```
    Console.WriteLine(doubleIt(10)
End Sub
```

حيث يأخذ المترجم تعبير لمدا وينشئ وظيفة جديدة بمحتوياته ويغير عبارة التصريح بحيث يأخذ تعبير لمدا عنوان الوظيفة الجديدة المولدة ففي هذه الحالة يتم توليد الوظيفة بنفس الأب الذي يحتوي على الطريقة التي تستخدم تعبير لمدا فإن كان TestLambda معرف في الفئة C فسوف يتم تعريف الوظيفة الجديدة في الفئة C أيضا ونلاحظ أن هذه الوظيفة غير قابلة للاستدعاء ويتم التصريح عنها باستخدام محدد الوصول Private

تعابير لمدا ورفع المتغيرات

في الأمثلة السابقة يشير جسم تعابير لمدا إلى متغيرات يتم تمريرها إلى تلك المتغيرات ومع ذلك تأتي قوة تعابير لمدا مع ثمار رفع المتغيرات وجميع تعابير لمدا مبنية على مبدأ متشابه. وتعبير لمدا يمكن أن يستخدم متغيرات مرتبطة أو متغيرات حرة لم يتم تعريفها ضمن التوقيع الخاص بتعبير لمدا فالمتغيرات الحرة ممكن أن يكون قد تم التصريح عنها في الإجراء المستدعي للتعبير فقد تكون متغيرات محلية أو محددات ممررة لذلك الإجراء والتعابير المرتبطة تكون تلك التي تم التصريح عنها في جسم التعبير أو عناصر في الفئة المحتوية للتعبير لمدا متضمنا الفئة الأب لتلك الفئة. وهذا هام من أجل التمييز بين المتغيرات المرتبطة والحرة في تعابير لمدا الخاصة بك لأنها تؤثر على دلالة تعبير لمدا والكود الذي يتم توليده وبالتالي يؤثر على صحة برنامجك وهذا مثال يحتوي على تعابير لمدا تستخدم متغيرات مرتبطة وأخرى حرة

```
Function MakeLambda() As Func(Of Integer, Integer)
    Dim y As Integer = 10
    Dim addTen As Func(Of Integer, Integer) = Function(ByVal x) x + y
    Return addTen
End Function

Sub UseLambda()
    Dim addTen = MakeLambda()
    Consloe.WriteLine(addTen(5))
End Sub
```

فهذا الكود سيقوم بطباعة 15 على نافذة الكونسول عندما يتم استدعاء UseLambda ولكن يمكن أن تسأل نفسك كيف يعمل هذا؟ تحدد الوظيفة MakeLambda المتغير y كمتغير محلي والتعبير لمدا يستخدم y ولكن التعبير لمدا يتم إعادته كنوع معاد من الوظيفة MakeLambda والوظيفة UseLambda تحصل على التعبير لمدا من الوظيفة MakeLambda وتنفذ التعبير لمدا ويبدو الأمر كما لو أن المتغير y قد تم تذكره من قبل التعبير لمدا. ففترة حياة المتغير y تنتهي مع نهاية الطريقة MakeLambda فعندما نحصل على التعبير لمدا من MakeLambda فسوف تصبح MakeLambda خارج المجال ويجب إزالة المساحة التي تحجزها في المكدس وبطريقة ما يعلق هذا المتغير مع تعبير لمدا وهذا ما يعرف برفع المتغير Variable Lifting ففي هذه الحالة يدعى المتغير y بالمتغير المرفوع وكما ترى فالمتغيرات المرفوعة تعتبر ميزة برمجية قوية فالمترجم يقوم بالكثير من العمل من أجل تمكينك من إمساك حالة المتغير حيث يحفظها خارج مجال فترة حياتها الطبيعية فعندما يصادف المترجم تعابير لمدا تستخدم متغيرات حرة يقوم برفع المتغير إلى فئة تدعى Closure بحيث تكون فترة حياة هذه الفئة تمتد إلى ما بعد فترة حياة المتغيرات الحرة المستضافة داخلها ويقوم المترجم بإعادة كتابة الوصول إلى المتغيرات في الطرق ليتم الوصول إلى نسختها الموجودة في الفئة Closure

دعنا نسير مرة أخرى عبر المثال MakeLambda

```
Dim MakeLambda() As Func(Of Integer, Integer)
    Dim y As Integer = 10
    Dim addTen As Func(Of Integer, Integer) = Function(ByVal x) x + y
        Return addTen
End Function
```

وكما قمنا بالتحليل سابقا فالمتغير x مرتبط بمحدد التعبير لمدا ولكن المتغير y تعبير حر ويقوم المترجم بالكشف عن ذلك ويتابع بإنشاء الفئة Closure التي تلتقط المتغيرات الحرة كما في تعريف تعبير لمدا

```
Public Class _Closure$__1
    Public y As Integer
```