ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ «МОРДОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИМ. Н.П. ОГАРЁВА»

Направление: Информатика и вычислительная техника

**ЛАБОРАТОРНАЯ РАБОТА**

по дисциплине

«Программная Инженерия»

ЛР – 02069964 – Программная инженерия – 09 – 19

Лабораторная работа № 9

Выполнили: Проверил:

студенты 441 группы

05.12.2019 \_\_\_\_\_\_\_\_\_\_\_

С. О. Леушкин С.А. Федосин

Д.Д. Абдуллаев

Саранск 2019

Лабораторная работа 9

Создание объектов СОМ в Delphi

**Цели.**

* ознакомиться с технологией СОМ;
* научиться создавать СОМ-сервера и клиенты СОМ-объекта в Delphi XE.

**Порядок выполнения работы.**

##### **1. Создание внутреннего сервера СОМ**

Процесс создания внут­реннего сервера СОМ можно представить как последовательность следующих этапов:

1. Создание динамической библиотеки.
2. Создание объекта СОМ.
3. Создание интерфейсов объекта СОМ.
4. Задание кода методов интерфейса.

Код программы: Main.pas

unit main;

interface

uses

Winapi.Windows, Winapi.Messages, System.SysUtils, System.Variants, System.Classes, Vcl.Graphics,

Vcl.Controls, Vcl.Forms, Vcl.Dialogs, Vcl.StdCtrls, TestComServer\_Lab9\_TLB;

type

TForm1 = class(TForm)

ListBox1: TListBox;

GroupBox1: TGroupBox;

GroupBox2: TGroupBox;

Label1: TLabel;

Label2: TLabel;

Label3: TLabel;

Label4: TLabel;

Label5: TLabel;

Label6: TLabel;

Label7: TLabel;

Label8: TLabel;

Button1: TButton;

Button2: TButton;

GroupBox3: TGroupBox;

Label9: TLabel;

Label10: TLabel;

Label11: TLabel;

Label12: TLabel;

Button3: TButton;

procedure FormShow(Sender: TObject);

procedure Button1Click(Sender: TObject);

procedure Button2Click(Sender: TObject);

procedure Button3Click(Sender: TObject);

private

{ Private declarations }

public

Stat: ITestComLab9;

MinMax : IMinMax;

NumAverage: INumAverage;

{ Public declarations }

end;

var

Form1: TForm1;

implementation

{$R \*.dfm}

procedure TForm1.Button1Click(Sender: TObject);

var

V: OleVariant;

i: Integer;

begin

{ Создаем вариантный массив и заполняем созданный массив значениями из списка}

V:=varArrayCreate([0,ListBox1.Items.Count-1],varSingle);

for i:=0 to ListBox1.Items.Count-1 do

V[i]:=StrToFloat(ListBox1.Items[i]);

{ Вызываем методы интерфейса ITestCom и выводим результаты на форму}

Label2.Caption:=FloatToStrF(Stat.Avg(V),ffGeneral,4,2);

Label4.Caption:=FloatToStrF(Stat.Variance(V),FfGeneral,2,2);

end;

procedure TForm1.Button2Click(Sender: TObject);

var

V: OleVariant;

i: Integer;

begin

{ Создаем вариантный массив и заполняем созданный массив значениями из списка}

V:=varArrayCreate([0,ListBox1.Items.Count-1],varSingle);

for i:=0 to ListBox1.Items.Count-1 do

V[i]:=StrToFloat(ListBox1.Items[i]);

{ Вызываем методы интерфейса IMinMax и выводим результаты на форму}

Label6.Caption:=FloatToStrF(MinMax.Min(V),ffGeneral,4,2);

Label8.Caption:=FloatToStrF(MinMax.Max(V),ffGeneral,2,2);

end;

procedure TForm1.Button3Click(Sender: TObject);

var

V: OleVariant;

i: Integer;

begin

{ Создаем вариантный массив и заполняем созданный массив значениями из списка}

V:=varArrayCreate([0,ListBox1.Items.Count-1],varSingle);

for i:=0 to ListBox1.Items.Count-1 do

V[i]:=StrToFloat(ListBox1.Items[i]);

Label11.Caption:=FloatToStrF(NumAverage.Av\_geom(V),ffGeneral,4,2);

Label12.Caption:=FloatToStrF(NumAverage.Av\_harm(V),ffGeneral,2,2);

end;

procedure TForm1.FormShow(Sender: TObject);

begin

Stat:=CoTestComLab9.Create;

Stat.QueryInterface(IMinMax,MinMax);

Stat.QueryInterface(INumAverage,NumAverage);

end;

end.

Код программы TestComLab9.pas:

unit TestComLab9;

{$WARN SYMBOL\_PLATFORM OFF}

interface

uses

Windows, ActiveX, Classes, ComObj, TestComServer\_Lab9\_TLB, StdVcl, Variants;

type

TTestComLab9 = class(TTypedComObject, ITestComLab9)

protected

function Avg(Param1: OleVariant): Single; stdcall;

function Variance(Param1: OleVariant): Single; stdcall;

function Max(Param1: OleVariant): Single; stdcall;

function Min(Param1: OleVariant): Single; stdcall;

function Av\_geom(Param1: OleVariant): Single; stdcall;

function Av\_harm(Param1: OleVariant): Single; stdcall;

end;

implementation

uses ComServ;

function TTestComLab9.Avg(Param1: OleVariant): Single;

var

N1,N2,i : Integer;

A : Single;

begin

A:=0;

N1:=varArrayLowBound(Param1,1);

N2:=varArrayHighBound(Param1,1);

for i:=N1 to N2 do

A:=A+Param1[i];

result:=A/(N2-N1+1);

end;

function TTestComLab9.Variance(Param1: OleVariant): Single;

var

N1,N2,i : Integer;

A,D : Single;

begin

A:=Avg(Param1);

D:=0;

N1:=varArrayLowBound(Param1,1);

N2:=varArrayHighBound(Param1,1);

for i:=N1 to N2 do

D:=sqr(Param1[i]-A);

result:=D/(N2-N1);

end;

function TTestComLab9.Max(Param1: OleVariant): Single;

var

N1, N2, I : Integer;

M : Single;

begin

N1:= VarArrayLowBound(Param1,1);

N2:= varArrayHighBound(Param1,1);

M:=Param1[N1];

for i:=N1 to N2 do

if Param1[i]>M then M:=Param1[i];

result:=M;

end;

function TTestComLab9.Min(Param1: OleVariant): Single;

var

N1,N2,i : Integer;

M : Single;

begin

N1:=varArrayLowBound(Param1,1);

N2:=varArrayHighBound(Param1,1);

M:=Param1[N1];

for i:=N1 to N2 do

if Param1[i]<M then M:=Param1[i];

result:=M;

end;

function TTestComLab9.Av\_geom(Param1: OleVariant): Single;

var

Xg:Single;

i,N1,N2:Integer;

Begin

Xg:=1;

N1:=varArrayLowBound(Param1,1);

N2:=varArrayHighBound(Param1,1);

for i:=N1 to N2 do

Xg:=Xg\*Param1[i];

Xg:=Exp(Ln(Xg)/((N2-N1)-1));

Result:=Xg;

End;

function TTestComLab9.Av\_harm(Param1: OleVariant): Single;

var

i,N1,N2:Integer;

sum:Single;

begin

sum:=0;

N1:=varArrayLowBound(Param1,1);

N2:=varArrayHighBound(Param1,1);

for i:=N1 to N2 do

sum:=sum+(1/Param1[i]);

Result:=(N2-N1)/sum;

end;

initialization

TTypedComObjectFactory.Create(ComServer, TTestComLab9, Class\_TestComLab9,

ciMultiInstance, tmSingle);

end.

Код программы TestComServer\_Lab9\_TLB.pas:

unit TestComServer\_Lab9\_TLB;

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* //

// WARNING

// -------

// The types declared in this file were generated from data read from a

// Type Library. If this type library is explicitly or indirectly (via

// another type library referring to this type library) re-imported, or the

// 'Refresh' command of the Type Library Editor activated while editing the

// Type Library, the contents of this file will be regenerated and all

// manual modifications will be lost.

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* //

// $Rev: 52393 $

// File generated on 05.12.2019 15:57:57 from Type Library described below.

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* //

// Type Lib: D:\Students\LAB9\TestComServer\_Lab9 (1)

// LIBID: {F50EAB0B-6D38-43D8-ACBB-7824C5790A13}

// LCID: 0

// Helpfile:

// HelpString:

// DepndLst:

// (1) v2.0 stdole, (C:\Windows\SysWOW64\stdole2.tlb)

// SYS\_KIND: SYS\_WIN32

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* //

{$TYPEDADDRESS OFF} // Unit must be compiled without type-checked pointers.

{$WARN SYMBOL\_PLATFORM OFF}

{$WRITEABLECONST ON}

{$VARPROPSETTER ON}

{$ALIGN 4}

interface

uses Winapi.Windows, System.Classes, System.Variants, System.Win.StdVCL, Vcl.Graphics, Vcl.OleServer, Winapi.ActiveX;

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*//

// GUIDS declared in the TypeLibrary. Following prefixes are used:

// Type Libraries : LIBID\_xxxx

// CoClasses : CLASS\_xxxx

// DISPInterfaces : DIID\_xxxx

// Non-DISP interfaces: IID\_xxxx

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*//

const

// TypeLibrary Major and minor versions

TestComServer\_Lab9MajorVersion = 1;

TestComServer\_Lab9MinorVersion = 0;

LIBID\_TestComServer\_Lab9: TGUID = '{F50EAB0B-6D38-43D8-ACBB-7824C5790A13}';

IID\_ITestComLab9: TGUID = '{14CCDFDF-3DE4-46BE-859B-9426343FBB94}';

CLASS\_TestComLab9: TGUID = '{EB57B2A5-9B0A-4417-9475-D04321D3968C}';

IID\_IMinMax: TGUID = '{6ECF45A5-B4FA-4BD6-8DD6-586F216F3BC5}';

IID\_INumAverage: TGUID = '{16D1C3B2-5ED7-420E-AE56-C45343ACD1DB}';

type

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*//

// Forward declaration of types defined in TypeLibrary

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*//

ITestComLab9 = interface;

IMinMax = interface;

IMinMaxDisp = dispinterface;

INumAverage = interface;

INumAverageDisp = dispinterface;

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*//

// Declaration of CoClasses defined in Type Library

// (NOTE: Here we map each CoClass to its Default Interface)

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*//

TestComLab9 = ITestComLab9;

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*//

// Interface: ITestComLab9

// Flags: (0)

// GUID: {14CCDFDF-3DE4-46BE-859B-9426343FBB94}

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*//

ITestComLab9 = interface(IUnknown)

['{14CCDFDF-3DE4-46BE-859B-9426343FBB94}']

function Avg(Param1: OleVariant): Single; stdcall;

function Variance(Param1: OleVariant): Single; stdcall;

end;

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*//

// Interface: IMinMax

// Flags: (4416) Dual OleAutomation Dispatchable

// GUID: {6ECF45A5-B4FA-4BD6-8DD6-586F216F3BC5}

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*//

IMinMax = interface(IDispatch)

['{6ECF45A5-B4FA-4BD6-8DD6-586F216F3BC5}']

function Min(Param1: OleVariant): Single; stdcall;

function Max(Param1: OleVariant): Single; stdcall;

end;

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*//

// DispIntf: IMinMaxDisp

// Flags: (4416) Dual OleAutomation Dispatchable

// GUID: {6ECF45A5-B4FA-4BD6-8DD6-586F216F3BC5}

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*//

IMinMaxDisp = dispinterface

['{6ECF45A5-B4FA-4BD6-8DD6-586F216F3BC5}']

function Min(Param1: OleVariant): Single; dispid 201;

function Max(Param1: OleVariant): Single; dispid 202;

end;

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*//

// Interface: INumAverage

// Flags: (4416) Dual OleAutomation Dispatchable

// GUID: {16D1C3B2-5ED7-420E-AE56-C45343ACD1DB}

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*//

INumAverage = interface(IDispatch)

['{16D1C3B2-5ED7-420E-AE56-C45343ACD1DB}']

function Av\_harm(Param1: OleVariant): Single; stdcall;

function Av\_geom(Param1: OleVariant): Single; stdcall;

end;

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*//

// DispIntf: INumAverageDisp

// Flags: (4416) Dual OleAutomation Dispatchable

// GUID: {16D1C3B2-5ED7-420E-AE56-C45343ACD1DB}

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*//

INumAverageDisp = dispinterface

['{16D1C3B2-5ED7-420E-AE56-C45343ACD1DB}']

function Av\_harm(Param1: OleVariant): Single; dispid 201;

function Av\_geom(Param1: OleVariant): Single; dispid 202;

end;

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*//

// The Class CoTestComLab9 provides a Create and CreateRemote method to

// create instances of the default interface ITestComLab9 exposed by

// the CoClass TestComLab9. The functions are intended to be used by

// clients wishing to automate the CoClass objects exposed by the

// server of this typelibrary.

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*//

CoTestComLab9 = class

class function Create: ITestComLab9;

class function CreateRemote(const MachineName: string): ITestComLab9;

end;

implementation

uses System.Win.ComObj;

class function CoTestComLab9.Create: ITestComLab9;

begin

Result := CreateComObject(CLASS\_TestComLab9) as ITestComLab9;

end;

class function CoTestComLab9.CreateRemote(const MachineName: string): ITestComLab9;

begin

Result := CreateRemoteComObject(MachineName, CLASS\_TestComLab9) as ITestComLab9;

end;

end.

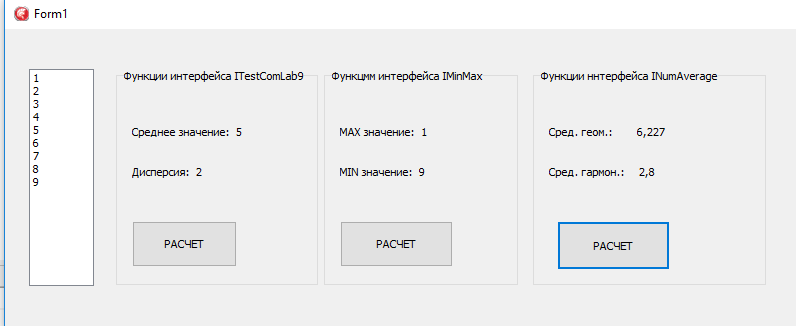


Рисунок 1 – Результат выполнения программы COM

Приложение будет работать корректно, только при успешной регистрации на сервере Active X.

### Контрольные вопросы.

1. Как расшифровывается аббревиатура COM?

**-** COM – компонент модели объекта.

2. Перечислите этапы создания внутреннего сервера COM.

- Создание динамической библиотеки.

- Создание объекта СОМ.

- Создание интерфейсов объекта СОМ.

- Задание кода методов интерфейса.

3. Как создать динамическую библиотеку сервера COM?

- Динамическая библиотека, которая является внутренним сервером СОМ, создается в виде отдельного проекта Delphi. Для создания такой библиотеки следует выбрать команду File | New | Other… главного меню Delphi XE и затем выбрать на вкладке ActiveX открывшегося окна хранилища объектов значок AciveX Library.

4. Как создать объект COM?

**-** Для этого выберите команду File | New | Other… главного меню Delphi XE и выберите на вкладке ActiveX открывшегося окна диалога New Items значок COM Object.

5. Как создать новый интерфейс объекта COM?

- Обычно для создания библиотек типов используется специальный язык IDL (Interface Description Language — язык описания интерфейсов). Синтаксис данного языка похож на синтаксис языка C++. Однако в Delphi при создании библиотеки типов обычно используется синтаксис языка Object Pascal.

При разработке библиотек типов в Delphi, как правило, не требуется писать код вручную, так как код библиотеки типов генерируется автоматически, а для внесения необходимых изменений и дополнений используется специальный редактор.

6. Как получить указатель на основной интерфейс объекта COM?

- Для создания экземпляра объекта СОМ используется метод Create составного класса (CoClass) объекта. Данный метод возвращает указатель на основной интерфейс объекта СОМ.

7. Как получить указатель на не основной интерфейс объекта COM?

- Для получения указателя на какой-либо другой интерфейс следует воспользоваться методом QueryInterface основного интерфейса.