограждат с ( )
- Подзаявките като резултат могат да връщат:
  - единствена скаларна стойност – константа
  - списък от стойности
  - релация

Подзаявки

- Нека R е релация, а L списък от стойности:
  - EXISTS (R) – Връща стойност TRUE, ако в R има кортежи
  - s IN (L) – Връща стойност TRUE, ако s е сред елементите на списъка
  - s NOT IN (L) - Връща стойност TRUE, ако s НЕ е сред елементите на списъка
  - s > ALL(L) - Връща стойност TRUE, ако s е по-голям от всички елементи на списъка
  - s > ANY(L) - Връща стойност TRUE, ако s е по-голям поне от един от елементите на списъка

Подзаявки – константна стойност

```
SELECT *
FROM STARSIN
WHERE STARNAME = (SELECT NAME
                  FROM MOVIESTAR
                  WHERE GENDER = 'F' AND NAME LIKE 'S%');
```

	MOVIETITLE	MOVIEYEAR	STARNAME
1	The Usual Suspects	1995	Sandra Bullock

Подзаявки – списък

```
SELECT *
FROM STARSIN
WHERE STARNAME IN (SELECT NAME
                   FROM MOVIESTAR
                   WHERE GENDER = 'F');
```

	MOVIETITLE	MOVIEYEAR	STARNAME
1	Star Wars	1977	Kim Basinger
2	Terms of Endearment	1983	Jane Fonda
3	The Usual Suspects	1995	Sandra Bullock

Корелативни подзаявки

Подзаявка в която имаме зависимост от главната заявката се наричат корелативна под заявка

```
SELECT DISTINCT M1.TITLE, M1.YEAR
FROM MOVIE M1
WHERE M1.STUDIONAME IN ( SELECT M2.STUDIONAME
                         FROM MOVIE M2
                         WHERE  UPPER(M2.TITLE)
                         LIKE '%WAR%'
                         AND    M1.YEAR < M2.YEAR );
```

	TITLE	YEAR
1	Logan's run	1976
2	Star Wars	1977

Подзаявки в FROM клаузата

```
SELECT MOVIETITLE, S.STARNAME, T.BIRTHDATE
FROM STARSIN AS S, (SELECT NAME, BIRTHDATE
                    FROM MOVIESTAR
                    WHERE GENDER = 'M') AS T
WHERE S.STARNAME = T.NAME;
```

	MOVIETITLE	STARNAME	BIRTHDATE
1	Star Wars	Alec Baldwin	1977-07-06
2	Star Wars	Harrison Ford	1955-05-05
3	Star Wars: Empire Strikes Back	Harrison Ford	1955-05-05
4	Terms of Endearment	Jack Nicholson	1949-05-05
5	The Usual Suspects	Jack Nicholson	1949-05-05