**1、jdbc**

java程序数据和关系型数据库进行传输的框架，提供了**各种接口**，来操作各种数据库，mysql，oracle，sqlserver等，使用的对应的数据库的时候，需要提供驱动jar（各种

数据库操作的实现类）。

**---》好处**

统一了连接各种数据库的代码-connection，statement，resultset等接口

**---》不足**

操作数据库时步骤太复杂，加载驱动，获取连接，获取语句处理器，编写sql，设置参数，处理并封装返回结果，关闭各种资源。

**2、hibernate**

对jdbc的**（重量级封装），目标-操作数据库不需要使用sql（方便不会sql的开发者）**是一个ORM（对象关系映射）框架，**把数据库表（table）的关系，映射到java类（po类）之间的关系**。用面向对象的思维来操作关系式数据库。

**核心功能**

a）自动根据po类及关系建立出对应的表，并设置表的关系

po类：one to one，one to many，many to many

b）直接用save，get，delete，update 可以直接转化为对应的sql语句

c）提供了面向对象查询语句hql-解决了各数据库的兼容性

session.createQuery("hql");

d）提供jdbc基本封装的sql

session.createSQLquery("sql");

。。。

**3、mybatis(原名ibatis)**

对jdbc的**（轻量级级封装），目标-把sql和java代码分离，封装sql的参数转换，封装sql的结果集的转换。（方便sql的修改）**

是一个ORM（对象关系映射）框架，**没有完全映射**！

**把sql参数映射为对象，结果集映射为对象。**

**核心功能**

a）给执行的sql设置参数可以使用对象（javabean，map...），sql中可以动态取出（#{},${}）

b）获取sql执行的结果，可以直接映射到对象（javabean，map...）--可以自定义映射（配置映射关系）

c）动态的sql，可以在xml中动态拼接sql（方便业务需求的更改）（if，choose，foreach）

**。。。**

**4、jdbc、mybatis、hibernate的对比**

**性能对比**

jdbc > mybatis > hibernate

**学习成本对比**

mybatis > jdbc > hibernate

**开发速度对比-基本的增，删，改，查**

hibernate > mybatis >jdbc

**开发速度对比-复杂的查询（多表连接，分组，子查询）**

mybatis 约等于 hibernate > jdbc

**代码的维护-》sql代码的灵活度（修改）**

mybatis（**sql分离到xml**） > hibernate（sql在java中） ，jdbc

**5、选择**

a）业务需求比较固定，不是特别复杂

---》hibernate

线下的系统（数据量不大），开发人员技术高（否则容易性能问题）

b）业务需求经常变，sql要经常改，业务比较复杂

---》mybatis

数据量大，比较适合统计分析的系统（经常需要改查询的条件）

**6、总结**

**hibernate入门难，进阶更难，性能还不是特别好--（数据库的基本操作简单）**

**mybatus 入门简单，进阶简单，性能通常比hibernate好**