## 第一题:

- (一)、动态sql是可以完成sql语句的动态组装,以适应多变的业务逻辑
- (二)、动态sql有:

<if>, <choose>, <when>, <otherwise>, <foreach>, <include>, <trim>, <set>, <br/>bind>

# (三)、原理:

通过对#{}的解析,将#{}使用?进行代替,并解析出#{}里面的值存储到list中, 设置PreparedStatement参数的时候根据上面的list获取实体中相应的属性进行赋值 第二题:

mybatis仅支持 association 关联对象和 collection 关联集合对象的延迟加载, association 指的就 是一对一, collection 指的就是一对多查询。原理就是使用CGLIB 创建目标对象的代理对象 就用目标方法的时候进入拦截器调用invoke方法,发现为null才执行事先保存好的关联。ql。 第三题:

mybatis有5种执行器,分别

是SimpleExecutor、BatchExecutor、ReuseExecutor、CachingExecutor,他们的区别是: SimpleExecutor是最简单的执行器,根据对应的sql直接执行即可,不会做一些额外的操作;拼接完SQL之后,直接交给 StatementHandler去执行。BatchExecutor通过批量操作来优化性能。通常需要注意的是批量更新操作,由于内部有缓存的实现,使用完成通记得调用flushStatements来清除缓存;ReuseExecutor每次开始读或写操作,优先从缓存中获取对应的 Statement 对象。如果不存在,才进行创建。执行完成后一个关闭设 Statement 对象;CachingExecutor支持二级缓存的 Executor 的实现类

### 第四题:

mybatis的一级缓存和二级缓存的底层数据结构都是HashMap,一级缓存是sqlsession级别的缓存,一级缓存只在当前sqlsession范围有效,二级缓存是mapps 发别的,同一个mapper儿级别下多个sqlsession可以共用二级缓存,执行插入、更新、删除都会是一级缓存和二级缓存失效。

mybatis的插件原理是在四大对象创建出来的时候,不是直接返回,而 是interceptorChain.pluginAll(parameterHandler);获得到所有的interceptor,调用 interceptor.plugin(target);返回target对象,插件为目标对象创建代理对,AOP我的的插件插件可以为四大对象代理处代理对象,代理对象可以上截到四大对象的每一个执行; 自定义插件需要实现mybatis插件接口interceptor,把实现类配置到mybatis配置文件的plugins。 >plugin中即可

#### 编程题:

