# 开发文档

### 开发环境

本程序采用Microsoft VS2012进行开发。

### 开发步骤

* 1. **建立单文档工程文档**

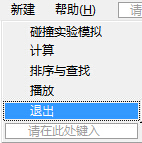
首先，使用VS2012新建一个MFC工程，命名为MFCApplication6。在选择MFC工程参数时，选择为单文档文件以及一些其它常用的选项。需要说明的是，由于最开始并没有事先建立好数据库，因此最开始的并没有选择提供数据库支持。后来再需要调用数据库时，采用CDataBase类对数据库进行操作。

* 1. **界面设计与实现**

程序的主界面如下：



界面相对来说比较简单，但是可以很明显的通过界面看到程序的功能。在菜单栏中，主要是“新建”和“帮助”两个子菜单。



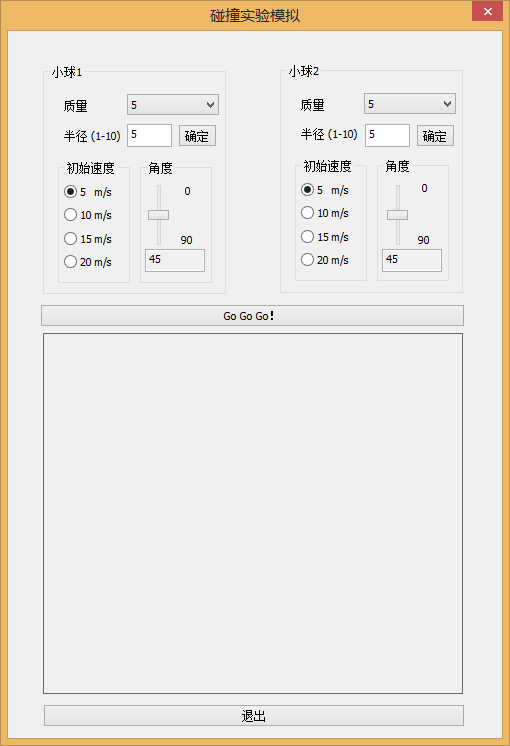
工具栏的5个按钮也对应着5项基本功能：

* 碰：碰撞实验模拟
* 算：计算
* 排：排序与查找
* 播：播放
* 关：关于

具体的每一个功能的详细情况在大作业介绍中都有说明。

* 1. **“碰撞实验模拟”对话框设计与实现**

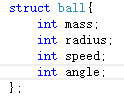
建立一个新的对话框，ID为IDD\_SIMULATION，并对对话框进行设计：



对应的控件信息如下：

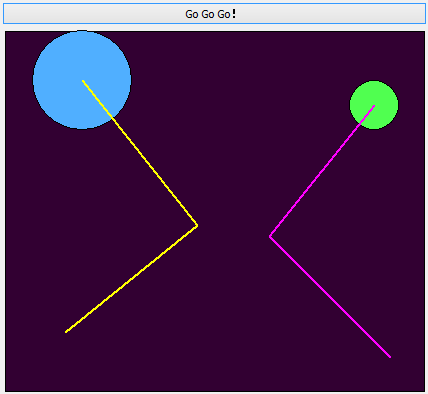
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | 类型 | 变量 | Captain | 响应函数 | 功能描述 |
| IDC\_BUTTON1 | CButton | m\_button\_go | Go Go Go! | OnBnClickedButton1 | 开始实验模拟 |
| IDC\_BUTTON2 | CButton | m1\_radius\_sure | 确定 | OnBnClickedButton2 | 更改小球1的半径 |
| IDC\_BUTTON3 | CButton | m2\_radius\_sure | 确定 | OnBnClickedButton3 | 更改小球2的半径 |
| IDC\_COMBO1 | CComboBox | m1\_cb |  | OnCbnSelchangeCombo1 | 选择小球1的质量 |
| IDC\_COMBO2 | CComboBox | m2\_cb |  | OnCbnSelchangeCombo2 | 选择小球2的质量 |
| IDC\_DRAW | CStatic | m\_pic |  |  | 显示模拟动画的区域 |
| IDC\_EDIT1 | CEdit | m1\_radius\_edit |  |  | 输入小球1的半径 |
| IDC\_EDIT2 | CEdit | m2\_radius\_edit |  |  | 输入小球2的半径 |
| IDC\_EDIT5 | CEdit | m1\_angle\_edit |  |  | 显示小球1的角度 |
| IDC\_EDIT6 | CEdit | m2\_angle\_edit |  |  | 显示小球1的角度 |
| IDC\_RADIO1 | CButton | m1\_rad1 | 5 m/s | OnBnClickedRadio1 | 选择小球1初始速度 |
| IDC\_RADIO2 | CButton | m1\_rad2 | 10m/s | OnBnClickedRadio2 | 选择小球1初始速度 |
| IDC\_RADIO3 | CButton | m1\_rad3 | 15m/s | OnBnClickedRadio3 | 选择小球1初始速度 |
| IDC\_RADIO4 | CButton | m1\_rad4 | 20m/s | OnBnClickedRadio4 | 选择小球1初始速度 |
| IDC\_RADIO5 | CButton | m2\_rad1 | 5 m/s | OnBnClickedRadio5 | 选择小球2初始速度 |
| IDC\_RADIO6 | CButton | m2\_rad2 | 10m/s | OnBnClickedRadio6 | 选择小球2初始速度 |
| IDC\_RADIO7 | CButton | m2\_rad3 | 15m/s | OnBnClickedRadio7 | 选择小球2初始速度 |
| IDC\_RADIO8 | CButton | m2\_rad4 | 20m/s | OnBnClickedRadio8 | 选择小球2初始速度 |
| IDC\_SLIDER1 | CSliderCtrl | m1\_slider |  | OnHScroll  OnVScroll | 滑动选择小球1的角度 |
| IDC\_SLIDER2 | CSliderCtrl | m2\_slider |  | OnHScroll  OnVScroll | 滑动选择小球2的角度 |
| IDCANCEL | CButton |  | 退出 |  | 退出对话框 |

两个小球的属性通过一系列的控件进行输入。每个小球都是一个ball类型的struct结构：



点击“Go Go Go！”按钮后，开始进行模拟仿真。为了防止模拟出现问题，在模拟进行中会临时禁用“Go Go Go！”按钮。

模拟分为两部分，第一部分是碰撞前，此时选择每隔10ms让小球向前移动一个速度单位，并重新绘制小球的位置和运动轨迹；第二部分是碰撞后，通过物理公式（这个公式推导真的很恶心，推导了好久好久才算出来）进行计算得到碰撞后的新速度，继续重新绘制小球的位置和运动轨迹。小球在运动过程中颜色会进行变化。



* 1. **“计算”对话框设计与实现**

建立一个新的对话框，ID为IDD\_CALC，并对对话框进行设计：



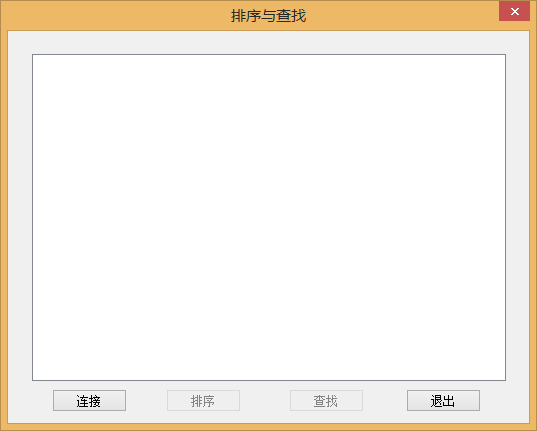
对应的控件信息如下：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | 类型 | 变量 | Captain | 响应函数 | 功能描述 |
| IDC\_BUTTON1 | CButton | m\_button1 | 平均值 | OnBnClickedButton1 | 计算平均值 |
| IDC\_BUTTON2 | CButton | m\_button2 | 方差 | OnBnClickedButton2 | 计算方差 |
| IDC\_BUTTON3 | CButton | m\_button3 | 均方根 | OnBnClickedButton3 | 计算均方根 |
| IDC\_EDIT1 | CEdit | m\_edit1 |  |  | 显示数据1 |
| IDC\_EDIT2 | CEdit | m\_edit2 |  |  | 显示数据2 |
| IDC\_EDIT3 | CEdit | m\_edit3 |  |  | 显示数据3 |
| IDC\_EDIT4 | CEdit | m\_edit4 |  |  | 显示计算结果 |
| IDC\_SCROLLBAR1 | CScrollBar | m\_sb1 |  | OnHScroll  OnVScroll | 滑动选择数据1 |
| IDC\_SCROLLBAR2 | CScrollBar | m\_sb2 |  | OnHScroll  OnVScroll | 滑动选择数据2 |
| IDC\_SLIDER1 | CSliderCtrl | m\_slider |  | OnHScroll  OnVScroll | 滑动选择数据3 |

界面上主要有三个滑块，用来设置三个数据的值。三个按钮分别调用已经写好的GetMean()、GetVariance()和GetRMS()函数类计算三个数的平均值、方差和均方根。

* 1. **“排序与查找”对话框设计与实现**

建立一个新的对话框，ID为IDD\_SORT，并对对话框进行设计：



对应的控件信息如下：

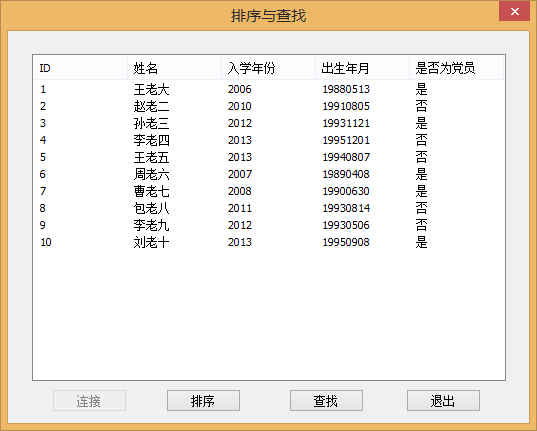
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | 类型 | 变量 | Captain | 响应函数 | 功能描述 |
| IDC\_BUTTON1 | CButton | m\_button1 | 连接 | OnBnClickedButton1 | 连接数据库 |
| IDC\_BUTTON2 | CButton | m\_button2 | 排序 | OnBnClickedButton2 | 数据排序 |
| IDC\_BUTTON3 | CButton | m\_button3 | 查找 | OnBnClickedButton3 | 数据查找 |
| IDC\_LIST1 | CListCtrl | m\_accesslist |  | OnLvnColumnclickList1 | 显示数据的区域 |
| IDCANCEL | CButton |  | 退出 |  | 退出对话框 |

由于在最初建立单文档文件时没有选择连接数据库，因此需要自己建立链接。

在stdafx.h文件的最后加上这一行：

#import "C:\Program Files\Common Files\System\ADO\msado15.dll" no\_namespace rename ("EOF", "adoEOF")

然后利用CDataBase类建立一个数据库的类m\_database，并在用户点击“连接”按钮时，连接数据库。使用相应的数据库操作函数将数据库的信息插入界面中的CListCtrl控件（View属性为Report）中，形成类似Excel的列表视图。



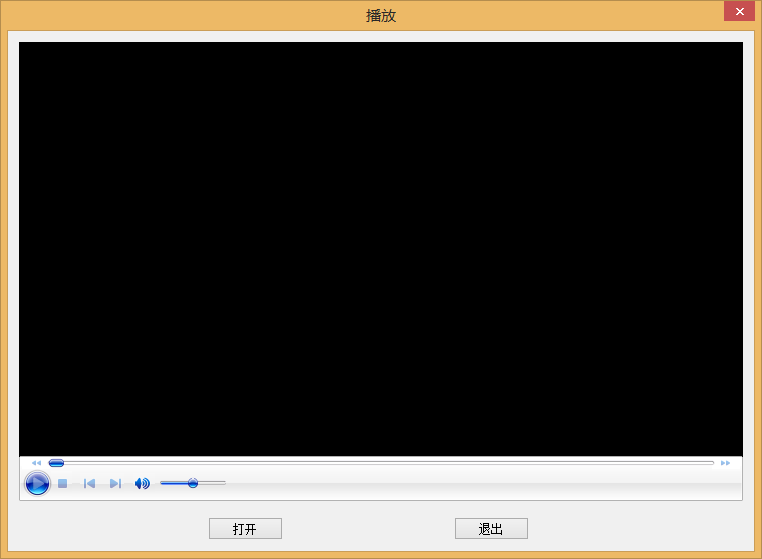
为了防止数据库连接出现错误，在数据库成功连接后将禁用“连接”按钮，同时启用“排序”和“查找”按钮。

对于“排序”按钮，为了增加排序效果，点击“排序”按钮后提示“请直接点击列名称进行排序！”的消息。在点击每一个列名称时，都会按列关键字进行行排序。这里主要是实现了OnLvnColumnclickList1事件响应函数并通过写排序函数进行排序。

对于“查找”按钮，点击“查找”时会出现一个对话框，用于提醒用户输入需要查找的信息。用户点击确定后会进行查找，找到会提示用户并将对应的列高亮显示。

* 1. **“播放”对话框设计与实现**

建立一个新的对话框，ID为IDD\_PLAY，并对对话框进行设计：



对应的控件信息如下：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | 类型 | 变量 | Captain | 响应函数 | 功能描述 |
| IDC\_BUTTON1 | CButton | m\_open | 打开 | OnBnClickedButton1 | 打开媒体文件 |
| IDC\_OCX1 | CWMPPlayer4 | m\_player |  |  | 播放媒体文件 |
| IDCANCEL | CButton |  | 退出 |  | 退出对话框 |

需要使用Windows Media Player中的CWMPPlayer4的类文件中相应的接口函数，不过不知道为什么在加入MFC对应的ActiveX控件时无法找到Windows Media Player控件，因此从网上下载了相应的CWMPPlayer4.cpp文件并添加进入该工程，并在DoDataExchange函数中加入语句：

DDX\_Control(pDX, IDC\_OCX1, m\_player);//绑定播放器控件

打开文件使用CFileDialog类，媒体文件描述为：

"媒体文件(\*.wmv,\*.mp3,\*.avi,,\*.rm,\*.rmvb,\*.mkv,\*.mp4)|\*.wmv;\*.mp3;\*.avi;\*.rm;\*.rmvb; \*.mkv;\*.mp4|"

打开后调用m\_player.put\_URL函数即可进行播放。