

主键，不少人以为自己懂了，却不透彻...

原创 58沈剑 架构师之路 2019-10-30

MySQL是被广泛应用的数据库，**InnoDB**又是用得最广的存储引擎，**主键**又是对InnoDB性能影响最大的因素之一，主键选择对，可以极大提升性能。

InnoDB**选择什么列作为主键**，有一些最佳实践：

- (1) **不能为空**的列；
- (2) **不能重复**的列；
- (3) **很少改变**的列；

画外音：行是按照聚集索引物理排序的，如果主键频繁改变，物理顺序会改变，性能会急剧降低。

- (4) **经常被检索**(where key=XXX)的列

画外音：被检索的列上要建立索引，如果该索引是聚集索引，能够避免回表，性能提升几乎一倍。

- (5) **不是太长**的列

画外音：普通索引叶子节点会存储主键值，如果主键值太长，会增加普通索引的大小。

聚集索引，普通索引底层结构如何？

详见《InnoDB聚集索引，普通索引的索引差异》。

什么是回表，什么是索引覆盖，如何避免回表？

详见《如何避免回表查询？什么是索引覆盖？》。

主键太长为啥特别影响性能？

详见《数据库，主键为何不宜太长长长？》。

参考上述最佳实践，业务上经常采用这样的一些列作为主键：

- **用户ID**：uid
- **消息ID**：msgid
- **订单ID**：oid
- ...

这些列，能够很好的满足**非空、唯一、不变、经常被查询、长整型**等最佳实践，性能较高。

画外音：这些ID建议业务侧使用snowflake生成，即全局唯一，又趋势递增。

如果没有这样的业务属性，也可以使用自增ID(auto_inc_id)作为主键，自增ID能够满足**非空，唯一，不变，长整型**等最佳实践，性能也比较高。

画外音：自增ID不宜暴露给上游，否则分库扩展时有大坑。

但是，上面毕竟是理论，落到实操层面，我们真的掌握得这么透彻吗？下面五个小习题，看大家对InnoDB主键到底掌握到什么程度。

练习一：建表时，可不可以不声明主键？

```
(1) create table user(  
    name varchar(10)  
)engine=innodb;  
(2) insert into user values('shenjian');  
(3) insert into user values('shenjian');
```

画外音：建表时，不声明主键，插入两个相同的元素。

提问，连续执行上面的语句，执行结果是：

- A 建表语句(1)报错
- B 插入语句(2)报错
- C 插入语句(3)报错
- D 均不报错

练习二：建表时，可不可以不声明主键非空？

```
(1) create table user(  
    id int,  
    name varchar(10),  
    primary key(id)  
)engine=innodb;  
(2) insert into user(name) values('shenjian');  
(3) insert into user(name) values('shenjian');
```

画外音：建表时，不声明非空，插入两个相同的元素。

提问，连续执行上面的语句，执行结果是：

- A 建表语句(1)报错
- B 插入语句(2)报错
- C 插入语句(3)报错
- D 均不报错

练习三：建表时，可不可以选择多个字段做主键？

```
(1) create table user(  
    id int not null,  
    name varchar(10) not null,  
    primary key(id, name)  
)engine=innodb;  
(2) insert into user values(1, 'shenjian');  
(3) insert into user values(1, 'zhangsan');  
(4) insert into user values(2, 'shenjian');
```

画外音：建表时，声明联合主键(a,b)，插入若干元素，有些a重复，有些b重复。

提问，连续执行上面的语句，执行结果是：

- A 建表语句(1)报错
- B 插入语句(2)报错
- C 插入语句(3)报错
- D 插入语句(3)报错
- E 均不报错

练习四：可不可以主动插入自增主键？

```
(1) create table user(  
    id int auto_increment,  
    name varchar(10) not null,  
    primary key(id)  
)engine=innodb;  
(2) insert into user(name) values('shenjian');  
(3) insert into user(id, name) values(10,'shenjian');  
(4) insert into user(name) values('shenjian');
```

画外音：建表时，自增ID为主键，插入若干元素，有些包含自增ID，有些不包含。

提问，连续执行上面的语句，执行结果是：

- A 建表语句(1)报错
- B 插入语句(2)报错
- C 插入语句(3)报错
- D 插入语句(3)报错
- E 均不报错

练习五：建表时，可不可以使用联合自增主键？

```
(1) create table user(  
    id int auto_increment,  
    name varchar(10) not null,
```

```
primary key(name, id)
)engine=innodb;
(2) insert into user(name) values('shenjian');
(3) insert into user(id, name) values(10,'shenjian');
(4) insert into user(name) values('shenjian');
```

画外音：建表时，声明联合主键(a,b)，并且有一个是自增ID，插入若干元素，包含自增ID，有些不包含。

提问，连续执行上面的语句，执行结果是：

- A 建表语句(1)报错
- B 插入语句(2)报错
- C 插入语句(3)报错
- D 插入语句(3)报错
- E 均不报错

你的答案是什么，真的掌握透彻了么？

相关文章：

《索引，一文搞定》

[阅读原文](#)

喜欢此内容的人还喜欢

我是产品，你猜，我喜欢和怎样的研发合作？

架构师之路

香港电影，这次要一雪前耻了

独立鱼电影

那些跨出舒适圈的人，在圈外遇见了什么？

周源远