.NET Framework 4

V.S.Net 2010

Part one

Project 403

Name : Mohammed Ahmed Reyad Mahran

NickNameNew : Opreyad36333

NickNameOld : OpMrayed20953

Email : IT_M.Reyad@yahoo.com

٢٠١٥/٥/١٦

مقدمة

تم إعداد هذا البحث لتقديمه لاكاديمية المجموعة العربية للكمبيوتر.

واتقدم بالشكر للاكاديمية على المستوى الذى وصلت اليه

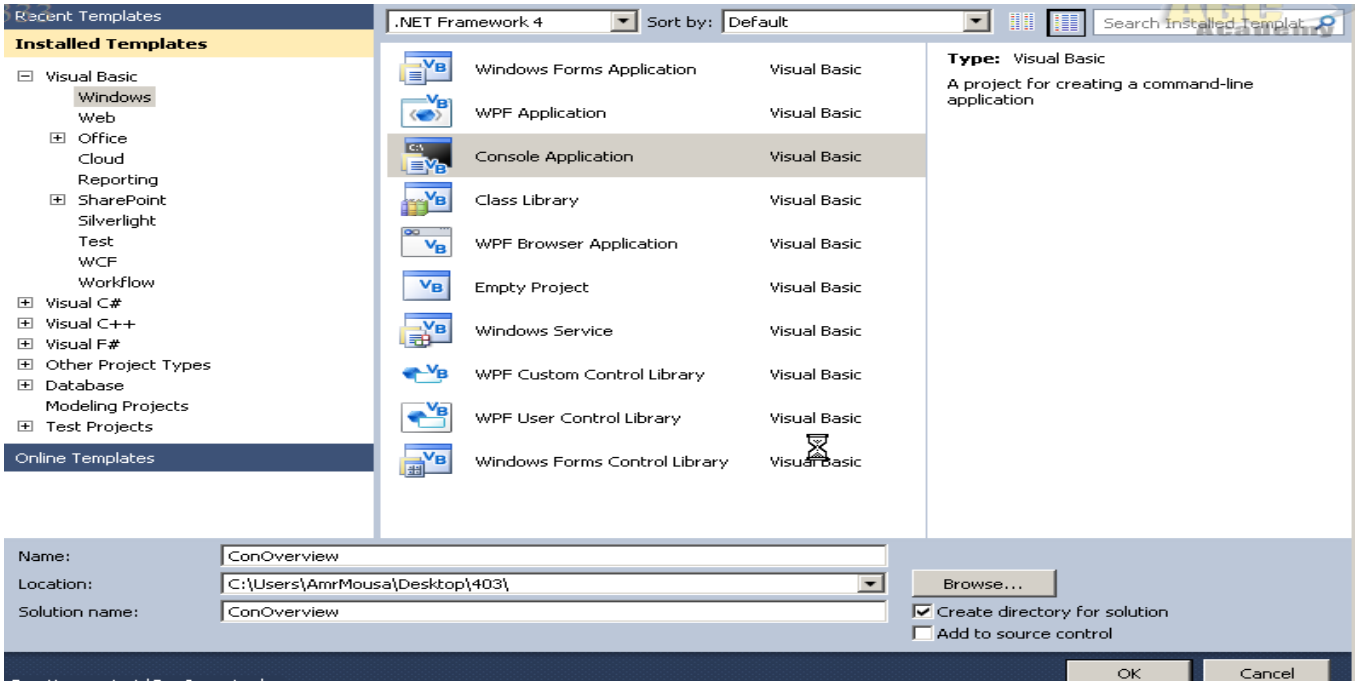
محتويات

الفصل	العنوان	محتوياته	رقم الصفحة
١	Console Application	○ مقدمة ○ المتغيرات ○ Enumerations	٤ ٥ ٧
٢	Array المصفوفة	○ انشاء مصفوفة ○ حلقة التكرار Loop ○ الدوال المصفوفة (ReDim - & Sorting Reverse - & ReDim with PreServe Erase - (BinarySearch ○ التعامل مع قواعد بيانات من خلال المصفوفة	٨ ٨ ٩ ١٢
٣	String النصوص	○ الدوال النصوص ○ String Builder	١٥ ١٧
٤	Functions الدوال	○ انشاء الدوال ○ انشاء معاملات اختيارية داخل الدوال Optional ○ Overloading ○ ByVal & ByRef ○ التعامل الداله و الساب مع المصفوفة (Array)	١٩ ٢٠ ٢١ ٢١ ٢٣
٥	Exceptions	○ مقدمة ○ بتفجير خطا	٢٤ ٢٥
٦	Directory Info	○ المجلدات والتعامل معها وقراءة ما بداخلها ○ للتشفير المجلدات والملفات ○ انشاء وحذف ونقل ○ Environment ○ CMD	٢٧ ٢٨ ٢٩ ٢٩ ٣٠
٧	UDT User Define Types	○ مفهوم Structure	٣١
٨	التعامل مع الملفات	○ مقدمة ○ Stream ○ Binary	٣٤ ٣٤ ٣٨
٩	عمل مكتبة DLL	○ مقدمة ○ مكتبة التشفير ○ مكتبة نطق الكلمة	٤١ ٤١ ٤٣

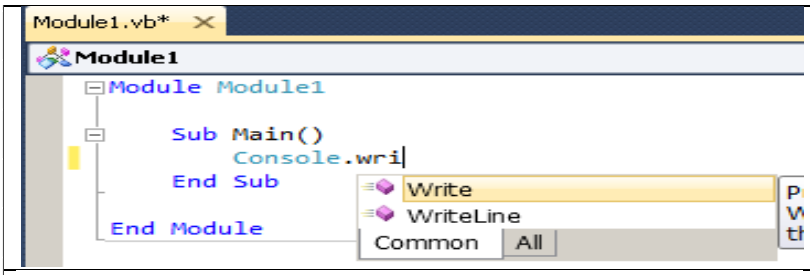
الفصل الاول : Console Application

< مقدمة :

- هى شاشة دوس سوف نقوم من خلالها استخدام لشرح .NET Framework.
- سوف نستخدم لفهم بعض المفاهيم قبل شرح .NET Framework.
- لفتح مشروع جديد من Console Application



- بعض الاوامر الخاصة :



الامر Write :
هى تطبع المكتوب ويظل المؤشر فى نفس السطر

الامر WriteLine :
هى تطبع المكتوب وينزل المؤشر فى نفس السطر جديد

```
Sub Main()

    Dim X As Integer = 10
    Dim y As Integer = 200
    Dim z As Integer = 300

                                        opreyad36333
    Console.Write("X value =  {0} Y value = {1}  Z value = {2}", X,y,z)
```

▲ 1 of 2 ▼ Write(format As String, **ParamArray arg() As Object**)
Writes the text representation of the specified array of objects to the s
arg: *An array of objects to write using format.*

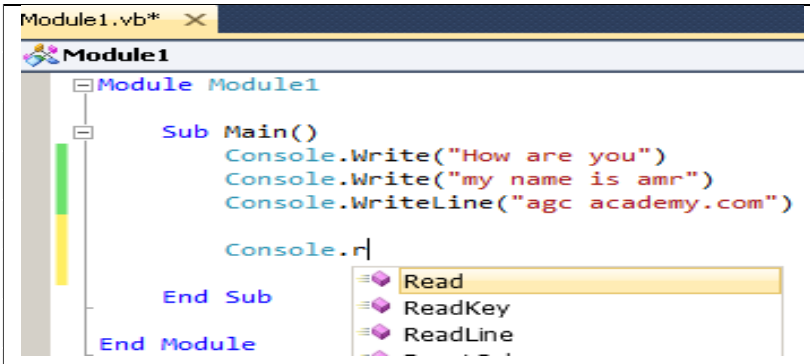
```
End Sub
```

كما يمكن داخل اقواس هذه الدالة استخدام هذه الاقواس {} و التى يكتب بداخلها ارقام تبدأ من صفر و تزداد هذه الارقام على حسب المتغيرات الموجوده بعد الفاصلة ,

الامر Read :
تستخدم هذه الدالة للقرأة من المستخدم و ويظل المؤشر فى نفس السطر

الامر ReadLine :
تستخدم هذه الدالة للقرأة من المستخدم مع وضع مؤشر الماوس فى السطر الذى يليه

الامر ReadKey :
هى تقرا الحرف فقط



الاوامر التالية خاصة بشاشة البرنامج
الامر الاول : لون الخلفية
الامر الثانى : تنظيف الشاشة
الامر التالت : عرض والطول الشاشة
الامر الرابع : عنوان الذى يظهر فى شريط العنوان
الامر الخامس : لون الخط
الامر السادس : مكان وابعاد المؤشر فى الشاشة البرنامج
الامر السابع والثامن : تم شرحه

```
Sub Main()

    Console.BackgroundColor = ConsoleColor.DarkBlue
    Console.Clear()
    Console.SetWindowSize(40, 10)
    Console.Title = "AGC Academy"
    Console.ForegroundColor = ConsoleColor.DarkRed

    Console.SetCursorPosition(10, 10)

    Console.WriteLine("www.agcacademy.com")



    Console.ReadKey()

End Sub
```

> المتغيرات :

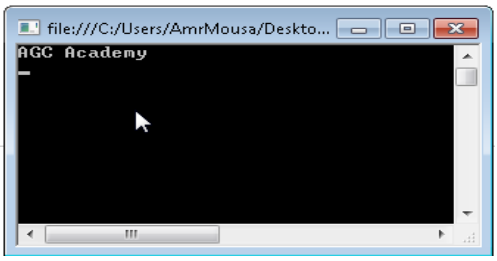
- **انواع المتغيرات :**

النوع	القيمة	يستخدم فى
Byte	0 to 255	وحدة ثنائية
Int16 Short	-32768 to 32767	عدد قصير
System.Int32 (Integer/int)	-2147483648 to 2147483647	عدد صحيح
System.UInt32 (UInteger/uint)	0 to 4294967295	عدد صحيح موجب
System.Int64 (Long/long)	-9223372036854775808 to 9223372036854775807	عدد طويل
System.Double (Double/double)	-1.79769313486232E+308 to 1.79769313486232E+308	عدد مزدوج
System.Decimal (Decimal/decimal)	-79228162514264337593543950335 to 79228162514264337593543950335	عدد عشرى
System.Single (Single/float)	-3.402823E+38 to 3.402823E+38	عدد مفرد
Boolean	True or False / Yes or No	منطقى
Char	-	حرف
String		نصى
System.DateTime (Date/date)	1/1/0001 12:00:00 AM to 12/31/9999 11:59:59 PM	تاريخ ووقت
Object	-	عام ويتعمل مع جميع المتغيرات والقيم

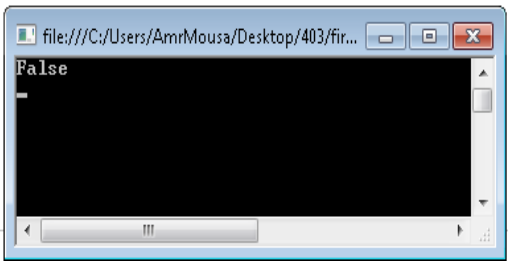
- **التعامل مع المتغيرات :**

Dim Private Public	اى اسم المتغير	AS	اى نوع من البيانات	لتعريف اى متغير
				قمنا بتعريف متغير اسم B نوع البيانات بيانات نصية واعطينا له قيمة ثم طبعنها ومع قراه الحرف المنتظر
				قمنا بتعريف متغير اسم B نوع البيانات منطقية ثم طبعنها من غير اعطائه قيمة لانه بداخه قيمة افتراضية
				لذلك اذا اردنا تنظيف القيمة الافتراضية نستخدم هذا الكود عند تعريف المتغير
				اذا اردنا التاكد من المتغير له قيمة نستخدم HasValue
				اذا اردنا الجمع بين النصين نستخدم +

```
Sub Main()
    Dim B As String
    B = "AGC Academy"
    Console.WriteLine(B)

    Console.ReadKey()

End Sub

Module
```




```
Sub Main()

    Dim B As Boolean
    Console.WriteLine(B)


    Console.ReadKey()

End Sub

Module
```




```
Sub Main()

    Dim B As Nullable(Of Boolean) = Nothing
    Console.WriteLine(B)


    Console.ReadKey()

End Sub
```

```
Sub Main()

    Dim B As Nullable(Of Boolean) = Nothing

    If B.HasValue Then
        Console.WriteLine(B)

    Else

        Console.WriteLine("Nullable value")

    End If


    Console.ReadKey()

End Sub
```

```
Sub Main()

    Dim X As String = "AGC Academy"
    Dim Url As String = "www.agcit.com"



    Console.WriteLine("Company Name: " + X + "URL :" + Url)
    Console.ReadKey()

End Sub
```

< Enumerations :

- هى التعداد يعنى هى تجميعه لمجموعة من الثوابت ويختار المبرمج من بينهم من خلال الكلاس الذى قمت بتصميمه
- مثال على ذلك

```
Sub Main()

    Console.BackgroundColor =


End Sub

Module
```

- كيفية التعامل معها

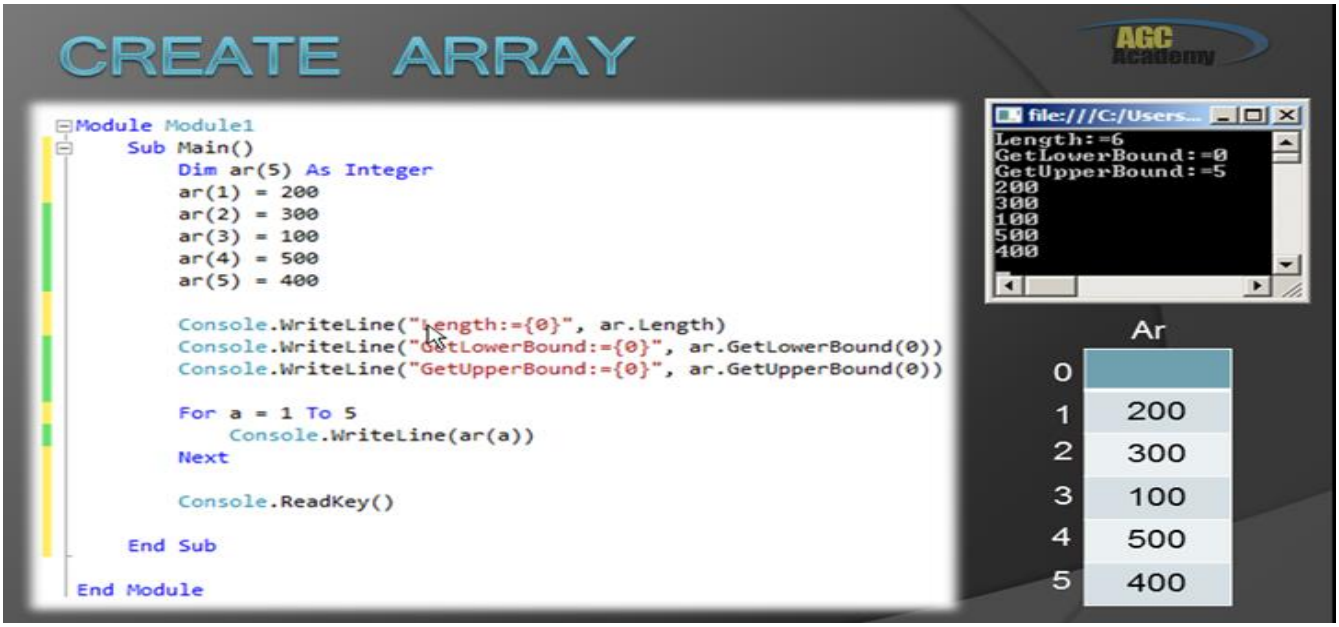
`Enum MyColor As Integer`
 	
`W = 15`
`B = 1`
`R = 12`
 	
`End Enum`
	كيفية انشاءها

`Console.BackgroundColor = MyColor.R`
 	
`Console.Clear()`
 	
`Console.ReadKey()`
	كيفية استخدامها

الفصل التانى : المصفوفة Array

> انشاء مصفوفة :

- الفرق بين المصفوفة و المتغير ان المتغير يقوم بحفظ قيمه متغير واحد فقط اما المصفوفة تقوم بحفظ اكثر من متغير داخل مصفوفة لها اسم واحد
- هى نفس طريقة انشاء المتغيرات ولكن بنضع جانب اسم المتغير قوسين ( ) نحدد فيه طول وعرض المصفوفة
- او هى عباره عن متغير ولكن مقسم الى عده اقسام يبدا من الاندكس صفر الى طول المصفوفه التى تم تحديدها
- مثلا يعنى المصفوفه اللى عندنا طولها ٤ اى يوجد بداخلها خمس قيم من صفر الى اربعه = خمس قيم
- المطلوب فى المثال القادم (اولا / طول المصفوفه - ثانيا / الحصول على اعلى قيمه فى المصفوفه - ثالثا / الحصول على اقل قيمه فى المصفوفه)



> حلقة التكرار Loop :

- هنا بقى لو عايز اطبع عناصر المصفوفه من خلال حلقة تكرار for/Next

```
For A As Integer = 0 To 4

    Console.WriteLine(M(A))

Next

Console.ReadKey()
```

- دى مصفوفه و لكن قيمها ليست ثابته وانما نحصل عليها من خلال Console.ReadLine
- واللى بخده فى متغير --- وهكذا ونستخدم حلقة تكرار لطباعه عناصرها

```
Console.Write("Enter Array No 0 : ")

M(0) = Console.ReadLine
```

```
Console.Write("Enter Array No 1 : ")
M(1) = Console.ReadLine

Console.Write("Enter Array No 2 : ")
M(2) = Console.ReadLine

Console.Write("Enter Array No 3 : ")
M(3) = Console.ReadLine

Console.Write("Enter Array No 4 : ")
M(4) = Console.ReadLine

Console.Clear()
For A As Integer = 0 To 4
    Console.WriteLine(M(A))
Next
Console.ReadKey(
```

> الدوال المصفوفة (ReDim - Sorting & Reverse - ReDim with PreServe & Erase - BinarySearch) :

- هنا انا عايز اعلن مصفوفه بس طولها متغير وباخد قيمتها الخاصه بالطول فى متغير انتجر
- وبعلن عن مصفوفه وباعيد انشائها وبخلى طولها هو المتغير الانتجر
- (اعاده طول المصفوفه يكون بالكلمه ReDim)

```
Dim T As Integer
Console.Write("Enter Length Of Array :-   ")
T = Console.ReadLine

Dim M() As Integer
ReDim M(T)
```

- بعلن عن دوراه لتحمل قيم المصفوفه او لطباعتها وطول المصفوفه هو المتغير الانتجر

```
For A As Integer = 0 To T
```

- الكود دا بطبع قيم الاندكس الخاص بالمصفوفة وبكتب قيم المصفوفه وبحملها مرة اخرى

```
Console.Write("Enter Array No {0} : ", A)
```

```
        M(A) = Console.ReadLine
    Next
```

- وهنا بطبع قيم المصفوفه

```
    Console.Clear()
    For A As Integer = 0 To T
        Console.WriteLine(M(A))
    Next

    Console.ReadKey()
```

- سبق ايضاح الفكره التاليه وهى مصفوفه وبيتم تحملها وعرضها من خلال حقة تكرار

```
    Dim D(4) As String
    D(0) = "Mahmoud"
    D(1) = "Mohamed"
    D(2) = "Ibrahim"
    D(3) = "ElMetwaly"
    D(4) = "Abou El Sooud"
    For i As Integer = 0 To D.Length - 1
        Console.WriteLine(D(i))
    Next
    Console.WriteLine("--------------------------------------")
```

- الجديد هنا بقى دا كود عمل فرز للمصفوفه وبعيد طبعتها من تانى

```
    System.Array.Sort(D)
    For i As Integer = 0 To D.Length - 1
        Console.WriteLine(D(i))
    Next
    Console.WriteLine("--------------------------------------")
    Dim S As String
    Console.Write("Enter Search Name :   ")
    S = Console.ReadLine
```

- اذا اردنا مسح المصفوفة نستخدم الدالة Erase

```
Erase D
```

- الكود التالى دا كود البحث K = System.Array.BinarySearch(D, S)
- فى المصفوفه وباخد قيمه البحث فى متغير K ودالله البحث فى المصفوفه كما واضح تاخد قيمتين
  ١- اسم المصفوفه
  ٢- كلمه البحث ودى فى المتغير اس

```
Console.WriteLine("-------------------------------------")

Dim K As Integer

K = System.Array.BinarySearch(D, S)

Console.WriteLine(K)

Console.WriteLine("-------------------------------------")

Console.ReadKey()
```

- سبق الحديث انه لو اردنا اعاده تعريف المصفوفه نستخدم ReDim
- طب لو انا اعادت تعريف المصفوفه وزودنا طولها من ٣ الى ٥ كما بالمثال وطبعنا المصفوفه مش هيطبع غير القيم الرابعه والخامسه ومش هيحافظ على السابق – طب لو اردنا اننا نحافظ على القيم السابقه
- نستخدمReDim Preserve

```
Dim F(3) As String

F(0) = "A"

F(1) = "B"

F(2) = "C"

F(3) = "D"

ReDim Preserve F(5)

F(4) = "E"

F(5) = "F"

For i As Integer = 0 To F.Length - 1

    Console.WriteLine(F(i))

Next

Console.WriteLine("-------------------------------------")

Console.ReadKey()
```

> التعامل مع قواعد بيانات من خلال المصفوفة :

- مثال
- طبعا معروف اننا لما بنتعامل مع الادو كلاسيك
- بندرجها من الخصائص Microsoft Active Data Object X

```
Module Module1
```

- هنا انا بعلن عن متغيرات CN لحجز كلاس الكنكشن بتاع الاتصال بقاعده البيانات
- ومتغير RS لحجز كلاس ال Recordset لفتح الجدول داخل قاعده البيانات

```
    Dim CN As New ADODB.Connection

    Dim RS As New ADODB.Recordset
```

- دا كود فتح الاتصال بقاعده البيانات اكسس

```
        CN.Open("Provider=Microsoft.Jet.OLEDB.4.0;Data Source= AGC.mdb;Persist Security
Info=False")
```

- المتغير الاسترنج اللى جى دا بضع بداخله جمله سكلت

```
  Dim S As String

        S = "select Sname ,Seience,Math,Math + Seience as [Total] from Degree"
```

- بفتح الجدول الفلان اللى قائم على جمله سلكت ونوع الفتح واللوك كورسر

```
        RS.Open(S, CN, ADODB.CursorTypeEnum.adOpenDynamic,
ADODB.LockTypeEnum.adLockOptimistic)


        Console.WriteLine("     Degree Of Students ")

        Console.WriteLine("----(((((((((*)))))))))----")
```

- هنا بقى انا عايز اعرض الداتاوطبعا السجلات بداخل RS
- فبعرض سجل سجل من خلال حلقة التكرار Do While

```
        Do While Not RS.EOF

            Console.Write("Name :- {0}  Math :- {1} Seience :- {2}   ",
RS("Sname").Value, RS("Math").Value, RS("Seience").Value)

            Console.WriteLine("   Total :- {0}", RS("total").Value)

            RS.MoveNext()

        Loop

        Console.ReadKey()
```

- الكود التالى OpenDataBase() دا عباره عن سب بداخله كود الاتصال بقاعده البيانات

```
        OpenDataBase()
```

- وهنا حلقة التكرا ايضا مثل السالف ذكرها ولكن الدوراه هنا For / Next

```
        For i As Integer = 0 To RS.RecordCount - 1

            Console.Write("Name :- {0}  Math :- {1} Seience :- {2}   ",
RS("Sname").Value, RS("Math").Value, RS("Seience").Value)

            Console.WriteLine("   Total :- {0}", RS("total").Value)

            RS.MoveNext()

        Next

        Console.WriteLine("------------------------")

        Console.Write("Total Records Is :-  ")

        Console.WriteLine(RS.RecordCount)

        Console.ReadKey()
```

- الكود التالى قمنا فيه بالتالى الاعلان عن متغير T انتجر ليحمل عدد السجلات
- حلقة التكرار للمرور على قيم الجدول اللى بداخل Rs
- اعلنا عن مصفوفتين سترنج واحد للاسم والثانى للاجمالى
- ثم قمنا باعاده الاعلان عنهم على ان يكون طول تلك
- المصفوفات هو طول السجلات فبالجدول ثم قمنا بعمل حلقة التكرار ليتم تحميل تلك المتغيرات ببياناتهم من الجدول ويليها دواره لعرضهم

```
        OpenDataBase()

        Dim T As Integer

        T = RS.RecordCount - 1


        For i As Integer = 0 To T

            Console.Write("Name :- {0}  Math :- {1} Seience :- {2}   ",
RS("Sname").Value, RS("Math").Value, RS("Seience").Value)

            Console.WriteLine("   Total :- {0}", RS("total").Value)

            RS.MoveNext()

        Next

        Console.WriteLine("------------------------")

        Dim Rname(), Rtotal() As String

        ReDim Rname(T)

        ReDim Rtotal(T)


        RS.MoveFirst()

        For U As Integer = 0 To T
```

```
            Rname(U) = RS("Sname").Value

            Rtotal(U) = RS("Total").Value

            RS.MoveNext()

        Next


        For i As Integer = 0 To T

            Console.WriteLine(Rname(i) & "  <->  " & Rtotal(i))

        Next

        Console.ReadKey()

    End Sub
```

- دا سب فتح قاعده البيانات

```
    Sub OpenDataBase()


        CN.Open("Provider=Microsoft.Jet.OLEDB.4.0;Data Source= AGC.mdb;Persist Security
Info=False")

        Dim S As String

        S = "select Sname ,Seience,Math,Math + Seience as [Total] from Degree"

        RS.CursorLocation = ADODB.CursorLocationEnum.adUseClient

        RS.Open(S, CN, ADODB.CursorTypeEnum.adOpenDynamic,
ADODB.LockTypeEnum.adLockOptimistic)

    End Sub

End Module
```

الفصل الثالث : النصوص String

> الدوال للنصوص :

- Left / تعنى قراءه النص بطول كذا او عرض اول كذا حرف من الشمال
- Right / نفس السابقه ولكن من اليمين
- InStr / هنا نسال عن نص معين ويعود برقم اى رقم النص فى ترتيب النص
- Mid / تعنى اقرا من نصف النص من اول الرقم كذا وحتى الرقم كذا
- Replace / تسعى استبدل النص كذا بكذا
- Len / تعبر عن الطول
- UCase / تعنى جعل حاله الاحرف كابتل وعكسها سمول وهو الداله Lcase
- Trim / ازاله المسافات فى النص من اليمين واليسار
- RTrim / ازاله المسافات من اليمين فقط ولازالتها من اليسار LTrim

```
    Dim S As String = ("   I'm Mahmoud - Student In AGC Academy   ")
    Console.WriteLine(S)
    Console.WriteLine(Left(S, 10))
    Console.WriteLine(Right(S, 10))
    Console.WriteLine(InStr(S, "AGC"))
    Console.WriteLine(Mid(S, 5, 7))
    Console.WriteLine(Replace(S, "AGC", Chr(100)))
    Console.WriteLine(Len(S))
    Console.WriteLine(UCase(S))
    Console.WriteLine(LCase(S))
    Console.WriteLine(Trim(S))
    Console.WriteLine(RTrim(S))
    Console.WriteLine(LTrim(S))
    Console.ReadKey()
End Sub
```

- **الداله Lset : هى تاخذ معاملان الاول اسم النص او المتغير النصى والثانى انتجر وهو الطول**
- **ببساطه انا بقول ياخذ طول قدره كذا بغض النظر عن طول النص فلو اخذت الداله الرقم ٦ كطول وانا كاتب نص طوله ١٠ فهنا مش هيكتب الا ٦ احرف فقط فالداله تلك تقوم بتظليل الطول المحدد بغض النظر عن النص قصير ام لا**
- **والفرق بين Lset – Rset هو الاتجاه من اليمين ام اليسار**

```
Dim X As String
      Console.WriteLine("Write A word Less 20 .:")
      X = Console.ReadLine()

      Dim R As String
      R = LSet(X, 5)
      Console.WriteLine(R)
      '--------------------------------------------------
      Dim Q As String
      Q = RSet(X, 5)
      Console.WriteLine(Q)
      Console.ReadKey()
```

- **مثال اخر: هنا نوضح طريقه اخد المسافات**
- **وهو اما انا اكتب + Space(12) + بدل ما اكتب + " ------ " +**

```
      Dim Q As String = "Mahmoud"
      Dim Q1 As String = "Mohamed "
      'Dim Q2 As String = Q + Space(12) + Q1
      Dim Q2 As String = Q + " ------ " + Q1
      Console.WriteLine(Q2)
      Console.ReadKey()
```

< String Builder :

- هو عباره عن كلاس مخصص للتعامل النصوص --- وعلى الرغم من امكانيه التعامل مع كل خصائص الاسترنج بلدر من كائن استرنج الا ان الاسترنج بلدر مميز فى سرعه التعامل وسهوله التعامل

```
Sub main()
```

- طريقه الاعلان عن الاسترنج بلدر

```
Dim SB As New System.Text.StringBuilder
```

- عندما نريد اضافه سطر نصى جديد نستخدم الخاصيه AppendLine
- اما الخاصيه Append فقط تعنى سطر مع عدم اضافه انتر

```
SB.AppendLine("Allah Is My God")

SB.Append("Mohamed Is My Prophet")
```

- Space(3) تعنى اضافه ثلاث مسافات وممكن ان اتاخد يدويا بالشكل التالى & "----" &

```
SB.AppendLine(Space(3) & "----" & "In The Name Of Allah")

Console.WriteLine("Enter URL .:")

SB.AppendLine(Console.ReadLine)

Console.Clear()
```

- Replace تعنى استبدال الحرف كذا بالحرف كذا

```
SB.Replace("M", "@")

SB.Replace("o", "#")

SB.Replace("h", "!")

Console.WriteLine(SB.ToString())

SB.Replace("@", "M")

SB.Replace("#", "o")

SB.Replace("!", "h")

Console.WriteLine(SB.ToString())

Console.ReadKey()
```

- Remove تعنى حذف من اندكس بداية الحرف كذا حتى طول اندكس كذا

```
SB.Remove(5, 2)
```

- **المثال التالى عباره عن فكره تشفير نصى (توضيح الفكره اولا - عندنا خاصيه اسمها Chr وهى عباره عن انه لكل حرف على الكيبورد رقم وكل علامه خاصه رقم واجمالى الارقام تلك ٢٥٥ وفى الحقيقه ان كل الحروف والارقام تنحصر حتى العدد ١٢٢ ومابعد ذلك رموز وشفرات - فاعتمدنا فى الفكره هنا على اننا نجيب رقم الحرف الفلانى ونضيف له ١٢٥ - وبعد كده نحول الرقم الاجمالى للحرف المقابل – ملحوظة ان اقصى رقم ٢٥٥ وانا كنت بقول ان الحروف والارقام تنحصر حتى ١٢٢ - فلو جمعنا ١٢٢+١٢٥ = ١٤٧ اى لم نتجاوز ال ٢٥٥ وايضا ضمناننا دخلنا منطقه الشيفرات الغريبه)**

```
        Dim SB As New System.Text.StringBuilder

        SB.AppendLine("Allah Is My God")

        SB.AppendLine("Mohamed Is My Prophet")

        SB.AppendLine("PLS Send Me To Paradise")


        Console.WriteLine(SB.ToString())

        Console.WriteLine("^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^")
```

- **هنا نستبدل الاحرف اللى رقمها يقابل قيمه ال I**
- **بالاحرف اللى قيمتها مقابل للاى مضاف له ١٢٥**
- **وايضا نزيد الشعر بيتا ونضيف الشفره "!@"**

```
        For I As Integer = 46 To 122

            SB.Replace(Chr(I), (Chr(I + 125) & "@!"))

        Next

        Console.WriteLine(SB.ToString())

        Console.WriteLine("^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^")
```

- **هنا نعاود تشفير النص اللى خرج بنفس الشقره السابقه**
- **ليعود النص الى ماكان عليه**

```
        For I As Integer = 46 To 122

            SB.Replace(Chr(I + 125) & "@!", (Chr(I)))

        Next

        Console.WriteLine(SB.ToString())

        Console.WriteLine("^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^")

        Console.ReadKey()

    End Sub
```

الفصل الرابع : الدوال Functions

> انشاء الدوال :

- نكتب Function ثم اسم الدالة (نضع متغيرات ان وجدت) ثم نكتب خارج القوسين نوع القيمة التى تعود بها ونكتب فى الاسفل end Function
- ولكى تعود بقيمة نكتب داخل الدالة Return

```
Function Avg(ByVal x As Integer, ByVal y As Integer, ByVal z As Integer) As Integer
 Dim V As Integer

 V = (x + y + z) / 3

 Return V
End Function
```

- ثم نداء عليها با اسمها ونضع متغيرات اللى بداخلها
- ويجب انشاء متغير من نفس النوع التى تعود بها الدالة ونجعلها تساويها

```
Sub Main()

  Dim i As Integer
  i = Avg(10, 20, 30)


 Console.Write(i)
 Console.Read()

End Sub

Function Avg(ByVal x As Integer, ByVal y As Integer, ByVal z As Integer) As Integer
```

- مثال اخر يوضح ذلك (الدالة للتاكد من الايميل) وكيفية استخدامها :

مكتبة	استخدامها

Function ChkEmail(ByVal E As String) As Boolean
 	
If InStr(E, "@") < 1 Then
 	
Return False
End If
 	
If InStr(E, ".") < 1 Then
Return False
End If
 	
Return True
 	
End Function
	Sub Main()
	
	Dim I As String
	
	Console.Write("Enter Your EMail: ")
	I = Console.ReadLine
	
	Console.Write(ChkEmail(I))
	Console.ReadKey()
	
	End Sub

- مثال اخر يوضح ذلك (الدالة تشفير والدالة اخرى فك تشفير) وكيفية استخدامها :

للتشفير	فك تشفير

`Function MyEnc(ByVal Msg As String) As String`
`    Dim SB As New Text.StringBuilder`
`    SB.Append(Msg)`
`    Dim i As Integer`
`    For i = 45 To 122`
`        SB.Replace(Chr(i), Chr(i + 125) & "!@#")`
`    Next`
`    return SB.ToString`
`End Function`
	`Function MyDecryption(ByVal msg As String) As String`
	`    Dim SB As New Text.StringBuilder`
	`    SB.Append(msg)`
	`    Dim i As Integer`
	`    For i = 45 To 122`
	`        SB.Replace(Chr(i + 125) & "!@#", Chr(i))`
	`    Next`
	`    Return SB.ToString`
	`End Function`

```
Sub Main()

        Dim k As String
        k = "i love agc academy"

        Dim En As String = MyEnc(k)

        Dim De As String = MyDecryption(En)

        Console.WriteLine(En)
        Console.WriteLine(De)


        Console.ReadKey()
End Sub
```

> انشاء معاملات اختيارية داخل الدوال Optional:

- Optional : عندما توضع بجانب المعامل تصبح اختيارية اى ممكن ادخلها ام لا --- ومن الممكن وضع لها قيمة افتراضية من خلال ان نكتب بجوار المعامل = قيمة وللتوضيح من خلال المثال التالى :

```
Function MyEnc(ByVal msg As String, Optional ByVal key As String = "!@#") As String
    Dim SB As New Text.StringBuilder

    SB.Append(msg)

    Dim i As Integer
    For i = 45 To 122
        SB.Replace(Chr(i), Chr(i + 125) & key)
    Next

    Return SB.ToString
End Function
```

> Overloading

- هو يعتبر هذا المصطلح من مفاهيم الاساسية لـ OOP حيث يتيح لنا استخدام اكثر من Method بنفس الاسم ولكن بوجود شرط اساسي وهى
  ١- اختلاف عدد المعاملات
  ٢- لو فى نفس الاسم Method ونفس عدد المتغيرات ولكن لابد من اختلاف فى نوع البيانات المعامل
- وعلى سبيل المثال :

```
Function MyEnc(ByVal msg As String) As String ...
Function MyEnc(ByVal msg As String, ByVal key As String) As String ...
```

> ByVal & ByRef

- تمرير المعاملات للدوال او الاجراءات له طريقتان :
  ١- By Val وهنا المعامل مش بيتاثر بتغير القيمه
  ٢- By Ref وهنا المعامل بيتاثر بتغير القيمه

```
Sub main()
```

- اول شئ هنا انا اعلنت عن متغيران (A / انتجر = ١٠ - A1 / انتجر = ٢٠)

```
    Dim A As Integer = 10
    Dim A1 As Integer = 20
```

- وهنا ندهت على الاجراء Testy وهو اجراء له معاملان

```
    testy(A, A1)
```

- والاجراء كما يلى

```
Sub testy(ByRef X As Integer, ByVal X1 As Integer)
    Console.WriteLine("X = " & X & " - " & "X1 = " & X1)
    X = X + 1
    X1 = X1 + 1
End Sub
```

- ولا ننسى ان قيمه الاكس بى رف والاكس ون بى فال
- وهنا بطبع قيمه السب A و A1 (والناتج هيكون الاكس = ١١ - والاكس ون = ٢٠ اى دون تغير)

```
    Console.WriteLine("A = " & A & " - " & "A1 = " & A1)
    Console.ReadKey()
    '''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''
    Dim A As String
```

```
        A = Console.ReadLine
        Dim X As String = A
        'SubAlterMail(X)
```

- هنا انا بنده على الداله FunAlterMail - والسب دا ببدل فيه www بالقيمه HTTP://www
- والسب دا بياخد معامل استرنج اللى فيه بكتب اسم الموقع بس المعامل بر رف وبناء على ذلك سيحدث التغير ولو بى فال لن يحدث تغير

```
        FunAlterMail(X)
        Console.WriteLine(X)
        Console.ReadKey()
                    ================================================
    Function SubAlterMail(ByRef STR As String) As String
        Dim SB As New Text.StringBuilder
        SB.AppendLine(STR)
        SB.Replace("www", "HTTP://www")
        STR = SB.ToString
      Return STR
End Function
                    ================================================
        Dim S As Integer
        S = 10
        Valy(S)
        Console.WriteLine(S)
        Console.ReadKey()
    End Sub
                    ================================================
    Sub Valy(ByVal X As Integer)
        X = 20
    End Sub
                    ================================================
    Sub testy(ByRef X As String)
        Console.WriteLine(X)
        X = "AHMED"
    End Sub
```

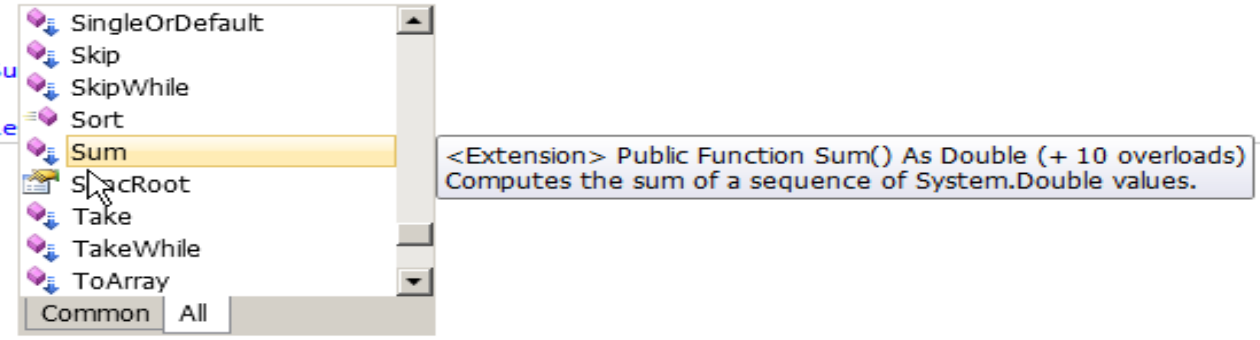
- دا السب بديل الداله SubAlterMail

```
Sub FunAlterMail(ByRef STR As String)

    Dim SB As New Text.StringBuilder

    SB.AppendLine(STR)

    SB.Replace("www", "HTTP://www")

    STR = SB.ToString

End Sub
```

> التعامل الداله و السابب مع المصفوفة (Array)

- لدينا مصفوفة اسمها K لديها بعض الداول كما يلى :

```
Sub Main()

    Dim k(5) As Double
    k(1) = 100
    k(2) = 300
    k(3) = 250
    k(4) = 200
    k(5) = 800

    k.
```



- من الممكن تجميع المصفوفة وقسمتها على طولها – ١

```
Module Module1
    Sub Main()

        Dim k(5) As Double
        k(1) = 100
        k(2) = 300
        k(3) = 250
        k(4) = 200
        k(5) = 800

        Console.WriteLine(Avg(k))
        Console.ReadKey()

    End Sub

    Function Avg(ByVal Ar() As Double) As Double
        Dim i As Double
        i = Ar.Sum() / (Ar.Length - 1)
        Return i

    End Function

End Module
```

الفصل الخامس : Exception

> مقدمة :

- موضوع الاستثناء يعنى وقت حدوث الخطا
- والاخطاء عديده منها ماهو خطا يحدث وقت التشغيل واخطاء فى الكود والاخطاء المنطقيه
- معالجات الاخطاء الخاصه بالكود دى تعالج داخيا فى الكود فى وضع التصميم
- اما الاخطاء المنطقيه او وقت التشغيل مثل محاوله القسمه على صفر مثلا المهم لو الاخطاء دى حصلت وقت تشغيل البرنامج البرنامج هيعمل اغلاق وعموما هو شئ غير مقبول على الاطلاق وحل الموضوع دا بيكمن فى استخدام صيغه الاكسبشن وهو كما يلى

```
Try
```

- هنا يوضع كود التصميم او المحتمل حدوث الخطا به

```
Catch ex As Exception
```

- هنا نضع علاج الخطا المحتمل عن الكود السابق

```
End Try
```

- اول الاكواد التى نتعامل معها هو برنامج القسمه (ولكن ماذا لو ادخل المستخدم فى المقسوم عليه صفر الناتج سيكون خطا مش منطقى لانه لايمكن القسمه على صفر)
- وهنا نستخدم مجموعه التراى كاتش لعلاج الخطا دة

```
Module Excep

    Sub main()

        Try
10:

            Console.WriteLine("ENTER FIRST NUM .:")

            Dim X As Integer

            X = Console.ReadLine


            Console.WriteLine("ENTER Second NUM .:")

            Dim X1 As Integer

            X1 = Console.ReadLine


            Dim X2 As Integer

            X2 = X / X1


            Console.WriteLine(X2)

            Console.ReadKey()

        Catch ex As Exception
```

- لو الخطا حصل بقول يطلع رساله بالخطا ده
- Err.Number / تعنى اخراج رقم الخطا فالـ Err هو كائن به العديد من الخصائص منها (رقم الخطا / ووصف الخطا)
- ثم طلبنا من البرمبت هل تريد الاستمرار ام لا
- فلو استمر فاستخدمنا الكود GoTo X ومعناها اذهب الى الجزء الفلانى بخلاف ذلك يخرج من البرنامج

```
            Console.WriteLine("INVALID DIVIDE")
            Console.WriteLine(Err.Number)
            Console.WriteLine(Err.Description)

            Console.WriteLine("Choise  1- contonue      2-End")

            Dim H As Integer
            H = Console.ReadLine

            If H = 1 Then
                Console.Clear()
                GoTo 10
            Else
                End
            End If
        End Try

        Dim C As Integer
10:
```

> **بتفجير خطا:**

- البرنامج التالى مطلوب فيه التالى (ان يدخل المستخدم الرقم السرى - واعلانا عن متغير يستقبل القيمه المكتوبه - وبقول للمتغير دا لو انت مش بتساوى القيمه ١٢٣ - وكمان الرقم السرى كتب مرتان خطا اخرج رساله بوصف الخطا - فقد اعلنا عن متغير انتجر لمعرفه عدد مرات كتابه الرقم السرى خطا - باضافه واحد كل مره يحدث بها خطا - وان كتب الرقم صحيحا اول مره يرحب بالعميل فى الاى جى سى - وان كتب اول مره خطا يوضح ذلك ويعيد السب من الاول مادام لم يتجاوز ثلاث مرات خطا وفى المره الثالثه يقوم بتفجير خطا - فتفجير الخطا معناها اننا نقوم بصنع الخطا ولكن اى نوع خطا فى الواقع كل خطا له رقم فنقوم بالنداء على الرقم مع تفجير الخطا - وشكل الكود (Err.Raise(3)

```
        Console.WriteLine("ENTER YOUR PASSWORD .:. ")
        Dim X As String = Console.ReadLine
```

```
        If X <> 123 Then
            C = C + 1
            If C < 3 Then
                Console.WriteLine("ERROR PASSWORD")
                GoTo 10
            Else
                Err.Raise(3)
                Console.WriteLine(Err.Description)
            End If
        Else
            Console.WriteLine("Welcome In AGC")
            Console.ReadKey()
        End If
```

> **المثال التالى منه ببساطه نقدر نعرف رقم كل ووصفه حتى نتمكن من استخدام تلك الاخطا فى عمل فير لها كما نرى**

```
For I As Integer = 1 To 30
            On Error Resume Next
            Err.Raise(I)
            Console.WriteLine(I & Space(3) & Err.Description)
            Err.Clear()
        Next
        Console.ReadKey()
    End Sub
End Module
```

الفصل السادس : DirectoryInfo

< المجلدات والتعامل معها وقراءة ما بداخلها :

- اولا لابد من اضافة المكتبة لانها بتسهل التعامل مع الملفات والمجلدات وهى التالي

```
Imports System.IO
```

- هنا سنتحدث عن المجلدات والتعامل معاها وقراءه مابداخلها

```
Module Directoryy
    Sub main()
        On Error Resume Next
```

- هنا بعد ادراج المكتبه IO نقوم بالاعلان عن متغير يحمل بداخله الكلاس DirectoryInfo ولابد ان نمر المسار الخاص بالمجلد وهنا نقول له بارتيشن C
- بحاجه الى مصفوفه نضع بداخلها تلك المجلدات لذا نعلن عن متغير مصفوفه نوعها DirectoryInfo لاننا هنقرا مجلدات ونمرر لها مسار تلك المجلدات والمسار دا هنجيبه من المتغير الاول اللى حددنا فيه المسار

```
        Dim WDir As New DirectoryInfo("c:\\")
        Dim ArrDir() As DirectoryInfo
        ArrDir = WDir.GetDirectories
```

- عشان نعمل حقلة تكرار لازم نحدد طولها وطبعا هيكون طول المصفوفه ونحمل فى الدواره رقم المصفوفه ومقابلها من المجلد

```
        For I As Integer = 0 To ArrDir.Length - 1
            Console.WriteLine(I & Space(5) & ArrDir(I).ToString)
            '& Space(5) & ArrDir(I).LastAccessTime)
```

- هنا بقى انا عايز كمان المجلدات اللى جوه كل مجلد على ال C

```
            Dim WDir1 As New DirectoryInfo(WDir.ToString + ArrDir(I).ToString)
            Dim ArrDir1() As DirectoryInfo
            ArrDir1 = WDir1.GetDirectories
            For II As Integer = 0 To ArrDir1.Length - 1
                Console.WriteLine(II & Space(5) & ArrDir1(II).ToString & Space(5) & 
ArrDir(I).LastAccessTime)
            Next
            Console.WriteLine("''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''")
        Next
```

- **مثال بسيط لما تم شرحه ولكن هنا بعد قراءه المجلدات ونحمل الملفات اللى جوه المجلدات دى مش المجلدات**

```
        Dim Windir As New DirectoryInfo("c:\\windows")
        Dim DRS() As DirectoryInfo = Windir.GetDirectories

        For I As Integer = 0 To DRS.Length - 1
            Console.WriteLine(DRS(I).Name)
            Console.WriteLine(")))))))))))))))))))))))))))))))))))))))))))))")

            Dim F() As FileInfo = (DRS(I).GetFiles)

            For B As Integer = 0 To F.Length - 1
                Console.WriteLine(F(B).FullName)
            Next
            Console.WriteLine(";;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;")
        Next
        Console.ReadKey()
```

> للتشفير المجلدات والملفات :

- **هنا نقوم بتشفير الملفات من خلال الكلاس FileInfo ولابد ان نمرر مسار الملف**

```
        Dim F As New FileInfo("G:\\Testy.txt")
```

- **التشفير على الملف والمستخدم اللى فى الويندوز يعنى المستخدم الحالى هو بس اللى يقدر يفتح الملف**

```
        F.Encrypt()
        F.Decrypt()
```

- **هنا نسال عن وجود الملف ونسخه او حذفه**

```
        If F.Exists Then
            F.CopyTo("c:\\Test\Testy.txt")
            F.Delete()
        Else
            Console.WriteLine("Not Found")
        End If
        Console.ReadLine()
    End Sub
End Module
```

< انشاء وحذف ونقل الملفات :

- قمنا باضافة مكتبة IO وانشاء متغير من المكتبة ونقومنا بتحديد مسار وبعدها نتاكد اذا كان هذا الملف موجود اما لا فاذا كان لم يكن موجود ينشئها

```
Imports System.IO

Module Module1

    Sub Main()
        Dim Dr As New DirectoryInfo("E:\AGCTest")

        If Dr.Exists = False Then
            Dr.Create()
        Else
```

- فاذا كان موجودة فيقوم بنقلها الى مكان اخر

```
Dr.MoveTo("E:\\AGCTEST")
```

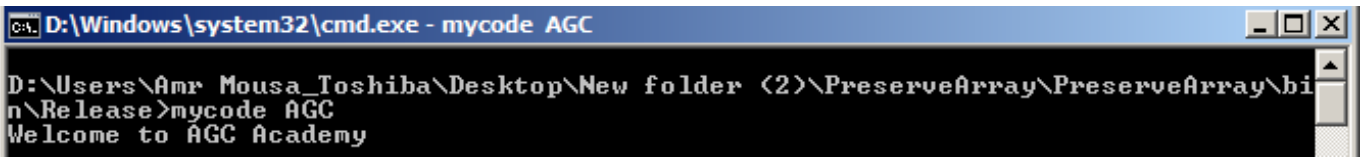
- اوحذفها مثلا :

```
            Dr.Delete()
        End If

    End Sub
```

< Environment :

- هى كلاس تعطى بها العديد من Method التى تمكن من خلالها معرفة تفاصيل دقيقة عن نظام التشغيل
- CommandLine : هى تستخدم لعدم فتح احد برنامجى الا من خلال كلمة سر حيث اننا نقوم بفتح برنامج CMd ثم ندخل المسار البرنامج ثم كلمة سر وبعدها يفتح البرنامج او الملف المشفر

```
Sub Main()

    Dim X As String
    X = Environment.CommandLine

    If InStr(X, "AGC") > 0 Then
        Console.WriteLine("Welcome to AGC Academy")
        Console.ReadKey()
        End
    End If

    Err.Raise(8)


    Console.ReadKey()


End Sub
```



< CMD :

- هذة الاوامر يتم استخدامها للتعامل مع الملفات بدون استخدام اى المكتبة.
- مثل هذة الاوامر هى :
- الامر انشاء مجلد

```
MkDir("E:\\NewOne")
```

- الامر حذف مجلد

```
RmDir("E:\\Newone")
```

- الامر اعادة تسمية مجلد

```
Rename("E:\mytextfile.txt", "E:\\myT.txt")
```

- الامر حذف مجلد

```
Kill("E:\\myT.txt")
```

- الامر مسار مجلد الحالى

```
Dim a As String = CurDir()
Console.WriteLine(a)
Console.ReadKey()
```

الفصل السابع : UDT
User Define Types

> مفهوم Sturcture :

- فى الواقع عندما نعلن عن متغير من نوع انتجر فهو يقبل ارقام ونوع نص فهو يقبل نصوص
- والاستركتشر هو متغير من نوع (اسم الاستركتشر) بس دا برده مش عارفين نوعه ايه
- هنا بقى نقول ان الاستركتشر دا عباره عن متغيرات ودوال واجراء مجمعين فى تركيب واحد
- ونستغل التركيب فى عمل تركيبات كامله لكل عمليات الاضافه والحذف والاستدعاء لكل جدول على حده مثلا
- عموما المثال القادم يوضح فكره المثال
- احنا عندنا جدول فيه درجات الطلاب وانا عايز اعمل تركيب بمجرد النداء عليه وتمرير بيانات طالب ما يقوم بعرض درجاته والنسبه وهل هو ناجح ام لا

```
Module Stud
```

- اول هام اعلنا عن المتغيرات الخاصه بفتح كائن الاتصال بقاعده البيانات وكائن فتح الجدول

```
    Dim CN As New ADODB.Connection
    Dim RS As New ADODB.Recordset
Structure NewUDT
```

- اعلنا فى البدايه عن متغيرات عامه تحمل اسم الطالب والمواد المقررة عليه

```
        Dim RealName As String
        Dim Math As Single
        Dim Physics As Single
        Dim Chemistry As Single
```

- دى اول داله داخل التركيب لحساب اجمالى الدرجات

```
        Function Totaly() As Double
            Totaly = Chemistry + Physics + Math
        End Function
```

- دى داله حساب النسبه المئويه ولكن الجميل هنا اننا بنقووم بالنداء وعلى داله التوتال دون الاعلان عنها لانها فى نفس التركيب فهى معرفه

```
        Function Percent() As Single
            Percent = (Totaly() * 100) / 30
        End Function
```

- ودى داله التقدير العام وبرده اعتمادنا على داله البرسنت اللى هى معتمده على داله التوتال

```
        Function Rating() As String
```

```
        If Percent() >= 50 And Percent() <= 65 Then Rating = "Pass"
        If Percent() > 65 And Percent() <= 80 Then Rating = "Good"
        If Percent() > 80 And Percent() <= 90 Then Rating = "Very Good"
        If Percent() >= 90 Then Rating = "Excellent"
    End Function
```

- **ودى داله البحث وجواها بفتح سب الاتصال بقاعده البيانات**

```
    Function FindStud(ByVal S As String) As Boolean
        OpenDB()
        RS.MoveFirst()
        RS.Find("RealName='" + S + "'")

        If RS.EOF = True Or RS.BOF = True Then
            Return False
        Else
```

- **هنا لو الداله بتعود بقيمه ترو بنوصل المتغيرات العامه فى لاتركيب بما يخص اسم الطالب والمواد الموجوده**

```
            FindStud = True
            RealName = RS("RealName").Value
            Math = RS("Math").Value
            Chemistry = RS("Chemestry").Value
            Physics = RS("Physics").Value
        End If
```

- **النداء على سب اغلاق قاعده البيانات**

```
        CloseDB()
    End Function
End Structure
```

- **هنا انتهى التركيب هنستخدمه ازاى هنشوف ده فى Sub Main**

- دى سب فتح الاتصال بقاعده البيانات

```
    Sub OpenDB()

        CN.Open("Provider=SQLOLEDB.1;Integrated Security=SSPI;Persist Security
Info=False;Initial Catalog=Students;Data Source=PC")

        RS.Open("Stud", CN, ADODB.CursorTypeEnum.adOpenDynamic,
ADODB.LockTypeEnum.adLockOptimistic)

    End Sub
```

- سب اغلاق قاعده البيانات

```
    Sub CloseDB()

        RS.Close()

        CN.Close()

    End Sub
```

- هنا بقى السب مين وفيه بطلب من العميل اسم الطالب وبنادى على الداله الخاصه بالبحث وبمرر ليها اسم الطالب المكتوب ولو الداله برترن ترو يعنى لقى الطالب يعرض بياناته ولو غير موجود يعرض انه مش موجود

```
    Sub main()
       Dim NS As New NewUDT
        Console.WriteLine("Enter RealName .: ")
        Dim H As String
        H = Console.ReadLine
        If NS.FindStud(H) = False Then
            Console.WriteLine("No Data Found")
        Else
            Console.Clear()
            Console.WriteLine("Real Nmae .: {0}", NS.RealName)
            Console.WriteLine("Math      .: {0}", NS.Math)
            Console.WriteLine("Chemistry .: {0}", NS.Chemistry)
            Console.WriteLine("Physics   .: {0}", NS.Physics)
            Console.WriteLine("Total     .: {0}/30", NS.Totaly)
            Console.WriteLine("Percent   .: {0}", NS.Percent)
            Console.WriteLine("Rating    .: {0}", NS.Rating)
        End If
        Console.ReadKey()
    End Sub
End Module
```

الفصل الثامن : التعامل مع الملفات

> مقدمة :

- فى التعامل مع الملفات ويوجد العديد من الكلاسات وكل كلاس يقوم بمجموعة من الوظائف
- ملفات عندى منها انواع :

١- TextFile : نكتب فى ملفات نصية

٢- BinrayFile : دى عشان تشفير

> TextFile:

- طبعا لابد من ادارج النيم سبيس I/O للتعامل مع StreamClass
- مثال الكتابه داخل ملف تكست :

```
Imports System.IO

Module StreamClass

    Sub main()
```

- هنا قمنا بالاعلان عن متغير يحمل بداخله كلاس الاستريم ريت
- وهو مخصص للكتابه واضافه البيانات

```
        Dim SW As StreamWriter
```

- هنا شورنا على مسار الملف المراد انشاءه وخليناه فى الدسك توب
- والامر CreateText بيقوم انشاء ملف واذا كان موجود يحذف القديم وينشئه من جديد

```
        SW = File.AppendText("C:\Documents and Settings\mah\Desktop\AGC.txt")
```

- هنا اعلنا عن متغيران استرنج هناخد جواه اللى المستخدم كتبهم

```
        Dim X, R As String

        X = Console.ReadLine

        R = Console.ReadLine
```

- هنا هنجمع قيمه المتغيران الاسترنج ونضفهم فى الملف التكست

```
        SW.WriteLine(R + X)

        SW.Flush() ' تستخدم لعمل تحديث ولسه فى كلام تانى لسه هيسجل

        SW.Close() ' هيعمل تحديث وهغلق الملف
```

- مثال القراءه داخل ملف تكست :

```
        Dim SR As StreamReader = File.OpenText("C:\Documents and
Settings\mah\Desktop\AGC.txt")
        Do While Not SR.EndOfStream
            'Console.WriteLine(SR.ReadLine)
```

- ده هيقرا سطر

```
           'Console.WriteLine(SR.ReadToEnd)
```

- ده هيقرا الملف كله من اوله حتى اخره

```
            'Console.WriteLine(SR.Read)
```

- ريد فقط سيقوم بقراء حرف حرف وكمان هيقرا رقم الاسكى المقابل مش الحرف نفسه

```
            Console.WriteLine(Chr(SR.Read))
```

- هنا بقى قولت له هات الكركتر بتاع الرقم الاسكى

```
        Loop
        Console.ReadKey()
```

==========

```
        Dim SR As StreamReader = File.OpenText("C:\Documents and
Settings\mah\Desktop\AGC.txt")
        Dim H As String = ""
        Do While Not IsNothing(H)
            H = SR.ReadLine
            Console.WriteLine(H)
        Loop
        Console.ReadKey()
```

- استخدام الاستريم مباشر للكتابه الامثله القادمه سنتعامل مع الاستريم مباشر دون استخدام الفيل بس هنا لازم ناخد نسخه من كلاس الاستريم

```
        Dim SW As New StreamWriter("C:\Documents and Settings\mah\Desktop\AGC.txt", True)
        Dim D As String
1:
        D = Console.ReadLine
        SW.WriteLine(D)
        SW.Flush()
        Console.WriteLine(" 1 = Continue  /  2 = End")
```

```
Dim H As Integer
H = Console.ReadLine
If H = 1 Then GoTo 1
If H = 2 Then End
SW.Close()
```

- استخدام الاستريم مباشر للقراءه

```
Dim SR As New StreamReader("C:\Documents and Settings\mah\Desktop\AGC.txt")
Do While Not SR.EndOfStream 'وصل المؤشر لنهاية الملف
    Console.WriteLine(SR.ReadLine())
Loop
Console.ReadKey()
```

- قراءه ملف فى مصفوفه

```
Dim SR As New StreamReader("C:\Documents and Settings\mah\Desktop\AGC.txt")
```

- هنا انا عايز اعرف طول المصفوفه فبعمل دوراره تشيل اللى جوه الملف النصى وكل ملف الدواره بتلف
- بكون اعلنت عن متغير انتجر وبقول للدواره ان طولك هو طول المتغير الانتجر دا لحد ميخلص قراءه
- وطبعا لازم اعمل اعاده لتعريف المصفوفه واستخدم كلمه بريسرف عشان يحافظ على الطول السابق

```
Dim AR() As Double
Dim I As Integer
Do While Not SR.EndOfStream
    I = I + 1
    ReDim Preserve AR(I)
    AR(I) = SR.ReadLine()
Loop
SR.Close()
```

- هنا بكون عرفت طول المصفوفة بخليه يقرا اللى فى المصفوفه وكان ممكن اخلى الكود فى الدواره اللى فاتت ولكن كنوع من التقسيم

```
For I = 1 To AR.Length - 1
    Console.WriteLine(AR(I))
Next
Console.WriteLine("++++++++++++++++++++++++++++++++++++++++++++")
```

- اعاده الفرز للمصفوفه

```
System.Array.Sort(AR)
```

```
        For I = 1 To AR.Length - 1
            Console.WriteLine(AR(I))
        Next
```

- اعاده الكتابه داخل الملف التكست بالترتيب

```
        Dim SW As New StreamWriter("C:\Documents and Settings\mah\Desktop\AGC.txt",
False)
        For I = 1 To AR.Length - 1
            SW.WriteLine(AR(I))
        Next
        SW.Close()
        Console.WriteLine("++++++++++++++++++++++++++++++++++++++++++++")
        Console.WriteLine("Text File Updated")


        Console.ReadKey()
    End Sub
End Module
```

< BinrayFile :

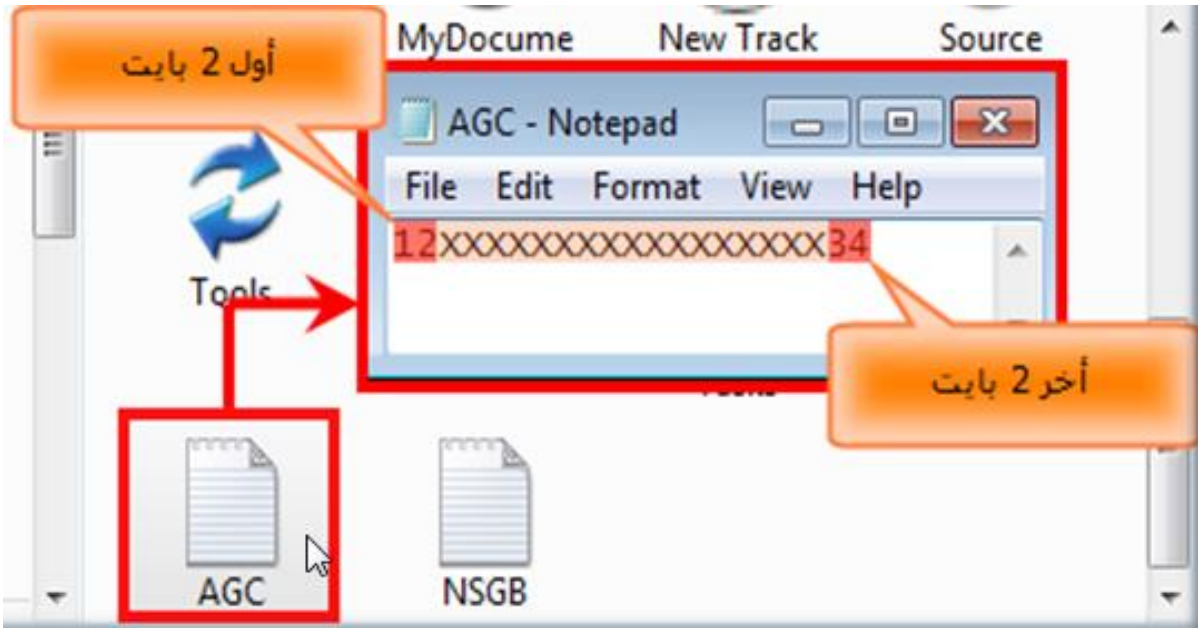
- كل حرف بياخذ مساحة فى الرام بايت تقريبا يعنى كل حرف بياخذ مساحة ٨ بت وبالتالى لابد معرفة وحدات القياس

وحدات قياس الذاكرة	
Bit	أقل وحدة قياس فى الذاكرة
Byte	1 Bit * 8
KB	1 Byte * 1024
MB	1 KB * 1024
GB	1 MB * 1024
TB	1 GB * 1024

- اى ملف مبنى من الشكل ده مجموعة بت (اى ان كل مربع من دول عبارة عن بت) وكل بت تعملها بترتيب معين تطلع صورة وتعملها بترتيب معين تطلع ملف

1	0	0	1	1	0	0	0	1	0
1	0	0	0	1	0	0	0	0	0
0	1	0	1	0	0	1	1	0	0
0	1	1	1	0	1	0	0	1	0
0	0	0	0	1	1	0	1	0	1
0	1	1	1	1	0	0	0	0	1
0	0	1	1	1	0	1	0	1	0
1	1	0	1	0	1	0	0	1	1

- صورة توضح ما سبق:



- هنا فى فكره جديده هى عباره عن تشفير الملفات ولازم اول هام ندرج مكتبه IO
- والحدث اللى فى سب هى استخدام للداله فى الاسفل

```
Imports System.IO

Module BF

    Sub main()

        Dim S As String

        Dim S1 As Integer
```

- دى متغيرات عشان ناخذ قيم المعاملات للداله

```
        Console.WriteLine("ENTER PATH FILE    .: ")

        S = Console.ReadLine


        Console.WriteLine("ENTER Len ENCRYPT .: ")

        S1 = Console.ReadLine

        Console.WriteLine(BinFile(S, S1))

        Console.ReadKey()

    End Sub
```

==========

```
    Function BinFile(ByVal FN As String, ByVal L As Integer) As String
```

- التاكد من طول البيت الخاص بالتشفير وان لا يقل عن 2 بايت

```
        If L < 2 Then

            Console.Clear()

            Return "Len Less Two Character"

        End If

        '-----------------------------------------
```

- التاكد من مسار الملف ان كان موجود ام لا

```
        If File.Exists(FN) = False Then

            Console.Clear()

            Return "File Not Found In This Pass"

        End If

        '-----------------------------------------
```

- قمنا باخذ نسخه من كلاس الفيل استريم وحملنا بداخله ملف تكست وجعل خاصيه الفيل مود اوبن

```
Dim FS As New FileStream(FN, FileMode.Open)
```

- الاعلان عن متغيران مصفوفه عشان اشيل جواهم طول البيت الخاصه بالملف المراد تبدلهم

```
Dim H(L) As Byte

Dim F(L) As Byte
```

- هنا بقول للفيل استريم من مكان من واقف وبطول اتنين بيت حط القيمه دى فى اتش

```
FS.Read(H, 0, L)
```

- هنا بقول للفيل استريم خلى الكورس يقف فى اخر الملف ويرجع لورا اتنين بيت

```
FS.Seek(-L, SeekOrigin.End)
```

- هنا بحمل مكان ما الكورس واقف وبطول اتنين بيت واحط الناتج فى اف

```
FS.Read(F, 0, L)
```

- لحد هنا انا حملت فى كل متغير اتنين بيت
- الباقى بقى انى ابدل القيم دى

==========

- الكود التالى انا هوجه الكورس لاخر الملف ورجه خطوتين او بايتين لورا عشان بعد كده اكتب بداخل الاتش اللى حصل عليه من اخر بايتين فى المف

```
FS.Seek(-L, SeekOrigin.End)

FS.Write(H, 0, L)
```

- الكود هنا بخلى المؤشر يقف فى اول المف ويبدا من الصفر
- عشان يجعل قيمه الاف الاتنين بيت دول

```
FS.Seek(0, SeekOrigin.Begin)

FS.Write(F, 0, L)
```

- هنا قمنا باغلاق الفيل استريم

```
        FS.Close()

        Console.Clear()

        Return "Encryption Done"

    End Function

End Module
```

الفصل التاسع : عمل مكتبة DLL

> مقدمة :

- فى الواقع صناعه المكتبه من اسهل ما يمكن ان نتحدث عنه
- ولكن المهم ان يتم تجهيز الكلاسات الخاصه بتلك المكتبه وبداخلها الاجراءات والدوال

> مكتبة التشفير :

- قمنا بعمل اجراء يقوم بالنداء صوتيا على مايتم كتابته ووصعنها بداخل كلاس وقمنا بعمل داله لتشفير الملفات ووضعناها داخل كلاس وكلا الكلاسين داخل مكتبه ولكن كيف يتم بدء المكتبة
- من خلال فتح مشروع جديد ونختار Make Class وبداخله نضع اكواد مانريد كمايلى :

```
Imports System.IO

Public Class Encryp

    Function BinFile(ByVal FN As String, ByVal L As Integer) As String
```

- التاكد من طول البيت الخاص بالتشفير وان لا يقل عن 2 بايت

```
        If L < 2 Then

            Console.Clear()

            Return "Len Less Two Character"

        End If

        '-----------------------------------------
```

- التاكد من مسار الملف ان كان موجود ام لا

```
        If File.Exists(FN) = False Then

            Console.Clear()

            Return "File Not Found In This Pass"

        End If

        '-----------------------------------------
```

- قمنا باخذ نسخه من كلاس الفيل استريم وحملنا بداخله ملف تكست وجعل خاصيه الفيل مود اوبن

```
        Dim FS As New FileStream(FN, FileMode.Open)
```

- الاعلان عن متغيران مصفوفه عشان اشيل جواهم طول البيت الخاصه بالملف المراد تبدلهم

```
        Dim H(L) As Byte

        Dim F(L) As Byte
```

- هنا بقول للفيل استريم من مكان من واقف وبطول اتنين بيت حط القيمه دى فى اتش

```
FS.Read(H, 0, L)
```

- هنا بقول للفيل استريم خلى الكورس يقف فى اخر الملف ويرجع لورا اتنين بيت

```
FS.Seek(-L, SeekOrigin.End)
```

- هنا بحمل مكان ما الكورس واقف وبطول اتنين بيت واحط الناتج فى اف

```
FS.Read(F, 0, L)
```

- لحد هنا انا حملت فى كل متغير اتنين بيت
- الباقى بقى انى ابدل القيم دى

- الكود التالى انا هوجه الكورس لاخر الملف ورجه خطوتين او بايتين لورا
- عشان بعد كده اكتب بداخل الاتش اللى حصل عليه من اخر بايتين فى المف

```
FS.Seek(-L, SeekOrigin.End)

FS.Write(H, 0, L)

'************************************************
```

- الكود هنا بخلى المؤشر يقف فى اول المف ويبدا من الصفر
- عشان يجعل قيمه الاف الاتنين بيت دول

```
FS.Seek(0, SeekOrigin.Begin)

FS.Write(F, 0, L)
```

- هنا قمنا باغلاق الفيل استريم

```
FS.Close()

Console.Clear()

Return "Encryption Done"

    End Function
End Class

Public Class SPK
```

> مكتبة نطق الكلمة :

- دا سب التحدث صوتيا وفيه انا بعلن عن متغير نوعه توبجكت ومنه بكريت اوبجكت وبنده على ال API الخاص بالصوتيات

```
    Sub SPEAKWORD(ByVal MSG As String)
        Dim SP As Object
        SP = CreateObject("SAPI.SPVOICE")
        SP.Speak(MSG)
    End Sub
End Class
```

- ثم نقوم بعمل AGCDLL من قائمه Debug ونستدعى المكتبه فى المشروع من قائمه Project نختار Properties ومنها Browse ومنها نختار المكتبه ونقوم بالنداء عليها كمايلى وان كان لها اشكال عده للنداء

```
Module TestClass
    Sub main()
```

- اخذ متغيرات تحمل نسخ كل كلاس ثم تمرير القيم

```
        Dim EN As New AGCDLL.Encryp
        Dim SPP As New AGCDLL.SPK
        Console.WriteLine(EN.BinFile("c:\\AGC.TXT", 3))
        SPP.SPEAKWORD(EN.BinFile("c:\\AGC.TXT", 1))
        Console.ReadKey()
    End Sub
End Module
```

المراجع

اكاديمية المجموعة العربية للكمبيوتر

الخاتمة

اهداء وشكر

اكاديمية المجموعة للكمبيوتر على مستوى الذى وصلت اليه

وخاصة الدكتور / عمرو موسى

والمهندس / احمد سمير

والعاملين بالفرع القبة