几个比较经典的规则：

建立索引的规则：

1、利用最左前缀：Mysql会一直向右查找直到遇到范围操作（>，<，like、between）就停止匹配。比如a=1 and b=2 and c>3 and d=6；此时如果建立了（a,b,c,d）索引，那么后面的d索引是完全没有用到，当换成了（a,b,d,c）就可以用到。

**2、不能过度索引：在修改表内容的时候，索引必须更新或者重构，所以索引过多时，会消耗更多的时间。**

3、尽量扩展索引而不要新建索引

**4、最适合的索引的列是出现在where子句中的列或连接子句中指定的列。**

**5、不同值较少的列不必要建立索引（性别）。**

普遍的规则：

**1、表的主键、外键必须有索引；**  
  
2、数据量超过300的表应该有索引；  
  
3、经常与其他表进行连接的表，在连接字段上应该建立索引；  
  
4、经常出现在Where子句中的字段，特别是大表的字段，应该建立索引；  
  
5、索引应该建在选择性高的字段上；  
  
6、索引应该建在小字段上，对于大的文本字段甚至超长字段，不要建索引；  
  
7、复合索引的建立需要进行仔细分析；尽量考虑用单字段索引代替：  
  
   A、正确选择复合索引中的主列字段，一般是选择性较好的字段；  
  
   B、复合索引的几个字段是否经常同时以AND方式出现在Where子句中？单字段查询是否极少甚至没有？如果是，则可以建立复合索引；否则考虑单字段索引；  
  
   C、如果复合索引中包含的字段经常单独出现在Where子句中，则分解为多个单字段索引；  
  
   D、如果复合索引所包含的字段超过3个，那么仔细考虑其必要性，考虑减少复合的字段；  
  
   E、如果既有单字段索引，又有这几个字段上的复合索引，一般可以删除复合索引；  
  
8、频繁进行数据操作的表，不要建立太多的索引；  
  
9、删除无用的索引，避免对执行计划造成负面影响；  
  
   以上是一些普遍的建立索引时的判断依据。一言以蔽之，索引的建立必须慎重，对每个索引的必要性都应该经过仔细分析，要有建立的依据。因为太多的索引与不充分、不正确的索引对性能都毫无益处：在表上建立的每个索引都会增加存储开销，索引对于插入、删除、更新操作也会增加处理上的开销。另外，过多的复合索引，在有单字段索引的情况下，一般都是没有存在价值的；相反，还会降低数据增加删除时的性能，特别是对频繁更新的表来说，负面影响更大