

엑셀 사용법

【열려있는 엑셀 제어】

```
ex := ComObjActive("Excel.Application") ; 열려있는 엑셀을 ex로 선언
ex.sheets("sheet1").select ; ※ 시트명 선택 ※ 2줄로 열린엑셀 사용가능
ex.range("A1").select ; A1셀 선택하기 = ex.cells(1,1).select
msgbox, % ex.Activecell.value ; 선택셀(A1)의 값 출력
msgbox, % ex.Activecell.Offset(1,2).value ; 선택셀(A1) 기준 1행, 2열에 위치한 셀값(C2) 출력
ex.Activecell.Offset(2,0).select ; 선택셀(A1) 기준 2행 아래 위치한 셀선택(A3)
msgbox, % ex.cells(4,2).value ; 5행,3열의 셀값(B4) 출력
```

【엑셀 생성】

```
ex := ComObjCreate("Excel.Application") ; 엑셀을 ex로 선언하고 열기 (엑셀제어 가능)
ex.visible := True ; 엑셀을 보이게 ↔False ; ex.Workbooks .Add ; 새문서
ex.Workbooks.Add ; 새문서 생성
ex.range("A1").value := 12300 ; A1셀에 12300 입력
ex.cells(2,1).value := "테스트" ; A2셀에 테스트 입력 ※ ex.cells(행,열).value
ex.DisplayAlerts := false ; 경고창 없애기(다른 이름으로 저장시 함께 사용)
ex.ActiveWorkbook.Saveas(A_ScriptDir . "Wabc_test.xlsx") ; 저장(스크립트 폴더)
ex.Quit ; 엑셀종료
```

【기존 엑셀파일 열기 + 제어】

```
ex := ComObjCreate("Excel.Application")
ex.visible := True ; 엑셀 보이게 ↔False ; ex.Workbooks .Add ; 새문서
Path = %A_ScriptDir%WTESTWTEST_EXCEL.xlsm ; or FileSelectFile, path
ex.Workbooks.Open(Path) ; 엑셀 열기
ex.sheets("Sheet1").select ; 시트 선택
msgbox, % ex.range("A2").value ; A2셀값 표시 /ex.range("A4").value:=100 ; 값입력
```

IfWinNotExist, TEST_EXCEL ; 만약 타이틀을 가진 윈도우창이 없다면

```
{
    ex := ComObjCreate("Excel.Application") ; 엑셀을 ex로 선언
    Path = %A_ScriptDir%WTESTWTEST_EXCEL.xlsm ; 파일경로
    ex.visible := True ; 보이게(↔False) ; ex.Workbooks .Add ; 새문서
    ex.Workbooks.Open(Path) ; 파일 열기
}
```

IfWinExist, TEST_EXCEL

```
    ex := ComObjActive("Excel.Application") ; 열려있는 엑셀 객체선언
    ex.sheets("DB").select ; 시트 선택
    Anum:=endRow(ex,"A") ; A열 끝행(작업기준)
    Arr:=ex.Range(First_Col(ex) . First_Row(ex) . ":" Last_Col(ex) . Last_Row(ex)).value ; Arr[행,열]
    MsgBox, % "특정 배열값 : " Arr[Anum,3] " `n`n A열 끝행수 : " Anum
```

【엑셀 제어 테스트】

```

path = %A_ScriptDir%WTESTWTEST_EXCEL.xlsm ; 경로
ex := ComObjCreate("Excel.Application") ; 엑셀객체 생성
ex.visible := True ; 보이게(↔False)
ex.Workbooks.Open(path) ; 경로 파일 열기
ex := ComObjActive("Excel.Application") ; 열린화일 제어+ComObjActive와 쌍
ex.sheets("임시").Select ; 시트 선택 / ; 위까지는 엑셀 기본설정
; ■ 셀값 입력
MyVar := "AHK Excel TEST" ; 변수 선언 및 변수값 할당 (AHK Excel TEST)
ex.Range("A1").value := MyVar ; A1셀에 변수값 입력 ( " ")
ex.Range("A2").value := 100 ; A2셀에 숫자 100 입력
ex.Range("A2").Offset(1,0).value := 200 ; A3셀에 숫자 200 입력 ※ A2셀 기준 Offset(1행,0열)
ex.Range("A2").Offset(2,0).value := 300 ; A4셀에 숫자 300 입력 ※ A2셀 기준 Offset(2행,0열)
ex.Range("A5").Formula := "=SUM(A2:A4)" ; 수식 입력
ex.Range("C1:C4").value := "abc" ; 문자 입력(양쪽 따옴표)
Sleep, 2000 ; 2초 대기
; ■ 셀값 일괄변경 : While
while (ex.Range("C" . A_index).value != ""){
    ex.Range("C" . A_index).value := "New Value0" . A_index
}
msgbox, % "A1셀값 : " ex.cells(1,1).value
ex.ActiveWorkbook.Save() ; ■ 저장(동일 이름)
;; ■ 저장(다른이름)
;ex.DisplayAlerts := false
;ex.ActiveWorkbook.SaveAs(%A_ScriptDir%Wabc.xlsx)
; ■ 엑셀 종료
MsgBox,,, 셀값 입력 완료 `n`n5초 후 자동 종료, 5 ; `n은 줄바꿈
ex.ActiveWorkBook.Close
ex.Quit

```

※ 참고 함수

```

검색이동("문자") ; = find_text_move("문자") = find_text_enterN("문자")
input_value("문자") ; 셀값 입력(= clip_copy), 문자는 쌍따옴표
endRow(ex, "A") ; A열 마지막 행수 ※ 작업할 기준 + loop 등 사용
First_Row(ex) ; 연속된 범위의 첫번째 자료 행수(Row)
Last_Row(ex) ; " " 마지막 자료 행수(Row)
Last_Col(ex) ; " " " 열수(Column)

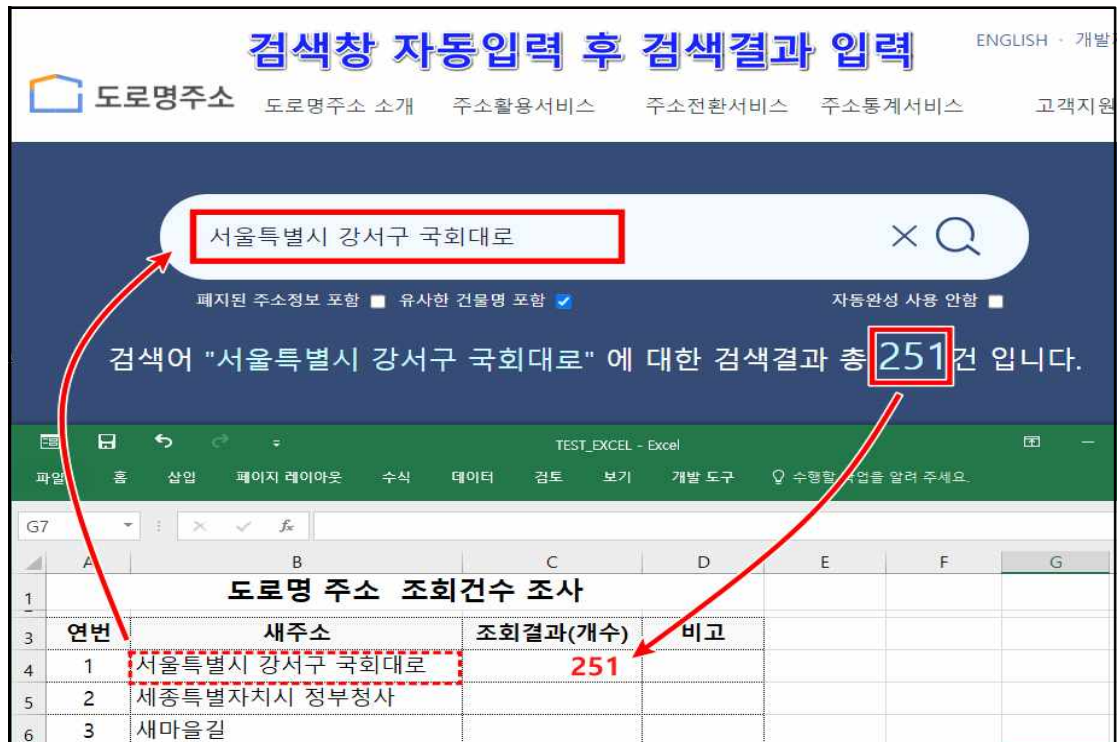
```

```

e_Row := ex.sheets("DB").UsedRange.Rows.Count ; 행수
e_Col := ex.sheets("DB").UsedRange.columns.Count ; 열 갯수
Arr := ex.sheets("DB").Range["A1:I" . endRow(ex,"A")].value ; Arr[행, 열] 배열
Msgbox, % Arr[1,2] ; Arr[행,열] 배열변수의 1행, 2열 값 출력
input_value(ex.Range("A" endRow(ex,"A")).text) ; A열 끝행의 값(text)
input_value(Arr[1,2]) ; 다른 곳에 값을 입력할 경우 활용(함수)
won := FormatNumber(Arr[2,3]) ; 금액 천원 표시(,)
Send, 금%won%원 ; 변수dd는 %로 감싸서 사용

```

※ 활용사례 1 : Excel, 클립보드 활용



^space:: ; ■ **Ctrl+Space** ■ 도로명주소 검색창에 활성화된 셀값 입력 후 검색

ex := ComObjActive("Excel.Application") ; 열려있는 엑셀을 ex로 선언

ex.sheets("TEST").select ; 작업할 시트 선택

WinActivate, 도로명주소 안내시스템 ; Title창 활성화(가장 앞으로 나오게)

Click, 300, 338, 3 ; 검색창에 3클릭 - 기존자료 덮어쓰기 위해 3클릭

Send, % ex.Activecell.value ; 선택 셀값 (B4) 입력

Send, {Enter} ; 엔터 (또는 검색버튼 클릭)

Sleep, 1000 ; 1초 대기

ex.Activecell.offset(1,0).select ; 선택셀(B4) 기준 행1칸 위치 셀선택(B5)

Click, 627, 461 ; 결과 나올 예상위치 클릭(2클릭 범위잡기 쉽게)

return

;

RButton:: ; ■ 범위선택(2Click) 후 **마우스 우버튼** 선택셀의 기준 행-1, 열1 위치 값입력

Send, ^c ; Ctrl+c (복사하기) ← 블록잡은 부분을 복사

ClipWait, 2 ; 클립보드 저장될때까지 대기

ex.Activecell.offset(-1,1).value := Clipboard ; 선택셀(B5) 기준 행-1,열1 위치 복사값 넣기

return

※ 활용사례 2 : Excel, 클립보드, Loop(반복), IF(조건), 정규식 활용

^F8:: ; Ctrl+F8 누르면 → Return 만날때까지 순차적으로 실행됨

ex := ComObjActive("Excel.Application") ; 열려있는 엑셀을 ex로 선언

ex.sheets("TEST").select ; 시트 선택

WinActivate, 도로명주소 안내시스템 ; Title 창 활성화(앞으로 나오게)

Sleep, 200

Click, 300, 338, 3 ; 검색창에 3클릭 - 기존자료 덮어쓰기 위해(좌표 확인★)

Send, % **ex.Activecell.value** ; 선택셀값(활성화된 셀값) 입력

Send, {Enter} ; 엔터키 (또는 검색 아이콘 클릭)

Sleep, 1000 ; 1초 대기(검색결과 나올때까지)

ex.Activecell.offset(1,0).select ; 아래셀 선택(선택셀(B4) 기준 1행 위치 셀선택(B5))

Click, 627, 461, 3 ; 결과가 나올 위치로 3클릭 범위복사(좌표 확인★)

; 클립복사 및 숫자만 추출하기

Clipboard = ; 클립보드 비우기

Send, ^c ; Ctrl+c (복사)

ClipWait, 2 ; 클립보드 저장될 때까지 대기

var := RegExReplace(clipboard, "WD") ; ■ 클립보드에서 숫자만 추출(정규식)

ex.Activecell.offset(-1,1).value := var ; 선택셀(B5) 기준 행-1, 열1 위치 셀값 넣기

return

; ※ 자동화 : Loop 반복문, if(ex.Range("A" . A_Index).value = "") 조건문 활용

; [참고]

loop

{ ; ■ 반복문 / 반복횟수 지정방법 : Loop, 3 { ~ }

if (A_index > 3) ; A_index 는 반복횟수 (1부터 1씩 증가 1,2,3..)

break ; Loop 탈출 / if 조건이 맞다면 Break로 반복문 탈출

MsgBox,,, %A_index%번째 생일 축하해,1 ; MsgBox 내 변수 : %변수%

}

; ■ 정규식: 문자열에서 숫자만 가져오기 / ★구글 : [ahk 정규식 숫자만](#)

price=금23,500원(금이만삼천오백원) ; 문자, 숫자 등 혼합된 문자열

var:=RegExReplace(price, "WD") ; WD는 숫자아닌 문자열임, 즉 숫자만 남겨라

MsgBox, 변수(var) 값 = %var%