

《MyBatis - 就业技能》

章节	Ch01 - MyBatis 入门
题目 1)	什么是框架技术，以及框架的优势
	是一个应用程序的半成品,提供可重用的公共结构,按一定规则组织的一组组件 优势： 1.不用再考虑公共问题 2.专心在业务实现上 3.结构统一，易于学习、维护
题目 2)	当前 java 主流框架以及作用
	1.spring 依赖注入容器 / AOP 实现 声明式事务，简化 Java EE 应用 黏合剂，将大家组装到一起 2.Spring MVC 结构最清晰的 MVC Model2 实现 高度可配置，支持多种视图技术，定制化开发 3.mybatis 用来替代原始的 dao 层 半自动化的 ORM 实现 动态 SQL、小巧灵活、简单易学
题目 3)	什么是持久化和 ORM
	持久化是程序数据在瞬时状态和持久状态间转换的过程 ORM：编写程序的时候，以面向对象的方式处理数据保存数据的时候，却以关系型数据库的方式存储
题目 4)	mybatis 基本要素
	核心对象： SqlSessionFactoryBuilder 用过即丢，推荐作用域范围：方法体内 SqlSessionFactory 最佳作用域范围：应用的全局作用域 生命周期与应用的生命周期相同 SqlSession 线程级一次操作 核心配置文件 mybatis-config.xml SQL 配置文件 mapper.xml
题目 5)	核心配置文件的标签及作用
	properties 通过外部指定的方式，实现动态配置(如数据源) settings 用来修改 MyBatis 运行时的行为方式(主要是 MyBatis 的一些全局配置属性的设置)

	<p>typeAliases 类型别名, 简化 java 类名</p> <p>environments 表示配置 MyBatis 的多套运行环境</p> <p>mappers 映射器, 定义 SQL 映射语句</p>
--	--

章节	Ch02 - SQL 映射文件
题目 1)	SQL 映射文件的几个顶级元素
	<p>mapper - namespace</p> <p>cache - 配置给定命名空间的缓存</p> <p>cache-ref - 从其他命名空间引用缓存配置</p> <p>resultMap - 用来描述数据库结果集和对象的对应关系</p> <p>sql - 可以重用的 SQL 块, 也可以被其他语句引用</p> <p>insert - 映射插入语句</p> <p>update - 映射更新语句</p> <p>delete - 映射删除语句</p> <p>select - 映射查询语句</p>
题目 2)	mybatis 的缓存机制
	<p>一级缓存(默认)</p> <p>二级缓存</p> <p>MyBatis 的全局 cache 配置</p> <p>在 Mapper XML 文件中设置缓存, 默认情况下是没有开启缓存的</p> <p>在 Mapper XML 文件配置支持 cache 后, 如果需要对个别查询进行调整, 可以单独设置 cache</p>
题目 3)	resultMap 和 resultType 的区别, 以及 resultMap 里的元素(顺序)
	<p>resultType: 对应的是 java 对象中的属性, 大小写不敏感</p> <p>resultMap 指的是定义好了的 id 的, 是定义好的 resultType 的引用</p> <p>resultMap: 属性 id: 唯一标识(resultMap 的名称)</p> <p>属性 type: Java 实体类</p> <p>标签 id: 一般对应数据库中该行的主键 id, 设置此项可提高 MyBatis 性能</p> <p>result: 映射到 JavaBean 的某个“简单类型”属性</p> <p>association: 映射到 JavaBean 的某个“复杂类型”属性, 比如 JavaBean 类</p> <p>collection: 映射到 JavaBean 的某个“复杂类型”属性, 比如集合</p>
题目 4)	resultMap 标签 association 用法 (一)
	<p>复杂的类型关联, 一对一</p> <p>内部嵌套</p> <p>映射一个嵌套 JavaBean 属性</p> <p>属性</p> <p>property: 映射数据库列的实体对象的属性</p> <p>javaType: 完整 Java 类名或者别名</p> <p>resultMap: 引用外部 resultMap</p> <p>子元素 id、result</p> <p>property: 映射数据库列的实体对象的属性</p> <p>column: 数据库列名或者别名</p>
题目 5)	resultMap 标签 association 用法 (二)
	复杂类型集合, 一对多

	<p>内部嵌套</p> <p>映射一个嵌套结果集到一个列表</p> <p>属性</p> <p>property: 映射数据库列的实体对象的属性</p> <p>ofType: 完整 Java 类名或者别名(集合所包括的类型)</p> <p>resultMap: 引用外部 resultMap</p> <p>子元素</p> <p>id、result</p> <p>property: 映射数据库列的实体对象的属性</p> <p>column: 数据库列名或者别名</p>
--	---

章节	Ch03 - 动态 SQL
题目 1)	#{}和\${}的区别是什么
	<p>1)#{}是预编译处理, \${}是字符串替换。</p> <p>2)Mybatis 在处理#{ }时, 会将 sql 中的#{ }替换为?号, 调用 PreparedStatement 的 set 方法来赋值;</p> <p>3)Mybatis 在处理\${ }时, 就是把\${ }替换成变量的值。</p> <p>4)使用#{ }可以有效的防止 SQL 注入, 提高系统安全性。</p>
题目 2)	动态 SQL 的主要元素
	if、trim、where、set、choose (when、otherwise)、foreach
题目 3)	如何获取自动生成的(主)键值?
	insert 方法总是返回一个 int 值 - 这个值代表的是插入的行数。而自动生成的键值在 insert 方法执行完后可以被设置到传入的参数对象中。
题目 4)	什么是 ognl 表达式
	ognl 是一种功能强大的表达式语言, 通过它简单一致的表达式语法, 可以存取对象的任意属性, 调用对象的方法, 遍历整个对象的结构图, 实现字段类型转化等功能。
题目 5)	Mybatis 是如何进行分页的? 分页插件的原理是什么?
	<p>Mybatis 使用 RowBounds 对象进行分页, 它是针对 ResultSet 结果集执行的内存分页, 而非物理分页, 可以在 sql 内直接书写带有物理分页的参数来完成物理分页功能, 也可以使用分页插件来完成物理分页。</p> <p>分页插件的基本原理是使用 Mybatis 提供的插件接口, 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, 根据 dialect 方言, 添加对应的物理分页语句和物理分页参数。</p>

章节	Ch04 - 继承与封装
题目 1)	当实体类中的属性名和表中的字段名不一样, 怎么办 ?
	<p>a.通过在查询的 sql 语句中定义字段名的别名, 让字段名的别名和实体类的属性名一致</p> <p>b.通过<resultMap>来映射字段名和实体类属性名的一一对应的关系</p>
题目 2)	mybatis 中如何获得自动生成主键的值
	insert 方法总是返回一个 int 值 - 这个值代表的是插入的行数。而自动生成的键值在 insert 方法执行完后可以被设置到传入的参数对象中
题目 3)	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?

	<p>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。</p> <p>Mybatis 提供了 9 种动态 sql 标签： trim where set foreach if choose when otherwise bind。</p> <p>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。</p>
题目 4)	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？
	<p>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。</p>
题目 5)	

章节	Ch05 - MyBatis 综合练习
题目 1)	oracle 数据库分页
	<p>a.最内层 sql，查询要分页的所有数据</p> <p>b.第二层 sql，通过 rownum 伪列确定显示数据的上限，并且给查询的数据添加 rownum 伪列的值</p> <p>c.最外层 sql，设置显示数据的下限</p> <pre> select * from (select a.*,rownum r from(select * from 表名 where 条件 order by 列) a where rownum<=页数*条数) b where r> (页数-1) *条数 </pre>
题目 2)	Oracle 中经常使用到的函数
	<p>length 长度、</p> <p>lower 小写、</p> <p>upper 大写、</p> <p>to_date 转化日期、</p> <p>to_char 转化字符、</p> <p>to_number 转化数字、</p> <p>Ltrim 去左边空格、</p> <p>rtrim 去右边空格、</p> <p>substr 截取字符串、</p> <p>add_month 增加或减掉月份、</p>
题目 3)	oracle 数据库的优化
	<ol style="list-style-type: none"> 1) 调整数据库结构的设计：需要考虑是否使用分区功能、是否建立索引等 2) 调整数据库的 SQL 语句 3) 调整服务器的内存分配 4) 调整硬盘 I/O 5) 调整操作系统参数
题目 4)	数据库事务的特性是指什么

	<p>1) 原子性 (Atomic): 事务中的各项操作, 要么全做要么全不做, 任何一项操作的失败都会导致整个事务的失败</p> <p>2) 一致性 (Consistent): 事务结束后系统状态是一样的</p> <p>3) 隔离性 (Isolated): 并发执行的事务彼此无法看到对方的中间状态</p> <p>4) 持久性 (Durable): 事务完成后, 即使发生灾难性的故障, 通过日志和同步备份可以在故障发生后重建数据</p>
题目 5)	mysql 和 oracle 数据库的区别
	<p>1) 应用方面: MySQL 是中小型应用的数据库, 一般用于个人项目或中小型网站及论坛。Oracle 属于大型数据库, 一般在具有相当规模的企业应用。</p> <p>2) 自动增长的数据类型方面: MySQL 有自动增长的数据类型。Oracle 没有自动增长的数据类型, 需要建立一个自增序列</p> <p>3) group by 用法: MySQL 中 group by 在 select 语句中可以随意使用, 但是在 Oracle 中如果查询语句中有组函数, 那其他列名必须是组函数处理过的或者是 group by 子句中的列, 否则报错</p> <p>4) 引号方面: MySQL 中用双引号包起字符串, Oracle 中只可以用单引号包起字符串</p>

章节	Ch06 - Oracle 数据库入门
题目 1)	SQL 语句分类
	<p>DQL (数据查询语言) select</p> <p>DML (数据操作语言) insert、delete、update</p> <p>DDL (数据定义语言) create、drop、alter</p> <p>DCL (数据控制语言) grant: 把权限授予用户、revoke: 把权限从用户收回</p> <p>TPL (TCL, 事务控制语言): commit、rollback</p>
题目 2)	Oracle 中经常使用到的函数
	<p>length 长度、lower 小写、upper 大写、to_date 转化日期、to_char 转化字符、to_number 转化数字</p> <p>Ltrim 去左边空格、rtrim 去右边空格、substr 截取字符串、add_month 增加或减掉月份</p>
题目 3)	怎样创建一个索引, 索引使用的原则, 有什么优缺点
	<p>create index 索引名 on 表名 (列名)</p> <p>原则:</p> <p> 建议索引列建立 not null 约束</p> <p> 经常与其他表进行连接的表, 在连接列上建立索引</p> <p>优缺点:</p> <p> 创建索引能大大加快检索速度, 加强表与表的连接, 但是创建索引很占用空间</p>
题目 4)	MySQL 数据库与 Oracle 数据库有什么区别
	<p>1) 应用方面: MySQL 是中小型应用的数据库, 一般用于个人项目或中小型网站及论坛。Oracle 属于大型数据库, 一般在具有相当规模的企业应用。</p> <p>2) 自动增长的数据类型方面: MySQL 有自动增长的数据类型。Oracle 没有自动增长的数据类型, 需要建立一个自增序列</p> <p>3) group by 用法: MySQL 中 group by 在 select 语句中可以随意使用, 但是在 Oracle 中如果查询语句中有组函数, 那其他列名必须是组函数处理过的或者是 group by 子句中的列, 否则报错</p>

	4) 引号方面: MySQL 中用双引号包起字符串, Oracle 中只可以用单引号包起字符串
题目 5)	