



www.oppadu.com

오피두엑셀 무료 E-Book

## 1. 예제파일 이해하기 (직원 급여명세서 기본양식)

예제파일은 3 개의 시트를 포함합니다.

- ✓ **세율표** : [2019 년 기준] 소득세 및 4 대보험의 요율이 작성된 시트입니다.
- ✓ **급여자료** : 세율표를 기준으로 각 직원별 월소득에 따른 공제액과 실수령액을 계산합니다.
- ✓ **급여명세서** : 월과 사원번호를 선택하여 직원의 급여명세서를 출력합니다. 자료는 급여자료시트를 기준으로 출력합니다.

급여자료 및 급여명세서에는 **INDEX/MATCH 함수**가 사용되었습니다. 함수에 대한 자세한 설명은 관련포스트를 참고하세요.

공식: `=SUM(G2:O2)*INDEX(세율표!$B$4:$D$10,MATCH(SUM(G2:O2)*12,세율표!$B$4:$B$10,1),2)-INDEX(세율표!$B$4:$D$10,MATCH(SUM(G2:O2)*12,세율표!$B$4:$B$10,1),3)/12`

기본	직책수	연장수	휴일수	상여	소득	주인	고용보	국민연	장기요	건강보
5,500,000	300,000			150,000	70,000	100,980	39,130	270,900	16,254	194,446
4,500,000	200,000			150,000	70,000	745,800	74,580	31,980	221,400	13,284
3,000,000	50,000			150,000	70,000	400,500	40,050	21,255	147,150	8,829

[급여자료] 시트의 소득세 및 각종 보험지급액은 [세율표] 시트를 참고하여 계산합니다.

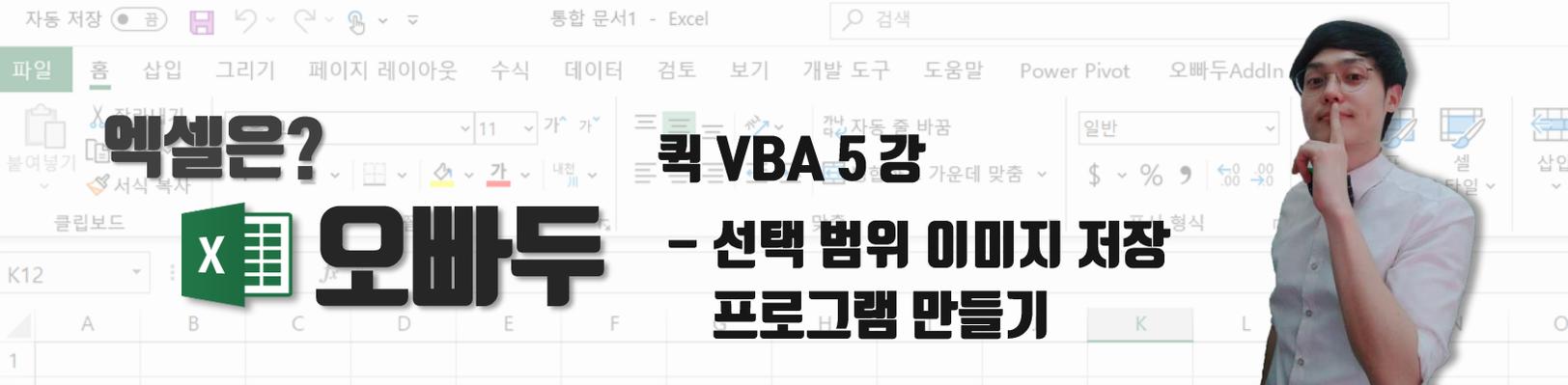
2019년 8월 급여 명세서				날짜/직원선택	
사 번 :	OPD0003	성 명 :	형님두	년도	2019
부 서 :	중무부	직 책 :	대리	월	8
지급방식	지급액	공제항목	공제액	사번	OPD0003
기본급	<code>=INDEX(Table1[#모두],MATCH(1,--(Table1[#모두],[사원번호])=급여명세서!\$C\$3)*--(Table1[#모두],[지급년도]=급여명세서!\$H\$3)*--(</code>				
직책수당	50,000	수민세	45,300		
연장수당	-	고용보험	23,530		
휴일수당	350,000	국민연금	162,900		
상여금	-	장기요양	9,774		

[급여명세서] 시트의 각 직원별 지급액/공제액은 INDEX/MATCH 다중조건 함수를 사용하여 참조합니다.

## 2. 본 강의에 사용된 VBA 사용자 지정함수

본 강의에서는 **총 4 개의 추가 사용자지정함수**를 사용하였습니다. 각 함수에 대한 자세한 설명은 관련 포스트를 참고하세요.

- FileExists 함수 : 파일경로의 기존 파일존재여부를 확인합니다.
- GetDesktopPath 함수 : 컴퓨터의 바탕화면 경로를 출력합니다.
- FileSequence 함수 : 파일이름이 중복되지 않도록 순번을 매깁니다.
- ValidFileName 함수 : 지정한 파일이름의 사용가능여부를 확인합니다.



www.oppadu.com

오빠두엑셀 무료 E-Book

**FileExists 함수 :: 사용자지정함수**

설명 : 지정한 파일경로의 파일존재여부를 확인합니다.

구문 := FILEEXISTS ( 파일경로 )

[오빠두 함수 상세설명 링크](#)

```
Public Function FileExists(ByVal path_ As String) As Boolean
FileExists = (Dir(path_, vbDirectory) <> "")
End Function

Sub 파일존재여부확인 (Path As String)
'#####
'Path 로 입력된 경로에 파일이 존재하는지 여부를 확인하여 안내메세지를 띄웁니다.
'#####

If FileExists(Path) = True Then
MsgBox "해당 파일이 이미 존재합니다." & vbNewLine & _
"파일경로 : " & vbNewLine & _
Path
Else
MsgBox "해당 파일이 존재하지 않습니다."
End If

End Sub
```

**GetDesktopPath 함수 :: 사용자지정함수**

설명 : 사용자 컴퓨터의 바탕화면경로를 출력합니다.

구문 := GETDESKTOPPATH ( [백슬래쉬표시] )

[오빠두 함수 상세설명 링크](#)

```
Public Function GetDesktopPath(Optional BackSlash As Boolean = True) As String
'// 사용자의 바탕화면 경로를 출력합니다.

Dim oWShell As Object

Set oWShell = CreateObject("WScript.Shell")

If BackSlash = True Then
GetDesktopPath = oWShell.SpecialFolders("Desktop") & "\\"
Else
GetDesktopPath = oWShell.SpecialFolders("Desktop")
End If
```



www.oppadu.com

오빠두엑셀 무료 E-Book

```

End If

Set oWSHShell = Nothing

End Function

Sub 바탕화면경로_()

MsgBox "현재 사용중인 컴퓨터의 바탕화면 경로는" & vbCrLf & _
      "[[ " & GetDesktopPath & " ]]" & vbCrLf & _
      "입니다."

End Sub

```

**FileSequence 함수 :: 사용자지정함수**

설명 : 파일이름이 중복되지 않도록 파일이름에 순번을 매깁니다.

구문 := FILESEQUENCE( 파일경로, [시작번호] )

[오빠두 함수 상세설명 링크](#)

```

Function FileSequence(FilePath As String, Optional Sequence As Long = 1) as String

Dim Ext As String: Dim Path As String: Dim newPath As String
Dim Pnt As Long

Pnt = InStrRev(FilePath, ".")
Path = Left(FilePath, Pnt - 1)
Ext = Right(FilePath, Len(FilePath) - Pnt + 1)

newPath = Path & Sequence & Ext

Do Until FileExists(newPath) = False
  Sequence = Sequence + 1
  newPath = Path & Sequence & Ext
Loop

FileSequence = newPath

End Function

Sub 파일저장 ()

Dim strPath As String
Dim newPath As String

strPath = GetDesktopPath & "복사본.xlsx"
newPath = FileSequence(strPath, 1)

```



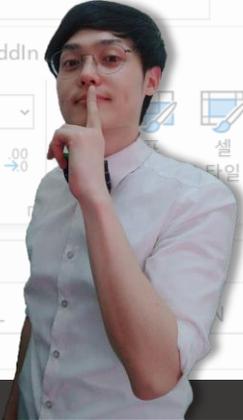
엑셀은?



오빠두

퀵 VBA 5강

- 선택 범위 이미지 저장 프로그램 만들기



www.oppadu.com

오빠두엑셀 무료 E-Book

```
ThisWorkbook.SaveAs newPath

MsgBox newPath & "경로로 파일저장을 완료하였습니다."

End Sub
```

**ValidFileName 함수 :: 사용자지정함수**

설명 : 지정한 파일이름의 사용가능 여부를 확인합니다.

구문 := VALIDFILENAME ( 파일경로 )

[오빠두 함수 상세설명 링크](#)

```
Function ValidFileName (FileName As String) As Boolean

Dim Arr As Variant: Dim Val As Variant
Dim Pnt As Long

Arr = Array("/", "\", ":", "*", "?", " ", "<", ">", "|")

If InStr(1, FileName, ":\") > 0 Then
    Pnt = InStrRev(FileName, "\")
    FileName = Right(FileName, Len(FileName) - Pnt)
    Debug.Print FileName
End If

ValidFileName = True

For Each Val In Arr
    If InStr(1, FileName, Val) > 0 Then ValidFileName = False: Exit Function
Next

End Function

Sub 파일이름체크 ()

If ValidFileName (Sheet1.Range("F2").Value) = True Then
    MsgBox "사용가능한 파일이름입니다."
Else
    MsgBox "사용 불가능한 파일이름입니다.", vbCritical
End If

End Sub
```



### 3. 매크로 동작원리 이해하기

선택한 범위를 이미지로 저장하는 매크로는 **아래의 9 단계로 작성**합니다.

엑셀은 이미지를 바로 출력/저장하는 기능을 지원하지 않습니다. 따라서 **이미지를 빈 차트안에 붙여넣기 한 뒤, 차트를 그림형태로 출력**하는 방법을 사용합니다.

1. 선택된 범위를 그림형태로 복사
2. 임시 워크시트 추가
3. 추가된 임시시트에 이미지 붙여넣기
4. 이미지의 높이와 넓이 확인후 이미지 제거
5. 추가된 임시시트에 빈 차트 생성
6. 차트의 높이와 넓이를 이미지와 동일하게 변경
7. 차트안에 이미지 붙여넣기
8. 차트를 원하는 파일경로에 이미지형태로 출력
9. 임시 워크시트 삭제

### 4. VBA 명령문 작성하기

범위를 이미지로 저장하는 매크로 명령문을 작성합니다. 명령문 각 단계별로 알아보겠습니다. 우선 Rng\_To\_Image 라는 명령문을 아래와 같이 **인수를 포함**하여 만들어주세요.

```
Sub Rng_To_Image(rngSelection As Range, _
    Optional FileName As String = "엑셀이미지", _
    Optional SavePath As String = "", _
    Optional AddSequence As Boolean = True)

End Sub
```

명령문에 추가된 인수들은 아래와 같습니다.

- ✓ rngSelection : 사용자가 선택한 범위입니다.
- ✓ FileName : 이미지로 저장될 파일명입니다. (기본값: 엑셀이미지)
- ✓ SavePath : 이미지를 저장할 파일경로입니다. (기본값: 빈칸)
- ✓ AddSequence : 이미지를 덮어쓰기 할지 또는 순번에 따라 저장할지 결정합니다.



# 엑셀은?



# 오빠두

## 퀵 VBA 5강

### - 선택 범위 이미지 저장 프로그램 만들기



www.oppadu.com

오빠두엑셀 무료 E-Book

#### 4-A. 변수생성

아래와 같이 변수를 생성합니다.

- ✓ NewWs : 임시로 추가할 워크시트입니다.
- ✓ PicRange : 선택한 범위를 그림형태로 붙여넣기 한 이미지 개체입니다.
- ✓ MyObj : 이미지를 붙여넣기 할 차트 개체입니다.
- ✓ PicH, PicW : 이미지의 높이와 넓이입니다.
- ✓ FilePath : 이미지가 저장될 파일명을 포함한 전체 파일경로입니다.

```
Dim NewWs As Worksheet
Dim picRange As Object: Dim MyObj As Chart
Dim PicH As Double: Dim PicW As Double
Dim FilePath As String
```

#### 4-B. 파일이름 점검 및 저장할 파일경로 생성

윈도우에는 **파일이름으로 사용불가능한 특수기호**가 있습니다. 사용자가 지정한 파일이름이 윈도우에서 사용가능한지 파일이름을 점검합니다.

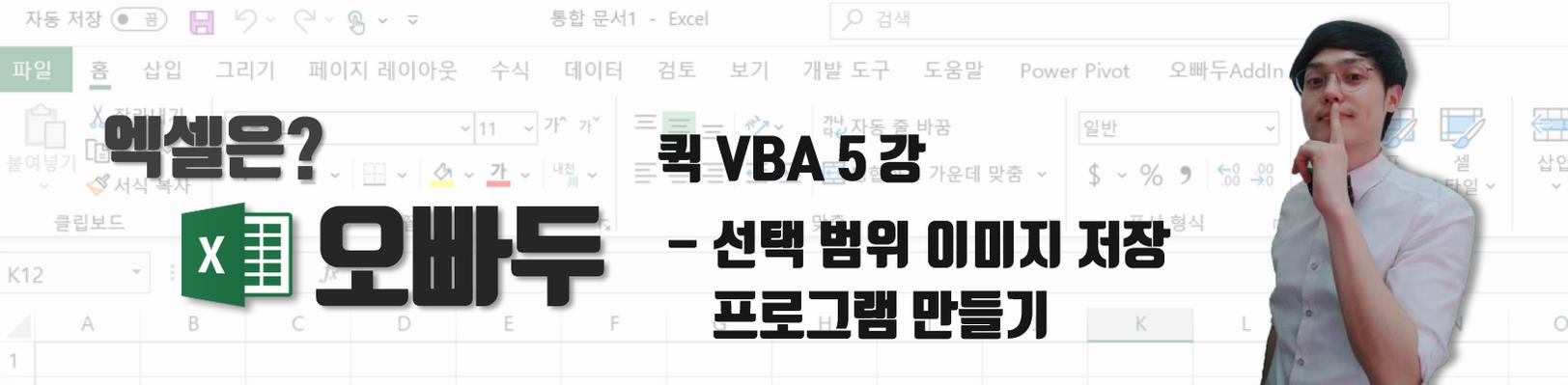
파일이름에 이상이 없을 시, 전체 파일경로를 생성합니다. 명령문의 보조 인수 중 SavePath 가 빈칸일 경우, [GetDesktopPath 함수](#)를 통해 바탕화면에 파일이 저장되도록 만들어줍니다.

AddSequence 인수가 True 일경우, 파일 이름을 순번에 따라 저장합니다. 이전 단계에서 붙여넣기 한 [FileSequence 함수](#)를 통해 파일이름에 순번을 매겨줍니다.

```
If ValidFileName (FileName) = False Then MsgBox "올바른 파일명을 사용하세요.": End

If SavePath = "" Then SavePath = GetDesktopPath
FilePath = SavePath & FileName & ".png"

If AddSequence = True Then
    FilePath = FileSequence(FilePath, 1)
End If
```



#### 4-C. 선택한 범위를 그림형태로 복사

선택된 범위를 그림형태로 복사합니다. 본 예제에서는 사용자가 선택한 범위를 rngSelection 으로 받아왔지만, 별도의 범위를 임의로 설정할 수도 있습니다.

```
rngSelection.CopyPicture xlScreen, xlPicture
```

#### 4-D. 임시시트 생성 및 그림 붙여넣기

그림을 붙여넣기 할 임시시트를 생성한 뒤, 시트에 이미지를 붙여넣기 합니다.

```
Set NewWs = ActiveWorkbook.Sheets.Add
NewWs.Paste
```

#### 4-E. 이미지의 높이와 넓이 확인 후 이미지 제거

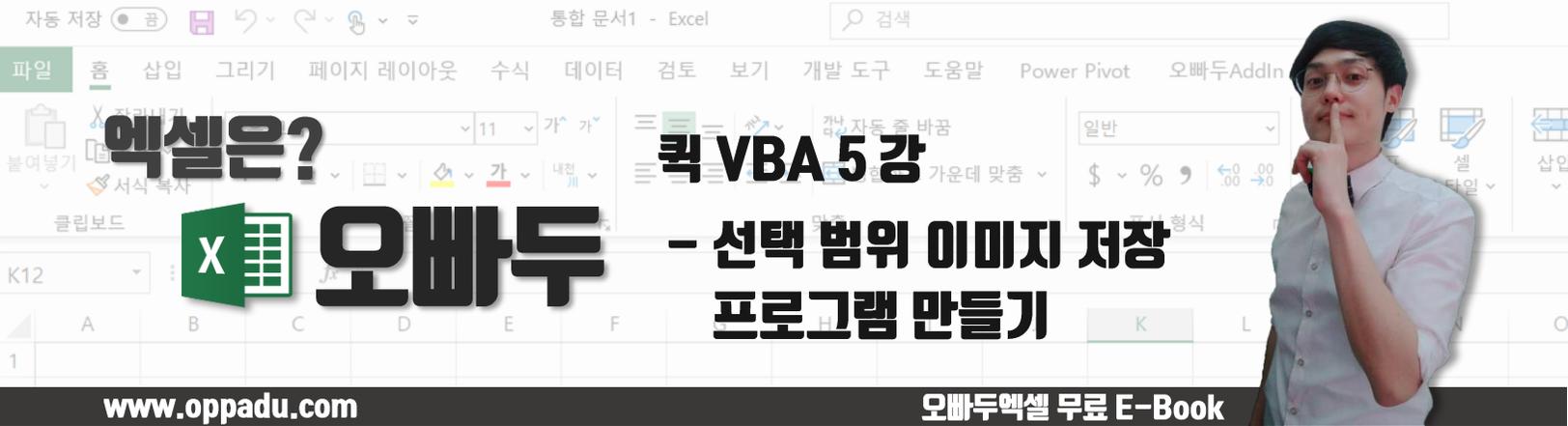
붙여넣기 한 이미지의 높이와 넓이를 확인한 뒤, 이미지를 삭제합니다.

```
Set picRange = NewWs.Shapes.Item(1)
With picRange
    PicH = .Height
    PicW = .Width
    .Delete
End With
```

#### 4-F. 빈 차트 생성 및 높이와 넓이 조정

임시 시트에 빈 차트를 생성한 뒤, 차트의 높이와 넓이를 이미지와 동일하게 조정합니다.

```
With NewWs.Shapes.AddChart2
    .Height = PicH
    .Width = PicW
End With
```



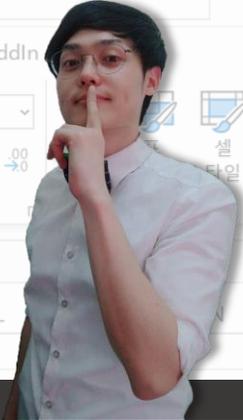
# 엑셀은?



# 오빠두

## 퀵 VBA 5강

### - 선택 범위 이미지 저장 프로그램 만들기



www.oppadu.com

오빠두엑셀 무료 E-Book

#### 4-G. 빈 차트안에 그림 붙여넣기

차트 안에 그림을 붙여넣기 합니다. 차트안에 그림을 붙여넣기 할 때에는 반드시 **Select** 를 통해 차트가 선택이 된 상태에서 **Paste** 로 붙여넣기 해야만 매크로가 동작한다는 점을 주의하세요.

```
Set MyObj = NewWs.Shapes.Item(1).Chart
MyObj.ChartArea.Select
MyObj.Paste
```

#### 4-H. 차트를 이미지로 저장

차트를 이미지로 저장합니다. 이전 단계에서 만들어준 파일 경로에 PNG 형태의 이미지로 저장합니다.

```
MyObj.Export FilePath, "PNG"
```

#### 4-I. 임시시트 삭제

매크로 실행을 위해 추가했던 임시 워크시트를 삭제합니다. 엑셀은 기본적으로 시트를 삭제할 시 안내메세지를 출력합니다. 안내메세지 출력을 방지하기 위해 시트 삭제 전 **DisplayAlerts 속성**을 False 로 변경합니다.

```
Application.DisplayAlerts = False
NewWs.Delete
Application.DisplayAlerts = True
```

### 5. 이미지 저장 매크로 테스트하기

작성한 명령문이 잘 동작하는지 테스트합니다. 아래와 같이 Test 명령문을 작성한 뒤 실행합니다.

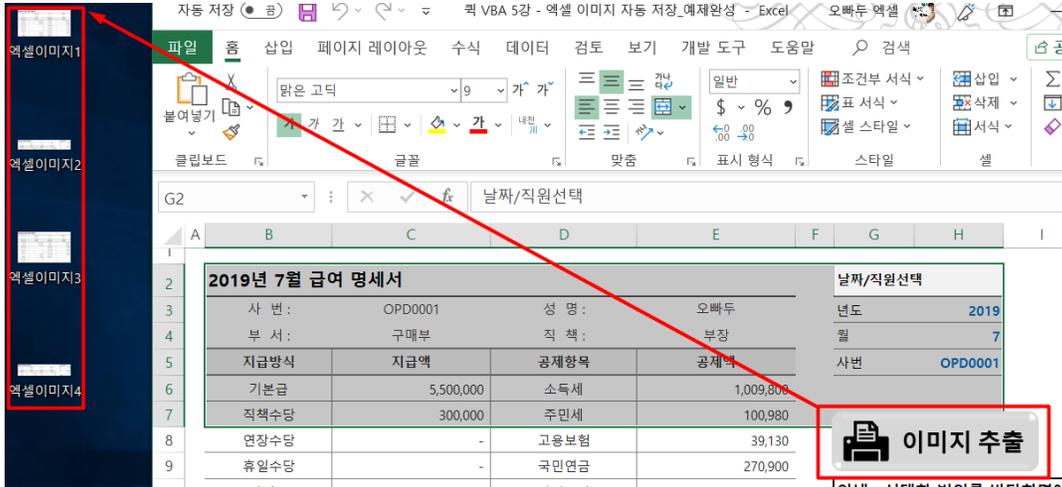
```
Sub Test ()
Rng_To_Image Selection
End Sub
```



www.oppadu.com

오편두엑셀 무료 E-Book

매크로를 실행하면 아래와 같이 바탕화면에 선택된 범위가 이미지로 저장되는 것을 확인할 수 있습니다.



시트에 도형을 추가한 뒤, Test 매크로를 지정하여 매크로가 잘 동작하는지 테스트합니다.

## 6. 강의에 사용된 전체 명령문

```
Option Explicit

Sub Test ()

Rng_To_Image Selection

End Sub

Function ValidFileName (FileName As String) As Boolean

Dim Arr As Variant: Dim Val As Variant
Dim Pnt As Long

Arr = Array("/", "\", ":", "*", "?", " ", "<", ">", "|")

If InStr(1, FileName, ":\"") > 0 Then
    Pnt = InStrRev(FileName, "\")
    FileName = Right(FileName, Len(FileName) - Pnt)
    Debug.Print FileName
End If

ValidFileName = True

For Each Val In Arr
    If InStr(1, FileName, Val) > 0 Then ValidFileName = False: Exit Function
Next
```



```
End Function
```

```
Sub Rng_To_Image(rngSelection As Range, _  
    Optional FileName As String = "엑셀이미지", _  
    Optional SavePath As String = "", _  
    Optional AddSequence As Boolean = True)
```

```
Dim NewWs As Worksheet  
Dim picRange As Object: Dim MyObj As Chart  
Dim PicH As Double: Dim PicW As Double  
Dim FilePath As String
```

```
If ValidFileName(FileName) = False Then MsgBox "올바른 파일명을 사용하세요.": End
```

```
If SavePath = "" Then SavePath = GetDesktopPath  
FilePath = SavePath & FileName & ".png"
```

```
rngSelection.CopyPicture xlScreen, xlPicture
```

```
Set NewWs = ActiveWorkbook.Sheets.Add  
NewWs.Paste
```

```
Set picRange = NewWs.Shapes.Item(1)  
With picRange  
    PicH = .Height  
    PicW = .Width  
    .Delete  
End With
```

```
With NewWs.Shapes.AddChart2  
    .Height = PicH  
    .Width = PicW  
End With
```

```
Set MyObj = NewWs.Shapes.Item(1).Chart
```

```
MyObj.ChartArea.Select  
MyObj.Paste
```

```
If AddSequence = True Then  
    FilePath = FileSequence(FilePath, 1)  
End If
```

```
MyObj.Export FilePath, "PNG"
```

```
Application.DisplayAlerts = False
```



```
NewWs.Delete
Application.DisplayAlerts = True

End Sub

Public Function FileExists(ByVal path_ As String) As Boolean

    FileExists = (Dir(path_, vbDirectory) <> "")

End Function

Public Function GetDesktopPath(Optional BackSlash As Boolean = True) As String

Dim oWShell As Object

Set oWShell = CreateObject("WScript.Shell")

If BackSlash = True Then
    GetDesktopPath = oWShell.SpecialFolders("Desktop") & "\"
Else
    GetDesktopPath = oWShell.SpecialFolders("Desktop")
End If

Set oWShell = Nothing

End Function

Function FileSequence(FilePath As String, Optional Sequence As Long = 1) As String

Dim Ext As String: Dim Path As String: Dim newPath As String
Dim Pnt As Long

Pnt = InStrRev(FilePath, ".")
Path = Left(FilePath, Pnt - 1)
Ext = Right(FilePath, Len(FilePath) - Pnt + 1)

newPath = Path & Sequence & Ext

Do Until FileExists(newPath) = False
    Sequence = Sequence + 1
    newPath = Path & Sequence & Ext
Loop

FileSequence = newPath

End Function
```