

# OLCDURATION

Updated: 31 Mar 2016

Use **OLCDURATION** to calculate the duration for a bond that has an odd last coupon. The duration is calculated as the first derivative of the price of the bond with respect to yield multiplied by -1, divided by the dirty price of the bond multiplied by 1 plus the yield divided by the frequency.

$$DURATION = \frac{-\frac{\partial P}{\partial y}}{P_{dirty}} \left( 1 + \frac{Y}{F} \right)$$

## Syntax

```
Public Shared Function OLCDURATION(  
    ByVal Settlement As Date,  
    ByVal Maturity As Date,  
    ByVal LastCouponDate As Date,  
    ByVal Rate As Double,  
    ByVal Yld As Double,  
    ByVal Redemption As Double,  
    ByVal Frequency As Double,  
    ByVal Basis As String,)
```

## Arguments

### *Settlement*

the settlement date of the security. *Settlement* is an expression that returns a **Date**, or of a type that can be implicitly converted to **Date**.

### *Maturity*

the maturity date of the security. *Maturity* is an expression that returns a **Date**, or of a type that can be implicitly converted to **Date**.

### *LastCouponDate*

the last coupon date of the security. The period from the last coupon date until the maturity date defines the odd interest period. The quasi-maturity date is assumed to occur at a regular periodic interval as defined by *Frequency* and *Basis*. *LastCouponDate* is an expression that returns a **Date**, or of a type that can be implicitly converted to **{paramtype}**.

### *Rate*

the bond's annual coupon rate. *Rate* is an expression that returns a **Double**, or of a type that can be implicitly converted to **Double**.

### *Yld*

the yield for the maturity date passed into the function. *Yld* is an expression that returns a **Double**, or of a type that can be implicitly converted to **Double**.

### Redemption

the redemption value of the bond assuming a par value of 100. *Redemption* is an expression that returns a **Double**, or of a type that can be implicitly converted to **Double**.

### Frequency

the number of coupon payments per year. For annual payments, *Frequency* = 1; for semi-annual, *Frequency* = 2; for quarterly, *Frequency* = 4; for bi-monthly, *Frequency* = 6; for monthly, *Frequency* = 12. For bonds with Basis = "A/364" or 9, you can enter 364 for payments made every 52 weeks, 182 for payments made every 26 weeks, 91 for payments made every 13 weeks, 28 for payments made every 4 weeks, 14 for payments made every 2 weeks, and 7 for weekly payments. *Frequency* is an expression that returns a **Double**, or of a type that can be implicitly converted to **Double**.

### Basis

the type of day count to use.

Basis	Day count basis
0, "BOND"	US (NASD) 30/360
1, "ACTUAL"	Actual/Actual
2, "A360"	Actual/360
3, "A365"	Actual/365
4, "30E/360 (ISDA)", "30E/360", "ISDA", "30E/360 ISDA", "EBOND"	European 30/360
5, "30/360", "30/360 ISDA", "GERMAN"	30/360 ISDA
6, "NL/ACT"	No Leap Year/ACT
7, "NL/365"	No Leap Year /365
8, "NL/360"	No Leap Year /360
9, "A/365"	Actual/364
10, "BOND NON-EOM"	US (NASD) 30/360 non-end-of-month
11, "ACTUAL NON-EOM"	Actual/Actual non-end-of-month
12, "A360 NON-EOM"	Actual/360 non-end-of-month
13, "A365 NON-EOM"	Actual/365 non-end-of-month
14, "30E/360 NON-EOM", "30E/360 ICMA NON-EOM", "EBOND NON-EOM"	European 30/360 non-end-of-month
15, "30/360 NON-EOM", "30/360 ISDA NON-EOM", "GERMAN NON-EOM"	30/360 ISDA non-end-of-month
16, "NL/ACT NON-EOM"	No Leap Year/ACT non-end-of-month
17, "NL/365 NON-EOM"	No Leap Year/365 non-end-of-month
18, "NL/360 NON-EOM"	No Leap Year/360 non-end-of-month
19, "A/365 NON-EOM"	Actual/364 non-end-of-month

*Basis* is an expression that returns a **String**, or of a type that can be implicitly converted to **String**.

## Return Type

Double

## Remarks

- If *Basis* < 0 or *Basis* > 19 an error will be returned.
- If *Maturity* <= *Settlement* 0 is returned.
- If *Settlement* is NULL, *Settlement* current system processing date
- If *Rate* is NULL, *Rate* = 0
- If *Yld* is NULL, *Yld* = 0
- If *Frequency* is NULL, *Frequency* = 2
- If *Basis* is NULL, *Basis* = 0.
- *Rate* is entered as a decimal value; 1.0% = 0.01
- *Yld* is entered as a decimal value; 1.0% = 0.01
- If *Maturity* is NULL an error will be returned.
- If *IssueDate* is NULL an error will be returned.
- If *LastCouponDate* is NULL an error will be returned.
- If *Frequency* invalid an error will be returned.

## See Also

- CFCONVEXITY - Convexity of a series of cash flows
- CFDURATION - Duration of a series of cash flows
- CFMDURATION - Modified duration of a series of cash flows
- CONVEXITY - Convexity of an option free bond
- DURATION - Duration of a security
- MDURATION - Macauley Duration
- OFCCONVEXITY - Convexity of a bond with and odd first coupon
- OFCDURATION - Duration of a bond with an odd first coupon
- OFCMDURATION - Modified duration of a bond with an odd first coupon
- OFLCONVEXITY - Convexity of a bond with an odd first and odd last coupon
- OFLDURATION - Duration of a bond with an odd first and odd last coupon
- OFLMDURATION - Modified duration of a bond with an odd first and odd last coupon
- OLCCONVEXITY - Convexity of a bond with an odd last coupon
- OLCMDURATION - Modified duration of a bond with an odd last coupon
- RPICONVEXITY - Convexity of a bond paying regular periodic interest
- RPIDURATION - Duration of a bond paying regular periodic interest
- RPIMDURATION - Modified duration of a bond paying regular periodic interest
- STEPCONVEXITY - Convexity of a stepped-coupon bond
- STEPDURATION - Duration of a stepped-coupon bond
- STEPMDURATION - Modified duration of a stepped-coupon bond