

数据库作业二说明

一个简单的Mobile-DB命令行小工具😋

Java 8 Hibernate 5.02 license MIT

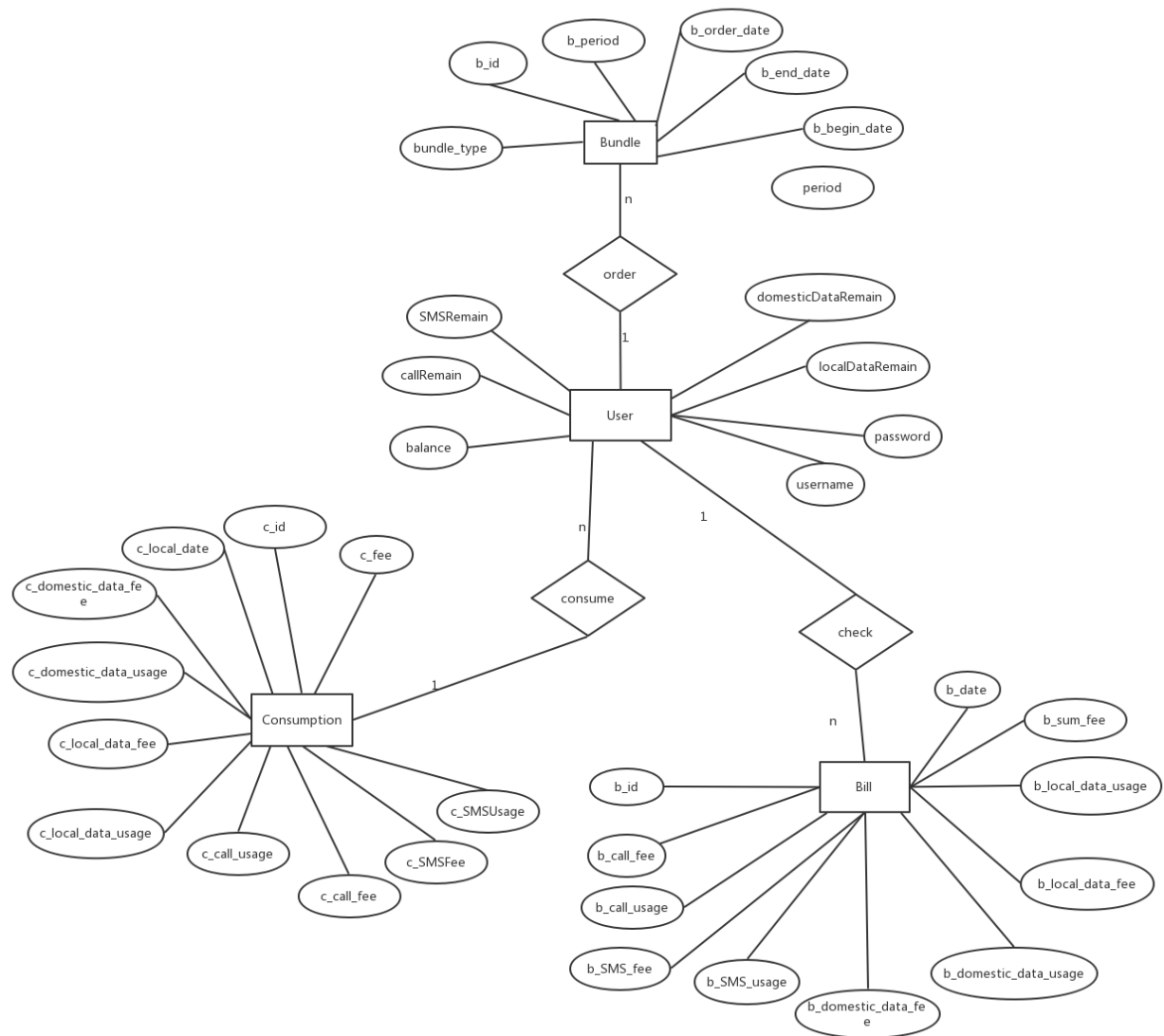
161250096 潘羽 [项目地址](#)

十分感谢您检查作业，助教小哥哥小姐姐辛苦了! 🙏🙏🙏

- [数据库作业二说明](#)
 - [一个简单的Mobile-DB命令行小工具😋](#)
 - [ER图](#)
 - [需求](#)
 - [额外支持](#)
 - [配置说明](#)
 - [设计说明](#)
 - [领域类模型](#)
 - [系统构建](#)
 - [代码统计](#)
 - [一些操作和实现](#)
 - [优化思路](#)
 - [库表设计](#)
 - [关乎代码](#)
 - [操作及相关命令](#)
 - [增加用户](#)
 - [查询套餐](#)
 - [订购套餐](#)
 - [订购套餐并立即生效](#)
 - [订购套餐并次月生效](#)
 - [退订套餐并立即生效](#)
 - [退订套餐并次月生效](#)
 - [资费生成](#)
 - [通话](#)
 - [短信](#)
 - [本地流量](#)
 - [国内流量](#)
 - [生成账单](#)

- [额外支持](#)
 - [充值](#)
 - [导出账单为txt文件](#)
- [Demo](#)
 - [增加用户](#)
 - [查询套餐](#)
 - [订购套餐\(立即生效\)](#)
 - [订购套餐\(次月生效\)](#)
 - [移除套餐\(立即生效\)](#)
 - [移除套餐\(次月生效\)](#)
- [中间过程](#)
 - [流程](#)
 - [命令执行段](#)
 - [实时电话费用生成](#)
 - [实时短信费用生成](#)
 - [实时本地流量费用生成](#)
 - [实时国内流量费用生成](#)
 - [月账单生成](#)
 - [导出账单](#)

ER图



需求

- 对某个用户进行套餐的查询(包括历史记录)、订购、退订(考虑立即生效和次月生效)操作 ✓
- 某个用户在通话情况下的资费生成 ✓
- 某个用户在使用流量情况下的资费生成 ✓
- 某个用户月账单的生成 ✓

额外支持

- 用户充值
- 用户导出账单

配置说明

- 运行DB2.sql脚本即可初始化数据库表和数据
- 数据库用户名root，密码1212hijkl，如果不符合可以在项目目录下的hibernate.cfg.xml文件修改用户名，密码

```
jdbc:mysql://localhost:3306/DB2?useUnicode=true&characterEnco
<!-- 数据库用户名 -->
<property name = "connection.username">root</property>

<!-- 密码（这里为空） -->
<property name = "connection.password">1212hjkl</property>

<!-- 数据库使用的方言 -->
```

设计说明

领域类模型

共有4个实体类:

- User: 用户实体类
- Budnle: 套餐实体类
- Consumption: 实时记录用户消耗的实体类
- Bill: 账单实体类

系统构建

1. Java8 + hibernate5.2 + MySQL 5.7.19 (for macos10.12 (x86_64))
2. 运行环境Mac OS X 16G
3. 所有实体类的构建依赖于util包下面的HQL工具类，支持CURD，包括单个和批量的增删改操作，语句精简到一句话，易用性强😁
4. 运行Main类的main方法后，只需命令行操作即可，emmm如果需要详细到每个操作的调用，请移步util包下的CommandLineTool

代码统计

```
py:~ py$ cd /Users/py/GitHub/未命名文件夹\ 2/db2/2/src
py:src py$ cloc ./
  23 text files.
  23 unique files.
   1 file ignored.

github.com/AlDanial/cloc v 1.74  T=0.12 s (180.8 files/s, 16448.8 lines/s)
-----
Language                  files      blank      comment      code
-----
Java                      22         334         218         1449
-----
SUM:                      22         334         218         1449
-----
py:src py$
```

一些操作和实现

- 用户套餐是每月初进行一次初始化，扣除用户余额并初始化用户的数据，挑选的时间是在夜深人静的凌晨...由util包下的TaskUtil完成，调用ScheduledExecutorService，由于初始化时候是先查询用户的套餐列表再更新数据所以不会产生问题
- 用户的次月生效对于订购时不更新用户除套餐外的数据，下个月月初初始化即可，对退订则是将套餐截止日期end_date改为本月最后一天
- 用户退订套餐我没有折算退费，因为我的套餐退订都没有退过费用💔，所有退订的套餐会将截止日期endDate设为前一天，方便查阅
- 用户的套餐叠加出现的情况，如果用户超出套餐使用，会扣费，如果此时叠加套餐，扣费不会返回

优化思路

库表设计

1. 注意到其实在整个过程中，更加侧重的是插入更新过程而非查询，而如刘嘉老师所说“用户也大多关注查询近期不会注意很久之前的记录”，进一步做下去一个考虑提高，我会考虑在高峰时候做个临时表存储以及所说的分批进行，逐次提交。

例如对于月底账单生成，月底用户查询账单的高峰会考虑临时抽出consumption这张表的数据，建个临时表，完成查询和插入后再放回去。

关乎代码

1. 其实能提升的最可见的方面是封装的HQLUtil，这个工具类是我去年软工二大作业封装的，简单好用，但其实它在一定程度上放弃了二级缓存(基于我每一次新的操作会开启一个新的session)，在demo上不会看出，但数据量巨大时候会有区别。可改进的方法是设法引入记录缓存的变量，模拟二级session。不过作为一个玩具在日常作业和小项目方面还是基本令人满意😊
2. 因为我也不知道大公司具体是如何处理定时高并发的操作，就写了个简单的定时服务，还需要多加学习

操作及相关命令

增加用户

```
add user [username] [password]
e.g add user 陈振宇 123456
```

对应代码段: dao.UserDAO.addUser(String username, String password)方法

查询套餐

```
search buddle [username]

e.g: search buddle 陈振宇
```

对应代码段: dao.BuddleDAO.listBundleByUsername(String username)

订购套餐

订购套餐并立即生效

```
order imm [buddleType] [username] [period]
```

e.g: order imm call 陈振宇 2
即用户陈振宇订购电话套餐2月并立即生效

对应代码段: dao.BuddleDAO.addBundleImmediately(BundleType bundleType, String username, int period)

订购套餐并次月生效

```
order next [buddleType] [username] [period]
```

e.g: order next sms 陈振宇 3
即用户陈振宇订购短信套餐3月并次月生效

对应代码段: dao.BuddleDAO.addBundleNextMonth(BundleType bundleType, String username, int period)

规定套餐类型在命令行中的输入

- call 电话套餐
- sms 短信套餐
- local 本地流量套餐
- dome 国内流量套餐

退订套餐并立即生效

```
deorder imm [username] [buddleId]
```

e.g: deorder imm [刘嘉] [4]
即用户刘嘉立即退订id为4的套餐

对应代码段: dao.BuddleDAO.removeBundleImmediately(String username, int buddleId)

退订套餐并次月生效

```
deorder next [username] [buddleId]
```

e.g: deorder next [刘嘉] [5]
即用户刘嘉退订id为4的套餐，次月生效

对应代码段: dao.BuddleDAO.removeBundleUntilNextMonth(String username, int buddleId)

资费生成

通话

```
fee call [username] [callUsage]
```

e.g: fee call 陈振宇 10

即用户陈振宇打了10分钟电话，显示话费费用和套餐提示

对应代码段: dao.ConsumptionDAO.addCallUsage(String username, double callUsage)

短信

```
fee sms [username] [SMSAmount]
```

e.g: fee sms 刘嘉 20

即用户刘嘉一次群发了20条短信，显示短信费用和套餐提示

对应代码段: dao.ConsumptionDAO.addSMSUsage(String username, int SMSAmount)

本地流量

```
fee local [username] [localDataUsage]
```

e.g: fee local 陈振宇 500

即用户陈振宇一次使用了500M本地流量，显示流量费用和套餐提示

对应代码段: dao.ConsumptionDAO.addLocalDataUsage(String username, double localDataUsage)

国内流量

```
fee dome[username] [domesticDataUsage]
```

e.g: fee dome 刘嘉 800

即用户刘嘉一次使用了800M流量，显示流量费用及套餐提示

对应代码段: dao.ConsumptionDAO.addDomesticDataUsage(String username, double domesticDataUsage)

生成账单

```
bill [dateToMonth] [username]
```

e.g: bill 2018-10 陈振宇

即显示用户陈振宇在2018年10月的账单详细信息，包括电话、短信、流量的使用及套餐情况，账户话费及

对应代码段: dao.BillDAO.addBill(String date, String username)

额外支持

充值

```
topup [username] [amount]
```

e.g: topup 刘嘉 100
即刘嘉充值100

导出账单为txt文件

```
export [date_to_month] [username]
```

e.g: export 2018-10 陈振宇
即可导出用户陈振宇2018年10月的账单

Demo

增加用户

```
add user 潘羽 admin
```

```
请输入命令...
add user 潘羽 admin
增加用户耗时 37ms
```

查询套餐

```
search buddle 潘羽
```

```
请输入命令...
search buddle 潘羽
无套餐
查询套餐时间24ms...
...
请输入命令...
```

订购套餐(立即生效)


```
order imm call 潘羽 1
```

```
请输入命令...
order imm call 潘羽 1
您已成功订购套餐(立即生效)...
订购套餐(立即生效)花费时间21ms
...
请输入命令...
```

此时查询用户潘羽的套餐信息

```
search buddle 潘羽
```

```
请输入命令...
search buddle 潘羽
套餐id: 1 开始日期: 2018-10-25 套餐类型: 电话套餐 结束日期: 2018-11-25
查询套餐时间13ms...
...
请输入命令...
```

此时从当天(2018-10-25)订购一个月到2018-11-25

订购套餐(次月生效)

```
order next sms 潘羽 2
```

```
请输入命令...
order next sms 潘羽 2
您已成功订购套餐(次月生效)...
订购套餐(次月生效)花费时间13ms
...
请输入命令...
```

此

时继续查询用户潘羽的套餐信息

```
search buddle 潘羽
```

```
请输入命令...
search buddle 潘羽
套餐id: 2 开始日期: 2018-11-01 套餐类型: 短信套餐 结束日期: 2018-10-31
套餐id: 11 开始日期: 2018-11-01 套餐类型: 短信套餐 结束日期: 2019-01-01
查询套餐时间6ms...
...
请输入命令...
```

此时多了一个套餐是从下个月11-01开始到2019-01-01结束的

此时查看一下用户的账户余额

```
check 潘羽
```

```
请输入命令...
check 潘羽
余额为-20.0元
...
```

余额为初始0 - 20 - 0 = -20

移除套餐(立即生效)

```
deorder imm 潘羽 1
```

```
取消套餐(立即)
取消套餐(立即)花费时间3005ms
...
请输入命令...
```

查看用户潘羽此时只有套餐二

```
请输入命令...
search buddle 潘羽
套餐id: 2 开始日期: 2018-11-01 套餐类型: 短信套餐 结束日期: 2019-01-01
查询套餐时间6ms...
...
请输入命令...
```

移除套餐(次月生效)

```
deorder next 潘羽 2!
```

```
请输入命令...
deorder next 潘羽 2
取消套餐(次月生效)
取消套餐(次月生效)花费时间6ms
...
```

查看用户潘羽此时订单结束日期为2018-10-31

```
...
请输入命令...
search buddle 潘羽
套餐id: 2 开始日期: 2018-11-01 套餐类型: 短信套餐 结束日期: 2018-10-31
查询套餐时间5ms...
...
```

中间过程

为了便于看出用户套餐对结果结果，我们增加用户刘嘉和陈振宇😄

流程

1. 增加用户刘嘉，陈振宇，并为他们各充值1000元
2. 刘嘉订购电话套餐 * 1(1个月立即生效)，短信套餐 * 1(2个月立即生效)
3. 陈振宇订购本地流量套餐 * 1(2个月立即生效)，国内流量套餐 * 1 (3个月次月生效)
4. 刘嘉进行通话100分钟，一次群发了300条短信
5. 陈振宇使用本地流量100M，国际流量1G

命令执行段

```
add user 刘嘉 123456
add user 陈振宇 123456
topup 刘嘉 1000
topup 陈振宇 1000
order imm call 刘嘉 1
order imm sms 刘嘉 2
order imm local 陈振宇 1000
order next dome 陈振宇 3000
fee call 刘嘉 100
fee sms 刘嘉 300
fee local 陈振宇 100
fee dome 陈振宇 1000
bill 2018-10 刘嘉
bill 2018-10 陈振宇
```

实时电话费用生成

```
fee call 刘嘉 100
```

```
请输入命令...
fee call 刘嘉 100
本次套餐减免花费0元，套餐还剩0 分钟
本次通话记录耗时13ms
...
```

实时短信费用生成

```
fee sms 刘嘉 300
```

```
请输入命令...
fee sms 刘嘉 300
套餐短信量已用完...
本次短信发送 300 条, 花费 10.0 元
本次短信记录耗时13ms
...
请输入命令...
```

实时本地流量费用生成

```
fee local 陈振宇 100
```

```
请输入命令...
fee local 陈振宇 100
本次套餐减免花费0元, 套餐还剩1900.0M
本次本地流量记录耗时6ms
...
```

实时国内流量费用生成

```
fee dome 陈振宇 1000
```

```
fee dome 陈振宇 1000
国内流量套餐用完
本次国内流量花费 0.0M, 花费3000.0 元
本次国内流量记录耗时11ms
...
```

月账单生成

```
bill 2018-10 刘嘉
```

$1000 - 20 - 10 - (300 - 200) * 0.1 = 960$

请输入命令...

```
bill 2018-10 刘嘉
```

2018-10账单

用户名: 刘嘉

-----套餐统计-----

通话时间: 100.0分钟 短信条数: 300条 本地流量: 0.0M 国内流量: 0.0M

-----费用-----

通话费用: 0.0元 短信费用: 10.0元 本地流量费用: 0.0元 国内流量费用: 0.0元

-----合计-----

总话费: 10.0 余额: 960.0

-----套餐剩余-----

话费套餐: 0分钟 短信套餐: 0条

本地流量套餐: 0.0M 国际流量套餐: 0.0M

本次查询订单耗时19ms

...

```
bill 2018-10 陈振宇
```

请输入命令...

```
bill 2018-10 陈振宇
```

2018-10账单

用户名: 陈振宇

-----套餐统计-----

通话时间: 0.0分钟 短信条数: 0条 本地流量: 100.0M 国内流量: 1000.0M

-----费用-----

通话费用: 0.0元 短信费用: 0.0元 本地流量费用: 0.0元 国内流量费用: 3000.0元

-----合计-----

总话费: 3000.0 余额: -2020.0

-----套餐剩余-----

话费套餐: 0分钟 短信套餐: 0条

本地流量套餐: 1900.0M 国际流量套餐: 0.0M

本次查询订单耗时19ms

...

$1000 - 20 - 0 - (1000 - 0) * 3 = -2020$

😬 这种欠巨多的... 😬

导出账单

```
export 2018-10 陈振宇
```

在当前目录下即可生成"2018-10陈振宇账单.txt"文件