Topic	Syntax

Example 1	----------
	Function example1(rate)
	  If rate < 0.01 Then
	  example1 = 0.01
	
	  ElseIf rate > 0.04 Then
	  example1 = 0.04
	
	  Else
	  example1 = rate
	  End If
	
	End Function
	---------- 	

Exercise 1	----------
	Function exercise1(x As Integer) As String
	  If x = 1 Then
	  exercise1 = "A"
	  ElseIf x = 2 Then
	  exercise1 = "B"
	  ElseIf x = 3 Then
	  exercise1 = "C"
	  ElseIf x > 3 Then
	  exercise1 = "D"
	  Else
	  exercise1 = "Value below 1"
	  End If
	End Function
	---------- 	

Exercise 2	----------
	Function exercise2(Bondtype As String, Facevalue As Double) As Double
	  Dim f As Double
	  Dim b As String
	  Dim r As Double
	
	  f = Facevalue
	  b = Bondtype
	
	  If UCase(b) = "AAA" Then
	  r = 0.03
	  ElseIf UCase(b) = "AA" Then
	  r = 0.04
	  ElseIf UCase(b) = "A" Then
	  r = 0.05
	  ElseIf UCase(b) = "B" Then
	  r = 0.06
	  Else
	  r = 0.08
	  End If
	
	  exercise2 = f * r

	End Function ----------

Exercise 3 – do at home	----------
	Function exercise3(x As Double, y As Double) As Double
	  If x > 10 Then
	  If y > 5 Then
	  exercise3 = 1
	  Else
	  exercise3 = 2
	  End If
	  ElseIf x < -10 Then
	  If y > 5 Then
	  exercise3 = 3
	  Else
	  exercise3 = 4
	  End If
	  ElseIf y > 5 Then
	  If x = y Then
	  exercise3 = 5
	  Else
	  exercise3 = 6
	  End If
	  Else
	  exercise3 = 7
	  End If
	
	End Function

Example 2	----------
	Function NewPV(CF, r)
	  Dim i As Integer
	  Dim Temp
	
	  Temp = 0
	  For i = 1 To 5
	  Temp = Temp + CF / (1 + r) ^ i
	  Next i
	
	  NewPV = Temp
	End Function
	---------- 	

Exercise 4	----------
	Function BetterNewPV(CF as double, r as double, n as integer) as double
	  Dim i As Integer
	  Dim Temp as double
	
	  Temp = 0
	  For i = 1 To n
	  Temp = Temp + CF / (1 + r) ^ i
	  Next i
	  BetterNewPV = Temp
	End Function
	---------- 	

Example 3	---------- Function example3(Capital As Double, deposit As Double, rate As Double) As Double

	Dim n As Integer Dim amount As Double n = 0 amount = 0 Do While Capital >= amount (or: Do until Capital <= amount) n = n + 1 amount = amount * (1 + rate) + deposit Loop example3 = n End Function ----------

Example 3 (extended)	----------
	Function example3extended(Capital As Double, deposit As Double, rate As Double, **prob As Double**) As Double
	Dim n As Integer
	Dim amount As Double, **draw As Double**, **z_score As Double**
	Dim lottery As Boolean
	
	n = 0
	amount = 0
	lottery = False
	z_score = WorksheetFunction.NormSInv(prob)
	
	Do While (Capital >= amount) **And (lottery = False)**
	
	n = n + 1
	amount = amount * (1 + rate) + deposit
	draw = WorksheetFunction.NormSInv(Rnd())
	If draw < z_score Then
	lottery = True
	End If
	Loop
	example3extended = n
	End Function
	---------- 	

Exercise 5	---------- Function exercise5(loan As Double, rate As Double, payment As Double) As Double Dim n As Integer n = 0 Do Until loan <= 0 n = n + 1 loan = loan * (1 + rate) - payment Loop exercise5 = n End Function ----------
Exercise 6 – do at home	---------- Function exercise6(S As Double, X As Double, T As Double, v As Double, r As Double, q As Double, CallPut As String) As Double

	Dim d1 As Double
	Dim d2 As Double
	
	d1 = (Log(S / X) + (r - q + v ^ 2 / 2) * T) / (v * Sqr(T))
	d2 = d1 - v * Sqr(T)
	
	If UCase(CallPut) = "C" Then
	exercise6 = S * WorksheetFunction.NormSDist(d1) * Exp(-q * T) - X * Exp(-r * T) * WorksheetFunction.NormSDist(d2)
	ElseIf UCase(CallPut) = "P" Then
	exercise6 = X * Exp(-r * T) * WorksheetFunction.NormSDist(-d2) - S * WorksheetFunction.NormSDist(-d1) * Exp(-q * T)
	Else
	MsgBox "CallPut must be C for call or P for put"
	End If
	End Function
	---------- 	

Example 4	---------- Sub example4() Dim i As Integer Dim j As Integer For i = 1 To 20 For j = 1 To 20 Cells(i, j) = i * j Next j Next i Cells(21, 21).Value = WorksheetFunction.Sum(Range("A1:T20")) End Sub ----------
Example 5	---------- Sub example5() Dim i As Integer Dim arr(1 To 100, 0) As Integer For i = 1 To 100 arr(i, 0) = i Next i Range("a1:a100") = arr End Sub
Exercise 7	---------- Sub exercise7() Dim i As Integer, j As Integer Dim arr(1 To 20, 1 To 20) For i = 1 To 20 For j = 1 To 20 arr(i, j) = i * j Next j Next i Range("a1:t20") = arr Range("a21") = Application.WorksheetFunction.Sum(arr) End Sub ----------

Example 6	---------- Sub example6() Dim i As Integer Dim output(3 To 36, 0) Dim dataind() As Variant 'Defined as variant since it's not possible to read a range into other arrays than of type variant For i = 1 To 34 dataind = Range("a" & i & ":a" & i + 2).Value 'read three numbers into the array dataind output(i + 2, 0) = Application.WorksheetFunction.Average(dataind) Next i Range("d3:d36") = output End Sub ----------

Example 7	----------
	Sub Call_upper_bound()
	Dim optiondata As Range
	Dim numstrikes As Integer
	Dim arbitrage As Boolean
	Dim i As Integer
	Dim spot As Double, T As Double, r As Double
	
	spot = Worksheets("Option_Data").Range("B3").Value
	T = Worksheets("Option_Data").Range("B5").Value
	r = Worksheets("Option_Data").Range("B7").Value
	
	Set optiondata = Worksheets("Option_Data").Range("A11", "C193")
	numstrikes = optiondata.Rows.Count
	arbitrage = False
	i = 1
	
	While ((i <= numstrikes) And (arbitrage = False))
	If optiondata.Cells(i, 2).Value > spot Then
	arbitrage = True
	MsgBox "Call Upper bound violated at " & optiondata.Cells(i, 2).Address
	Else
	i = i + 1
	End If
	Wend
	If arbitrage = False Then
	MsgBox "no call upper bound is violated"
	End If
	End Sub
	---------- 	

Exercise 8	---------- Sub Put_upper_bound() Dim optiondata As Range Dim numstrikes As Integer Dim arbitrage As Boolean Dim i As Integer Dim spot As Double, T As Double, r As Double

	spot = Worksheets("Option_Data").Range("B3").Value
	T = Worksheets("Option_Data").Range("B5").Value
	r = Worksheets("Option_Data").Range("B7").Value
	
	Set optiondata = Worksheets("Option_Data").Range("A12", "C193")
	numstrikes = optiondata.Rows.Count
	arbitrage = False
	i = 1
	
	While ((i <= numstrikes) And (arbitrage = False))
	  If optiondata.Cells(i, 3).Value > optiondata.Cells(i, 1).Value * Exp(-r * T) Then
	  arbitrage = True
	  MsgBox "Put Upper bound violated at " & optiondata.Cells(i, 3).Address
	  Else
	  i = i + 1
	  End If
	Wend
	  If arbitrage = False Then
	  MsgBox "No put upper bound is violated"
	  End If
	End Sub
	---------- 	

Advanced exercise 9 – do at home	----------
	
	Sub regression()
	
	Dim ry As range, rx As range
	Dim count As Integer, rangestart As Integer, rangeend As Integer
	
	count = 0
	rangestart = 1
	rangeend = 60
	Dim beta(1 To 300, 1)
	
	  Do While count <= 299
	  count = count + 1
	  rangestart = rangestart + 1
	  rangeend = rangeend + 1
	
	  Set ry = range("c" & rangestart & ":c" & rangeend)
	  Set rx = range("b" & rangestart & ":b" & rangeend)
	
	  beta(count, 0) = Application.WorksheetFunction.Slope(ry, rx)
	
	  Loop
	
	range("e2:e302") = beta
	
	End Sub
	---------- 	