数据库事物

今日目标：  
1.了解事物的作用

2.掌握事物的四大特性（面试）

3.了解事物的三个读写问题

4掌握Mysql开启和结束事物

5了解事物的四个隔离级别

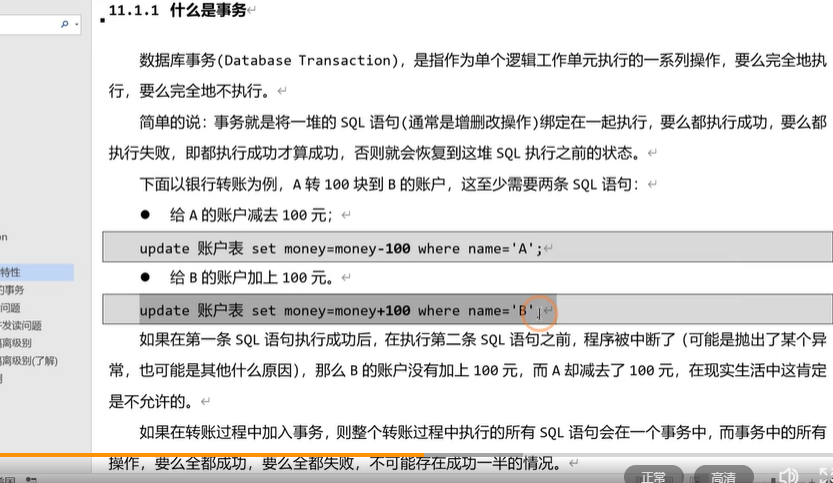
什么是事物

数据库事物是指作为单个里哦及单元执行的一系列的操作

要么完全地执行要么完全不执行

简单的来说事物就是将一堆的mysql语句（通常指的是增删改查的操作）绑定在一起

执行要么全部执行成功 要么全部执行失败即 执行成功才算是成功否则就会恢复到sql执行这些命令之前的状态



事物的四个特性

事物的四大特性ACID是指

原子性(Atomictty)：事物中所有的操作是不可再分割的原子单位事物中所有的操作

要么全部执行成功要么全部执行失败

一致性：事物执行后数据库的状态与其他业务的规则要保持一直如果站桩业务无论事物执行成功与否，参与转账的两个账户金额之和应该是不变的

A1000 1000-100=900 1000

B 1000 1000+100=1100 1000

隔离性

个零星是指再并发操作中不同事物之和应该隔离开来使每个并发的

事物是不会互相干扰的也就是说，再食物中查看数据更新时

数据所处的状态要么是另一事物修改它之前的状态

要么是另一事物修改它之后的状态事物是不会查看到中间状态的数据

例如再A事物中查看另一个B事物正在修改A的账户中A的账户金额要查看到B事物之前的A的账户金额要么查看到B事物之后A的账户金额

事物1 查询A、B 的账户金额之和

事物2 A转账给B 100元

A-100=900

B+100=1100

事物和事物之间完全隔离需要数据设置了较高的事务隔离级别

在数据库的事务级别足够高的情况下事物和事物之间是完全隔离的

在十五的隔离级别较低的情况下可能会出现在事物中能看到另外一个事物

正在进行中的事物的状态

持久性:

一旦事物提交成功事物中所有的数据操作都必须要被持久化

到数据库中即使提交事物中数据库拿上崩溃再数据库重启的时候

也必须要保证通过某种机制来恢复数据

开启事务------

A :1000 -100=900 在日志中记录事务提交之后将会更新A账户的金为900

B：1000+100=1100在日志中记录事务提交之后将会更信B账户的金额为1100

查询A、B账户的金额

结束事务----回滚、提交

Mysql中的事务

在默认的情况下 Mysql每执行一条SQL语句都是一个单独的事物如果需要套在

一个事物中包含多条MYSQL语句那么需要开启事物和结束事物

开始事物start transaction

结束事物 commit （提交事务）或rollback回滚事务

在执行SQL语句之前限制性start transaction 这就开启了一个事务

（事务的起点）然后就可以去执行多条SQL语句最后要结束事务

COMMIT 表示提交即事务中多了条SQL语句所作出的影响会

持久化到数据库中或者rollback表示回滚即回滚到事务的起点之前做的所有操作都被撤销！

下面演示A账户和B账户转账的例子

准备数据

---1.创建数据库jt\_db数据库（如果不存在才创建）

Create databases if not exists jt\_db charset utf-8;

Use jt\_db –选择jt\_ud 数据库

在jt\_db库由创建acc表(银行账户表)要求要有id（主键） name (姓名),money(账户金额)

Drop table if exists acc;

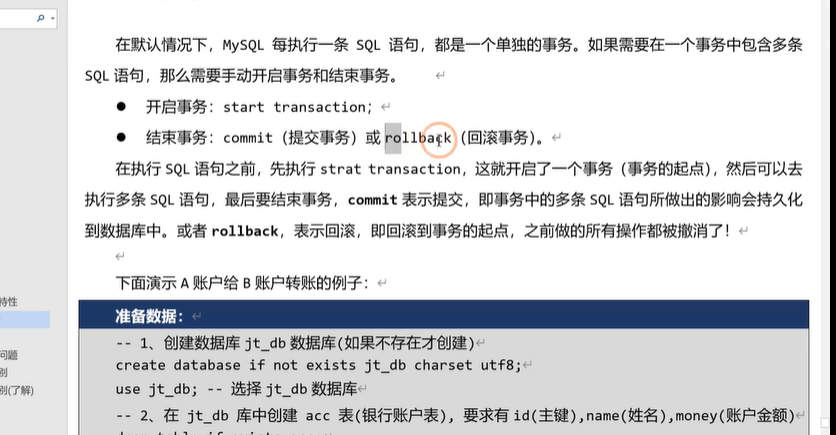
Create table acc(

Id int primary key auto\_increatment,

Name varchar(50)

Money double

)



Mybatis框架

Mybatis快速入门

Mybatis对数据库中的数据增删改查的操作

#{}占位符的应用

动态SQL的应用

Mybatis的Mapper接口开发

什么是Mybatis

Mybatis本是apache的一个开源项目iBatis 2010年这个项目由apache software

Foundation 迁移到了google code 并改名为mybatis 2013年11月迁移到Github

Mybatis是一个优秀的持久层框架它对jdbc的操作数据库的过程进行封装

使开发者只需要关注结果集检索等jdbc繁杂的过程代码

Mybatis通过xml或注解的方式将要执行的各种statmet(statement,pareparedStatemnt)

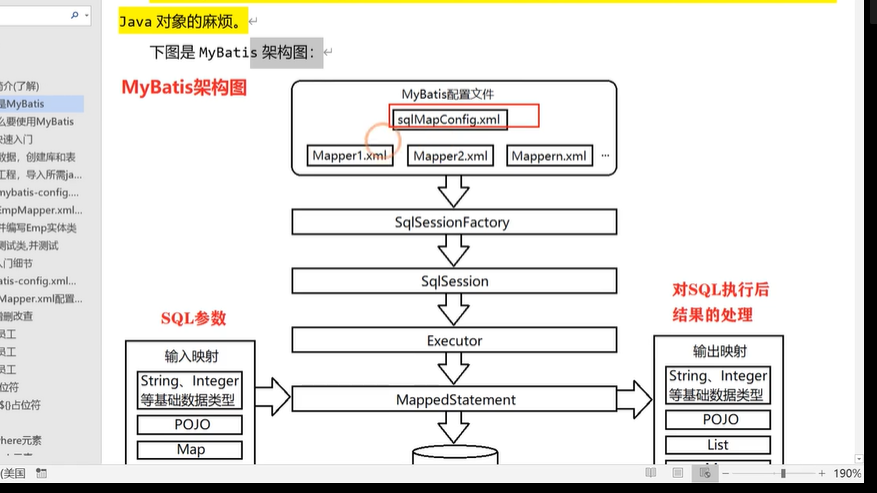
配置起来并通过java对象和statement中的sql进行映射最终执行的sql语句最后由

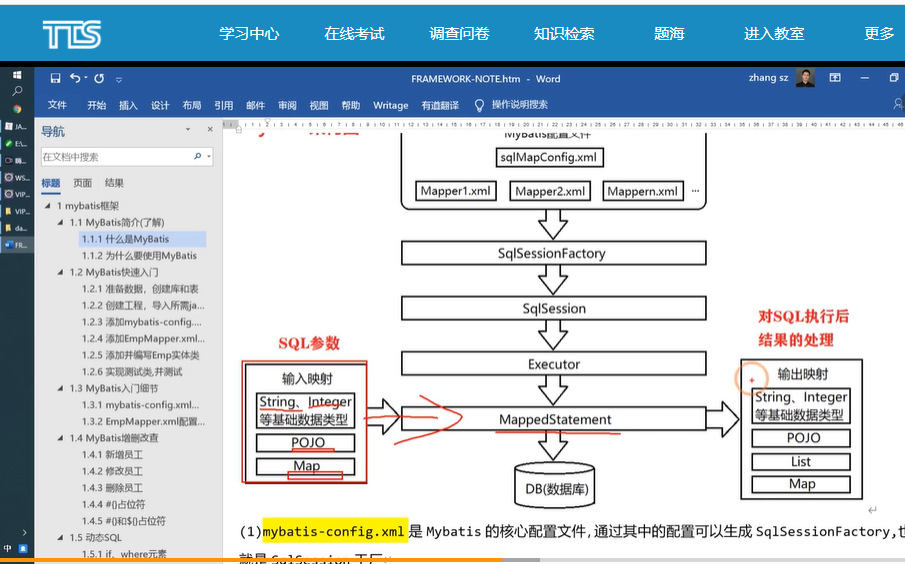
Mybatis框架执行sql并且将结果映射成java对象并且返回

总之Mybatis对JDBC访问数据的过程进行了封装

缉拿话了jdbc代码解决了jdbc将结果集封装为java对象的麻烦

下图是mybatis的架构图





为什么要使用mybits

使用JDBC 访问数据库有大量的重复代码（比如注册驱动获取链接 获取传输器

释放资源等）

JDBC自身是没有连接池的 会频繁的创建和关闭连接效率低下

Sql是写死在程序中的一旦修改SQL需要对类重新编译

对查询SQL职系那个后返回的的Result对象需要手动处理

有时会特别麻烦

使用Mybits框架访问数据库

Mybits对JDBC对了封装

可以简化JDBC代码

Mybits自身支持连接池也可以配置其他的连接池因此可以提高程序

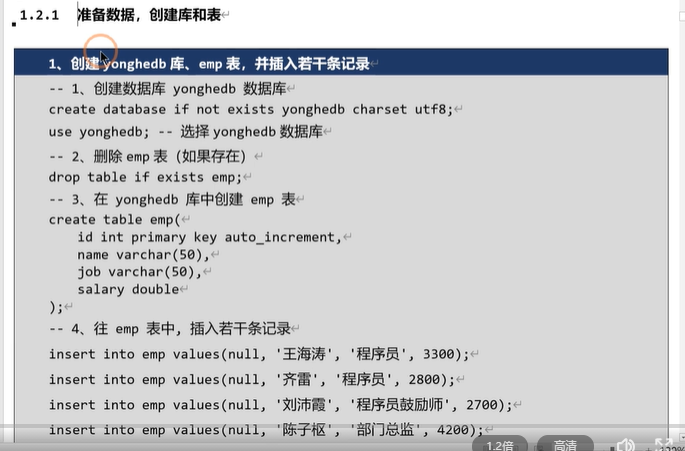
的效率

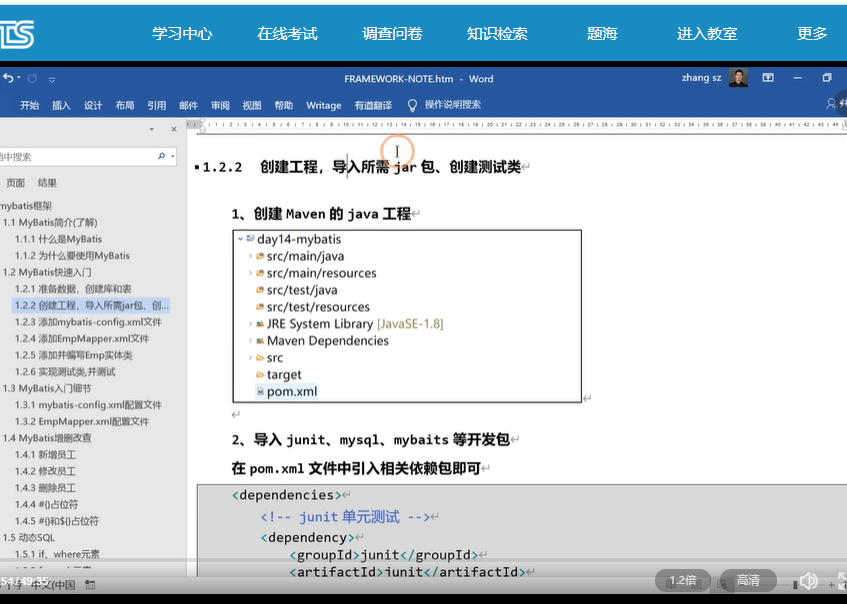
Mybits是将 SQL配置在mapper文件中修改SQL只是修改配置文件

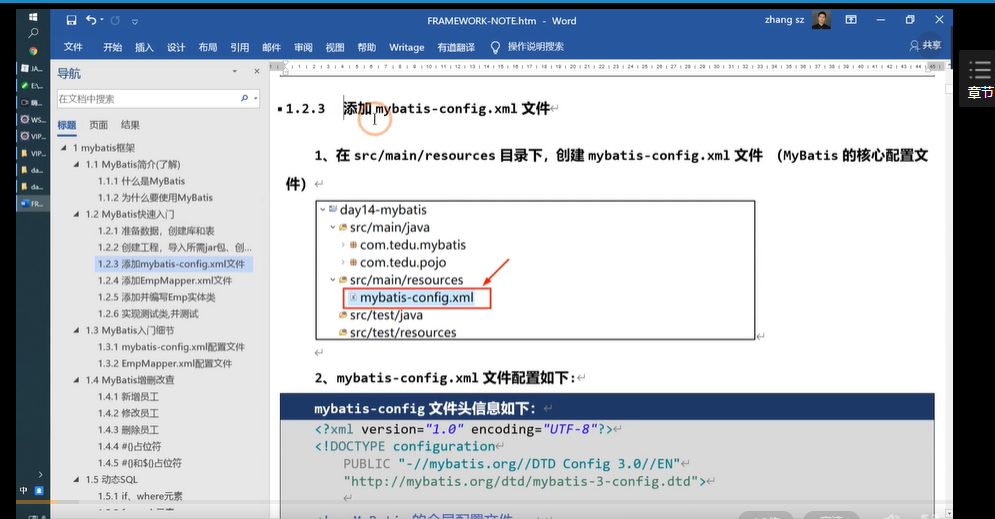
类不需要重新编译

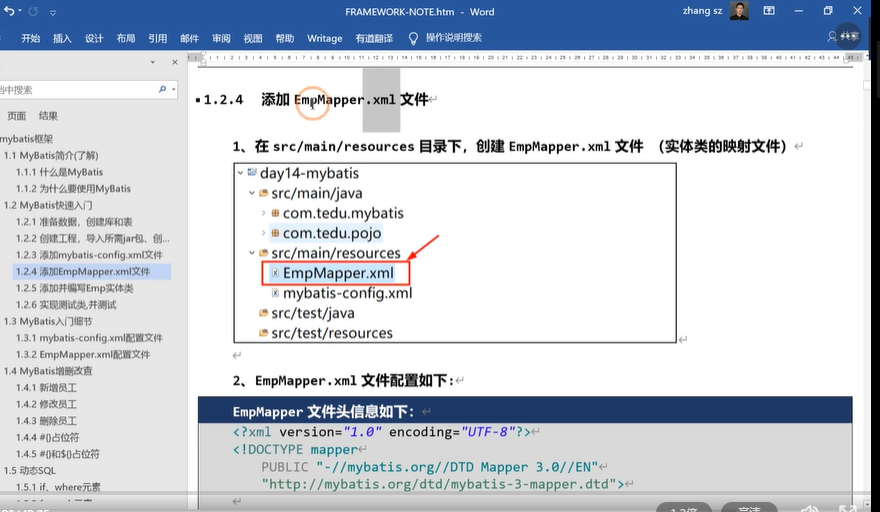
对查询SQL执行后返回的ResultSet对象Mybtis对帮我们处理转换成Java对象

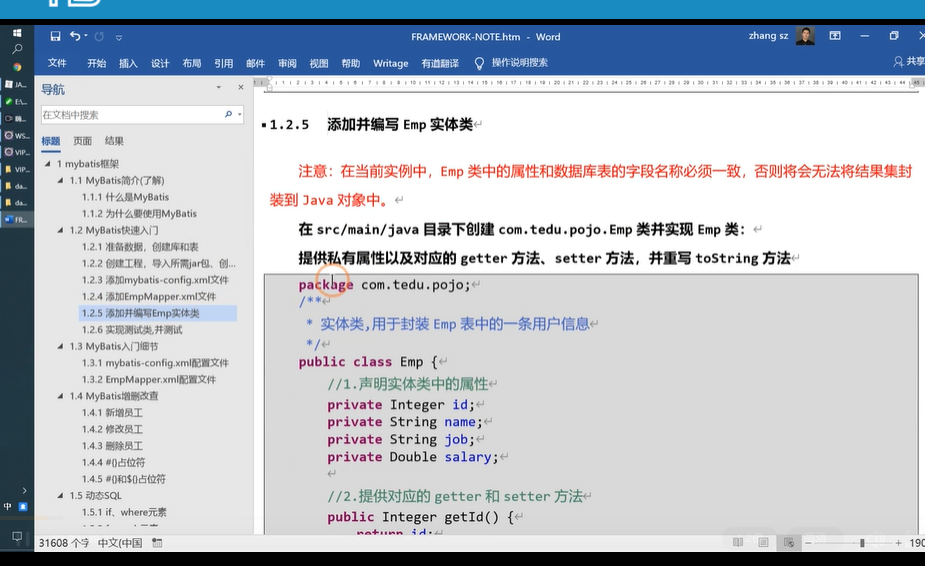
Mybitis快速人们

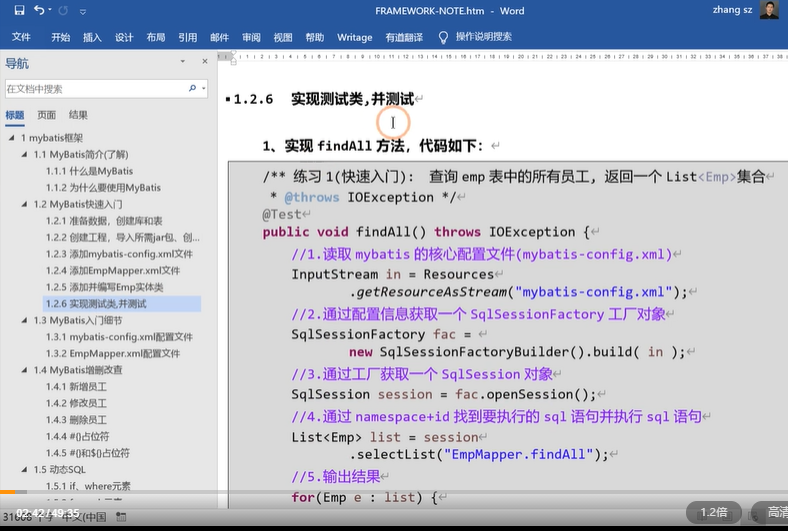












Mybaris增删改查

新增员工

编辑EmpMapper.xml文件，添加新增员工对应的SQL

联系2。新增员工信息：刘德华歌手8888

新增改的标签上不用指定ResuktType因为返回值都是Int类型

<update id=”insert”>

Insert into emp values(null.”刘德华”,“歌手”,8888)

</update>

辨析TestMybatis类添加TestInsert方法实现新增员工操作

联系2新增员工信刘德华 歌手8888

@Test

Public void testInsert（）{

//执行sql语句返回执行结果

}

Mybatis#{}占位符

在上面的增删改查 操作中SQL语的值是写死在SQLzhong 的在实际开发中此处

的值往往是用户提交过来的值因此我们需要将SQL中写死的值替换为占位符

使用占位符完全上卖弄的增删改查

#{}占位符相当于JDBC的问号占位符 是为SQL语句中的值进行展位如果

数值是自负床或者日期类型会进行转义处理

${}占位符是为SQL片段进行占位是将传过来的SQL片段直接凭借在SQL语句中

不会做任何的转义处理

练习5查询查询指定id的员工

Mapper文件配置

<select id=”findAll” resultType=”com.tedu.pojo.Emp”>

Select \* from emp where id=#{id}

</select>

Java代码实列

练习5 查询emp表中指定id的员工信息

@Test

Public void testFindById(){

//指定SQL语句，返回执行结果

Emp emp =sessipn.selectOne(

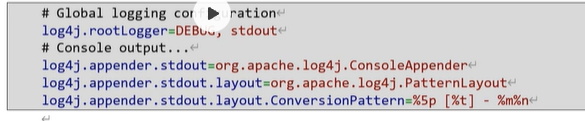
)

}

Mybatis占位符几个可以优化的地方

在项目中Log4j的配置文件用于打印日志的信息便于开发调试

在src或者相似的目录下创建log4j.properties如下



Mybatis默认使用log4j作为输出日志信息

只要将该文件放在指定的位置Log4j工具会自动 到指定的位置加载上述我呢见读取文件中的配置信息并使用!