

## 简答题部分

---

### 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？

答：动态 sql 是指在进行 sql 操作的时候，传入的参数对象或者参数值，根据匹配的条件，有可能需要动态的去判断是否为空，循环，进行 SQL 语句拼接的情况。

常用的标签：<if>、<foreach>、<where>、<set>、<choose>、<when>等

### 2、Mybatis是否支持延迟加载？如果支持，它的实现原理是什么？

答：Mybatis 仅支持 association 关联对象和 collection 关联集合对象的延迟加载，association 指的就是一对一查询，collection 指的就是一对多查询。

在 Mybatis配置文件中，可以通过配置是否启用延迟加载 lazyLoadingEnabled=true(或false)。它的原理是，使用 CGLIB 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。

### 3、Mybatis都有哪些Executor执行器？它们之间的区别是什么？

答：Mybatis有三种基本的Executor执行器，SimpleExecutor、ReuseExecutor、BatchExecutor，CachingExecutor。

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

ReuseExecutor的特点：每执行一次update或select操作，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 的特点：每执行一次 update 操作，将所有 sql 都添加到批处理任务中，等待统一执行，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。

CachingExecutor：运用了装饰者模式的 Executor，给其他 Executor 增加缓存功能，开启二级缓存有效。

### 4、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）？

答：

一级缓存：

Mybatis的一级缓存是指SqlSession级别的，作用域是SqlSession，Mybatis默认开启一级缓存，在同一个SqlSession中，相同的Sql查询的时候，第一次查询的时候，就会从缓存中取，如果发现没有数据，那么就从数据库查询出来，并且缓存到HashMap中。如果下次还是相同的查询，就直接从缓存中查询，就不在去查询数据库，对应的就不在去执行SQL语句。当查询到的数据，进行增删改的操作的时候，缓存将会失效。在spring容器管理中每次查询都是创建一个新的sqlSession，所以在分布式环境中不会出现数据不一致的问题

二级缓存：

二级缓存是mapper级别的缓存，即多个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。这是与一级缓存最大的不同。第一次调用mapper下的sql的时候去查询信息，查询到的信息会存放到该mapper对应的二级缓存区域，第二次调用namespace下的mapper映射文件中，相同的SQL去查询，回去对应的二级缓存内取结果，使用需要开启cache标签，在select上添加useCache属性为true，在更新和删除时候需要手动开启flushCache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？

答：mybatis允许编写针对Executor、StatementHandler、ParameterHandler、ResultSetHandler四个接口的插件。

插件的运行原理是基于AOP的实现，通过实现Interceptor接口，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。