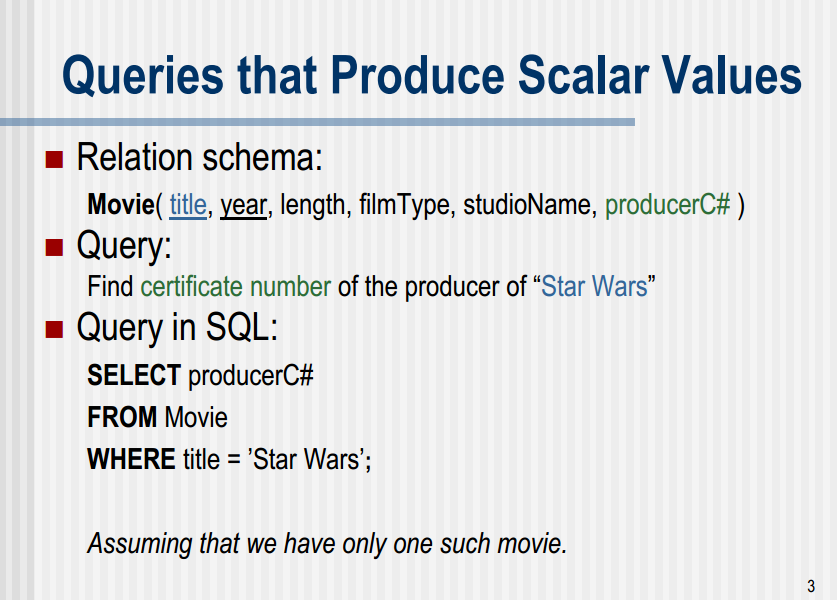
Scalar Value

SQL query作用到一组tuple上，最终产生一个relation表格

这些collection或许只有一个attribute或者只有一个value

同时满足的时候，我们说这个query产生了一个scalar value标量

//理解：就是nested query的前置知识，先要用一个query求出需要的值，然后复合query IN this relation



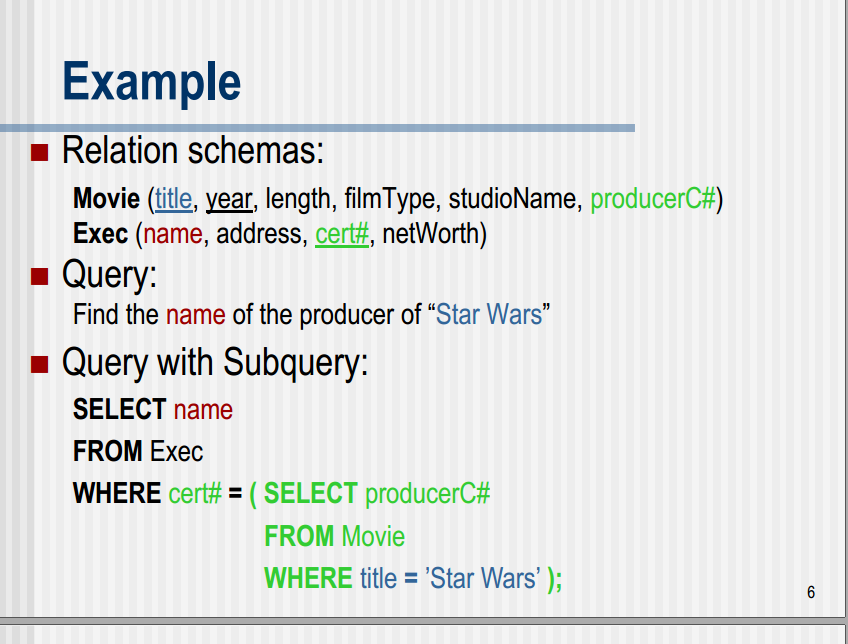
产生了一个标量

只有一个attribute producer C#

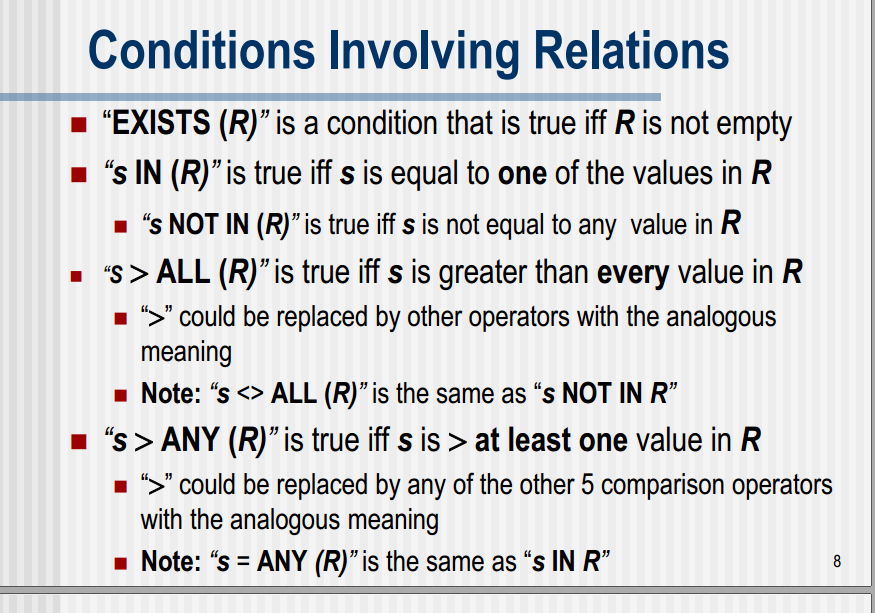
只有一个value



常规版本



NESTED版本，因为选出来的ProduceC#只有一个标准值



R指的是一个relation，

EXISTS代表着这个R不是EMPTY

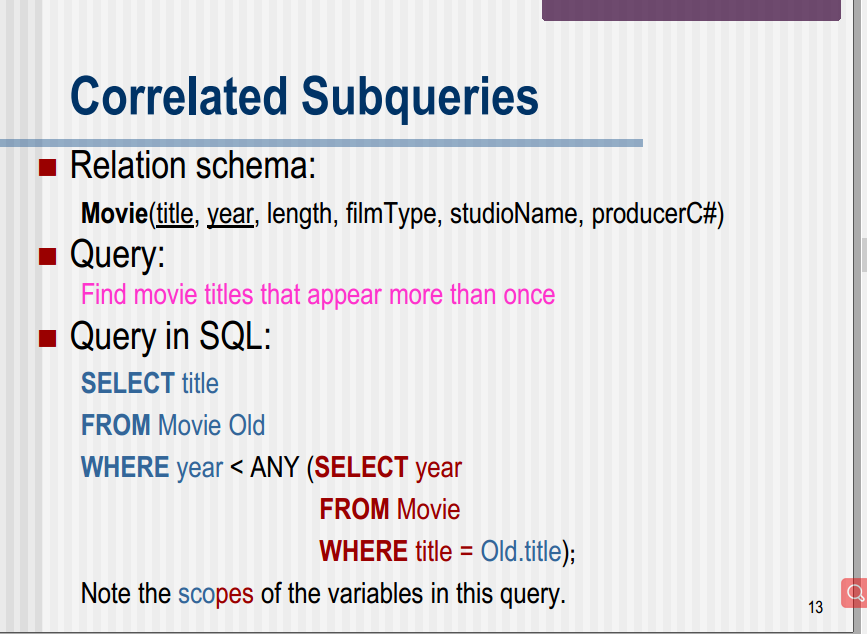
s IN (R)如果s是R的一部分

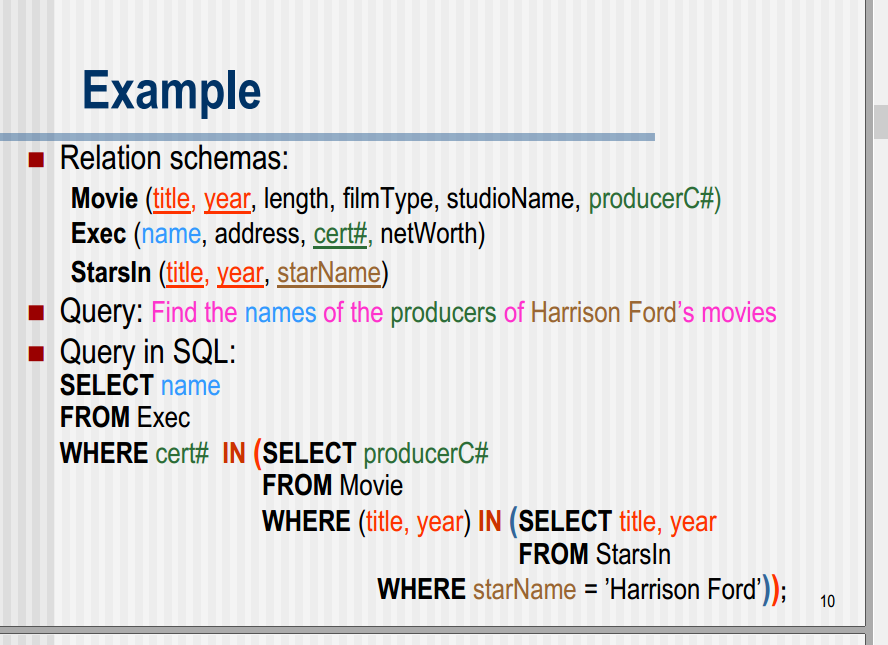
s>ALL (R)比R的所有value都大

ANY,至少大于一个R

Correlated subqueries

自己与自己的NEST





你也可以用括号将一组attribute作为一个attribute

Views

View是database里定义的一个table，但是并没有显示explicitly存储在database里，它是通过计算生成的

Logical data independence：

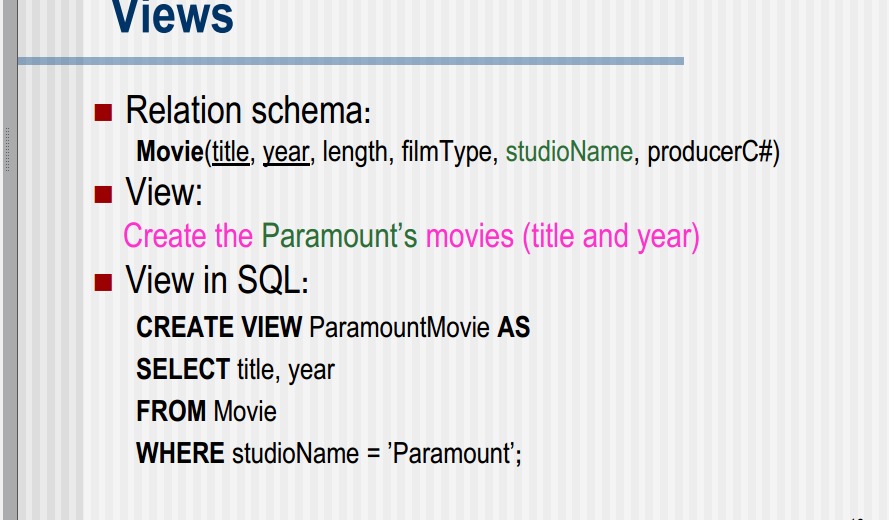
如果relation的结构变了，我们可以描述一个使用old schema定义一个view，这样应用仍然可以使用原来的old schema

安全：user只能看到有限的信息并且操作

　　1）简单：使用视图的用户完全不需要关心后面对应的表的结构、关联条件和筛选条件，对用户来说已经是过滤好的复合条件的结果集。

　　2）安全：使用视图的用户只能访问他们被允许查询的结果集，对表的权限管理并不能限制到某个行某个列，但是通过视图就可以简单的实现。

　　3）数据独立：一旦视图的结构确定了，可以屏蔽表结构变化对用户的影响，源表增加列对视图没有影响；源表修改列名，则可以通过修改视图来解决，不会造成对访问者的影响。



CREATE VIEW AS

只让你看title与year