

```

Public Function _Lambda$__1(ByVal x As Integer) As Integer
    Return x + Me.y
End Function
End Class

```

يمكنك رؤية أن متغير Closure يلتقط المتغير y ويخزنه في الفئة Closure ويتم تحويل المتغير الحر بعدها إلى متغير مرتبط داخل الفئة Closure كما يقوم المترجم بإعادة كتابة الطريقة التي تحتوي على التعبير لمدى لتبدو كما يلي

```

Function MakeLambda() As Func(Of Integer, Integer)
    Dim Closure As New _Closure$__1
    Closure.y = 10
    Return AddressOf Closure._Lambda$__1
End Function

```

يمكنك الآن رؤية كيف يقوم المترجم بإنشاء المتغير Closure ويعيد كتابة المتغير y الذي تم رفعه ضمن المتغير Closure ويضبط قيمته ويعيد ببساطة عنوان تعبير لمدى المخزن ضمن الفئة Closure ومن الهام ملاحظة أن المترجم يقوم برفع المتغيرات الحرة في تعابير لمدى فقط ويتم التقاط حالة المتغير في Closure الذي يبقى موجودا طالما أن تعبير لمدى بقي موجودا. انظر للمثال التالي

```

Sub Test()
    Dim y As Integer = 10
    Dim Lambda As Func(Of Integer, Integer) = Function(ByVal x) x + y
    y = 20
    Console.WriteLine(Lambda(5))
End Sub

```

ما هي القيمة التي تظهر عند تنفيذ الوظيفة السابقة؟ إن قلت 25 فقد أصبت. فلماذا 25 إذا؟ المترجم يقوم بالتقاط وإعادة كتابة جميع المتغيرات الحرة y إلى نسخة Closure كالتالي

```

Sub Test()
    Dim Closure As New $Closure_Compiler_Generated_Name$
    Closure.y = 10
    Dim Lambda = AddressOf Closure.Lambda_1
    Closure.y = 20
    Console.WriteLine(Lambda(5))
End Sub

```

في الوقت الذي يتم تنفيذ تعبير لمدى فيه تكون قيمة y قد تغيرت إلى 20 وبهذا فعندما يتم تنفيذ تعبير لمدى يعيد 5 + 20 وهذا هام جدا لأنه عندما نأتي للحديث عن الحلقات وأن المتغيرات الحرة يتم التقاطها في Closure وحيد قد ترى تصرفات غريبة. انظر للمثال التالي

```

Sub Test()
    For I = 1 To 5
        StartThread(Function() I + 10)
    Next
End Sub

```

افرض أن StartThread ينشئ مسارا جديدا ويطلع النتيجة على الكونسول وطالما أنه تم التقاطه إلى Closure فيمكن أن تكون الحلقة قد غيرت قيمة I في الوقت الذي يقوم المسار فيه باستدعاء تعبير لمدى وفي هذه الحالة فالبرنامج قد لا يطبع النتيجة المتوقعة وبدلا من ذلك عليك رؤية المتغير الملتقط داخل الحلقة

```

Sub Test
    For I = 1 To 5
        Dim x = I
        StartThread(Function() x + 10)
    Next

```