0.编写SQL脚本，创建必需的数据库结构

1.完整演示：注册新用户

1)服务器端：编写一个静态页面register.html,在客户端呈现出一个表单，辅助用户提交数据

2)客户端：使用浏览器通过网络访问Web服务器，获取register.html，输入注册信息，点击提交

3)服务器端：编写一个动态页面register.php，接收客户端提交的请求数据，保存入数据库，返回注册成功或失败

2.完整演示：删除指定编号的用户

1)服务器端：编写一个静态页面user\_delete.html,在客户端呈现出一个表单“请输入您要删除的用户编号”，辅助用户提交数据

2)客户端：使用浏览器通过网络访问Web服务器，获取user\_delete.html，输入想删除的用户编号，点击提交

3)服务器端：编写一个动态页面user\_delete.php，接收客户端提交的请求数据（uid），向数据库执行DELETE语句，返回删除成功或删除失败

3.完整演示：用户信息的修改

1)服务器端：编写一个静态页面user\_update.html,在客户端呈现出一个表单“请输入您要修改的用户编号、用户名、用户密码...”，辅助用户提交数据

2)客户端：使用浏览器通过网络访问Web服务器，获取user\_update.html，输入想修改的用户编号，以及修改后的信息，点击提交

3)服务器端：编写一个动态页面user\_update.php，接收客户端提交的请求数据，向数据库执行UPDATE xz\_user SET uname=?,upwd=? WHERE uid=?语句，返回修改成功或修改失败

4.完整演示：查询一个用户的信息

1)服务器端：编写一个静态页面user\_select.html,在客户端呈现出一个表单“请输入您要查询的用户编号”，辅助用户提交数据

2)客户端：使用浏览器通过网络访问Web服务器，获取user\_select.html，输入想查询的用户编号，点击提交

3)服务器端：编写一个动态页面user\_select.php，接收客户端提交的请求数据（uid），向数据库执行SELECT语句，返回查询失败或者成功查询后的用户信息

5.完整演示：查询所有用户的信息

1)服务器端：编写一个动态页面user\_list.php，向数据库执行SELECT语句，返回查询失败或者成功查询后的信息