

第十二周

2024.3.4

- 计划

-

- 做了

- 1. 实现map更新数据库，可直接传""，无需增加额外代码，且不影响业务。

- 遇到的问题(未解决)

- 如果我想要int数字类型的字段改为空怎么办

- 笔记

- 室内导航思路：beacon

【淘宝】<https://m.tb.cn/h.5FmpFdN8ibLrC4n?tk=c18PWOL0eR1 CZ0016> 「蓝牙打卡
蓝牙信标ibeacon定位NFC基站M1模块巡更考勤BLE室外B2-D」
点击链接直接打开 或者 淘宝搜索直接打开

- 保护高并发服务稳定主要有三把利器：缓存、降级和限流、熔断

- [拒绝宕机！一文详解分布式限流方案（附代码实现）\(qq.com\)](#)

- [服务降级、熔断、限流的区别 限流和降级区别-CSDN博客](#)

- 内部类应用外部类实例导致的内存泄漏：[Lambda 表达式到底会不会造成内存泄漏？\(qq.com\)](#)

- 浏览器查询json文件ctrl+shfit+f精准查找

- mysql树结构:

- [优雅的对树形结构进行高性能分页，闭包表才是yyds\(qq.com\)](#)

- [MySQL闭包表-CSDN博客](#)

- 队列应用：[多人同时导出 Excel 干崩服务器？我们来实现一个排队导出功能！\(qq.com\)](#)

- localdatetime的使用（与字符串的转换、计算、调整月份为第一天、转为其他类）：[Java8 以后的 LocalDateTime，你真的会用吗？\(qq.com\)](#)

- 字符集和编码的区别：

- [如何理解编码方式和字符集的区别？ - 白羽笑的回答 - 知乎](#)

- [java中文乱码浅析及解决方案\(qq.com\)](#)(编码不匹配补：字符缓存错误)

- 实现redis集群稳定高性能：[从0到1构建一个稳定、高性能Redis集群\(qq.com\)](#)

- 计数器的使用:

- 应用于高并发限流处理手段：[有什么解决高并发问题的思路？ - 知乎\(zhihu.com\)](#)

- [高并发下的计数器，为什么阿里不推荐使用 AtomicLong？\(qq.com\)](#)

- ssh用cmd远程连接时，会要输入密码

- 反射获取tag:

- [golang面试题：reflect（反射包）如何获取字段tag？为什么json包不能导出私有变量的tag？_golang面试题 反射tag-CSDN博客](#)

2024.3.5

- 计划
 - 改善导出功能(导入也可能会崩溃)
- 做了
 1. 解决excel文件下载不到本地问题(需要自己额外创建出最底下的文件夹)
 2. 学习rebbitmqq使用与原理
 - a. [消息队列 RabbitMQ 版 Go SDK -SDK 文档-文档中心-腾讯云 \(tencent.com\)](#)
 - b. [Go 学习笔记 \(57\) — Go 第三方库之 amqp \(RabbitMQ 生产者、消费者整个流程\)_go amqp-CSDN博客](#)
- 遇到的问题
- 笔记
 - orm遇到null/0的处理：
 - [当gorm更新的字段内容是0值时，不会更新的一种解决办法_gorm int零值处理-CSDN博客](#)
 - [gorm 更新数据时，0值会被忽略 - 立志做一个好的程序员 - 博客园 \(cnblogs.com\)](#)
 - [mybatis-plus update更新操作的三种方式 mybatisplus update\(\)用法-CSDN博客](#)
 - Java各个数据类型默认值：[java各数据类型的变量的默认值 java model不初始化的默认值是多少-CSDN博客](#)
 - 安卓开发：BV1g741157pD
 - 快速生成安卓/其他页面前端，更适合静态页面：
 - [前端代码自动生成器 前端代码生成器-CSDN博客](#)
 - [\(5 封私信 / 79 条消息\) 如何快速地开发一款 Android App? - 知乎 \(zhihu.com\)](#)
 - - [CodeFun](#)：这是一款UI设计稿智能生成源代码的工具，支持微信小程序、移动端H5和混合APP。您只需上传Sketch、PSD等设计稿，CodeFun会智能地生成可维护的前端代码123。
 - [PxCook](#)：这是一个高效易用的自动标注工具，可以生成前端代码。您可以使用它来提高工作效率，特别是在设计和开发协作方面4。
 - [Android Studio](#)：如果您想自己编写安卓应用程序，Android Studio是一个强大的工具。您可以使用Kotlin和Jetpack Compose来自定义您的应用程序。这需要一些学习成本，但它是一个非常灵活和强大的选项5。
 - goexcel：
 - [工作簿 · Excelize 简体字文档 \(xuri.me\)](#)
 - [Golang 生成 Excel 文档 - 掘金 \(juejin.cn\)](#)

2024.3.6

- 计划
 -
- 做了
 1. 学习mysql的sql执行顺序、原理：limit底层、join底层、in底层。（尝试解决limit慢问题）
 2. 配置aiAssitend(顾问)
 3. 思考3.4号未解决的问题
- 遇到的问题
 - Unresolved type 'Command':

- [Goland编辑器 提示 Unresolved type xxxxxx类似的错误_goland unresolved type-CSDN博客](#)
- MySQL server version for the right syntax to use near '.id WHERE base_project_cfg.delete_time IS NULL' at line 15:
 - 删掉 `base_project_cfg.id` 中的 ``
- 笔记
 - goland保持2021.1以上, plugin中下载codegeex
 - Mybatis-plus是批量更新所有字段, 还是只更新部分字段?
 - 一个是要看entity的配置, 如果没有特殊配置, 那就是实体对象不为""和null的则更新(补充: 为什么建议实体类里不建议用int, 而是int包装类, int为0也会被更新, 然后默认值就是0)
 - sql语句分页速度问题:
 - 回表的介绍: [为啥有些limit那么慢? limit 10000\g-CSDN博客](#)
 - [limit慢查询的原理和优化 为什么子查询能优化limit-CSDN博客](#)
 - [limit 10 offset 1000000为什么会很慢_presto offset 0 limit 10-CSDN博客](#)
 - [16、使用limit offset 分页时, 为什么越往后翻越慢? 如何解决? - 追风的小蚂蚁 - 博客园](#)
 - [mysql limit分页 \(偏移量\) 过大时优化问题 limit后面的范围非常大-CSDN博客](#)
 - 子查询和join时间复杂度:
 - [sql查询--比较left join、join和子查询的性能 left join和子查询效率对比-CSDN博客](#)
 - 结论: [MySQL关联查询 join 性能好, 还是in一下更快呢?\(qq.com\)](#)
 - 另外: 关于join的实现, 一般有两种, 分别是索引嵌套循环连接、hash连接。
 - 小结:

因为limit是一行一行匹配的, 然后匹配到对应的位移数后, 再将前面的位移数全部排出, 然后再利用获得的索引id, 进行回表查询数据, 如果我们只是去获取id, 那么就可以节省一个回表的操作;

另外, 因为in速度比join慢, 所以就自然结合上述的两个优点一起了, 但是实际上我们只是减少了回表的次数, 不能说节省了回表。(完美.....emmm)

2024.3.7

- 计划
 - 测试java实体类为int的, 如果是0, 是否会进行更新操作
 - 如果传的字段没有改变, 那也会更新吧:
 - [使用Mybatis-plus内置方法,更新一个字段还是全部字段, 三种应对情况下性能效率比较_mybatisplus更新一个字段-CSDN博客](#)
 - [MyBatis Plus 遇到的坑, update只能批量更新所有字段, 无法实现只更新部分字段-CSDN博客](#)
- 做了
 1. 学习并使用explain关键字、show warnings、执行时间含义, **有助于优化sql语句**
 2. 批量添加数据
 - a. 利用apifox批量请求
 - b. 利用协程(不是,为什么生成uuid不能放协程里?)
 - i. [Golang中用==判断是否相等 - elimsc - 博客园\(cnblogs.com\)](#)
 - ii. [Go 语言函数引用传递值 | 菜鸟教程\(runoob.com\)](#)

iii. 协程的使用示例: [Golang协程详解和应用 - 知乎 \(zhihu.com\)](#)

c. 自己写个不用协程的

- 遇到的问题

- mysql执行问题->优化表达=判断是否遇到派生表, Every derived table must have its own alias

- 笔记

- mysql查看sql执行详情:

- EXPLAIN解释:

- 只能用于select语句, 另外: 在database中执行=EXPLAIN + show warnings两个命令一起看了

- EXPLAIN 字段详细解读: [MySQL优化从执行计划开始\(explain超详细\) - 知乎 \(zhihu.com\)](#)

- sql执行顺序、语句结构, EXPLAIN 简单介绍、show warnings简单介绍:

- [【MySQL优化】——看懂explain-CSDN博客](#)

- EXPLAIN和show warnings示例: [sql查询--比较left join、join和子查询的性能 left join和子查询效率对比-CSDN博客](#)

- EXPLAIN示例: [不要用 in + 子查询! \(qq.com\)](#)

- (所以具体字段对应到sql语句是什么意思呢呢???)

- show warnings: 查看具体sql优化后的语句(即实际执行语句)

- 执行时间解读:

- execution(执行阶段):

- ①在这里, MySQL 数据库引擎选择的索引、连接算法、访问表格的方式等信息, 以使查询能够以最有效的方式执行。
 - ②数据库执行SQL查询, 确定了符合条件的数据集, 但这些数据还没有被传送给用户。这个阶段主要是为了**定位和准备数据**。

- Fetching (获取阶段/数据检索阶段):

- ①从数据库中读取数据所花费的时间(有的说法说还和网络有关系,如果这个时间长,那需要检查网络原因),
 - ②在数据检索阶段, MySQL 根据执行计划的指示从表格中检索所需的数据, 并将其返回给用户。
 - ③在执行阶段完成后, 数据库开始从其存储中检索实际的数据, 并将它们传送给用户(用户可以是一个应用程序、一个脚本、或者直接是通过数据库查询与数据库交互的终端用户, 如cmd、idea等等)。这是**数据库将所选数据发送到用户端的过程**。

- sec就是秒的意思, 不足1s就是0 sec

- 补充: [MySQL查询结果中Duration Time和Fetch Time的区别 - 无敌师爷IT技术 Blog - 博客园 \(cnblogs.com\)](#)

- -- 查看BUFFER情况

```
select index_name, count(*)
from information_schema.INNODB_BUFFER_PAGE
where INDEX_NAME in ('val', 'primary')
and TABLE_NAME like '%base_project_cfg%'
group by index_name;
```

- 知网下载: <https://hotfou.com/7763.html>

2024.3.8

- 计划
 -
- 做了
 - 1.
- 遇到的问题
- 笔记
 - 提取不同文件的字符内容：
 - [TIKA - 内容提取\(Content Extraction\) 学习Apache Tika|WIKI教程\(iowiki.com\)](#)
 - [SpringBoot + Apache tika 轻松实现各种文档内容解析\(qq.com\)](#)

周小结

关键字：较优雅的实现更新空字符串、go-excel的小bug、rebbitm的学习、mysql的深入(explain关键字、show warnings命令)

- 做了
 - 主：
 - 实现了更新空字符串，且无额外编写函数（前面的版本使用了反射）
 - 学习了rebbitm的在go使用
 - 学习并深入使用mysql命令，改善分页查询速度(尝试协程添加，但是uuid不能在协程里，思考：多用户访问，是否导致uuid相同?会造成什么后果?)
 - 解决go-excel无法导入问题(需要等待，建议手动添加最底下的文件夹)
 - 次：
 - 学习了安卓、前端开发技巧
 - 学习了其他技术，如:redis集群稳定高性能
- 计划
 - aop、rpc的学习