Introduction to VB.net

Lecture 6

Nasrin Khodapanah

VB.NET

Agenda

- Constants
- Structures
- For Each
- Casting
- Try n Catch
- Objects
- Message Box – Input Box
- EXO

HERZING COLLEGE

Constants

Constants

Basically constants are used to:

- Keep code consistent
- Easy to change in time
- Standardize the code

Constants

```
Const ARRAYLENGTH As Integer = 10

...

Dim Array(ARRAYLENGTH ) As Integer
```

Structures

Structures

Structure is a container type, meaning it holds other types as members.

A container type is useful for gathering a set of data items, such as customer information, that are closely related but cannot be held in a single elementary type. A structure can contain data members to hold these items and code members to manipulate them.

Structures

EX: Book information for a bookstore

- Title
- Sub-title
- Author
- Pages
- Etc.

Structures

```
Structure Book
    Dim ID          As Integer
    Dim Title       As String
    Dim Author      As String
    Dim Pages       As Integer
End Structure
```

Structures

How to use it:

```
'Declare a simple structure
Dim book1 As Book
'Declare an array of structure
Dim books(9) As Book
```

Structures

How to use it:

books(0).ID = 0

books(0).Title = “The Feast of All Saints"

books(0).Author = "Anne Rice"

books(0).Pages = 571

books(1).ID = 1

books(1).Title = "A Brief History of Time"

books(1).Author = "Stephen Hawking"

books(1).Pages = 256

For Each

For Each

How to use it:

```
Console.WriteLine("Enter a name")
Dim myString As String = Console.ReadLine()

Dim mount As Integer = 0
For Each character As String In myString
   count =count + 1
Next

Console.WriteLine(myString & " has " & count & " characters")
```

Casting

Casting

Converts a variable from a type to another type.

```
Ctype(myString, Integer)

Dim myString As String = "667"
If Ctype(myString , Integer) = 667 Then
'...
End If
```

Try n Catch

Provides a way to handle some or all possible errors that may occur in a given block of code, while still running code.

```
Try
    num = CType(strng, Integer)
    'num = CInt(strng)

    Console.WriteLine(num)
Catch ex As Exception
    ' Handle the exception here
    Console.WriteLine("An error occurred: " & ex.Message)
End Try
```

Casting

CBool() : Return a Boolean.

CByte() : Return a Byte.

CChar() : Return a Caracter.

CDate() : Return a Date.

CDbl() : Return a Double.

CDec() : Return a Decimal.

CInt() : Return an Int.

CLng() : Return a Long.

CSng() : Return a Single.

CStr() : Return a String.

CUInt() : Return an Unsigned Integer.

CULng() : Return an Unsigned Long.

CUShort() : Return an Unsigned Short.

Objects (variable)

Objects

When we have no idea what the variable type is.

```
Dim myObject As Object
Do
        Console.Write("Enter number: ")
        myObject = Console.ReadLine()
Loop Until IsNumeric(myObject)
        myObject = CInt(myObject)
```

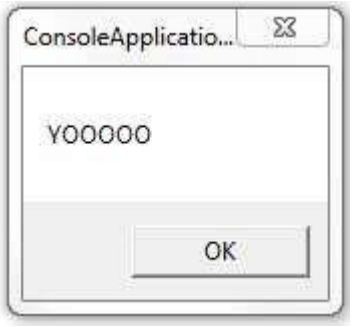
Message Box – Input Box

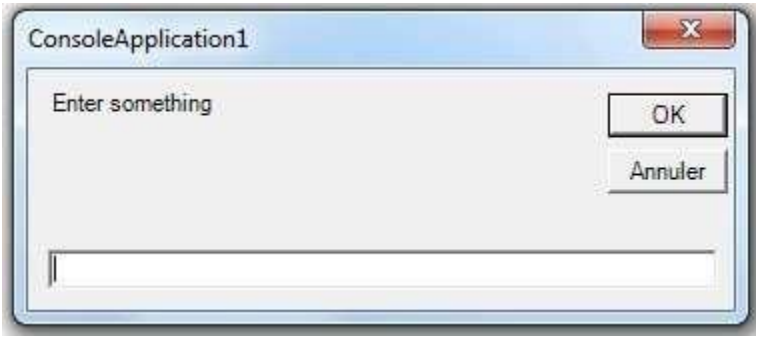
Message & Input Box

Well you've guessed right,

Message Box: To show a message

Input Box: To get something from the user with a message.





Message & Input Box

Return Value

Constant	Value
OK	1
Cancel	2
Abort	3
Retry	4
Ignore	5
Yes	6
No	7

EXO

EXO 6

Create a program to store students' information.

- First Name
- Last Name
- Program
- Teacher

Using Structures to manage the information. Give the user the option to add a new student, remove a student, or edit a student.

Use MessageBox and InputBox to get and show information.