

# 实验4 数据库应用开发实验

学号	19335162	姓名	潘思晗
----	----------	----	-----

## 一、实验目的

掌握使用程序设计语言访问数据库。

## 二、实验内容及结果

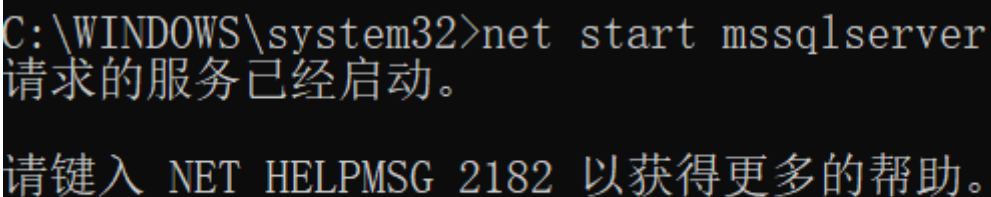
使用 *JDBC*或 *ODBC*或其它程序设计语言分别读取图书借阅数据库系统 *JY*的图书表 *book*、读者表*reader*、借阅记录表*record*表的所有数据并打印

选择使用ODBC连接Microsoft SQL server数据库。

### 1.配置本地环境

#### （1）启动**SQLSERVER**服务

以管理员身份运行cmd，输入`net start mssqlserver`启动。

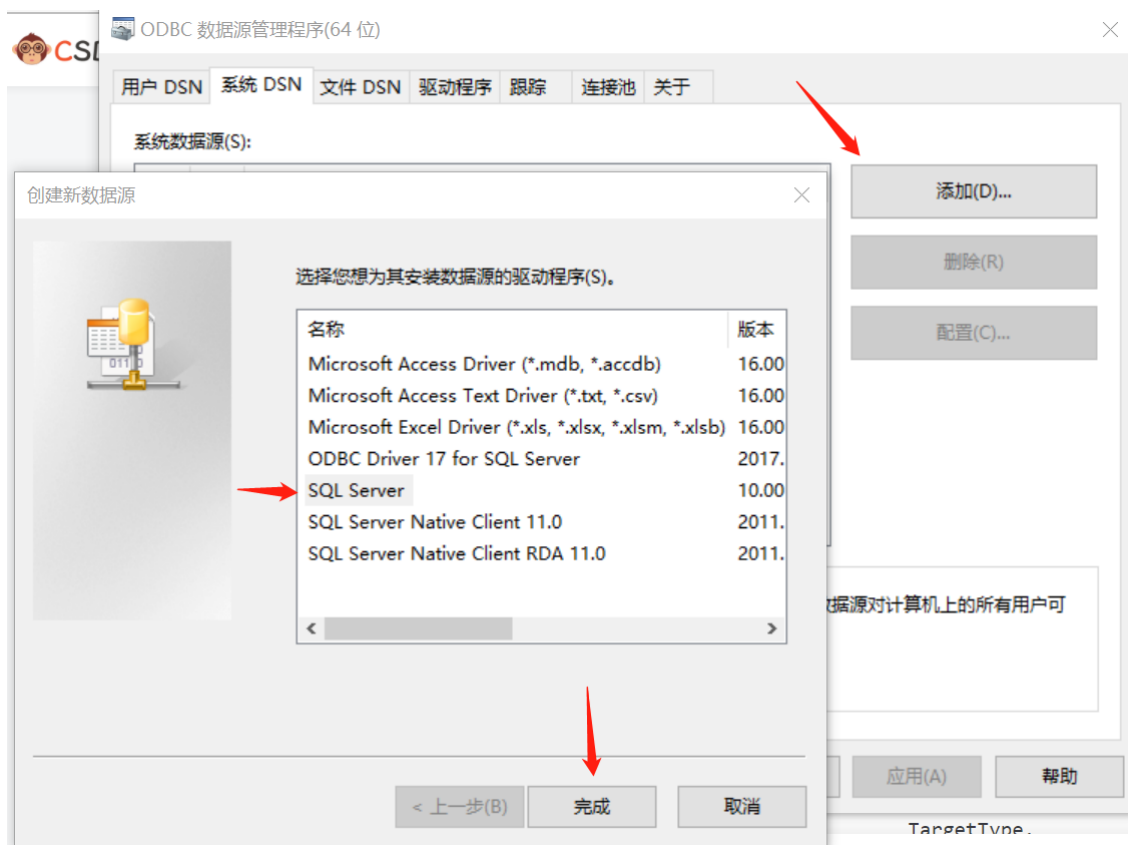


```
C:\WINDOWS\system32>net start mssqlserver
请求的服务已经启动。

请键入 NET HELPMSG 2182 以获得更多的帮助。
```

#### （2）配置系统**DSN**

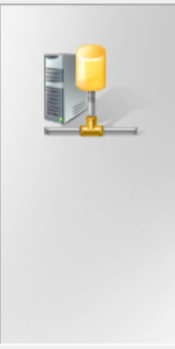
- 打开【ODBC数据源管理器】，在【ODBC数据源管理器】对话框中，单击【系统DNS】标签，单击【添加】按钮，弹出【创建新数据源】对话框，在驱动程序序列表中选择【SQL Server】，单击【完成】按钮，如下图所示



- 输入数据源的名称，此处我命名为Library，服务器名称为SQL Server中的服务器登录名称：



- 选择使用ID和密码的SQL Server验证，登录ID为sa，输入之前设置的密码



SQL Server 应该如何验证登录 ID 的真伪?

☐ 使用网络登录 ID 的 Windows NT 验证(W)。

☒ 使用用户输入登录 ID 和密码的 SQL Server 验证(S)。

要更改用于与 SQL Server 通讯的网络库，请单击“客户端配置”

客户端配置(T)...

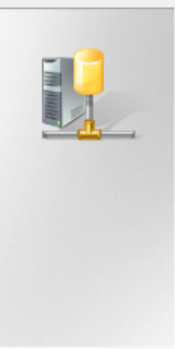
☒ 连接 SQL Server 以获得其它配置选项的默认设置(C)。

登录 ID(L):

密码(P):

< 上一步(B)    下一页(N) >    取消    帮助

- 更改默认的数据库为JY



☒ 更改默认的数据库为(D):

☐ 附加数据库文件名(H):

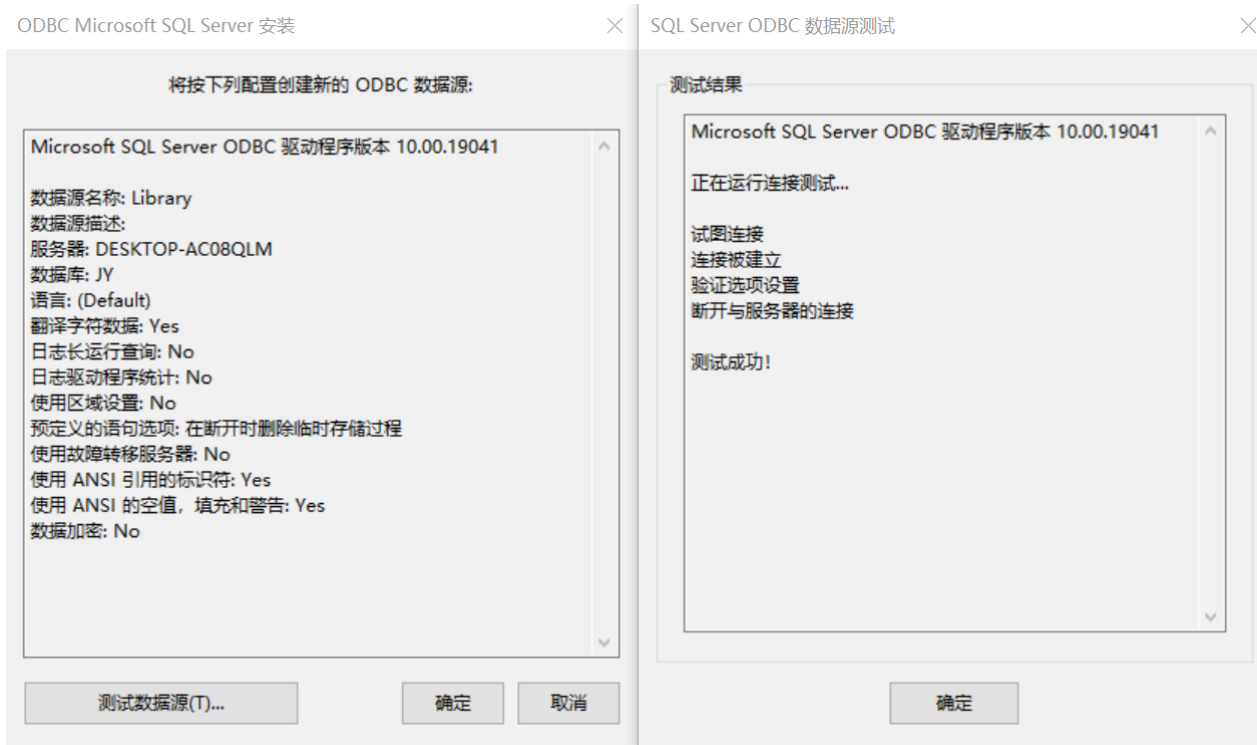
☒ 使用 ANSI 引用的标识符(U)。

☒ 使用 ANSI 的空值、填充及警告(A)。

☐ 若主 SQL Server 不可用，请使用故障转移 SQL Server(F)。

< 上一步(B)    下一页(N) >    取消    帮助

- 测试数据源，测试成功，即DNS添加成功



## 2. ODBC程序

- 开发工具: Visual Stdio 2019

打开visual studio, 建立新项目, 源文件新建项添加C++文件

## (1) 连接数据库

包含下列头文件：

```
#include <windows.h>
#include <sql.h>
#include <sqlext.h>
#include <sqltypes.h>
#include <odbcss.h>
#include <stdio.h>
#include <odbcinst.h>
#include <iostream>
using namespace std;

SQLHENV henv; //环境句柄
SQLHDBC hdbc; //连接句柄
SQLHSTMT hstmt; //语句句柄
SQLRETURN retcode; //返回值
```

建立连接：

```
//申请环境句柄
retcode = SQLAllocHandle(SQL_HANDLE_ENV, NULL, &henv);
retcode = SQLSetEnvAttr(henv, SQL_ATTR_ODBC_VERSION,
(SQLPOINTER)SQL_OV_ODBC3, SQL_IS_INTEGER);
//申请数据库连接句柄
retcode = SQLAllocHandle(SQL_HANDLE_DBC, henv, &hdbc);

retcode = SQLSetConnectAttr(hdbc, SQL_ATTR_AUTOCOMMIT,
(void*)SQL_AUTOCOMMIT_OFF, SQL_IS_POINTER);
if (!(retcode == SQL_SUCCESS || retcode ==
SQL_SUCCESS_WITH_INFO))printf("设置连接属性出错\n");

//连接数据库，xxxxxxx表示'sa'登陆密码
retcode = SQLConnect(hdbc, (SQLCHAR*)"Library", SQL_NTS,
(SQLCHAR*)"sa", SQL_NTS, (SQLCHAR*)"xxxxxxx", SQL_NTS);
if (!(retcode == SQL_SUCCESS || retcode == SQL_SUCCESS_WITH_INFO))
{
    printf("连接数据库失败!\n");
    return -1;
}
else
```

```
{
    printf("connect SQL success!");
}
```

## （2）读取JY的图书表book

使用 `SQLSQLExecDirect()` 执行 `use JY` 和 `select * from book` 语句，然后使用 `SQLGetData()` 函数获取表格数据

```
printf("\nTABLE_book:\n");
retcode = SQLAllocHandle(SQL_HANDLE_STMT, hdbc, &hstmt);
SQLCHAR sql1[] = "use JY"; //使用JY数据库
SQLCHAR sql2[] = "select * from book"; //查询book表的所有列
retcode = SQLExecDirect(hstmt, sql1, SQL_NTS); //SQLExecDirect, 此函数
用于执行SQL语句，执行sql1语句，即"use JY"
retcode = SQLExecDirect(hstmt, sql2, SQL_NTS); //执行sql2语句
if (retcode == SQL_SUCCESS || retcode == SQL_SUCCESS_WITH_INFO)
{
    SQLCHAR str1[50], str2[50], str3[50], str4[50], str5[50],
    str6[50], str7[50];
    INT64 len_str1, len_str2, len_str3, len_str4, len_str5,
    len_str6, len_str7;
    while (SQLFetch(hstmt) != SQL_NO_DATA)
    {
        SQLGetData(hstmt, 1, SQL_C_CHAR, str1, 50, &len_str1); //
        获取第一列数据
        SQLGetData(hstmt, 2, SQL_C_CHAR, str2, 50, &len_str2); //
        获取第二列数据
        SQLGetData(hstmt, 3, SQL_C_CHAR, str3, 50, &len_str3);
        SQLGetData(hstmt, 4, SQL_C_CHAR, str4, 50, &len_str4);
        SQLGetData(hstmt, 5, SQL_C_CHAR, str5, 50, &len_str5);
        SQLGetData(hstmt, 6, SQL_C_CHAR, str6, 50, &len_str6);
        SQLGetData(hstmt, 7, SQL_C_CHAR, str7, 50, &len_str7);
        //book表共7列

        printf("\n%s\t%s\t%s\t%s\t%s\t%s\t%s\n", str1, str2, str3,
        str4, str5, str6, str7);
    }
}
```

### （3）读取JY的读者表reader

同样的方法读取表reader

```
printf("\nTABLE_reader:\n");
retcode = SQLAllocHandle(SQL_HANDLE_STMT, hdbc, &hstmt);
SQLCHAR sql3[] = "select * from reader";//查询reader表的所有列
retcode = SQLExecDirect(hstmt, sql1, SQL_NTS);//执行sql1语句
retcode = SQLExecDirect(hstmt, sql3, SQL_NTS);//执行sql3语句
if (retcode == SQL_SUCCESS || retcode == SQL_SUCCESS_WITH_INFO)
{
    SQLCHAR str1[50], str2[50], str3[50], str4[50];
    INT64 len_str1, len_str2, len_str3, len_str4;
    while (SQLFetch(hstmt) != SQL_NO_DATA)
    {
        SQLGetData(hstmt, 1, SQL_C_CHAR, str1, 50, &len_str1);    //
        获取第一列数据
        SQLGetData(hstmt, 2, SQL_C_CHAR, str2, 50, &len_str2);    //
        获取第二列数据
        SQLGetData(hstmt, 3, SQL_C_CHAR, str3, 50, &len_str3);
        SQLGetData(hstmt, 4, SQL_C_CHAR, str4, 50, &len_str4);

        printf("\n%s\t%s\t%s\t%s\n", str1, str2, str3, str4);
    }
}
```

### （4）读取JY的借阅表record

读取表record

```
printf("\nTABLE_record:\n");
retcode = SQLAllocHandle(SQL_HANDLE_STMT, hdbc, &hstmt);
SQLCHAR sql4[] = "select * from reader";//查询record表的所有列
retcode = SQLExecDirect(hstmt, sql1, SQL_NTS);//执行sql1语句
retcode = SQLExecDirect(hstmt, sql4, SQL_NTS);//执行sql4语句
if (retcode == SQL_SUCCESS || retcode == SQL_SUCCESS_WITH_INFO)
{
    SQLCHAR str1[50], str2[50], str3[50], str4[50], str5[50],
    str6[50], str7[50];
    INT64 len_str1, len_str2, len_str3, len_str4, len_str5,
    len_str6, len_str7;
    while (SQLFetch(hstmt) != SQL_NO_DATA)
```

```

{
    SQLGetData(hstmt, 1, SQL_C_CHAR, str1, 50, &len_str1);    //
    获取第一列数据
    SQLGetData(hstmt, 2, SQL_C_CHAR, str2, 50, &len_str2);    //
    获取第二列数据
    SQLGetData(hstmt, 3, SQL_C_CHAR, str3, 50, &len_str3);
    SQLGetData(hstmt, 4, SQL_C_CHAR, str4, 50, &len_str4);
    SQLGetData(hstmt, 5, SQL_C_CHAR, str5, 50, &len_str5);

    printf("\n%s\t%s\t%s\t%s\t", str1, str2, str3, str4);
    if (len_str5<0)printf("NULL\n");
}
}

printf("processes finish\n");

```

## （5）释放连接

```

SQLFreeHandle(SQL_HANDLE_DBC, hdbc); //释放连接句柄
SQLFreeHandle(SQL_HANDLE_ENV, henv); //释放环境句柄

```

## 3.实验结果

### （1）图书表book

获取book的所有内容并打印


Microsoft Visual Studio 调试控制台

```

connect SQL success!
TABLE_book:
b0001      SQL Server 2012宝典      978-7-121-22013-5      廖梦怡  电子工业出版社  18      89.0000
b0002      职称英语专用教材      978-7-121-14800-2      孙若红  电子工业出版社  37      45.0000
b0003      中国通史      978-7-5388-53155      于海娣  黑龙江科学技术出版社  25      68.0000
b0004      丰子恺儿童文学选集      978-7-5007-8972-7      丰子恺  中国少年儿童出版社  40      22.5000
b0005      英语同义词辨析词典      978-7-5135-2294-6      赵同水  外语教学与研究出版社  6      55.0000
b0006      数据库基础与应用      978-7-304-06430-3      徐孝凯  中央广播电视大学出版社  5      35.0000
b0007      微积分初步      978-7-304-03742-0      赵坚  中央广播电视大学出版社  4      17.0000
b0008      ASP.NET从入门到精通      978-7-302-28753-7      明日科技  清华大学出版社  27      89.8000

```

对比SQL Server中表格内容：



b0001	SQL Server 2012宝典	978-7-121-22013-5	廖梦怡	电子工业出版社	18	89.00
b0002	职称英语专用教材	978-7-121-14800-2	孙若红	电子工业出版社	37	45.00
b0003	中国通史	978-7-5388-53155	于海娣	黑龙江科学技术出版社	25	68.00
b0004	丰子恺儿童文学选集	978-7-5007-8972-7	丰子恺	中国少年儿童出版社	40	22.50
b0005	英语同义词辨析词典	978-7-5135-2294-6	赵同水	外语教学与研究出版社	6	55.00
b0006	数据库基础与应用	978-7-304-06430-3	徐孝凯	中央广播电视大学出版社	5	35.00
b0007	微积分初步	978-7-304-03742-0	赵坚	中央广播电视大学出版社	4	17.00
b0008	ASP.NET从入门到精通	978-7-302-28753-7	明日科技	清华大学出版社	27	89.80

对比可知结果符合预期。

## （2）读者表reader

获取reader的全部内容

TABLE\_reader:

```

r0001      李德海  男      信息工程系
r0002      柳承运  男      信息工程系
r0003      安歌    女      涉外教育系
r0004      谢嫣然  女      涉外教育系
r0005      陈静玉  女      涉外教育系
r0006      李媛媛  女      经济管理系
r0007      胡锦涛  男      经济管理系
r0008      蔡明伟  男      行政管理系

```

对比SQL Server中表格内容：

r0001	李德海	男	信息工程系
r0002	柳承运	男	信息工程系
r0003	安歌	女	涉外教育系
r0004	谢嫣然	女	涉外教育系
r0005	陈静玉	女	涉外教育系
r0006	李媛媛	女	经济管理系
r0007	胡锦涛	男	经济管理系
r0008	蔡明伟	男	行政管理系

对比可知结果符合预期。

### （3）借阅表record

获取record的全部内容

TABLE\_record:

```
r0001      b0003      2014-01-12      2014-01-12      NULL
r0001      b0005      2014-01-26      2014-06-21      NULL
r0004      b0001      2014-03-02      2014-04-20      NULL
r0004      b0008      2014-03-26      2014-05-28      NULL
r0006      b0001      2014-04-16      2014-07-11      NULL
r0007      b0006      2014-05-08      2014-09-17      NULL
r0008      b0008      2014-06-29      2014-08-29      NULL
r0008      b0007      2014-08-15      2014-10-21      NULL
processes finish
```

对比SQL Server中表格内容：

r0001	b0003	2014-01-12	2014-01-12	NULL
r0001	b0005	2014-01-26	2014-06-21	NULL
r0004	b0001	2014-03-02	2014-04-20	NULL
r0004	b0008	2014-03-26	2014-05-28	NULL
r0006	b0001	2014-04-16	2014-07-11	NULL
r0007	b0006	2014-05-08	2014-09-17	NULL
r0008	b0008	2014-06-29	2014-08-29	NULL
r0008	b0007	2014-08-15	2014-10-21	NULL

对比可知结果符合预期。

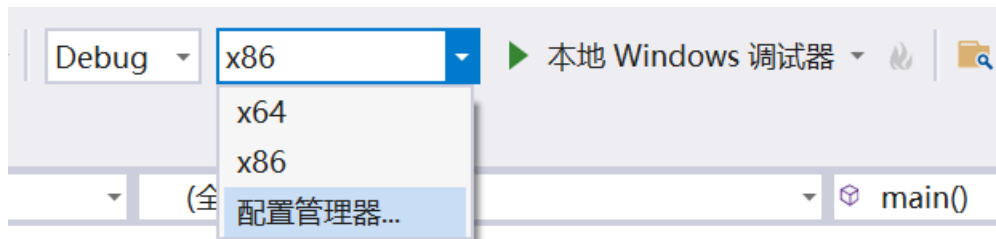
## 三、实验总结

本次实验，我学习了使用ODBC连接SQL Server数据库并读取和打印表格内容实验过程中遇到了很多问题，通过在网上查找资料的方式得到了解决。

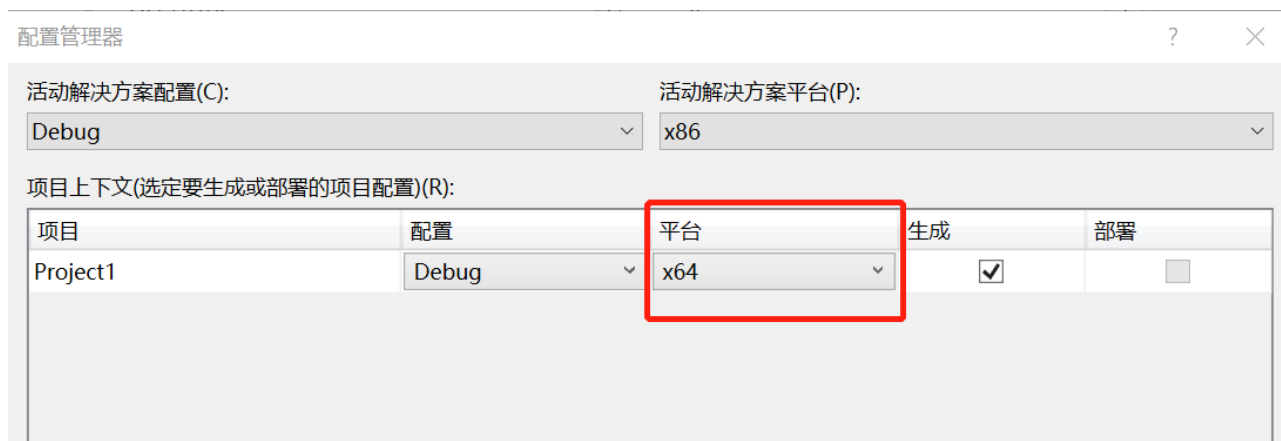
1.遇到错误“无法找到 **Visual Studio 2010** 的生成工具(平台工具集 =“v100”)”

解决方法：<https://www.cnblogs.com/youyou0/p/8995030.html>

首先在VS2019界面找到DEBUG右边的x86，点开选择配置管理器



在配置管理器中将平台修改为x64



2.更改为x64平台编译后，出现“SQLINTEGER \*”类型的实参与“SQLLEN \*”类型的形参不兼容”的报错

解决方法：[https://blog.csdn.net/qq\\_37457202/article/details/81261406](https://blog.csdn.net/qq_37457202/article/details/81261406)

将32位的SQLINTEGER修改为INT64

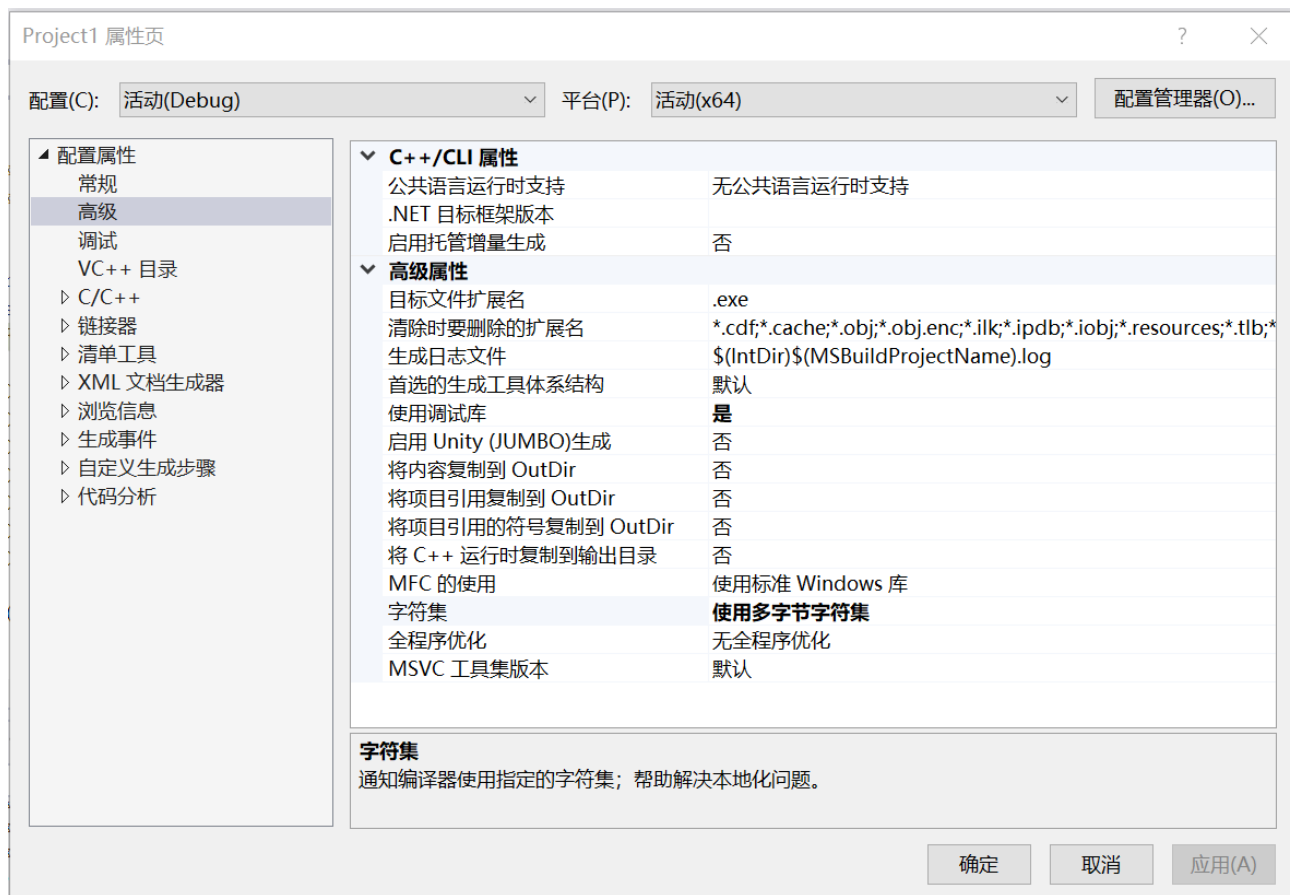
```
if (retcode == SQL_SUCCESS || retcode == SQL_SUCCESS_WITH_INFO)
{
    SQLCHAR str1[50], str2[50], str3[50], str4[50], str5[50], str6[50], str7[50];
    INT64 len_str1, len_str2, len_str3, len_str4, len_str5, len_str6, len_str7;
    while (SQLFetch(hstmt) != SQL_NO_DATA)
    {
        SQLGetData(hstmt, 1, SQL_C_CHAR, str1, 50, &len_str1); //获取第一列数据
        SQLGetData(hstmt, 2, SQL_C_CHAR, str2, 50, &len_str2); //获取第二列数据
        SQLGetData(hstmt, 3, SQL_C_CHAR, str3, 50, &len_str3);
        SQLGetData(hstmt, 4, SQL_C_CHAR, str4, 50, &len_str4);
        SQLGetData(hstmt, 5, SQL_C_CHAR, str5, 50, &len_str5);
        SQLGetData(hstmt, 6, SQL_C_CHAR, str6, 50, &len_str6);
        SQLGetData(hstmt, 7, SQL_C_CHAR, str7, 50, &len_str7); //book表共7列

        printf("\n%s\t%s\t%s\t%s\t%s\t%s\t%s\n", str1, str2, str3, str4, str5, str6, str7);
    }
}
```

3.遇到“SQLCHAR \*”类型的实参与“SQLWCHAR \*”类型的形参不兼容”报错

解决方法：

在调试属性--配置属性--高级--字符集中，设置为多字节字符集



更改完毕后，再次调试运行，结果成功。