

系统分析与设计课程设计实验报告

学号：

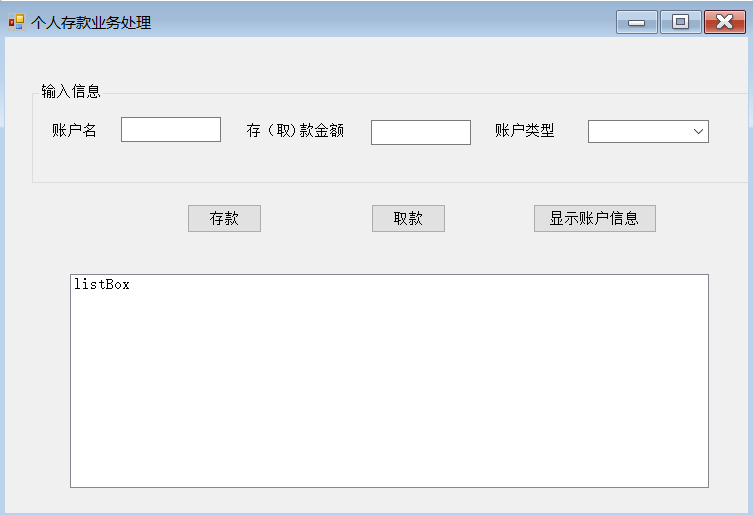
姓名：

指导老师：

完成时间：

# 实验三

## 实验结果



## 主要代码

FormMain.cs

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Text;

using System.Windows.Forms;

namespace BankCustoms

{

public partial class FormMain : Form

{

private enum CustomType

{

活期存款,

定期存款

}

public FormMain()

{

InitializeComponent();

//将存款类型值添入到下拉列表框中

comboBoxAccountType.Items.AddRange(Enum.GetNames(typeof(CustomType)));

}

//活期存款

private void CreateCustom(CustomType customType, string userName, double initMoney)

{

if (customType == CustomType.活期存款)

{

try

{

CheckingCustom newCustom = new CheckingCustom(userName, initMoney);

checkingCustomers.Add(userName, newCustom);

AddItemToListBox(string.Format("{0}开户{1}元", userName, initMoney), newCustom);

}

catch (Exception err)

{

MessageBox.Show(err.Message);

}

}

else

{

//定期存款

try

{

FixedCustom newCustom = new FixedCustom(userName, initMoney);

fixedCustomers.Add(userName, newCustom);

AddItemToListBox(string.Format("{0}开户{1}元", userName, initMoney), newCustom);

}

catch (Exception err)

{

MessageBox.Show(err.Message);

}

}

}

//将活期存款操作信息添加到状态信息列表中

private void AddItemToListBox(string prompt, CheckingCustom custom)

{

listBoxStatus.Items.Add(string.Format("{0}, 帐户：帐户名{1}，账号{2:d4}，当前余额{3:n2}元",

prompt, custom.AccountName, custom.AccountNumber, custom.AccountBalance));

}

//将定期存款操作信息添加到状态信息列表中

private void AddItemToListBox(string prompt, FixedCustom custom)

{

listBoxStatus.Items.Add(string.Format("{0}, 帐户：帐户名{1}，账号{2:d4}，当前余额{3:n2}元",

prompt, custom.AccountName, custom.AccountNumber, custom.AccountBalance));

}

//存款button操作

private void buttonDeposit\_Click(object sender, EventArgs e)

{

double money;

if (double.TryParse(textBoxMoney.Text, out money) == false)

{

MessageBox.Show("存款金额错误", "无法存款", MessageBoxButtons.OK, MessageBoxIcon.Question);

return;

}

if (comboBoxAccountType.SelectedItem.ToString() == "活期存款")

{

if (checkingCustomers.ContainsKey(textBoxUserName.Text) == false)

{

CreateCustom(CustomType.活期存款, textBoxUserName.Text, money);

}

else

{

try

{

CheckingCustom custom = checkingCustomers[textBoxUserName.Text];

custom.Diposit(money);

AddItemToListBox(string.Format("{0}存款{1}元", textBoxUserName.Text, money), custom);

}

catch (Exception err)

{

MessageBox.Show(err.Message);

}

}

}

else

{

if (fixedCustomers.ContainsKey(textBoxUserName.Text) == false)

{

CreateCustom(CustomType.定期存款, textBoxUserName.Text, money);

}

else

{

try

{

FixedCustom custom = fixedCustomers[textBoxUserName.Text];

custom.Diposit(money);

AddItemToListBox(string.Format("{0}存款{1}元", textBoxUserName.Text, money), custom);

}

catch (Exception err)

{

MessageBox.Show(err.Message);

}

}

}

}

//取款button操作

private void buttonWithdraw\_Click(object sender, EventArgs e)

{

double money;

if (double.TryParse(textBoxMoney.Text, out money) == false)

{

MessageBox.Show("取款金额错误", "无法取款", MessageBoxButtons.OK, MessageBoxIcon.Question);

return;

}

if (comboBoxAccountType.SelectedItem.ToString() == "活期存款")

{

if (checkingCustomers.ContainsKey(textBoxUserName.Text) == false)

{

MessageBox.Show("无此帐户 ", "取款失败", MessageBoxButtons.OK, MessageBoxIcon.Question);

}

else

{

try

{

CheckingCustom custom = checkingCustomers[textBoxUserName.Text];

//获取结算前活期存款余额

double balance = custom.AccountBalance;

//取款

custom.Withdraw(money);

AddItemToListBox(string.Format("{0}取款{1}元，结算利息{2:n2}元", textBoxUserName.Text, money, custom.AccountBalance + money - balance), custom);

}

catch (Exception err)

{

MessageBox.Show(err.Message);

}

}

}

else

{

if (fixedCustomers.ContainsKey(textBoxUserName.Text) == false)

{

MessageBox.Show("无此帐户", "取款失败", MessageBoxButtons.OK, MessageBoxIcon.Question);

}

else

{

try

{

FixedCustom custom = fixedCustomers[textBoxUserName.Text];

//获取结算前定期存款余额

double balance = custom.AccountBalance;

//取款

custom.Withdraw(money);

AddItemToListBox(string.Format("{0}取款{1}元，结算利息{2:n2}元", textBoxUserName.Text, money, custom.AccountBalance + money - balance), custom);

}

catch (Exception err)

{

MessageBox.Show(err.Message);

}

}

}

}

//显示账户信息

private void buttonShowCustoms\_Click(object sender, EventArgs e)

{

listBoxStatus.Items.Clear();

for (int i = 0; i < checkingCustomers.Count; i++)

{

AddItemToListBox("活期存款", checkingCustomers.Values[i]);

}

for (int i = 0; i < fixedCustomers.Count; i++)

{

AddItemToListBox("定期存款", fixedCustomers.Values[i]);

}

}

}

}

Custom.cs

using System;

using System.Collections.Generic;

using System.Text;

namespace BankCustoms

{

class Custom

{

private string accountName; //帐户名

//外部无法访问

protected string AccountName

{

get

{

return accountName;

}

set

{

if (string.IsNullOrEmpty(value) == true)

{

throw new Exception("帐户名不能为空 ");

}

else

{

for (int i = 0; i < value.Length; i++)

{

if (char.IsLetter(value[i]) == false)

{

throw new Exception("帐户名中不能有非法字符");

}

else

{

accountName = value;

}

}

}

}

}

private int accountNumber; //帐户号

protected int AccountNumber

{

get

{

return accountNumber;

}

set

{

accountNumber = value;

}

}

private double accountBalance; //帐户余额

protected double AccountBalance

{

get

{

return accountBalance;

}

set

{

if (value <= 0)

{

throw new Exception("帐户余额需为正数");

}

else

{

accountBalance = value;

}

}

}

//存款

public void Diposit(double money)

{

if (money <= 0)

{

throw new Exception("存款余额需为正数");

}

else

{

//修改余额

accountBalance += money;

}

}

//取款

public virtual void Withdraw(double money)

{

if (money <= 0)

{

throw new Exception("取款余额需为正数");

}

else

{

if (money > accountBalance)

{

throw new Exception("错误！取款数大于余额");

}

else

{

accountBalance -= money;

}

}

}

}

}

CheckingCustom.cs

using System;

using System.Collections.Generic;

using System.Text;

namespace BankCustoms

{

class CheckingCustom : Custom

{

private static int newAccountNumber = 1;

public float InterestRate //活期利率

{

get

{

return 0.005f;

}

}

public new string AccountName

{

get

{

return base.AccountName;

}

}

public new int AccountNumber

{

get

{

return base.AccountNumber;

}

}

public new double AccountBalance

{

get

{

return base.AccountBalance;

}

}

//构造函数

public CheckingCustom(string accountName, double accountBalance)

{

if (newAccountNumber > 4999)

{

throw new Exception("帐户号已用完");

}

else

{

base.AccountName = accountName;

base.AccountBalance = accountBalance;

base.AccountNumber = newAccountNumber;

newAccountNumber++;

}

}

//取款

public override void Withdraw(double money)

{

base.AccountBalance += InterestRate \* AccountBalance;

//调用Withdraw取款

base.Withdraw(money);

}

}

}

//设计器

namespace BankCustoms

{

partial class FormMain

{

private System.ComponentModel.IContainer components = null;

protected override void Dispose(bool disposing)

{

if (disposing && (components != null))

{

components.Dispose();

}

base.Dispose(disposing);

}

#region Windows 窗体设计器生成的代码

private void InitializeComponent()

{

this.comboBoxAccountType = new System.Windows.Forms.ComboBox();

this.buttonWithdraw = new System.Windows.Forms.Button();

this.groupBox2 = new System.Windows.Forms.GroupBox();

this.textBoxUserName = new System.Windows.Forms.TextBox();

this.textBoxMoney = new System.Windows.Forms.TextBox();

this.label6 = new System.Windows.Forms.Label();

this.label4 = new System.Windows.Forms.Label();

this.label5 = new System.Windows.Forms.Label();

this.listBoxStatus = new System.Windows.Forms.ListBox();

this.buttonDeposit = new System.Windows.Forms.Button();

this.buttonShowCustoms = new System.Windows.Forms.Button();

this.groupBox2.SuspendLayout();

this.SuspendLayout();

this.comboBoxAccountType.FormattingEnabled = true;

this.comboBoxAccountType.Location = new System.Drawing.Point(436, 19);

this.comboBoxAccountType.Name = "comboBoxAccountType";

this.comboBoxAccountType.Size = new System.Drawing.Size(114, 20);

this.comboBoxAccountType.TabIndex = 2;

this.buttonWithdraw.Location = new System.Drawing.Point(253, 88);

this.buttonWithdraw.Name = "buttonWithdraw";

this.buttonWithdraw.Size = new System.Drawing.Size(113, 23);

this.buttonWithdraw.TabIndex = 2;

this.buttonWithdraw.Text = "取款";

this.buttonWithdraw.UseVisualStyleBackColor = true;

this.buttonWithdraw.Click += new System.EventHandler(this.buttonWithdraw\_Click);

this.groupBox2.Controls.Add(this.comboBoxAccountType);

this.groupBox2.Controls.Add(this.textBoxUserName);

this.groupBox2.Controls.Add(this.textBoxMoney);

this.groupBox2.Controls.Add(this.label6);

this.groupBox2.Controls.Add(this.label4);

this.groupBox2.Controls.Add(this.label5);

this.groupBox2.Location = new System.Drawing.Point(15, 13);

this.groupBox2.Name = "groupBox2";

this.groupBox2.Size = new System.Drawing.Size(590, 52);

this.groupBox2.TabIndex = 18;

this.groupBox2.TabStop = false;

this.groupBox2.Text = "输入信息";

this.textBoxUserName.Location = new System.Drawing.Point(89, 19);

this.textBoxUserName.Name = "textBoxUserName";

this.textBoxUserName.Size = new System.Drawing.Size(75, 21);

this.textBoxUserName.TabIndex = 0;

this.textBoxMoney.Location = new System.Drawing.Point(275, 19);

this.textBoxMoney.Name = "textBoxMoney";

this.textBoxMoney.Size = new System.Drawing.Size(86, 21);

this.textBoxMoney.TabIndex = 1;

this.label6.AutoSize = true;

this.label6.Location = new System.Drawing.Point(377, 24);

this.label6.Name = "label6";

this.label6.Size = new System.Drawing.Size(53, 12);

this.label6.TabIndex = 0;

this.label6.Text = "帐户类型";

this.label4.AutoSize = true;

this.label4.Location = new System.Drawing.Point(42, 22);

this.label4.Name = "label4";

this.label4.Size = new System.Drawing.Size(41, 12);

this.label4.TabIndex = 0;

this.label4.Text = "帐户名";

this.label5.AutoSize = true;

this.label5.Location = new System.Drawing.Point(180, 24);

this.label5.Name = "label5";

this.label5.Size = new System.Drawing.Size(89, 12);

this.label5.TabIndex = 0;

this.label5.Text = "存（取）款金额";

this.listBoxStatus.FormattingEnabled = true;

this.listBoxStatus.ItemHeight = 12;

this.listBoxStatus.Location = new System.Drawing.Point(10, 131);

this.listBoxStatus.Name = "listBoxStatus";

this.listBoxStatus.Size = new System.Drawing.Size(605, 148);

this.listBoxStatus.TabIndex = 4;

this.buttonDeposit.Location = new System.Drawing.Point(104, 88);

this.buttonDeposit.Name = "buttonDeposit";

this.buttonDeposit.Size = new System.Drawing.Size(111, 23);

this.buttonDeposit.TabIndex = 1;

this.buttonDeposit.Text = "存款";

this.buttonDeposit.UseVisualStyleBackColor = true;

this.buttonDeposit.Click += new System.EventHandler(this.buttonDeposit\_Click);

this.buttonShowCustoms.Location = new System.Drawing.Point(407, 88);

this.buttonShowCustoms.Name = "buttonShowCustoms";

this.buttonShowCustoms.Size = new System.Drawing.Size(112, 23);

this.buttonShowCustoms.TabIndex = 3;

this.buttonShowCustoms.Text = "显示帐户信息";

this.buttonShowCustoms.UseVisualStyleBackColor = true;

this.buttonShowCustoms.Click += new System.EventHandler(this.buttonShowCustoms\_Click);

this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 12F);

this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;

this.ClientSize = new System.Drawing.Size(623, 293);

this.Controls.Add(this.buttonWithdraw);

this.Controls.Add(this.groupBox2);

this.Controls.Add(this.listBoxStatus);

this.Controls.Add(this.buttonDeposit);

this.Controls.Add(this.buttonShowCustoms);

this.Name = "FormMain";

this.Text = "个人存款业务处理";

this.groupBox2.ResumeLayout(false);

this.groupBox2.PerformLayout();

this.ResumeLayout(false);

}

private System.Windows.Forms.ComboBox comboBoxAccountType;

private System.Windows.Forms.Button buttonWithdraw;

private System.Windows.Forms.GroupBox groupBox2;

private System.Windows.Forms.TextBox textBoxUserName;

private System.Windows.Forms.TextBox textBoxMoney;

private System.Windows.Forms.Label label6;

private System.Windows.Forms.Label label4;

private System.Windows.Forms.Label label5;

private System.Windows.Forms.ListBox listBoxStatus;

private System.Windows.Forms.Button buttonDeposit;

private System.Windows.Forms.Button buttonShowCustoms;

}

}

using System;

using System.Collections.Generic;

using System.Windows.Forms;

namespace BankCustoms

{

static class Program

{

[STAThread]

static void Main()

{

Application.EnableVisualStyles();

Application.SetCompatibleTextRenderingDefault(false);

Application.Run(new FormMain());

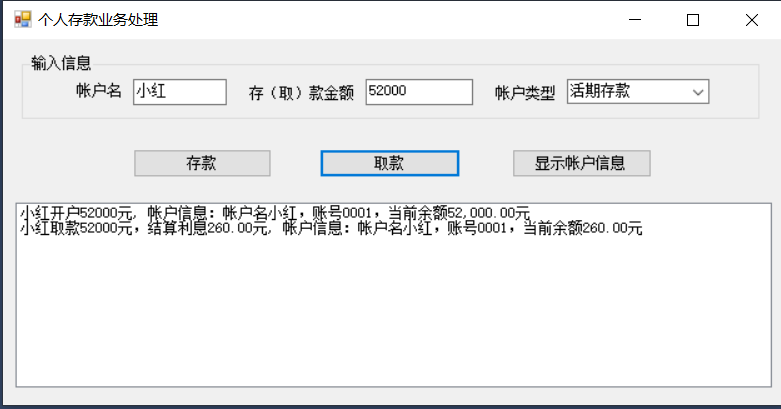
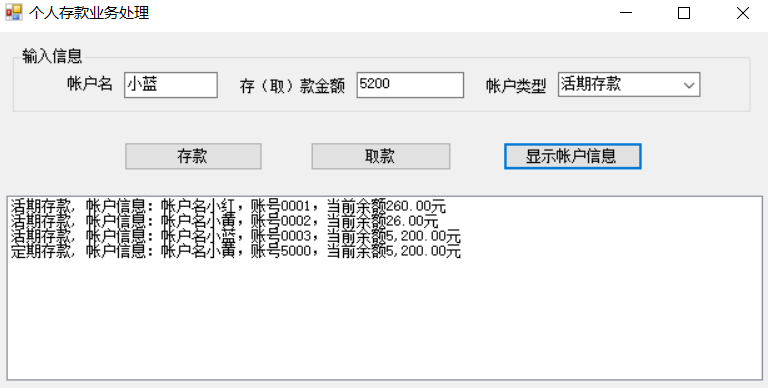
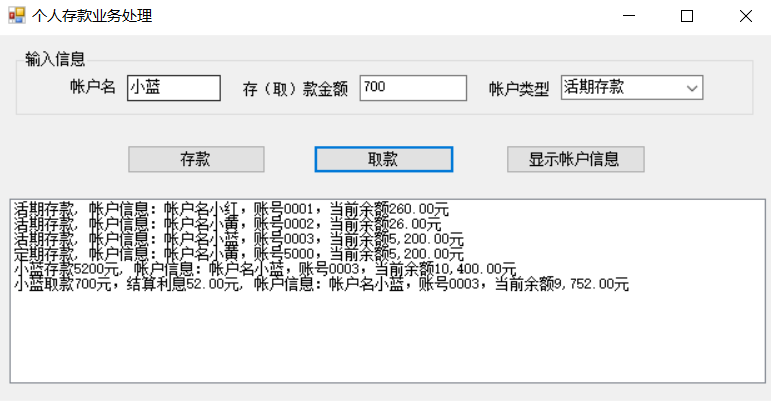
}

}

}

## 问题及解答

**(1). 画出有代表性的包含存款、取款和显示帐户信息的程序运行界面，运行界面可以使用多个图表示。**

**(2). 写出实验中遇到的问题及解决方法。如何将帐户信息保存到数据库中，简要回答（不需要写实现代码，只要求提供设计思路） 应该怎样实现存取款业务？**

**问题及解决方法：**

问题：账户存款追加问题

解决方法：创建活期存款和定期存款对象，分别调用Diposit方法和Withdraw方法。

解决方式：保存所有活期存款帐户，用帐户名做键，同一个人只能有一个活期帐号，static SortedList<string, CheckingCustom> checkingCustomers = new SortedList<string, CheckingCustom>();保存所有定期存款帐户，用帐户名做键，同一个人只能有一个定期帐号，static SortedList<string, FixedCustom> fixedCustomers = new SortedList<string, FixedCustom>();

设计CustomType函数，定义活期存款和定期存款对象；

设计CreateCustom函数，利用if条件判断存款类型进行存款对象开户创建；

设计AddItemToListBox函数，利用listBoxStatus.Items.Add将存款信息写入信息列表；

设计buttonDeposit\_Click和buttonWithdraw\_Click函数，创建custom对象，函数内部分别调用Diposit函数和Withdraw函数实现存取款业务；