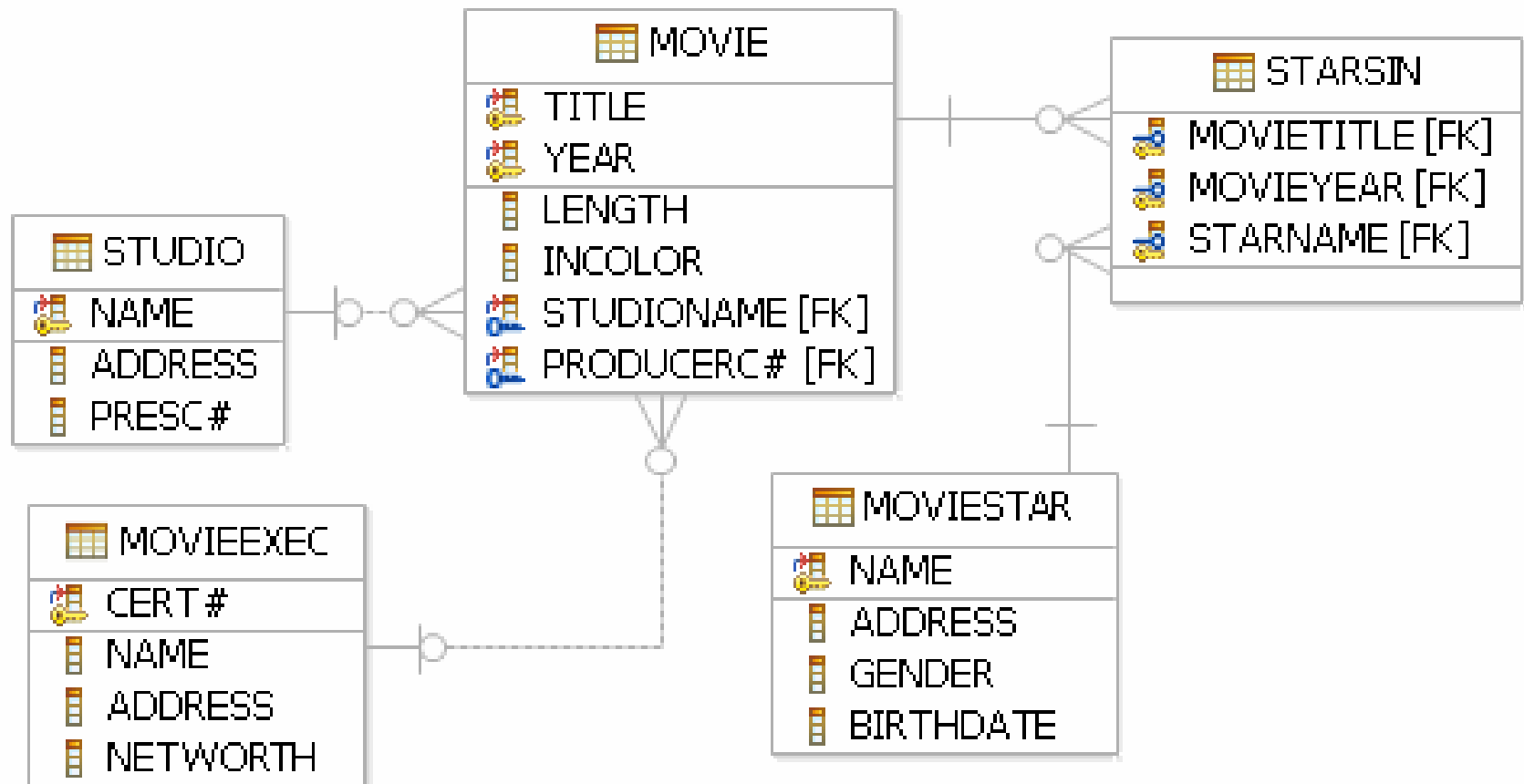




Подзаявки

Subqueries

# Схема на базата от данни - Movies



# 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
  - ▶  $s \text{ IN } (L)$  – Връща стойност  $TRUE$ , ако  $s$  е сред елементите на списъка
  - ▶  $s \text{ NOT IN } (L)$  - 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%') ;
```

	MOVIE TITLE	MOVIE YEAR	STAR NAME
1	The Usual Suspects	1995	Sandra Bullock



# Подзаявки – списък

---

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

	MOVIE TITLE	MOVIE YEAR	STAR NAME
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 MOVIE_TITLE, S.STARNAME, T.BIRTHDATE
FROM STARSIN AS S, (SELECT NAME, BIRTHDATE
                     FROM MOVIESTAR
                     WHERE GENDER = 'M') AS T
WHERE S.STARNAME = T.NAME;
```

	MOVIE_TITLE	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



# 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 STUDIO_NAME) AS STUDIOS,  
       COUNT(LENGTH) AS CNT_NOT_NULL_VALUES  
FROM MOVIE;
```

	ALL_ROWS	STUDIOS	CNT_NOT_NULL_VALUES
1	10	5	9

