

Бази от данни

Упражнение 4, втора част:

Съединения

Димитър Димитров

Пример за FULL JOIN

- За всеки актьор и/или продуцент да се изведе името му, рождената му дата и networth:
- Неудобен начин:

```
SELECT ms.name, me.name, ms.birthdate, me.networth  
FROM moviestar ms  
FULL JOIN movieexec me ON ms.name = me.name;
```

- Хубав начин:

```
SELECT COALESCE(ms.name, me.name) AS name,  
       ms.birthdate, me.networth  
FROM moviestar ms  
FULL JOIN movieexec me ON ms.name = me.name;
```

Заявки с повече
от един оператор JOIN

Първи пример

- За всяка държава да се изведат имената на корабите, които никога не са участвали в битка

```
select country, name  
from classes  
join ships on classes.class = ships.class  
left join outcomes on name = ship  
where outcomes.ship is null;
```

- Нека обърнем реда, в който свързваме таблиците:

```
select country, name  
from outcomes  
right join ships on ship = name  
join classes on classes.class = ships.class  
where outcomes.ship is null;
```

Втори пример (1)

- За всеки клас британски кораби да се изведат имената им (на класовете) и имената на всички битки, в които са участвали кораби от този клас.
Ако даден клас няма кораби или има, но те не са участвали в битка, също да се изведат (със стойност NULL за име на битката).

```
select distinct classes.class, battle  
from outcomes  
join ships on ship = name  
right join classes  
          on ships.class = classes.class  
where country = 'Gt.Britain';
```

- Има ли значение дали в SELECT клаузата ще посочим колоната `classes.class` или `ships.class`?

Втори пример (2)

- Нека пробваме да "обърнем" заявката, както направихме в първия пример:

```
select distinct classes.class,  
battle  
from classes  
left join ships  
      on classes.class = ships.class  
join outcomes on name = ship  
where country = 'Gt.Britain';
```

Втори пример (3)

- Нека пробваме да "обърнем" заявката, както направихме в първия пример:
- **`select distinct classes.class, battle
from classes
left join ships
on classes.class = ships.class
join outcomes on name = ship
where country = 'Gt.Britain';`**
- Този път резултатът е грешен!

Втори пример (4)

- Да се опитаме да поправим заявката, като направим и двете съединения ВЪНШНИ:

```
select distinct classes.class, battle
from classes
left join ships
      on classes.class = ships.class
LEFT join outcomes on ship = name
where country = 'Gt.Britain';
```


Втори пример (5)

- Да се опитаме да поправим заявката, като направим и двете съединения ВЪНШНИ:

```
select distinct classes.class, battle
from classes
left join ships
      on classes.class = ships.class
LEFT join outcomes on ship = name
where country = 'Gt.Britain';
```

Странно, отново е грешна!

ИЗВОД

- Когато в една заявка има повече от един оператор JOIN и поне един от тях е OUTER JOIN, редът на прилагането им има значение
- Ако са само INNER JOIN – няма
- Ако държим на LEFT JOIN в предишната задача, може да я решим така:

```
select distinct classes.class, battle
from classes
left join (select *
           from ships
           join outcomes on ship = name) t
  on classes.class = t.class
where country = 'Gt.Britain';
```

Условие в ON или в WHERE?

Първи пример – INNER JOIN

- За всеки клас да се изведат името му, държавата и имената на всички негови кораби, пуснати през 1916 г.

```
select classes.class, country, name  
from classes  
join ships on classes.class = ships.class  
where launched = 1916;
```

- Задачата може да се реши и така, макар и безсмислено:

```
select classes.class, country, name  
from classes  
join ships on classes.class = ships.class  
and launched = 1916;
```

Втори пример (1)

- Да допълним горната задача, като добавим и класовете, които нямат нито един кораб от 1916 - срещу тях да пише NULL:

```
select classes.class, country, name  
from classes  
left join ships  
      on classes.class = ships.class  
where launched = 1916;
```

- Дали е вярно горното решение?

Втори пример (2)

- Да допълним горната задача, като добавим и класовете, които нямат нито един кораб от 1916 - срещу тях да пише NULL:

```
select classes.class, country, name  
from classes  
left join ships  
      on classes.class = ships.class  
where launched = 1916;
```

- Горното решение не е вярно!
- Изпускаме например класовете, които нямат кораби – при тях launched има стойност NULL

Втори пример (3)

- Щом е така, не можем ли просто да добавим проверка за NULL?

```
select classes.class, country, name  
from classes  
left join ships  
      on classes.class = ships.class  
where launched = 1916 or launched is null;
```

- Все още е грешно!
 - Липсват класове, които имат кораби, но нито един от тези кораби не е от 1916
 - Хващаме излишни редове, ако launched по принцип позволява NULL

Втори пример (4)

- Хайде да видим и поне едно вярно решение:

```
select classes.class, country, name
from classes
left join ships
on classes.class = ships.class and launched = 1916;
```

- Или (излишно усложнено, но по-разбираемо):

```
select classes.class, country, name
from classes
left join (select *
           from ships
           where launched = 1916) ships1916
on classes.class = ships1916.class;
```


ИЗВОДИ

- При OUTER JOIN има разлика дали дадено условие ще бъде в ON или в WHERE
- При INNER JOIN няма, понеже се свежда до подмножество на декартово произведение
- При LEFT JOIN ползваме
 - ON, ако имаме условие за дясната таблица
 - WHERE, ако имаме условие за лявата, напр. класове, започващи с "X"

Допълнителен материал

- Представяне на дървовидна структура с таблица
- За всеки служител съхраняваме:
 - Уникален идентификатор - `employeeId`
 - Имена - `firstName`, `lastName`
 - Идентификатор на пряк началник – `reportsTo`
- Всеки служител е подчинен на друг служител от същата таблица
- Хората на най-високо ниво в йерархията имат стойност `NULL` за `reportsTo`
- За всеки служител да се изведат името му и името на прекия му началник. Да се изведат и тези служители, които нямат началник

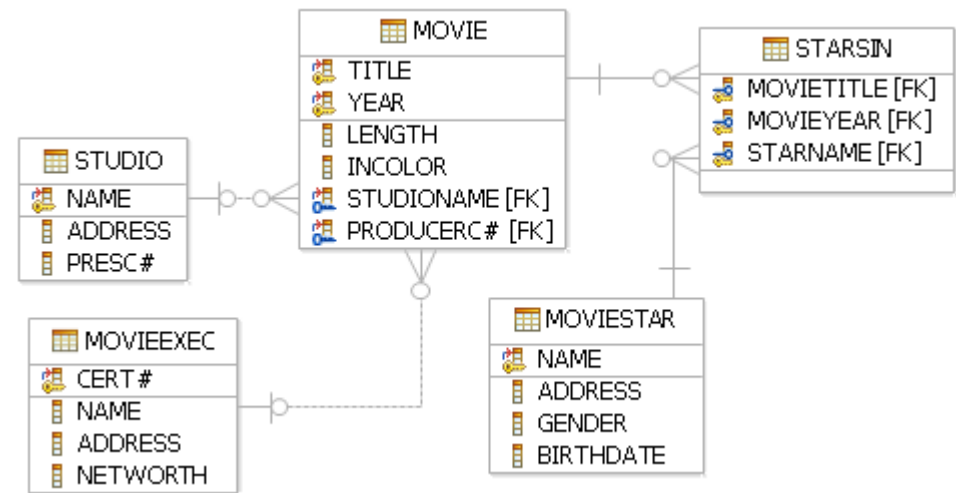
```
select e.firstName, e.lastName, b.firstName, b.lastName  
from Employees e  
left join Employees b on e.reportsTo = b.employeeId;
```

Въпроси?

Следват задачи

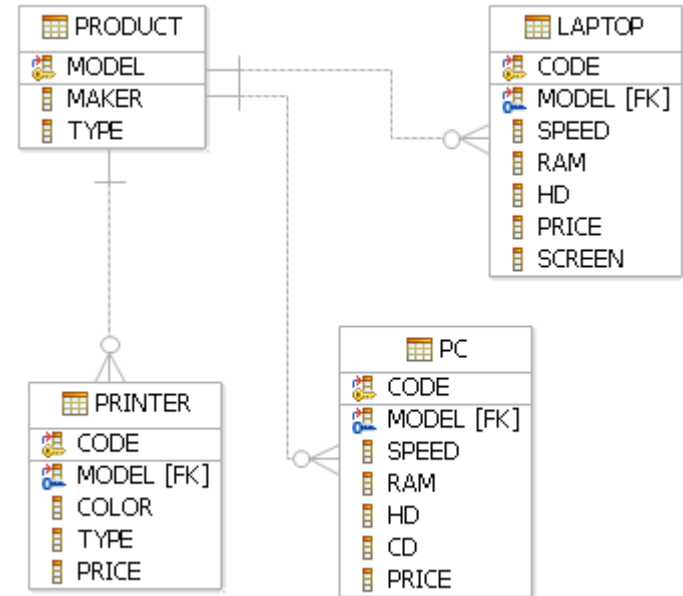
1. Задачи - Movies

1. Напишете заявка, която за всеки филм, по-дълъг от 120 минути, извежда заглавие, година, име и адрес на студио.
2. Напишете заявка, която извежда името на студиото и имената на актьорите, участвали във филми, произведени от това студио, подредени по име на студио.
3. Напишете заявка, която извежда имената на продуцентите на филмите, в които е играл Harrison Ford.
4. Напишете заявка, която извежда имената на актрисите, играли във филми на MGM.
5. Напишете заявка, която извежда името на продуцента и имената на филмите, продуцирани от продуцента на 'Star Wars'.
6. Напишете заявка, която извежда имената на актьорите, които не са участвали в нито един филм.



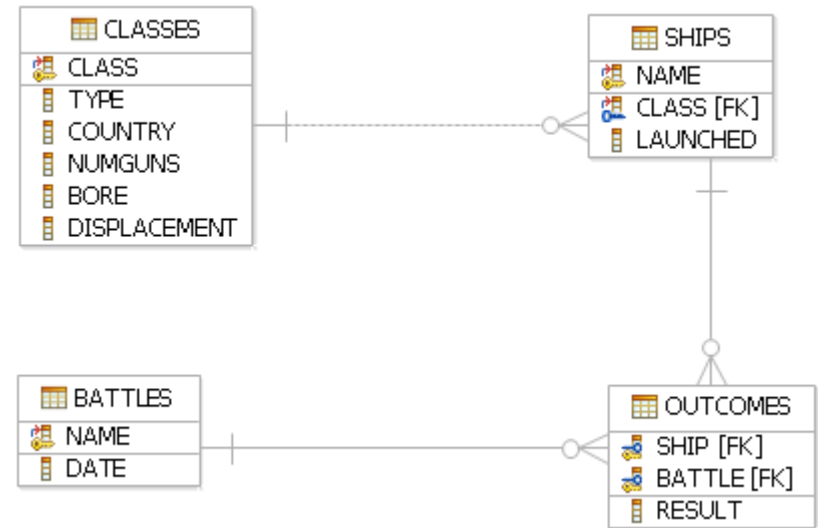
2. Задачи - PC

1. За всеки модел компютри да се изведат цените на различните конфигурации от този модел. Ако няма конфигурации за даден модел, да се изведе NULL. Резултатът да има две колони: model и price.
2. Напишете заявка, която извежда производител, модел и тип на продукт за тези производители, за които съответният продукт не се продава (няма го в таблиците PC, Laptop или Printer).



3. Задачи - Ships

1. Напишете заявка, която за всеки кораб извежда името му, държавата, броя оръдия и годината на пускане (launched).
2. Напишете заявка, която извежда имената на корабите, участвали в битка от 1942 г.



4. Общи задачи върху материала до момента

- Разгледайте документа с общи задачи