

엑셀 Live 21 강 - 보충자료

엑셀 웹 크롤링의 모든 것 http 요청 웹 스크랩 보조 명령문

2020.11.07



www.oppadu.com

목차

1. 웹 데이터 스크랩 보조 명령문 (GETHTTP 명령문).....	3
2. JSON 데이터 파싱 보조 명령문 (PARSEJSON 명령문).....	5
3. 문자열 잘라내기 (SPLITTER 명령문).....	9
4. 웹 상의 이미지를 시트에 삽입하기 (INSERTWEBIMAGE 명령문).....	10
5. 문자열을 텍스트파일로 추출하기 (EXPORTTEXT 명령문).....	11
6. 배열을 범위로 반환하기 (ARRAYTORNG 명령문)	12
7. 시트의 마지막 행 찾기 (GETLASTROW 명령문).....	13

1. 웹 데이터 스크랩 보조 명령문 (GetHttp 명령문)

http 요청을 사용해 웹에서 데이터를 스크랩합니다.

RequestHeader 는 반드시 짝수(쌍으로 이루어진 배열)로 입력해야 합니다.

예)

Sub Test

Dim htmlResult as Object: Dim strResult as String

Set htmlResult = GetHttp("https://www.naver.com", "Authorization, abc123, content-type, text/html; charset=UTF-8")

strResult = htmlResult.body.innerHTML

End Sub

```
Function GetHttp(URL As String, ParamArray RequestHeader() As Variant) As Object

'#####
'오빠두엑셀 VBA 사용자지정함수 (https://www.oppadu.com)
'▶ GetHttp 함수
'▶ 웹에서 데이터를 받아옵니다.
'▶ 인수 설명
'_____URL           : 데이터를 스크랩할 웹 페이지 주소입니다.
'_____RequestHeader   : RequestHeader 를 배열로 입력합니다. 반드시 짝수(한 쌍씩 이루어진)
개수로 입력되어야 합니다.
'▶ 사용 예제
'Dim HtmlResult As Object
'Set htmlResult = GetHttp("https://www.naver.com")
'Msgbox htmlResult.body.innerHTML
'#####

Dim oHTMLDoc As Object: Dim objHTTP As Object

Dim sUserAgent As String: sUserAgent = "Mozilla/5.0 (Linux; Android 6.0; Nexus 5 Build/MRA58N)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.183 Mobile Safari/537.36"

Application.DisplayAlerts = False

If Left(URL, 4) <> "http" Then URL = "http://" & URL
```

```

Set oHTMLDoc = CreateObject("HtmlFile")
Set objHTTP = CreateObject("MSXML2.ServerXMLHTTP")

objHTTP.setTimeouts 3000, 3000, 3000, 3000
objHTTP.Open "GET", URL, False
objHTTP.setRequestHeader "User-Agent", sUserAgent
If Not IsMissing(RequestHeader) Then
    Dim vRequestHeader As Variant
    For Each vRequestHeader In RequestHeader
        Dim uHeader As Long: Dim Lheader As Long: Dim steps As Long
        uHeader = UBound(vRequestHeader): Lheader = LBound(vRequestHeader)
        If (uHeader - Lheader) Mod 2 = 0 Then GET_HttpRequest = CVar(xlValue): Exit Function
        For i = Lheader To uHeader Step 2
            objHTTP.setRequestHeader vRequestHeader(i), vRequestHeader(i + 1)
        Next
    Next
End If
objHTTP.send

With oHTMLDoc
    .Open
    .Write objHTTP.responsetext
    .Close
End With

Set GetHttp = oHTMLDoc
Set oHTMLDoc = Nothing
Set objHTTP = Nothing

Application.DisplayAlerts = True

End Function

```

2. JSON 데이터 파싱 보조 명령문 (ParseJSON 명령문)

JSON 으로 받아온 데이터 중 원하는 필드만 추출합니다. 상황에 따라 배열의 첫번째 열을 특정 고유값으로 추가하거나, JSON 데이터 중 불필요한 부분만 선택하여 제거할 수 있습니다.

예제 1) ParseJSON("부동산정보", "주소, 이름, 가격")

서울시 강남구 1234	삼성 래미안	3 억 5 천
서울시 강남구 1394	힐스테이트	4 억
서울시 강남구 3920	e 편한세상	7 천
서울시 강남구 2020	자이 아파트	5 억

예제 2) ParseJSON("부동산정보", "주소, 이름, 가격", "강남구", ", ")

강남구	서울시 강남구 1234	삼성 래미안	3 억 5 천
강남구	서울시 강남구 1394	힐스테이트	4 억
강남구	서울시 강남구 3920	e 편한세상	7 천
강남구	서울시 강남구 2020	자이 아파트	5 억

Function ParseJSON(strJSON, strToParse, Optional strID, Optional strToRemove) As Variant

```
#####
'오빠두엑셀 VBA 사용자지정함수 (https://www.oppadu.com)
'▶ ParseJSON 함수
'▶ strJSON 데이터에서 선택한 데이터 값만 추출합니다.
'▶ 인수 설명
'_____strJSON      : JSON 데이터입니다.
'_____strToParse    : JSON 추출할 데이터 필드명입니다. 쉼표(.)로 구분하여 입력합니다.
'_____strID        : 추출한 데이터 배열의 열에 추가할 ID 입니다. (선택인수)
'_____strToRemove   : 추출한 데이터에서 제거할 문자열입니다. 쉼표(.)로 구분하여 입력합니다.
'▶ 사용 예제
'Dim v As Variant
'v = ParseJSON(JsonData, "Date, Name, Item")
#####
```

```

'-----
'변수 설정
'-----

Dim vaToParse As Variant: Dim vToParse As Variant
Dim vaToRemove As Variant: Dim vToRemove As Variant
Dim lngStart As Long
Dim objltn As Variant: Dim strltn As String: Dim tmpltm As String
Dim itmCnt As Long
Dim i As Long: Dim r As Long: Dim c As Long
Dim dictm As Object
Dim valtm As Variant: Dim valtems As Variant: Dim vaReturn As Variant
Dim iCol As Long: Dim maxCol As Long: Dim j As Long

Set dictm = CreateObject("Scripting.Dictionary")

'-----
'JSON 쿼리 분할
'-----

vaToParse = Split(strToParse, ",")
If Not IsMissing(strToRemove) Then vaToRemove = Split(strToRemove, ",")

lngStart = InStr(1, strJSON, "[")
strJSON = Right(strJSON, Len(strJSON) - lngStart)

objltn = Split(strJSON, "{")
itmCnt = UBound(objltn)

For i = 1 To itmCnt
    strltn = Split(objltn(i), "}")(0)
    strltn = Trim(strltn)
    If Left(strltn, 1) = "" Then strltn = Right(strltn, Len(strltn) - 1)
    If Right(strltn, 1) = "" Then strltn = Left(strltn, Len(strltn) - 1)
    iCol = Len(strltn) - Len(Replace(strltn, ":", ""))
    If iCol > maxCol Then maxCol = iCol

    If Not IsMissing(strID) And strID <> "" Then
        ReDim valtm(0 To iCol)
        valtm(0) = strID: j = 1
    Else
        ReDim valtm(0 To iCol - 1)
        j = 0
    End If

    On Error Resume Next

```

```

For Each vToParse In vaToParse
    If InStr(strltn, Trim(vToParse) & "":") > 0 Then
        tmpltm = Split(strltn, Trim(vToParse) & "":")(1)
        tmpltm = Split(tmpltm, ",")(0)
        tmpltm = Trim(tmpltm)
        If Left(tmpltm, 1) = "" Then tmpltm = Right(tmpltm, Len(tmpltm) - 1)
        If Right(tmpltm, 1) = "" Then tmpltm = Left(tmpltm, Len(tmpltm) - 1)
        tmpltm = Replace(tmpltm, "< ", "")
        If Not IsMissing(strToRemove) Then
            For Each vToRemove In vaToRemove
                tmpltm = Replace(tmpltm, Trim(vToRemove), "")
            Next
        End If
        valtm(j) = tmpltm
    ElseIf InStr(strltn, Trim(vToParse) & ".") > 0 Then
        tmpltm = Split(strltn, Trim(vToParse) & ".")(1)
        tmpltm = Split(tmpltm, ",")(0)
        tmpltm = Replace(tmpltm, vbTab, "")
        tmpltm = Trim(tmpltm)
        Do While Right(tmpltm, 1) = vbLf
            tmpltm = Left(tmpltm, Len(tmpltm) - 1)
        Loop
        If Left(tmpltm, 1) = "" Then tmpltm = Right(tmpltm, Len(tmpltm) - 1)
        If Right(tmpltm, 1) = "" Then tmpltm = Left(tmpltm, Len(tmpltm) - 1)
        tmpltm = Replace(tmpltm, "< ", "")
        If Not IsMissing(strToRemove) Then
            For Each vToRemove In vaToRemove
                tmpltm = Replace(tmpltm, Trim(vToRemove), "")
            Next
        End If
        valtm(j) = tmpltm
    End If
    j = j + 1
Next
On Error GoTo 0
dicltn.Add i, Array(valtm, 1)
Next

'-----
'Dictionary -> 배열 변환
'-----

r = dicltn.Count
c = UBound(vaToParse) + 1
If Not IsMissing(strID) And strID <> "" Then c = c + 1

```

```

If r = 0 Then ParseJSON = ""

valtems = dicltn.Items

ReDim vaReturn(1 To r, 1 To c)

On Error Resume Next
For i = 0 To r - 1
    For j = 0 To c - 1
        tmpltm = valtems(i)(0)(j)
        If IsNumeric(tmpltm) Then vaReturn(i + 1, j + 1) = CDBl(tmpltm) Else vaReturn(i + 1, j + 1) = tmpltm
    Next
Next
On Error GoTo 0

'-----
'결과값 리턴
'-----

ParseJSON = vaReturn

End Function

```


3. 문자열 잘라내기 (Splitter 명령문)

문자열에서 불필요한 부분을 잘라내거나 필요한 부분만 선택하여 추출합니다.

```
Function Splitter(v As Variant, Cutter As String, Optional Trimmer As String)

'#####
'오빠두엑셀 VBA 사용자지정함수 (https://www.oppadu.com)
'▶ Splitter 함수
'▶ Cutter ~ Trimmer 사이의 문자를 추출합니다. (Trimmer 가 빈칸일 경우 Cutter 이후 문자열을 추출합니다.)
'▶ 인수 설명
'_____v      : 문자열입니다.
'_____Cutter   : 문자열 절삭을 시작할 텍스트입니다.
'_____Trimmer : 문자열 절삭을 종료할 텍스트입니다. (선택인수)
'▶ 사용 예제
'Dim s As String
's = "{sa;b132@drama#weekend;aabbcc}"
's = Splitter(s, "@", "#")
'MsgBox s '--> "drama"를 반환합니다.
'#####

Dim vaArr As Variant

vaArr = Split(v, Cutter)(1)
If Not IsMissing(Trimmer) Then vaArr = Split(vaArr, Trimmer)(0)

Splitter = vaArr

End Function
```

4. 웹 상의 이미지를 시트에 삽입하기 (InsertWebImage 명령문)

인터넷 상의 이미지를 시트 위로 삽입합니다.

```
Sub InsertWebImage(targetRng As Range, ImgLink As String, Optional imgWidth As Double, Optional  
imgHeight As Double)  
  
'#####  
'오빠두엑셀 VBA 사용자지정함수 (https://www.oppadu.com)  
'▶ InsertWebImage 함수  
'▶ 웹 이미지를 셀 위에 삽입합니다.  
'▶ 인수 설명  
'_____targetRng           : 이미지를 삽입할 셀입니다.  
'_____imgLink             : 삽입할 이미지 링크입니다.  
'_____imgWidth            : 이미지 너비입니다. 기본값은 셀의 너비입니다. (선택인수)  
'_____imgHeight          : 이미지 높이입니다. 기본값은 셀의 높이입니다. (선택인수)  
'▶ 사용 예제  
'InsertWebImage  
sheet1.Range("A1"),"https://www.google.com/images/branding/googlelogo/1x/googlelogo_color_272x92d  
p.png"  
'#####  
  
Dim WS As Worksheet  
Dim shp As Shape  
Dim W As Double: Dim H As Double  
  
Set WS = targetRng.Parent  
If IsMissing(imgWidth) > 0 Then W = imgWidth Else W = targetRng.Width - 6  
If IsMissing(imgHeight) > 0 Then H = imgHeight Else H = targetRng.Height - 6  
  
Set shp = WS.Shapes.AddPicture(ImgLink, msoFalse, msoTrue, targetRng.Left + 3, targetRng.Top + 3, W,  
H)  
  
End Sub
```

5. 문자열을 텍스트파일로 추출하기 (ExportText 명령문)

문자열을 텍스트파일로 추출합니다.

```
Sub ExportText(InnerStrings As String, Optional fileName As String = "텍스트추출", Optional Path As String)

'#####
'오빠두엑셀 VBA 사용자지정함수 (https://www.oppadu.com)
'▶ Export_Text 함수
'▶ 문자열을 텍스트파일로 추출합니다.
'▶ 인수 설명
'_____InnerStrings   : 텍스트파일로 추출할 문자열입니다.
'_____fileName       : 텍스트 파일 이름입니다. 기본값은 "텍스트추출" 입니다. (선택인수)
'_____path           : 텍스트 파일을 생성할 경로입니다. 기본값은 바탕화면입니다. (선택인수)
'▶ 사용 예제
'ExportText "추출할 텍스트"
'#####

Dim TextFile As Integer
Dim FilePath As String

If IsMissing(Path) Then Path = Environ("USERPROFILE") & "\Desktop\"
FilePath = Path & fileName & ".txt"

TextFile = FreeFile

Open FilePath For Output As TextFile
Print #TextFile, InnerStrings
Close TextFile

End Sub
```

6. 배열을 범위로 반환하기 (ArrayToRng 명령문)

배열을 시트 위에 범위로 반환합니다.

```
Sub ArrayToRng(startRng As Range, Arr As Variant)

'#####
'오빠두엑셀 VBA 사용자지정함수 (https://www.oppadu.com)
'▶ ArrayToRng 함수
'▶ 배열을 범위 위로 반환합니다.
'▶ 인수 설명
'_____startRng   : 배열을 반환할 기준 범위(셀) 입니다.
'_____Arr        : 반환할 배열입니다.
'▶ 사용 예제
'Dim v As Variant
'ReDim v(0 to 1)
'v(0) = "a" : v(1) = "b"
'ArrayToRng Sheet1.Range("A1"), v
'#####

On Error GoTo SingleDimension:
startRng.Cells(1, 1).Resize(UBound(Arr, 1) - LBound(Arr, 1) + 1, UBound(Arr, 2) - LBound(Arr, 2) + 1) = Arr

Exit Sub
SingleDimension:
Dim tempArr As Variant: Dim i As Long
ReDim tempArr(LBound(Arr, 1) To UBound(Arr, 1), 1 To 1)
For i = LBound(Arr, 1) To UBound(Arr, 1)
    tempArr(i, 1) = Arr(i)
Next
startRng.Cells(1, 1).Resize(UBound(Arr, 1) - LBound(Arr, 1) + 1, 1) = tempArr

End Sub
```

7. 시트의 마지막 행 찾기 (GetLastRow 명령문)

시트의 마지막 행을 반환합니다.

```
Function GetLastRow(WS As Worksheet) As Long

'#####
'오빠두엑셀 VBA 사용자지정함수 (https://www.oppadu.com)
'▶ GetLastRow 함수
'▶ 시트에서 사용된 마지막 행을 반환합니다.
'▶ 인수 설명
'_____WS : 마지막 행을 조회할 대상 시트입니다.
'#####

With WS
GetLastRow = .UsedRange.Rows.Count + .UsedRange.Row - 1
End With

End Function
```