

```

:
TSrings
Lines      ListBox      Items
          . Tstrings      Memo
          TStringList.Create      TStrings
          . Free
Tstrings      Free
          . ( )
          :
          : 1
          :
          Edit      ListBox
          :
var a:TStrings;
Begin
a := TStringList.Create; //
a.Add(Edit1.text); //
Form1.Caption := a.Strings [0]; //      ( 0 )
ListBox1.Items.AddStrings(a); // ListBox      a
a.Free ;
end;

TsrtingList.Create
: 2
TStrings
:
:
Public      TStrings      A
:      OnCreat
A := TStringList.Create;

```


EconvertError



Exception1

Exception1

Var i :integer;

Begin

Try

i:= strtoint(edit1.text);

Except

Showmessage ('المربعاء is not a valid integer value.')

end;

Except

(

. Try

Try . . End

:

Edit

:

: 2

Edit

var i,j :integer;z:real;

Begin

Try

i:= strtoint(edit1.text);

j:= strtoint(edit2.text);

z:= i/j;

edit3.Text := floattostr(z);

Except

http://www.alshater.net

```
Showmessage ( ' ');
```

```
end;
```

```
    Edit1
```

```
    Except
```

```
End Except
```

```
Except Try
```

```
    Edit2
```

```
    / 0 / Edit2
```

```
    /232/ Edit1
```

```
( " ")
```

```
EzeorDivide
```

```
:
```

```
Var i,j :integer;z:real;
```

```
Begin
```

```
Try
```

```
i:= strtoint(edit1.text);
```

```
j:= strtoint(edit2.text);
```

```
z:= i/j;
```

```
edit3.Text := floattostr(z);
```

```
Except
```

```
On Econverterror Do Showmessage ( ' ');
```

```
On EZeroDivide Do shOwmessag(' ');
```

```
end;
```

```
Do EconvertError
```

```
: On Econverterror Do
```

```
...
```

```
: EmathError )
```

```
EzerDivide )
```

```
( .... EintError
```

```
( ErangeError
```

```
EconvertError
```

```
.....
```

```
: Try ... Finally 2
```

```
:
```

```
: 1
```

```
:
```

```
Var a:tstrings ;
```

```
i:integer;
```

```
begin
```

```
a:= Tstringlist.Create ; //
```

```

http://www.alshater.net
i:= strtoint(edit1.text);//
i:= i * 5;
a.Add(inttostr(i)); //
Form1.Caption := 'Done'
a.Free ;

```

```

Edit
a.Free
EConvertErrorr
) a
Form1.Caption := 'Done'
(
:

```

```

Var a:tstrings ;
i:integer;
begin
Try
a:= Tstringlist.Create ; //
i:= strtoint(edit1.text);//
i:= i * 5;

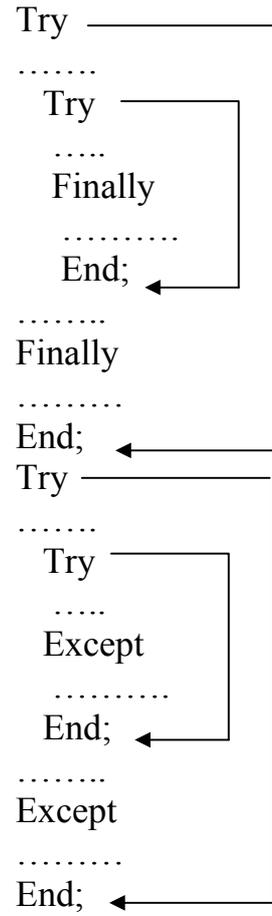
a.Add(inttostr(i));
Finally
Form1.Caption := 'Done' ;
a.Free ;
end;

```

```

Finally Run
. ( )
Finally Try Finally
. ( Finally Edit )
: Try
Try

```



) (Except) Try (Finally
 :
 (Bugs)
) (Debug

: 1

Run

Trace into



Run

Step Over

F7



. F8

Trace Into

Trace Into

Step Over

:

ListBox

```

Var s:string ;
i , j ,z :integer;
begin
for i := 0 to 500 do
begin
j:= i * 5;
z:= j * 2 ;
s:= inttostr ( j ) + ' ' + inttostr(z);
Form1.Caption := s;
listbox1.Items.Add(s)

```

Trace Into

F7

F7

F7

listbox1.Items.Add(s)

F7

I

I

....S Z

j:= i * 5

F7

I

s Z j

I

:

Trace Into

Step Over

Step Over

Application.Run

: Run To Cursor

Run To Cursor

F4

Run

:

:

Run To Cursor

Add Breakpoint

F5

Ok

Run

:
: 1

View

View → Debug Window → Breakpoints

Enable

: 2

: 3

(.... F5

View → Debug Window → Breakpoints

.... Breakpoint → Delete All

: 4

j =)

Condition

(500 j

Breakpoint Properties

(500

: 5

Breakpoint Properties

Pass count

100 Pass Count listBox1.Items.Add(s)

. 100

:

:

Debug → Add

1

Watch at Cursor

. Ctrl F5 Run

Add Watch 2

View → Debug Windows → watches View

Watch List 3

View → Debug Windows → :

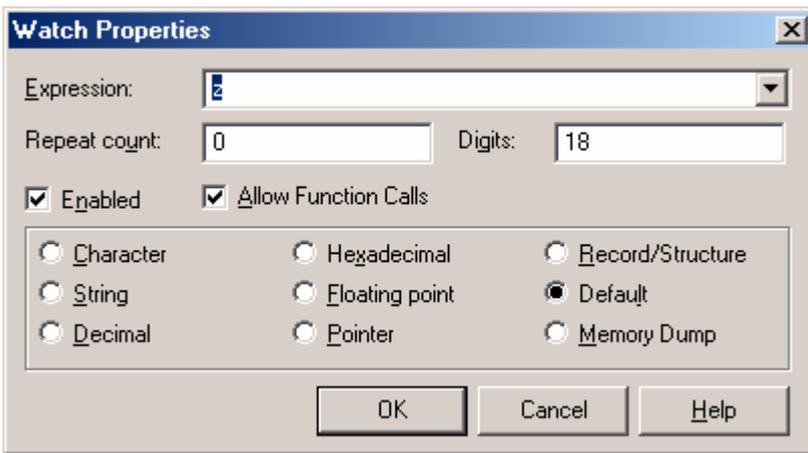
:

watches

Disable

. Delete

Edit Watch



Expression

i+j

: Repeat Count

Expression

/4 /

Repeat Count

Hexadecimal

Digits

. Floating Point

:

View → Debug Windows → call

Stack

Add (10);

:

:

Procedure Add (x);

:

Add

Begin

X := x +1;

If X < 50 then Add (x); // 50

End;