****一、简单题****

1、Mybatis动态sql是做什么的？都有哪些动态sql？简述一下动态sql的执行原理？

  Mybatis 动态 SQL ，可以让我们在 XML 映射文件内，以 XML 标签的形 式编写动态 SQL ，完成逻辑判断和动态拼接 SQL 的功能。

Mybatis 提供了 9 种动态 SQL 标签：<if/>、<choose/>、<when/>、<otherwise/>、<trim/>、<when/>、<set/>、<foreach/>、<bind/>。

其执行原理为，使用 OGNL 的表达式，从 SQL 参数对象中计算表达式的值,根据表达式的值动态拼接 SQL ，以此来完成动态 SQL 的功能。

1. Mybatis是否支持延迟加载？如果支持，它的实现原理是什么？

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。在Mybatis配置文件中，可以配置是否启用延迟加载lazyLoadingEnabled=true|false。

它的原理是创建目标对象的代理对象，当调用目标方法时，进入拦截器方法，比如调用a.getB().getName()，拦截器invoke()方法发现a.getB()是null值，那么就会单独发送事先保存好的查询关联B对象的sql，把B查询上来，然后调用a.setB(b)，于是a的对象b属性就有值了，接着完成a.getB().getName()方法的调用。这就是延迟加载的基本原理

1. Mybatis都有哪些Executor执行器？它们之间的区别是什么？

SimpleExecutor：每执行一次update或select，就开启一个Statement对象，用完立刻关闭Statement对象。

ReuseExecutor：执行update或select，以sql作为key查找Statement对象，存在就使用，不存在就创建，用完后，不关闭Statement对象，而是放置于Map内，供下一次使用。简言之，就是重复使用Statement对象。

BatchExecutor：执行update（没有select，JDBC批处理不支持select），将所有sql都添加到批处理中（addBatch()），等待统一执行（executeBatch()），它缓存了多个Statement对象，每个Statement对象都是addBatch()完毕后，等待逐一执行executeBatch()批处理。与JDBC批处理相同。

1. 简述下Mybatis的一级、二级缓存（分别从存储结构、范围、失效场景。三个方面来作答）？

一级和二级缓存都是hashMap结构保存数据

一级缓存是SqlSession级别的缓存。在操作数据库时需要构造 sqlSession对象，在对象中有一个数据结构（HashMap）用于存储缓存数据。不同的sqlSession之间的缓存数据区域（HashMap）是互相不影响的。

二级缓存是mapper级别的缓存，多个SqlSession去操作同一个Mapper的sql语句，多个SqlSession可以共用二级缓存，二级缓存是跨SqlSession的。

一级和二级，一般在执行完commit操作都要刷新缓存

5、简述Mybatis的插件运行原理，以及如何编写一个插件？

**mybatis可以编写针对Executor、StatementHandler、ParameterHandler、ResultSetHandler四个接口的插件，mybatis使用JDK的动态代理为需要拦截的接口生成代理对象，然后实现接口的拦截方法，所以当执行需要拦截的接口方法时，会进入拦截方法（AOP面向切面编程的思想）**

**1.编写Intercepror接口的实现类**

**2.设置插件的签名，告诉mybatis拦截哪个对象的哪个方法**

**3.最后将插件注册到全局配置文件中**

****二、编程题****

请完善自定义持久层框架IPersistence，在现有代码基础上添加、修改及删除功能。【需要采用getMapper方式】