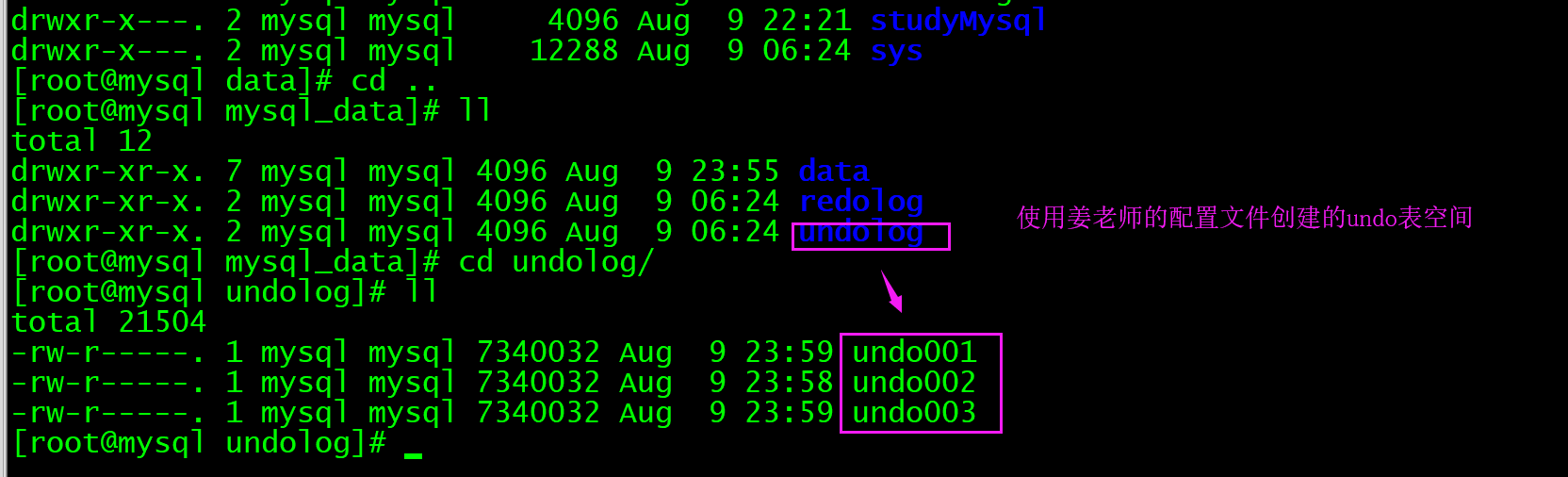
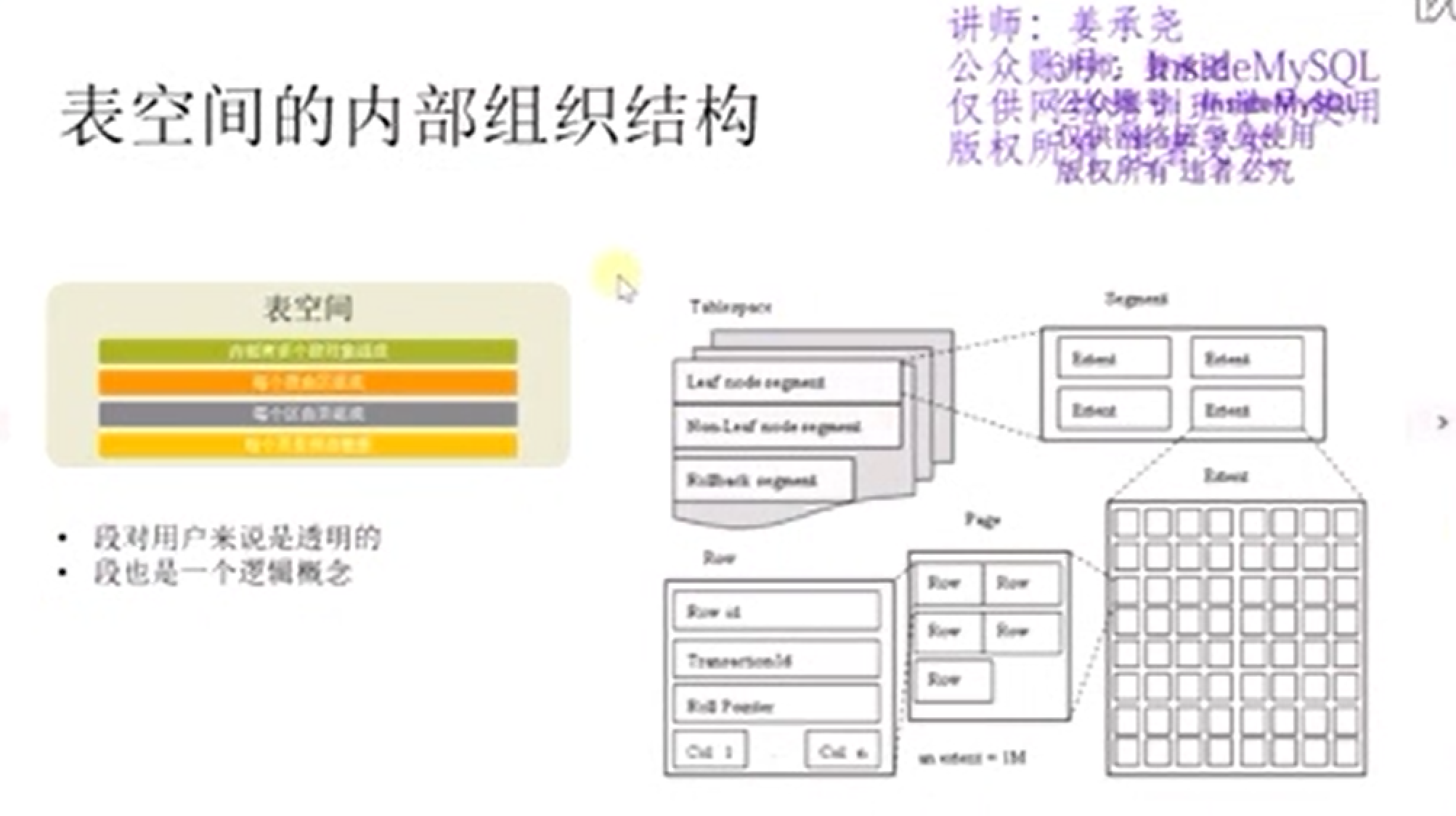
在日常工作中例行检查，查询当前线上是否存在没有主键的表存在？？？

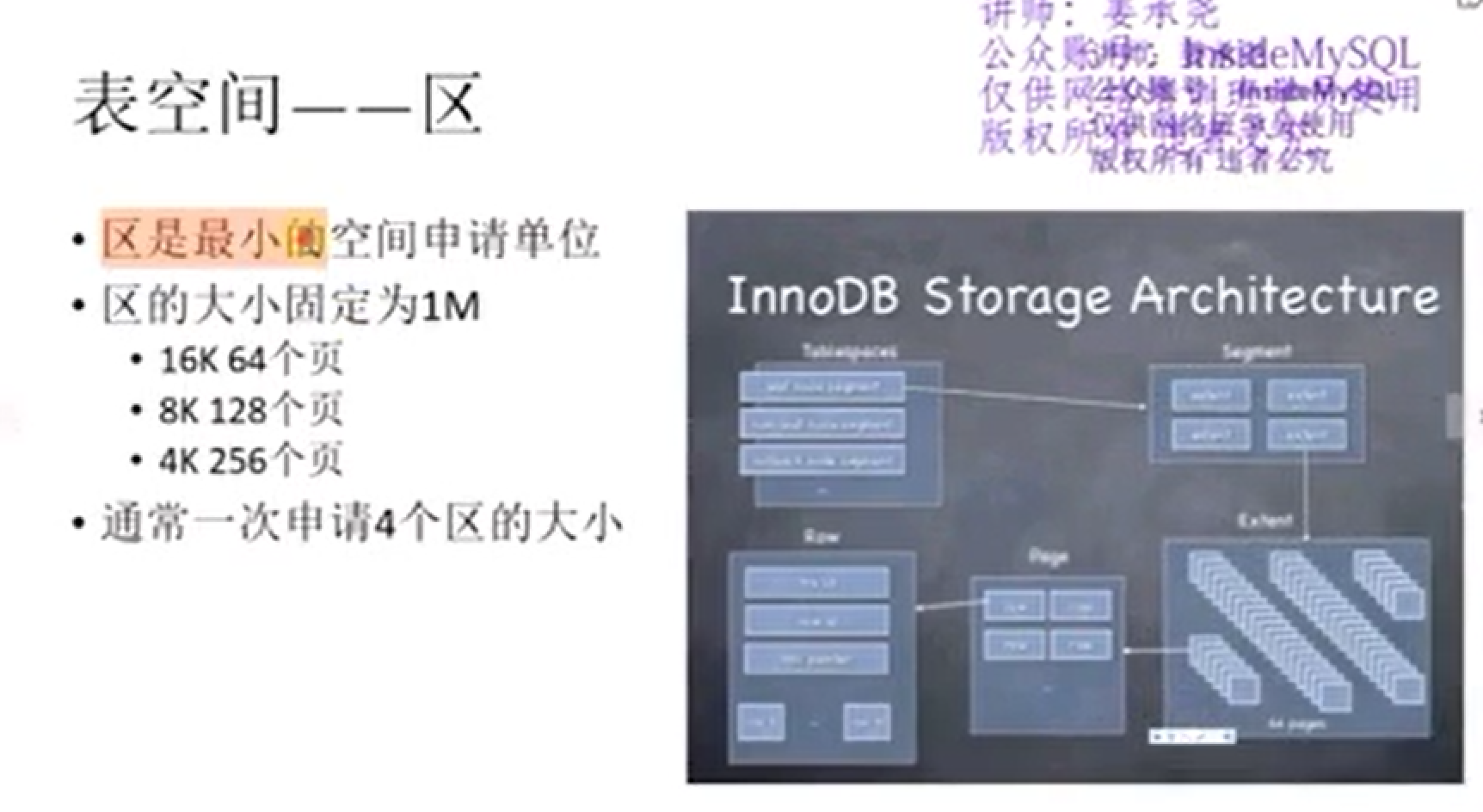
这里要用要information\_schema这个数据库，这个库中有很多元数据信息表。和索引组织表有关系。。。



表空间的内部结构



======区=====



为什么要以1m的大小来申请呢？

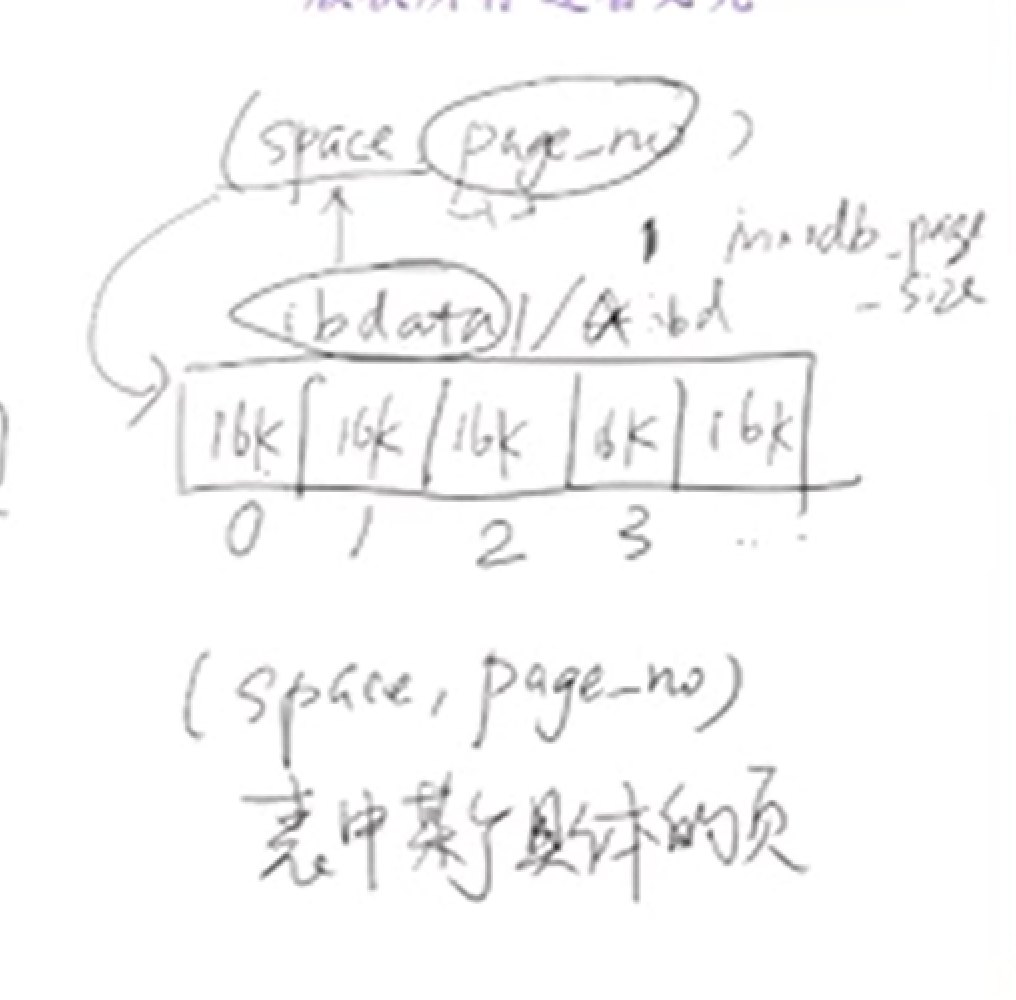
因为1m申请，1m以内得空间都是连续的，这个很重要。

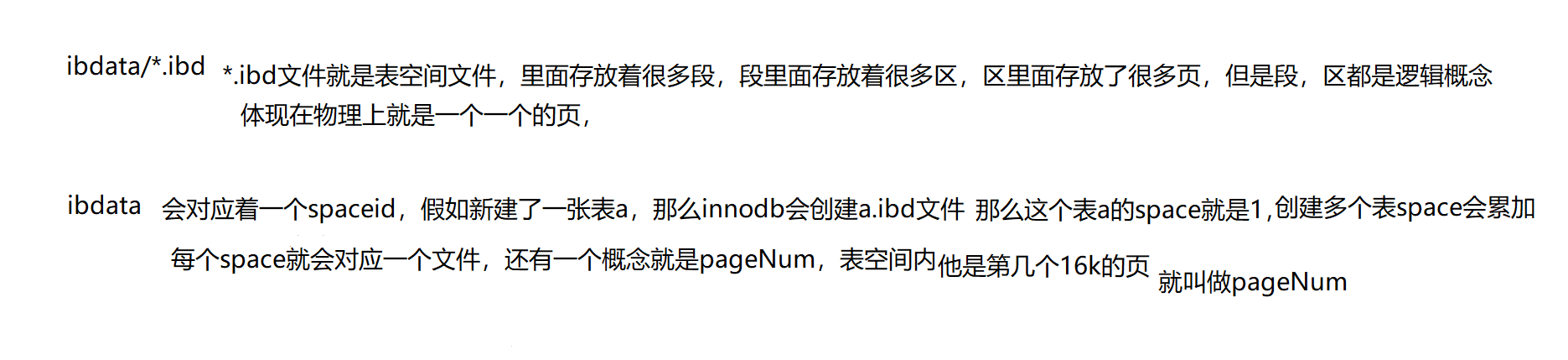
源码的角度去看，不是一次申请一个区，而是一次申请4个区的大小。

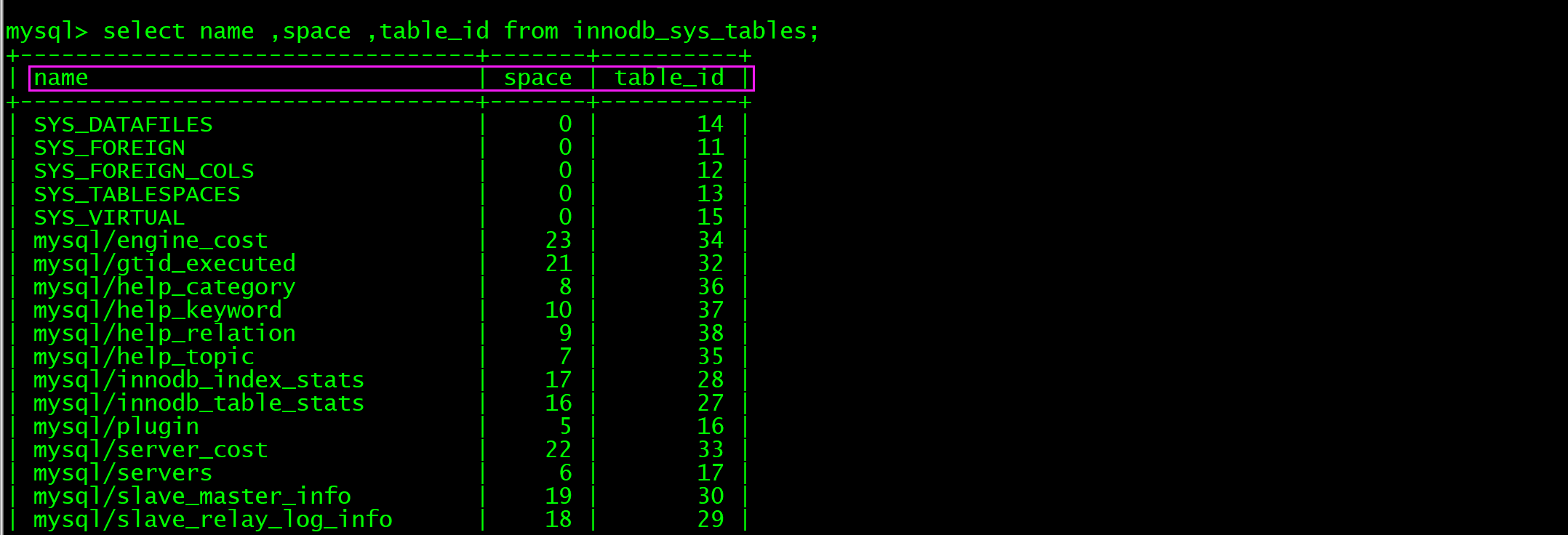
=======页========



在一个区中怎么定位一个页呢？



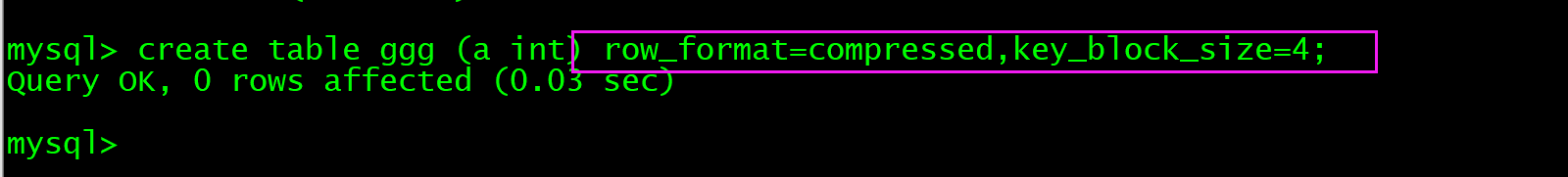




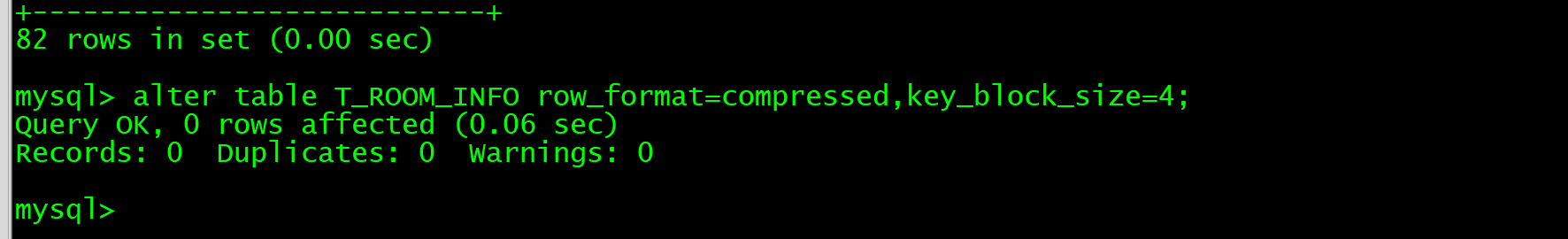
========================

Innodb\_page\_size 的大小是全局的，普通用户创建的表都是按照Innodb\_page\_size大小来创建的。

但是压缩表页的大小不是按照Innodb\_page\_size大小来创建的，是按照sql中的key\_block\_size指定的大小来创建的，单位是字节。



Innodb的压缩设计就是高性能的压缩。



开启压缩性能应该是变好，一次io读取的量比较小了。

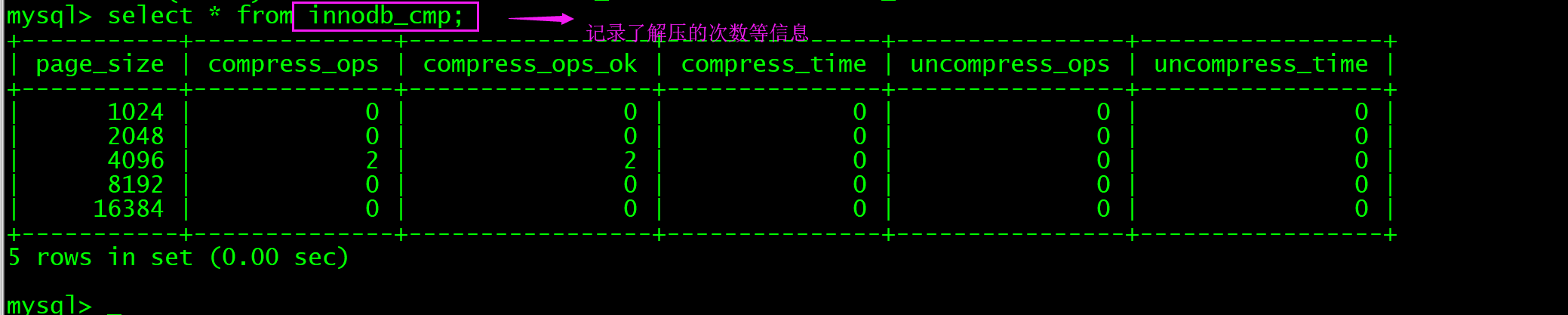
Row\_format=comparessed 压缩的对象是页，不是针对每条记录进行压缩。在读取页的时候进行解压，在写这个页的时候需要压缩，不需要每插入一条记录的时候进行解压压缩操作。

压缩的时候原来16k的页可以压缩成4k么？

如果可以压缩成4k，可以直接压缩成4k，那么假如只能压缩成8k呢，只能用2个4k的页，

假如是12k呢？只能用3个4k的页。

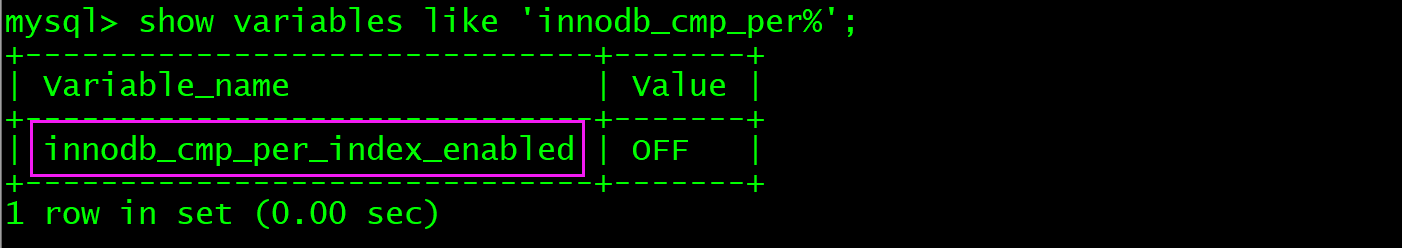
那么怎么判断能压成多少k？回再内存中先压缩一下，看看能压缩成多少。



还有innodb\_cmp\_reset这个表，当访问这个表的时候会把innodb\_cmp这个表清空，然后把innodb\_cmp里面的数据拿到innodb\_cmp\_reset表中。

=========补充innodb\_cmp不能和库表对应起来的缺陷========

1 首先把这个参数打开(对性能有影响，建议不要开启)



2 查询这个表就可以看到每个表每个索引的压缩情况。

