

### 1、什么是MyBatis?

答: MyBatis是一个可以自定义SQL、存储过程和高级映射的持久层框架。

### 2、讲下MyBatis的缓存

答: MyBatis的缓存分为一级缓存和二级缓存,一级缓存放在session里面,默认就有,二级缓存放在它的命名空间里,默认是不打开的,使用二级缓存属性类需要实现Serializable序列化接口(可用来保存对象的状态),可在它的映射文件中配置

### 3、Mybatis是如何进行分页的? 分页插件的原理是什么?

答:

1) Mybatis使用RowBounds对象进行分页, 也可以直接编写sql实现分页, 也可以使用Mybatis的分页插件。

2) 分页插件的原理: 实现Mybatis提供的接口, 实现自定义插件, 在插件的拦截方法内拦截待执行的sql, 然后重写sql。

举例: select \* from student, 拦截sql后重写为: select t.\* from (select \* from student) t limit 0, 10

### 4、简述Mybatis的插件运行原理, 以及如何编写一个插件?

答:

1) Mybatis仅可以编写针对ParameterHandler、ResultSetHandler、StatementHandler、Executor这4种接口的插件, Mybatis通过动态代理, 为需要拦截的接口生成代理对象以实现接口方法拦截功能, 每当执行这4种接口对象的方法时, 就会进入拦截方法, 具体就是InvocationHandler的invoke()方法, 当然, 只会拦截那些你指定需要拦截的方法。

2) 实现Mybatis的Interceptor接口并复写intercept()方法, 然后在给插件编写注解, 指定要拦截哪一个接口的哪些方法即可, 记住, 别忘了在配置文件中配置你编写的插件。

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

答:

1) Mybatis动态sql可以让我们在Xml映射文件内, 以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。

2) Mybatis提供了9种动态sql标签: trim|where|set|foreach|if|choose|when|otherwise|bind。

3) 其执行原理为, 使用OGNL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。

### 6、#{ }和\${ }的区别是什么?

答:

1) #{ }是预编译处理, \${ }是字符串替换。

2) Mybatis在处理#{ }时, 会将sql中的#{ }替换为?, 调用PreparedStatement的set方法来赋值;

3) Mybatis在处理\${ }时, 就是把\${ }替换成变量的值。

4) 使用#{ }可以有效的防止SQL注入, 提高系统安全性。

### 7、为什么说Mybatis是半自动ORM映射工具? 它与全自动的区别在哪里?

答: Hibernate属于全自动ORM映射工具, 使用Hibernate查询关联对象或者关联集合对象时, 可以根据对象关系模型直接获取, 所以它是全自动的。而Mybatis在查询关联对象或关联集合对象时, 需要手动编写sql来完成, 所以, 称之为半自动ORM映射工具。

8、Mybatis是否支持延迟加载？如果支持，它的实现原理是什么？

答：

1) 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。

2) 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，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。

9、MyBatis与Hibernate有哪些不同？

答：

1) Mybatis和hibernate不同，它不完全是一个ORM框架，因为MyBatis需要程序员自己编写Sql语句，不过mybatis可以通过XML或注解方式灵活配置要运行的sql语句，并将java对象和sql语句映射生成最终执行的sql，最后将sql执行的结果再映射生成java对象。

2) Mybatis学习门槛低，简单易学，程序员直接编写原生态sql，可严格控制sql执行性能，灵活度高，非常适合对关系数据模型要求不高的软件开发，例如互联网软件、企业运营类软件等，因为这类软件需求变化频繁，一旦需求变化要求成果输出迅速。但是灵活的前提是mybatis无法做到数据库无关性，如果需要实现支持多种数据库的软件则需要自定义多套sql映射文件，工作量大。

3) Hibernate对象/关系映射能力强，数据库无关性好，对于关系模型要求高的软件（例如需求固定的定制化软件）如果用hibernate开发可以节省很多代码，提高效率。但是Hibernate的缺点是学习门槛高，要精通门槛更高，而且怎么设计O/R映射，在性能和对象模型之间如何权衡，以及怎样用好Hibernate需要具有很强的经验和能力才行。

总之，按照用户的需求在有限的资源环境下只要能做出维护性、扩展性良好的软件架构都是好架构，所以框架只有适合才是最好。

10、MyBatis的好处是什么？

答：

1) MyBatis把sql语句从Java源程序中独立出来，放在单独的XML文件中编写，给程序的维护带来了很大便利。

2) MyBatis封装了底层JDBC API的调用细节，并能自动将结果集转换成Java Bean对象，大大简化了Java数据库编程的重复工作。

3) 因为MyBatis需要程序员自己去编写sql语句，程序员可以结合数据库自身的特点灵活控制sql语句，因此能够实现比Hibernate等全自动orm框架更高的查询效率，能够完成复杂查询。

11、简述Mybatis的Xml映射文件和Mybatis内部数据结构之间的映射关系？

答：Mybatis将所有Xml配置信息都封装到All-In-One重量级对象Configuration内部。在Xml映射文件中，标签会被解析为ParameterMap对象，其每个子元素会被解析为ParameterMapping对象。标签会被解析为ResultMap对象，其每个子元素会被解析为ResultMapping对象。每一个 