

索引的原理

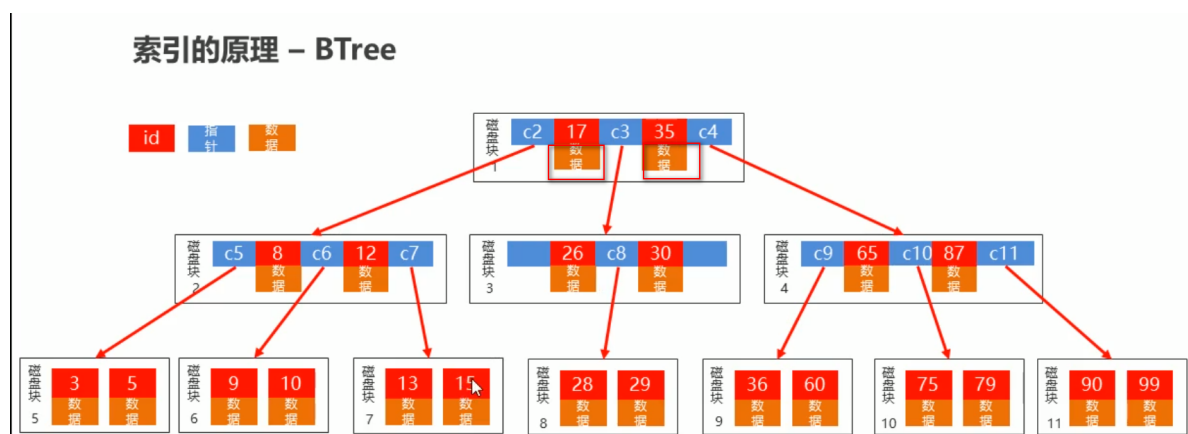
- 磁盘存储
- BTree
- B+Tree

索引的原理-磁盘原理

- 系统从磁盘读取数据到内存时 是以磁盘块（block）为基本单位；
- 位于同一个磁盘块中的数据，会被一次性全部读取出来，而不是需要什么取什么；
- InnoDB存储引擎中有页（Page）的概念,页是其磁盘管理的最小单位；
- InnoDB存储引擎中默认每个页的大小为16KB；
- InnoDB引擎若干个地址连接磁盘块，以此来达到页的大小 16KB，在查询数据时如果一个页中的每条数据都能有助于定位数据记录的位置，这将会减少磁盘I/O次数，提高查询效率；

索引的原理-BTree

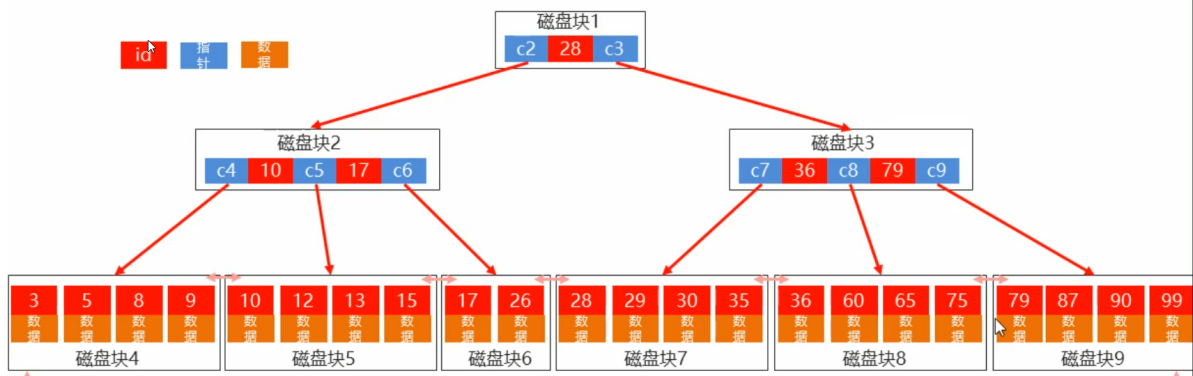
- 特点：
 - 每个磁盘块保存了 id 数据 指针（前后磁盘块的指针）
 - 每次读取数据时，前面磁盘块中的数据都会被读取出来，导致效率不高；



索引的原理-B+Tree

- 特点：
 - 只有在叶子磁盘块上才存储了数据；
 - 每次在读取数据时，前面的磁盘块中就不会读取数据，提高效率；
 - 每个叶子磁盘块都连接起来（方便了范围的查询）；

索引的原理 – B+Tree



总结

- BTree 数据结构
 - 每个节点中不仅包含key值，还有数据。会增加查询数据时磁盘的IO次数；
- B+Tree数据结构
 - 非叶子节点只存储key值；
 - 所有的数据存储在叶子节点；
 - 所有的叶子节点之间都有连接指针；
- B+Tree好处
 - 提高查询速度；
 - 减少磁盘的IO次数；
 - 树型结构较小；