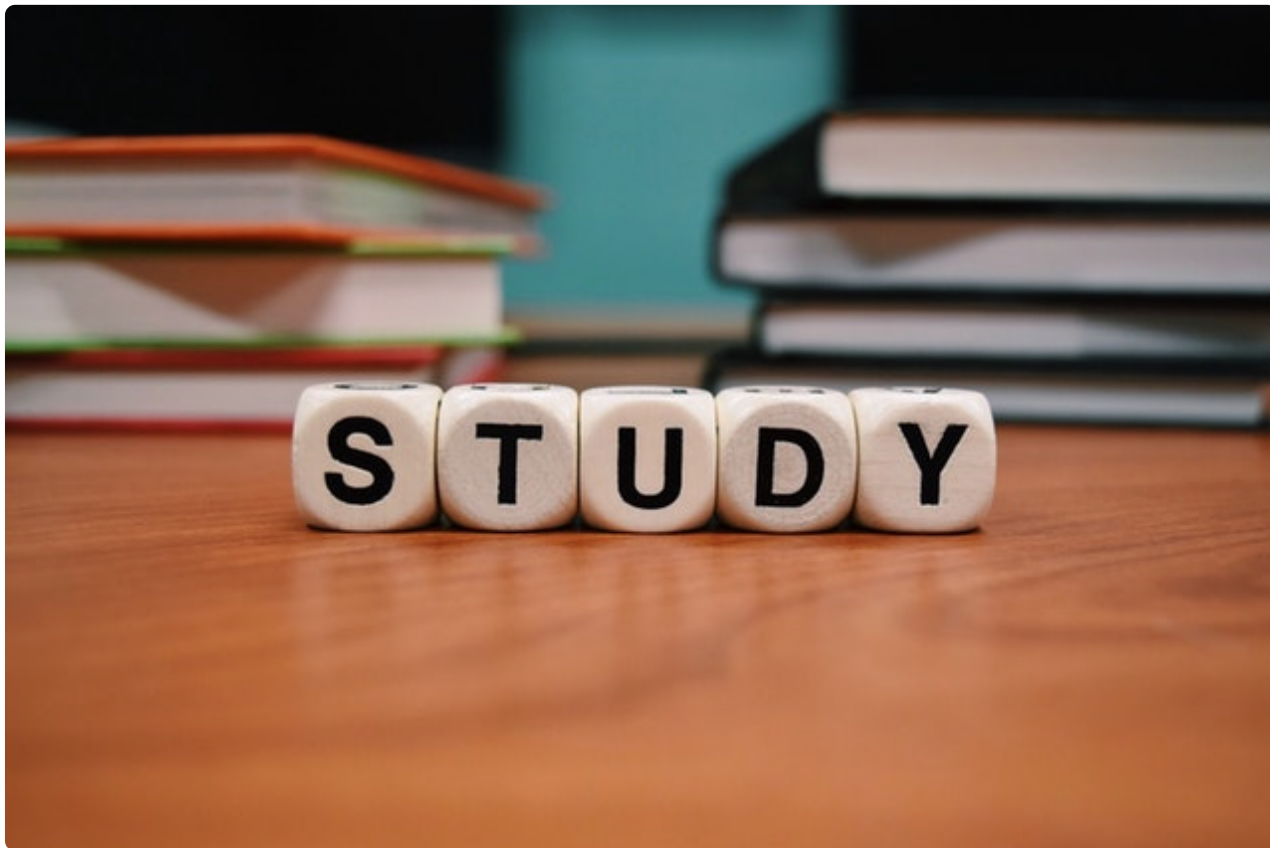


## 01 开篇词-你为什么要学习 MySQL?

更新时间: 2020-03-09 11:03:35



世上无难事,只要肯登攀。——毛泽东

你好,我是勤一,目前是一名后端研发工程师。我基本上每天都会和 **MySQL** 打交道,学习和积累了很多关于 **MySQL** 的知识。

首先,欢迎你来到我的专栏,在这个专栏里我会系统的带你学习 **MySQL**,由浅入深的带你学习 **MySQL** 的重点知识,并教会你如何提升 **MySQL** 的性能等等。那么,你可能会问:为什么我要系统的学习 **MySQL** 呢?

### 为什么我要系统的学习 **MySQL** ?

**MySQL** 的简单使用对我们的日常工作来说,肯定都不会陌生,甚至对于非技术岗的同学,也需要点基础的 **SQL** 知识,用来查询报表数据。但是在大多数的场景下,我们的 **SQL** 语句不会很复杂。**MySQL** 优化,服务器性能问题,**SQL** 查询优化等等这些不都是 **DBA** 应该操心的吗,跟我好像没什么关系啊?那我还有必要系统的学一遍 **MySQL** 吗?

答案是:有,身为一名开发工程师,你还真得好好学学 **MySQL** 这个东西。

我想在日常的开发中,你一定听过重构、优化等等"高大上"的名词。它们指得是什么呢?从字面意思上看就是代码的重构和优化。那代码为什么需要优化?

一个原因:性能太差。那到底是什么原因在影响代码的性能呢?简单来说,是一个叫 **IO** 的东西,**IO** 一般有两种:

- 网络 IO：它是说代码中存在网络调用，耗时的点主要存在于：服务器（响应、带宽限制、路由跳转延迟等等，一般为几十到几千毫秒，受环境的干扰比较大）。
- 磁盘 IO：它是说对磁盘的随机读写，而这里的随机读写通常都是数据库（例如 MySQL）的读写造成的。

而我们所说的优化就是针对这两种 IO 进行优化，但是由于网络 IO 受环境等不可预测的因素限制，可以做的优化比较有限。所以，优化的目标就集中在了磁盘 IO 的数据库读写上面。因此，系统的学习 MySQL，掌握高性能的使用与优化方法也就理所当然了。

另外，对于做后端的同学来说，你在面试中也已定遇见过 MySQL 相关的题目吧？面试官不会问你 MySQL 的 CRUD 是怎么做的。但是一定会问你：索引是怎么实现的、怎么对慢查询做优化、分库分表怎么实现等等诸如此类的问题。所以，系统的掌握 MySQL 在面试中尤其重要。

现在你还觉得系统的学一遍 MySQL 有必要吗？

## 我该如何系统的学习 MySQL 呢？

既然要学习 MySQL，那该怎么学？这是一个问题。

多年以前，我刚开始学习 Linux 的时候，脑子里也有这样的疑问。我想，该怎样学习 XXX 技术，这是一个所有程序员都绕不过去的经典问题。

怎样去学习并学好一门技术，你去网上搜一下会发现一大堆不靠谱的答案。看技术视频？时间有些不够，大部分已经工作的学员可能并没有成块的时间去看视频学习。看源码？更不靠谱，怎么看、从哪看、看完之后收获在哪这一系列问题还是没有答案。对于 MySQL 也是如此，这“家伙”也非常的庞大，用途也非常广，想要系统的学习掌握谈何容易。

我本人学习与使用 MySQL 多年，从 MySQL5.5 到 8.0 都有一定的使用、优化经验，这里我就从我的角度出发，谈一谈应该怎样系统的学习 MySQL。我将这个过程分为三个阶段：

**基础运用阶段：**这一阶段要去学习 MySQL 的理论知识，例如：常用的数据类型和适用场景、Schema 的设计规范等等。之后，以理论去指导实践，简单的用一用 MySQL，能理解基本的增删改查、聚合分组概念等等。同时，也要求能够自行去解决一些简单的问题，例如：SQL 语法错误、键冲突等等。

**能力提升阶段：**在能够熟练的应用之后，可以去看一些进阶知识点以及 MySQL 高级特性，例如：事务、锁并发、日志系统、索引优化等等。听起来，似乎这一阶段都是理论相关的。但是，也是理论指导实践的最佳诠释。在学习理论的过程中，你会惊讶的发现：原来 MySQL 还可以这样用、我之前的方法太低效了等等意想不到的收获。此时，你就可以用“新的知识”优化、重构之前的工作，以提升性能。同时，你的能力也在逐步提升。

**总结输出阶段：**在工作中，解决各种错误、性能问题的同时，一定要注意多做笔记，多做总结。并在这个过程中，将自己懂得的知识点、技巧、优化方案等等输出出去，帮助其他同学共同进步。正所谓，赠人玫瑰，手有余香。

为了能更好的帮助你渡过这三个阶段，我准备了这本专栏。我相信你在学习了这本专栏之后会在 MySQL 的道路上越走越远，并且收获满满！

下面我们一起来看下这本专栏是如何设计的：

## 本专栏的内容安排

本专栏一共设计了 7 章内容，涵盖了丰富的知识点，难度由浅入深，适用于不同能力水平的同学。这里，我将对这 7 章内容的安排进行说明。

- **MySQL 基础**：涵盖常用数据类型与使用建议、聚合与分组聚合、条件和系统函数等等知识点，旨在说明常用但是常常被忽略的知识点与技巧
- **MySQL 进阶**：对事务、锁并发、死锁、索引优化等等知识点进行讲解，旨在提升对 MySQL 的使用技能
- **MySQL 高级特性**：包含日志系统、视图、分区表、存储过程等等知识点的分析讲解，旨在更好的发挥 MySQL 的潜能
- **应用设计技巧与调优**：对 Binlog、SQL 查询优化、慢查询优化、服务器性能调优等等知识点进行讲解说明，旨在优化使用提升性能
- **MySQL 的实现原理**：分析说明 MySQL 的系统逻辑架构、SQL 解析 / 查询优化器的实现原理等等，学习这些能够对 MySQL 的底层与实现有所了解
- **实践应用**：包含两个系统的数据表完整设计，从业务思想出发，为解决问题设计了高可用的数据表结构，并对服务器及驱动进行合理配置
- **总结与思考**：包含总结、展望与思考三个方面，指出了重点内容、继续学习的方向指导，并留下了开放性的思考问题

## 学习之前的建议与提醒

最后，如果你想购买本专栏，你需要知道这些：

- MySQL 的安装、基本的增删改查、排序等语法是需要你事先理解掌握的
- 本专栏使用的 MySQL 版本是 5.7.28，使用 InnoDB 存储引擎。但是，MySQL 各个版本在语法、底层实现上差别不大，故本专栏的知识点适用于 MySQL 的常见版本（5.5、5.6、5.7）
- 对于专栏中留出的思考题，可以留言到问答区，我会回复解答
- 如果专栏中有不正确的地方，欢迎在留言区指正

}