1. 创建数据库和表tb\_book
2. 创建项目book4，导入库文件到lib目录，创建5个包

结构目录设计：Src

cn.book.dao 数据访问

cn.book.model 实体

cn.book.service 业务逻辑

cn.book.test 测试

cn.book.util 工具

cn.book.web 界面

1. 导入数据库连接的辅助工具类JdbcUtil.java

导入数据库连接池的配置文件c3p0-config.xml，并修改符合自己的连接参数

导入数据库连接池测试类TestCase.java，并运行，检查数据库连接是否正常。

1. 在model包中创建一个Book.java的实体类，包含以下属性：并生成getter和setter方法  
    **private** **int** id;

**private** String name;

**private** Double price;

**private** **int** bookCount;

**private** String author;

1. 创建BookDao.java接口，并添加以下抽象方法：

**public** **void** save(Book book);

**public** **void** delete(**int** id);

**public** **void** update(Book book);

**public** List<Book> queryAll();

**public** Book queryById(**int** id);

1. **在**cn.book.dao.impl**创建BookDao的实现类,BookDaoImpl.java**

**// 创建QueryRunner对象,执行SQL语句**

**private QueryRunner qr = new QueryRunner(JdbcUtil.getDataSource());**

1. 添加代码save()方法

String sql="insert into tb\_book001(name,price,bookCount,author) values(?,?,?,?)";

Object[] params={book.getName(),book.getPrice(),book.getBookCount(),book.getAuthor()};

qr.update(sql, params);

1. 添加测试用例TestCase.java  
    private BookDao dao=new BookDaoImpl();

Book book=**new** Book();

book.setName("KKK");

book.setPrice(23.0);

book.setBookCount(33);

book.setAuthor("AAA");

dao.save(book);