PYTHON EXCEL 이해하기

Moon Yong Joon

WINDOWS APPLICATION DISPATCH 처리

Moon Yong Joon

ie구동하기

Dispatch를 이용한 appl 호출

windows 내에서 application 호출 방법

```
import win32com.client, time
ie = win32com.client.Dispatch("InternetExplorer.Application")
ie.Visible = 1
ie.Navigate('http://www.naver.com')
time.sleep(3) #wait 3 sec.
print('You are surfing on', ie.Document.domain )
print('And now a list of the Links:' )
for i in ie.Document.links:
    print (i)
```

```
You are surfing on naver.com
And now a list of the Links:
http://www.naver.com/#news_cast2
http://www.naver.com/#themecast
http://www.naver.com/#time_square
http://www.naver.com/#cnsv_shbx
```

Python interpreter구동하기

Dispatch를 이용한 appl 호출

windows 내에서 python interpreter application 호출 방법

```
import sys
from win32com.client import Dispatch

d = Dispatch("Python.Interpreter")
print("2 + 5 =", d.Eval("2 + 5"))
d.Exec("print 'hi via COM'")

d.Exec("import sys")

print("COM server sys.version", d.Eval("sys.version"))
print("Client sys.version", sys.version)

print("COM server sys.path", d.Eval("sys.path"))
print("Client sys.path", sys.path)
```

python 실행 결과

windows 내에서 python interpreter application 실행 결과

```
2 + 5 = 7
COM server sys.version 2.7.10 (default, May 23 2015, 09:40:32) [MSC v.1500 32 bit (Intel)]
Client sys.version 3.5.2 |Anaconda 4.2.0 (64-bit)| (default, Jul  5 2016, 11:41:13) [MSC v.1
COM server sys.path ('C:\\Python27\\lib\\site-packages\\win32com\\server', 'C:\\Windows\\sys
\\DLLs', 'C:\\Python27\\lib', 'C:\\Python27\\lib\\plat-win', 'C:\\Python27\\lib\\lib-tk', 'C
te-packages', 'C:\\Python27\\lib\\site-packages\\win32', 'C:\\Python27\\lib\\site-packages\\
e-packages\\Pythonwin', 'C:\\Python27\\lib\\site-packages\\wx-2.8-msw-unicode')
Client sys.path ['', 'C:\\Program Files\\Anaconda3\\python35.zip', 'C:\\Program Files\\Anacc
conda3\\lib', 'C:\\Program Files\\Anaconda3', 'C:\\Program Files\\Anaconda3\\lib\\site-packa
\\lib\\site-packages\\Sphinx-1.4.6-py3.5.egg', 'C:\\Program Files\\Anaconda3\\lib\\site-pack
aconda3\\lib\\site-packages\\win32\\lib', 'C:\\Program Files\\Anaconda3\\lib\\site-packages\
conda3\\lib\\site-packages\\setuptools-27.2.0-py3.5.egg', 'C:\\Program Files\\Anaconda3\\lib
s', 'C:\\Users\\06411\\.ipython']
```

EXCEL 처리 – DISPATCH

Moon Yong Joon

excel 구동하기

excel 구동하기

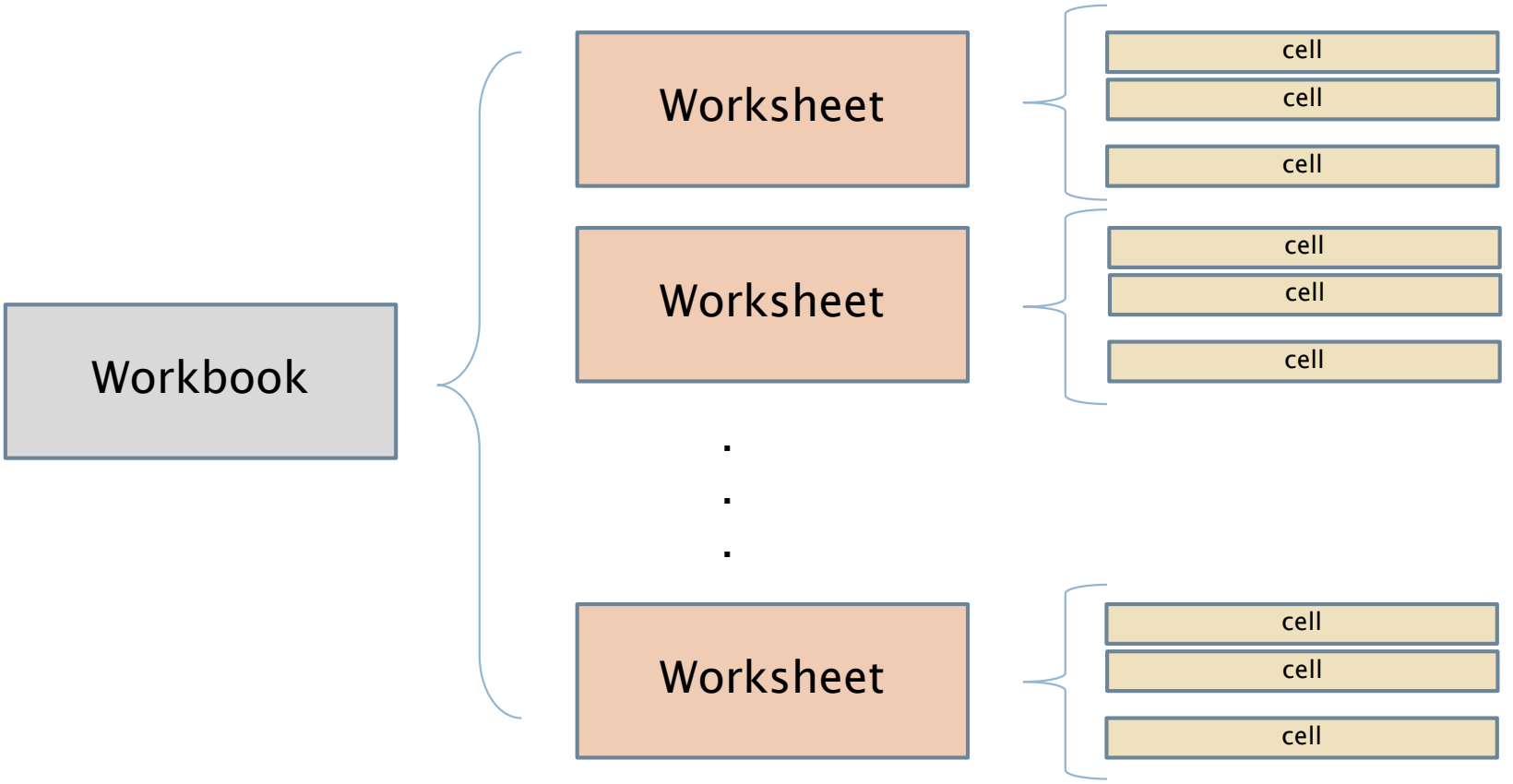
파이썬 모듈에서 excel을 구동하기

```
import win32com.client

excel = win32com.client.Dispatch("excel.Application")
excel.Visible = True
```

excel new

Excel 구조

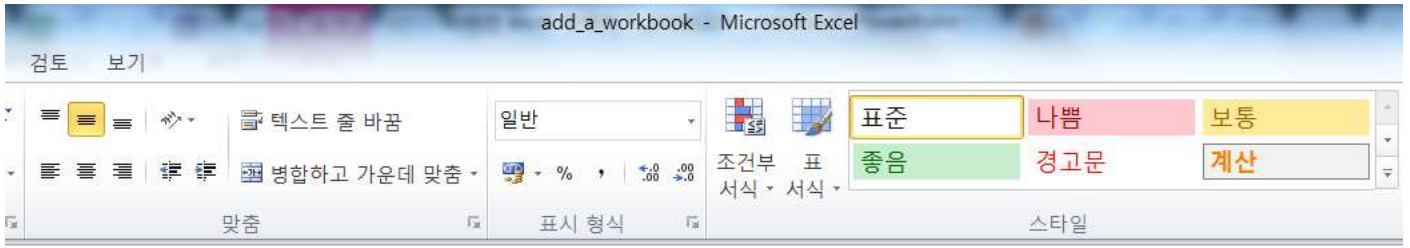
Workbook-> Worksheet -> cell 단위로 구성



workbook/worksheet 생성

excel을 구동하고 excel 파일인 workbook 만들고 그 내의 worksheet 생성해서 저장

```
import win32com.client

excel = win32com.client.Dispatch("Excel.Application")
excel.Visible = True

wb = excel.Workbooks.Add()
ws = wb.Worksheets("Sheet1")
wb.SaveAs('add_a_workbook.xlsx')
excel.Quit()
```

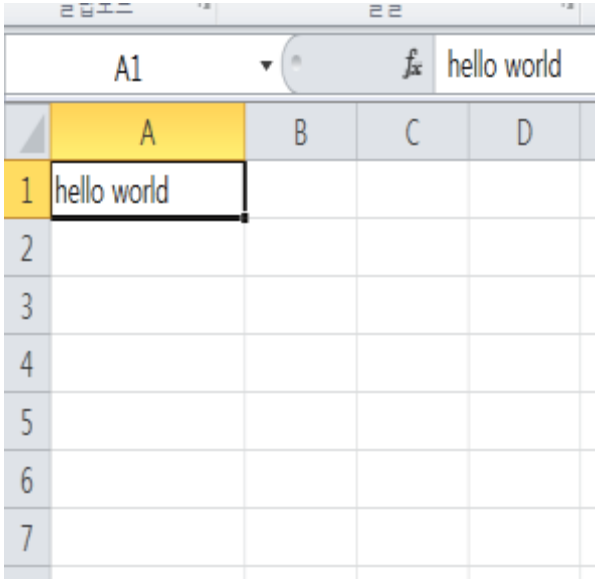


excel :workbook 처리

존재한 workbook 열기

workbooks.open(파일명)으로 기존 excel file 을 열기

```
#
# Open an existing workbook
#
import win32com.client as win32
excel = win32.gencache.EnsureDispatch('Excel.Application')
wb = excel.Workbooks.Open('t1est.xlsx')

excel.Visible = True
```

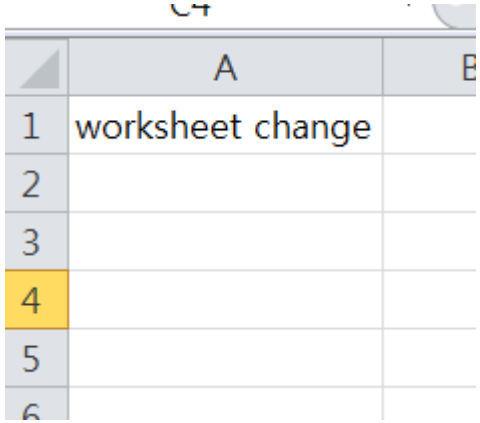


excel : worksheet처리

존재한 worksheet 열기

workbooks.open(파일명)으로 기존 excel file을 열고 worksheets(sheet명)으로 지정한 sheet로 열기

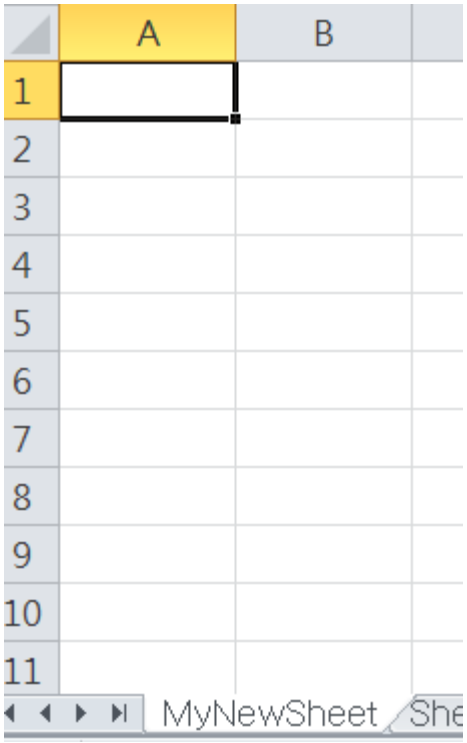
```
#
# Open an existing workbook/worksheet
#
import win32com.client as win32
excel = win32.gencache.EnsureDispatch('Excel.Application')
wb = excel.Workbooks.Open('t1est.xlsx')
ws = wb.Worksheets("change")
excel.Visible = True
```



기존 파일에 워크시트추가

기존 존재한 excel file을 기존 파일에 워크시트 추가

```
import win32com.client

excel = win32com.client.Dispatch("Excel.Application")
excel.Visible = True

wb = excel.Workbooks.Open('test1.xlsx')
ws = wb.Worksheets.Add()
ws.Name = "MyNewSheet"

wb.SaveAs('test1.xlsx')
excel.Quit()
```

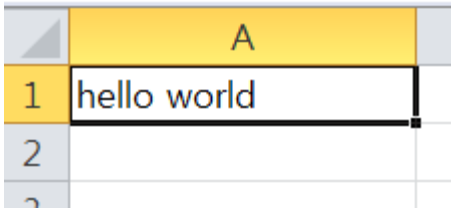


excel : cell 처리

excel : cell 갱신

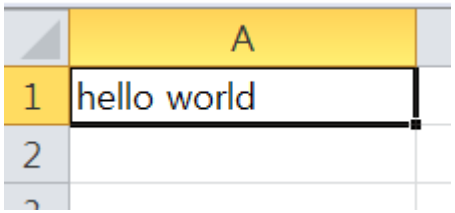
파이썬 모듈에서 excel을 구동하고 worksheets에 첫번째 cell에 값을 넣기

```
import win32com.client

excel = win32com.client.Dispatch("Excel.Application")
excel.Visible = True
wb = excel.Workbooks.Add()
ws = wb.Worksheets("Sheet1")
ws.Cells(1, 1).Value = "hello world"
wb.SaveAs('test.xlsx')
excel.Quit()
```



excel : cell 읽기

파이썬 모듈에서 excel을 구동하고 worksheets에 첫번째 cell의 값을 가져오기

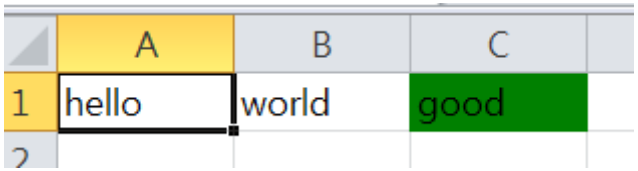



```
import win32com.client

excel = win32com.client.Dispatch("Excel.Application")
excel.Visible = True
wb = excel.Workbooks.Open('test.xlsx')
ws = wb.ActiveSheet
print(ws.Cells(1,1).Value)
excel.Quit()
```

```
hello world
```

excel : 여러 cell 처리

cell과 range를 기준으로 데이터를 넣고 내부에 색깔 입히기




```
import win32com.client

excel = win32com.client.Dispatch("Excel.Application")
excel.Visible = True
wb = excel.Workbooks.Add()
ws = wb.Worksheets("Sheet1")
ws.Cells(1, 1).Value = "hello"
ws.Cells(1,2).Value = "world"
ws.Range("C1").Value = "good"
ws.Range("C1").Interior.ColorIndex = 10

wb.SaveAs('test1.xlsx')
excel.Quit()
```

excel : cell 폭 조정

Cell에 대한 폭 조정, 특히 Range로 처리시 범주가 하나라도 “B:B”로 정의해야 사이즈 변경

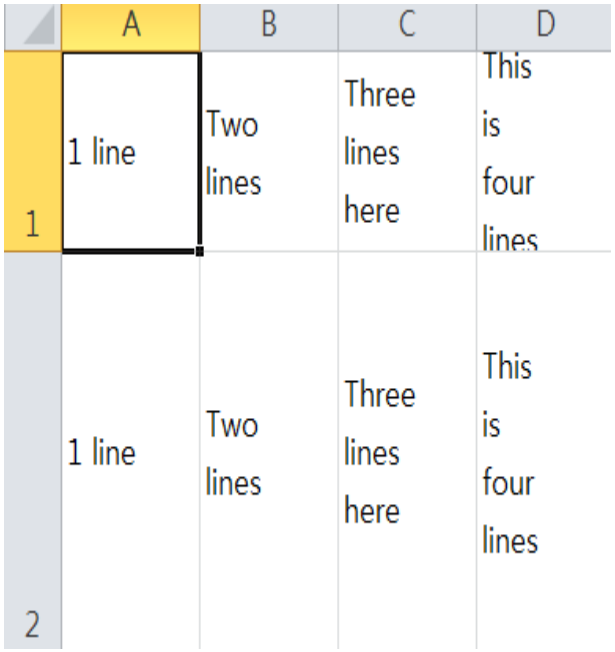
```
import win32com.client

excel = win32com.client.Dispatch("excel.Application")
excel.Visible = True
wb = excel.Workbooks.Add()
ws = wb.Worksheets("Sheet1")

ws.Range("A1:A10").Value = "A"
ws.Range("B1:B10").Value = "This is a very long line of text"
ws.Columns(1).ColumnWidth = 1
ws.Range("B:B").ColumnWidth = 27

wb.SaveAs('test17.xlsx')
excel.Quit()
```

	A	B
1	A	This is a very long line of text
2	A	This is a very long line of text
3	A	This is a very long line of text
4	A	This is a very long line of text
5	A	This is a very long line of text
6	A	This is a very long line of text
7	A	This is a very long line of text
8	A	This is a very long line of text
9	A	This is a very long line of text
10	A	This is a very long line of text

excel : cell 높이 조정

Cell에 대한 폭 조정, 특히 Range로 처리시 범주가 하나라도 “2:2”로 정의해야 사이즈 변경

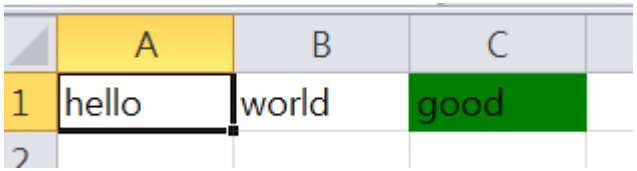
```
import win32com.client

excel = win32com.client.Dispatch("excel.Application")
excel.Visible = True
wb = excel.Workbooks.Add()
ws = wb.Worksheets("Sheet1")

ws.Range("A1:A2").Value = "1 line"
ws.Range("B1:B2").Value = "Two\nlines"
ws.Range("C1:C2").Value = "Three\nlines\nhere"
ws.Range("D1:D2").Value = "This\nis\nfour\nlines"
ws.Rows(1).RowHeight = 60
ws.Range("2:2").RowHeight = 120


wb.SaveAs('test20.xlsx')
excel.Quit()
```



excel :cell color

excel :color 정보

color 정보는 아래의 사이트 확인



http://dmcritchie.mvps.org/excel/colors.htm

Color Palette and the 56 Excel ColorIndex Colors - David McRitchie

dmcritchie.mvps.org/**excel**/colors.htm ▾ 이 페이지 번역하기

2011. 4. 5. - **Excel Color Index**, coloring of fonts, cell interiors. ... Each Microsoft **Excel** workbook has a palette of 56 colors that you can **apply** to cells, fonts, ...

excel :color 大리

color에 대한 index를 부여해서 각 cell에 색상을 넣음

```
import win32com.client

excel = win32com.client.Dispatch("excel.Application")
excel.Visible = True
wb = excel.Workbooks.Add()
ws = wb.Worksheets("Sheet1")
for i in range (1,21):
    ws.Cells(i,1).Value = i
    ws.Cells(i,1).Interior.ColorIndex = i

wb.SaveAs('test15.xlsx')
excel.Quit()
```

excel : cell offset

excel : offset

Cell를 기준으로 offset을 주고 값을 처리

```
import win32com.client

excel = win32com.client.Dispatch("excel.Application")
excel.Visible = True
wb = excel.Workbooks.Add()
ws = wb.Worksheets("Sheet1")

ws.Cells(1,1).Value = "Cell A1"
ws.Cells(1,1).Offset(2,4).Value = "Cell D2"

wb.SaveAs('test11.xlsx')
excel.Quit()
```

	A	B	C	D
1	Cell A1			
2				Cell D2
3				

excel : Range

excel : Range 처리

Range는 하나의 cell가 범위에 따른 값을 할당 가능

```
import win32com.client

excel = win32com.client.Dispatch("excel.Application")
excel.Visible = True
wb = excel.Workbooks.Add()
ws = wb.Worksheets("Sheet1")

ws.Cells(1,1).Value = "Cell A1"
ws.Cells(1,1).Offset(2,4).Value = "Cell D2"
ws.Range("A2").Value = "Cell A2"
ws.Range("A3:B4").Value = "A3:B4"
ws.Range("A6:B7,A9:B10").Value = "A6:B7,A9:B10"

wb.SaveAs('test10.xlsx')
excel.Quit()
```

	A	B	C	D
1	Cell A1			
2	Cell A2			Cell D2
3	A3:B4	A3:B4		
4	A3:B4	A3:B4		
5				
6	A6:B7,A9:B10	A6:B7,A9:B10		
7	A6:B7,A9:B10	A6:B7,A9:B10		
8				
9	A6:B7,A9:B10	A6:B7,A9:B10		
10	A6:B7,A9:B10	A6:B7,A9:B10		

excel : Range autofill 동일값

Range를 주고 그 범주에 AutoFill처리로 결과값을 추가할 경우 처리

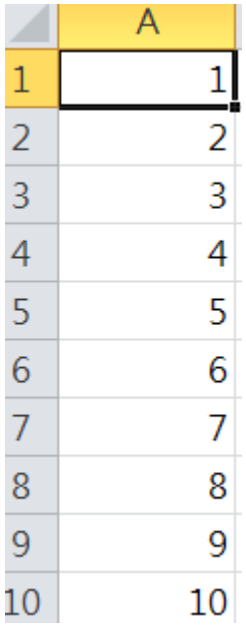
```
import win32com.client

excel = win32com.client.Dispatch("excel.Application")
excel.Visible = True
wb = excel.Workbooks.Add()
ws = wb.Worksheets("Sheet1")

ws.Range("A1").Value = 1
#ws.Range("A2").Value = 2
ws.Range("A1").AutoFill(ws.Range("A1:A10"))

wb.SaveAs('test14.xlsx')
excel.Quit()
```

	A
1	1
2	1
3	1
4	1
5	1
6	1
7	1
8	1
9	1
10	1

excel : Range autofill 증가값

Range를 주고 그 범주에 AutoFill처리로 결과값을 추가할 경우 처리

```
import win32com.client

excel = win32com.client.Dispatch("excel.Application")
excel.Visible = True
wb = excel.Workbooks.Add()
ws = wb.Worksheets("Sheet1")

ws.Range("A1").Value = 1
ws.Range("A2").Value = 2
ws.Range("A1:A2").AutoFill(ws.Range("A1:A10"))

wb.SaveAs('test13.xlsx')
excel.Quit()
```



	A
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	10

EXCEL 처리 – ENSUREDISPATCH

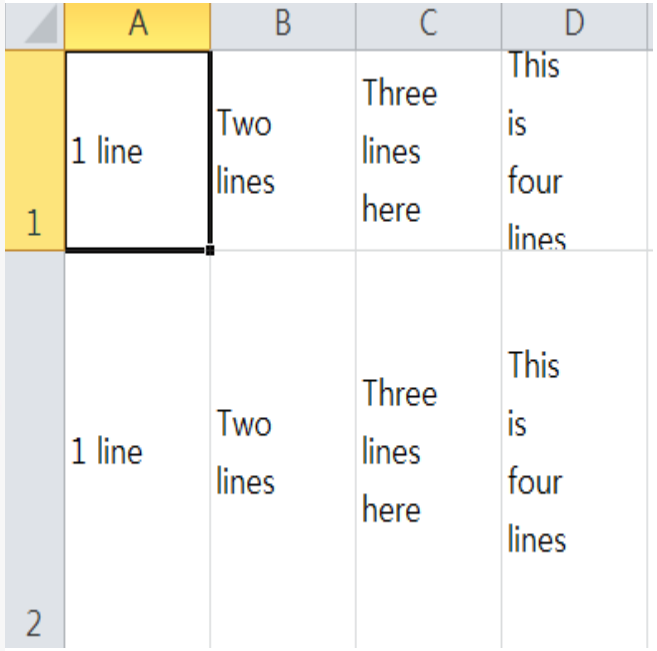
Moon Yong Joon

excel :cell내의 위치조정

excel : cell 위치 조정

Cell 내부의 위치조정은 반드시 EnsureDispatch 로 excel를 구동해야 하고 xlCenter를 할당

```
import win32com.client as win32
excel = win32.gencache.EnsureDispatch('Excel.Application')
wb = excel.Workbooks.Add()
ws = wb.Worksheets("Sheet1")
ws.Range("A1:A2").Value = "1 line"
ws.Range("B1:B2").Value = "Two\nlines"
ws.Range("C1:C2").Value = "Three\nlines\nhere"
ws.Range("D1:D2").Value = "This\nis\nfour\nlines"
ws.Rows(1).RowHeight = 60
ws.Range("2:2").RowHeight = 120
ws.Rows(1).VerticalAlignment = win32.constants.xlCenter
ws.Range("2:2").VerticalAlignment = win32.constants.xlCenter

# Alternately, you can autofit all rows in the worksheet
# ws.Rows.AutoFit()

wb.SaveAs('row_height.xlsx')
excel.Application.Quit()
```



OPENPYXL
모듈 처리

Moon Yong Joon

Excel 정보 가져오기

excel 파일 가져오기

test.xlsx 파일을 workbook으로 load해서 worksheet 읽어오기

```
import openpyxl
wb = openpyxl.load_workbook('test.xlsx')
print(wb.get_sheet_names())
```

```
['Sheet1', 'Sheet2', 'Sheet3']
```

	A	B	C
1	Title 1	Title 2	Title 3
2	1	a	08/18/07
3	2	b	08/19/07
4	3	c	08/20/07
5	4	d	08/21/07
6	5	e	08/22/07
7	6	f	08/23/07
8	7	g	08/24/07
9	8	h	08/25/07
10	9	i	08/26/07

Excel 파일 행과 열처리

worksheet 행,열 정보 가져오기

test.xlsx 파일을 workbook으로 load한 첫번째 worksheet 읽어와서 행,열 정보보기

```
import openpyxl
wb = openpyxl.load_workbook('test.xlsx')
ws = wb.get_sheet_by_name('Sheet1')
print(ws.max_column)
print(ws.max_row)
```

```
3
10
```

	A	B	C
1	Title 1	Title 2	Title 3
2	1	a	08/18/07
3	2	b	08/19/07
4	3	c	08/20/07
5	4	d	08/21/07
6	5	e	08/22/07
7	6	f	08/23/07
8	7	g	08/24/07
9	8	h	08/25/07
10	9	i	08/26/07

worksheet: 행 처리

worksheet 내의 행 처리 방법

```
import openpyxl
wb = openpyxl.load_workbook('test.xlsx')
ws = wb.get_sheet_by_name('Sheet1')
print(ws.rows)
for row in ws.rows :
    print(row)
```

```
<generator object get_squared_range at 0x0000000067B89E8>
(<Cell Sheet1.A1>, <Cell Sheet1.B1>, <Cell Sheet1.C1>)
(<Cell Sheet1.A2>, <Cell Sheet1.B2>, <Cell Sheet1.C2>)
(<Cell Sheet1.A3>, <Cell Sheet1.B3>, <Cell Sheet1.C3>)
(<Cell Sheet1.A4>, <Cell Sheet1.B4>, <Cell Sheet1.C4>)
(<Cell Sheet1.A5>, <Cell Sheet1.B5>, <Cell Sheet1.C5>)
(<Cell Sheet1.A6>, <Cell Sheet1.B6>, <Cell Sheet1.C6>)
(<Cell Sheet1.A7>, <Cell Sheet1.B7>, <Cell Sheet1.C7>)
(<Cell Sheet1.A8>, <Cell Sheet1.B8>, <Cell Sheet1.C8>)
(<Cell Sheet1.A9>, <Cell Sheet1.B9>, <Cell Sheet1.C9>)
(<Cell Sheet1.A10>, <Cell Sheet1.B10>, <Cell Sheet1.C10>)
```

	A	B	C
1	Title 1	Title 2	Title 3
2	1	a	08/18/07
3	2	b	08/19/07
4	3	c	08/20/07
5	4	d	08/21/07
6	5	e	08/22/07
7	6	f	08/23/07
8	7	g	08/24/07
9	8	h	08/25/07
10	9	i	08/26/07

worksheet: 열 처리

worksheet 내의 열 처리 방법

```
import openpyxl
wb = openpyxl.load_workbook('test.xlsx')
ws = wb.get_sheet_by_name('Sheet1')
print(ws.columns)
for column in ws.columns :
    print(column)
```

```
<generator object Worksheet._cells_by_col at 0x000000000
(<Cell Sheet1.A1>, <Cell Sheet1.A2>, <Cell Sheet1.A3>, <
Cell Sheet1.A8>, <Cell Sheet1.A9>, <Cell Sheet1.A10>)
(<Cell Sheet1.B1>, <Cell Sheet1.B2>, <Cell Sheet1.B3>, <
Cell Sheet1.B8>, <Cell Sheet1.B9>, <Cell Sheet1.B10>)
(<Cell Sheet1.C1>, <Cell Sheet1.C2>, <Cell Sheet1.C3>, <
Cell Sheet1.C8>, <Cell Sheet1.C9>, <Cell Sheet1.C10>)
```

	A	B	C
1	Title 1	Title 2	Title 3
2	1	a	08/18/07
3	2	b	08/19/07
4	3	c	08/20/07
5	4	d	08/21/07
6	5	e	08/22/07
7	6	f	08/23/07
8	7	g	08/24/07
9	8	h	08/25/07
10	9	i	08/26/07

Excel 파일 cell 정보

worksheet 내 cell 정보 조회

cell 정보를 가져와서 값 출력하기

```
import openpyxl
wb = openpyxl.load_workbook('test.xlsx')
ws = wb.get_sheet_by_name('Sheet1')
print(ws['A4'].value)
a2 = ws.cell(row=2,column=1)
print(a2.value)

a3 = ws.cell('A3')
print(a3.value)
```

```
3
1
2
```

```
C:\Program Files\Anaconda3\lib\site-packages\openpyxl\
s deprecated. Use ws[coordinate] instead
  warn("Using a coordinate with ws.cell is deprecated.
```

	A	B	C
1	Title 1	Title 2	Title 3
2	1	a	08/18/07
3	2	b	08/19/07
4	3	c	08/20/07
5	4	d	08/21/07
6	5	e	08/22/07
7	6	f	08/23/07
8	7	g	08/24/07
9	8	h	08/25/07
10	9	i	08/26/07

Excel 파일 생성하기

excel 내의 worksheet 생성

빈 workbook를 만들고 worksheet를 create_sheet로 만듦

```
import openpyxl as xl
wb = xl.Workbook()

print(wb)
# grab the active worksheet
ws = wb.create_sheet("Mysheet")
print(ws)
```

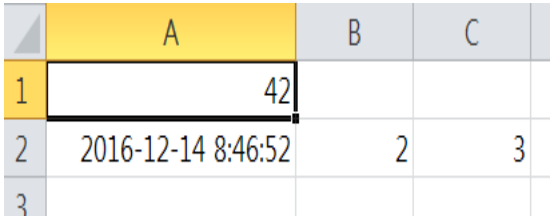
```
<openpyxl.workbook.workbook.Workbook object at 0x00000000067BB780>
<Worksheet "Mysheet">
```

excel 파일 생성

workbook/worksheet를 만들어 excel 파일 저장하기

```
import openpyxl as xl
wb = xl.Workbook()

print(wb)
# grab the active worksheet
ws = wb.create_sheet("Mysheet")
print(ws)

# Data can be assigned directly to cells
ws['A1'] = 42

# Rows can also be appended
ws.append([1, 2, 3])

# Python types will automatically be converted
import datetime
ws['A2'] = datetime.datetime.now()

# Save the file
wb.save(filename="sample2.xlsx")
```

```
<openpyxl.workbook.workbook.Workbook object at 0x000000000679AD30>
<Worksheet "Mysheet">
```



	A	B	C
1	42		
2	2016-12-14 8:46:52	2	3
3			

CSV
모듈 처리

Moon Yong Joon

csv: dict 타입 처리

csv : DictWriter

dict 파일을 입력받아 csv 파일 만들기

```
import csv
with open('dic_file.csv', 'w') as csvfile:
    fieldnames = ['first_name', 'last_name']
    writer =  csv.DictWriter(csvfile, fieldnames=fieldnames)
    writer.writeheader()
    writer.writerow({'first_name' : 'banana', 'last_name' : 'ssang'})
    writer.writerow({'first_name' : 'kong', 'last_name' : 'al'})
    writer.writerow({'first_name' : 'kong', 'last_name' : 'dal'})
```

	A	B
1	first_name	last_name
2		
3	banana	ssang
4		
5	kong	al
6		
7	kong	dal

csv : DictWriter(newline 조정)

dict 파일을 입력받아 csv 파일 만들때 newline='\r\n'이 기본이므로 '\n' 변경해야 라인이 한줄 줄어듦

```
import csv
with open('dic_file.csv','w',newline="\n") as csvfile :
    fieldnames = ['firstname', 'last_name']
    writer = csv.DictWriter(csvfile,fieldnames= fieldnames)
    writer.writeheader()
    writer.writerow({'firstname': 'banana', 'last_name':"song"})
    writer.writerow({'firstname': 'apple', 'last_name':"song"})
    writer.writerow({'firstname': 'cherry', 'last_name':"song"})
```

	A	B
1	firstname	last_name
2	banana	song
3	apple	song
4	cherry	song
5		

csv : DictReader

csv 파일을 읽어 dict 타입으로 출력하기

```
import csv
with open('dic_file.csv', 'r') as csvfile:
    reader = csv.DictReader(csvfile)
    for row in reader:
        print(row)
```

```
{'last_name': 'ssang', 'first_name': 'banana'}
{'last_name': 'al', 'first_name': 'kong'}
{'last_name': 'dal', 'first_name': 'kong'}
```

csv: list 타입 처리

csv : writer

list 파일을 입력받아 csv 파일 만들기

```
import csv

with open('persons1.csv', 'w', newline='') as csvfile:
    filewriter = csv.writer(csvfile, delimiter=',',
                            quotechar='|', quoting=csv.QUOTE_MINIMAL)
    filewriter.writerow(['Name', 'Profession'])
    filewriter.writerow(['Derek', 'Software Developer'])
    filewriter.writerow(['Steve', 'Software Developer'])
    filewriter.writerow(['Paul', 'Manager'])
```

	A	B
1	Name	Profession
2	Derek	Software Developer
3	Steve	Software Developer
4	Paul	Manager

csv : reader

csv 파일을 읽어 list 타입으로 출력하기

```
import csv
with open('persons1.csv', 'r') as f:
    reader = csv.reader(f)

    # read file row by row
    for row in reader:
        print(row)
```

```
['Name', 'Profession']
['Derek', 'Software Developer']
['Steve', 'Software Developer']
['Paul', 'Manager']
```

csv 파일 생성 후 읽기

csv : jupyter에서 생성

파일을 입력받아 csv 파일 만들기

```
%%writefile test.csv
"Title 1","Title 2","Title 3"
1,"a",08/18/07
2,"b",08/19/07
3,"c",08/20/07
4,"d",08/21/07
5,"e",08/22/07
6,"f",08/23/07
7,"g",08/24/07
8,"h",08/25/07
9,"i",08/26/07
```

```
Writing test.csv
```

	A	B	C
1	Title 1	Title 2	Title 3
2	1	a	08/18/07
3	2	b	08/19/07
4	3	c	08/20/07
5	4	d	08/21/07
6	5	e	08/22/07
7	6	f	08/23/07
8	7	g	08/24/07
9	8	h	08/25/07
10	9	i	08/26/07

csv : reader

csv 파일을 읽어 list 타입으로 출력하기

```
import csv

f = open('test.csv', 'r')
try:
    reader = csv.reader(f)
    for row in reader:
        print(row)
finally:
    f.close()
```

```
['Title 1', 'Title 2', 'Title 3']
['1', 'a', '08/18/07']
['2', 'b', '08/19/07']
['3', 'c', '08/20/07']
['4', 'd', '08/21/07']
['5', 'e', '08/22/07']
['6', 'f', '08/23/07']
['7', 'g', '08/24/07']
['8', 'h', '08/25/07']
['9', 'i', '08/26/07']
```

csv: dialect 등록 처리

csv :register_dialect

파일에 대한 dialect 정보를 등록해서 사용

```
help(csv.register_dialect)
```

```
Help on built-in function register_dialect in module _csv:

register_dialect(...)
    Create a mapping from a string name to a dialect class.
    dialect = csv.register_dialect(name[, dialect[, **fmtparams]])
```

```
import csv
csv.register_dialect(
    'mydialect',
    delimiter = ',',
    quotechar = '"',
    doublequote = True,
    skipinitialspace = True,
    lineterminator = '\r\n',
    quoting = csv.QUOTE_MINIMAL)
```

csv : 파일 생성

csv 파일 생성

```
%%writefile test2.csv
first_name ,     last_name,              city
Aleshia,         Tomkiewicz  ,           St. Stephens Ward
Evan,            Zigomalas,              Abbey Ward
France,          Andrade,                East Southbourne and Tuckton W
Ulysses,         Mcwalters ,             Hawerby cum Beesby
Tyisha,          Veness ,                Greets Green and Lyng Ward
Eric,            Rampy ,                 Desborough
Marg,            Grasmick,               Bargate Ward
Laquita ,        Hisaw,                  Chirton Ward
Lura,            Manzella ,              Staple Hill Ward
```

```
Overwriting test2.csv
```

csv : reader 읽기

reader로 읽은 파일의 데이터는 list 이므로 index로 처리 가능

```
with open('test2.csv', 'r') as mycsvfile:
    thedata = csv.reader(mycsvfile, dialect='mydialect')
    for row in thedata:
        print(row[0]+"\t \t"+row[1]+"\t \t"+row[2])
```

```
first_name              last_name               city
Aleshia         Tomkiewicz              St. Stephens Ward
Evan            Zigomalas               Abbey Ward
France          Andrade         East Southbourne and Tuckton W
Ulysses         Mcwalters               Hawerby cum Beesby
Tyisha          Veness          Greets Green and Lyng Ward
Eric            Rampy           Desborough
Marg            Grasmick                Bargate Ward
Laquita                 Hisaw           Chirton Ward
Lura            Manzella                Staple Hill Ward
```

csv: index 검색 처리

csv : 파일 생성

csv 파일 생성

```
%%writefile test3.csv
Name,     Phone numbers,    Address
Aleshia,  01835-703597,     St. Stephens Ward
Evan  ,   01937-864715,     Abbey Ward
France,   01347-368222,     East Southbourne and Tuckton W
Ulysses,  01912-771311 ,    Hawerby cum Beesby
Tyisha,   01547-429341,     Greets Green and Lyng Ward
Eric ,    01969-886290,     Desborough
Marg ,    01865-582516,     Bargate Ward
Laquita,  01746-394243,     Chirton Ward
Lura ,    01907-538509,     Staple Hill Ward
```

```
Writing test3.csv
```

csv : DictReader 읽기

DictReader로 파일을 생성하면 dict 타입을 지원하므로 헤더정보로 처리가 index가 가능

```
with open('test3.csv', 'r') as mycsvfile:
    dictofdata = csv.DictReader(mycsvfile, dialect='mydialect')
    for row in dictofdata:
        print(row['Name']+"\t "+row['Phone numbers']+"\t "+row['Address'])
```

```
Aleshia  01835-703597    St. Stephens Ward
Evan     01937-864715    Abbey Ward
France   01347-368222    East Southbourne and Tuckton W
Ulysses  01912-771311    Hawerby cum Beesby
Tyisha   01547-429341    Greets Green and Lyng Ward
Eric     01969-886290    Desborough
Marg     01865-582516    Bargate Ward
Laquita  01746-394243    Chirton Ward
Lura     01907-538509    Staple Hill Ward
```

list file로 csv 생성

csv : writer/reader 1

list로 데이터를 받아서 csv 파일 생성 및 읽기처리

```
arrayofdata=[['A','B', "C"],
             ['something','spam',2.334],
             ['anything','spam',0]]

with open('test4.csv', 'w', newline='') as mycsvfile:
    thedatawriter = csv.writer(mycsvfile, dialect='mydialect')
    for row in arrayofdata:
        thedatawriter.writerow(row)


with open('test4.csv', 'r') as mycsvfile:
    thedata = csv.reader(mycsvfile, dialect='mydialect')
    for row in thedata:
        print(row[0]+"\t \t"+row[1]+"\t \t"+row[2])
```

```
A               B               C
something               spam            2.334
anything                spam            0
```

csv : writer/reader 2

list로 데이터를 받아서 csv 파일 생성 및 읽기처리

```
arrayofdata=[['something','spam',2.334],
             ['anything','spam',0]]

with open('test5.csv', 'w', newline='') as mycsvfile:
    thedatawriter = csv.writer(mycsvfile, dialect='mydialect')
    for row in arrayofdata:
        thedatawriter.writerow(row)


with open('test5.csv', 'r') as mycsvfile:
    thedata = csv.reader(mycsvfile, dialect='mydialect')
    for row in thedata:
        print(row[0]+"\t \t"+row[1]+"\t \t"+row[2])
```

```
something               spam            2.334
anything                spam            0
```