作用：**保持数据的一致性、完整性**

**为何说外键有性能问题：**

1.**数据库需要维护外键的内部管理；**

2.外键等于把**数据的一致性事务实现，全部交给数据库服务器完成**；

3.有了外键，当**做一些涉及外键字段的增，删，更新操作之后，需要触发相关操作去检查，而不得不消耗资源**；

4.外键还会因为需**要请求对其他表内部加锁而容易出现死锁情况**；

[**数据库外键的使用以及优缺点**](http://www.cnblogs.com/dengyang/archive/2013/01/16/2862120.html)

摘录网上讨论共同观点： **主键和索引是不可少的，不仅可以优化数据检索速度，开发人员还省不其它的工作,**  
矛盾焦点：数据库设计是否需要外键。这里有两个问题：一个是如何保证数据库数据的完整性和一致性；二是第一条对性能的影响。

正方观点： 1，由数据库自身保证数据一致性，完整性，更可靠，因为程序很难100％保证数据的完整性，而用**外键即使在数据库服务器当机或者出现其他问题的时候，也能够最大限度的保证数据的一致性和完整性**。 eg：数据库和应用是一对多的关系，Ａ应用会维护他那部分数据的完整性，系统一变大时，增加了Ｂ应用，Ａ和Ｂ两个应用也许是不同的开发团队来做的。他们如何协调保证数据的完整性，而且一年以后如果又增加了C应用呢？ 2，有主外键的数据库设计可以增加ER图的可读性，这点在数据库设计时非常重要。 3，**外键在一定程度上说明的业务逻辑，会使设计周到具体全面。**

反方观点： 1，可以用触发器或应用程序保证数据的完整性 2，过分强调或者说使用主键／外键会平添开发难度，导致表过多等问题 3，**不用外键时数据管理简单，操作方便，性能高（导入导出等操作，在insert,   update,   delete   数据的时候更快）** eg:在海量的数据库中想都不要去想外键，试想，一个程序每天要insert数百万条记录，当存在外键约束的时候，每次要去扫描此记录是否合格，一般还不 止一个字段有外键，这样扫描的数量是成级数的增长！我的一个程序入库在3个小时做完，如果加上外键，需要28个小时！

结论： 1，**在大型系统中（性能要求不高，安全要求高），使用外键；在大型系统中（性能要求高，安全自己控制），不用外键**；**小系统随便，最好用外键。** 2，用外键要适当，不能过分追求 3，不用外键而用程序控制数据一致性和完整性时，应该写一层来保证，然后各个应用通过这个层来访问数据库。