

1. 코드 작성의 기본 설명

■ 함수 이름들

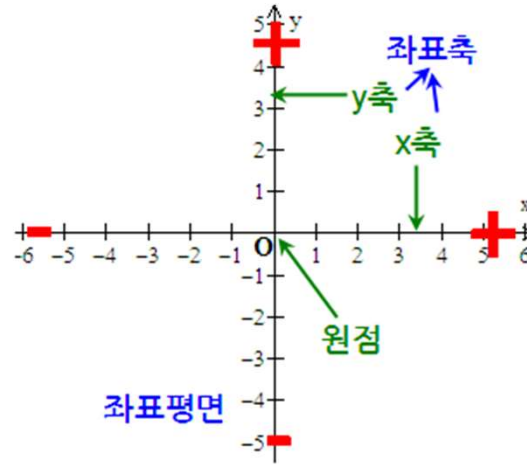
- ① 함수의 이름은 가능하면 쉽게, 별도로 외우지 않아도 연상이 가능한 방법으로 하려고 하였다. 그래서 3자리로 만들었으며, 어떤 이름은 약간 억지스러운 것도 있으니 이해 바랍니다
- ② 주소와 영역에 사용되는 것은 x,y로 만들었으며, 이때 x는 가로를 뜻하는 것으로 사용하였다. X축을 뜻하면 가로이지만, x좌표축의 표현 형식으로 보면 y를 나타내야 하지만, 여기서는 여러 고민중, x는 가로를 뜻하도록 하였으니 혼돈이 있으셔도 양해 바랍니다(즉, x를 숫자로 표현할때 x방향에 있는 몇번째로 할 것인지 xy좌표계로 할것인지를 고민하였다는 뜻이다)
- ③ 함수의 기본 내용은 외형은 엑셀로 조정을 하고, 내용이 변경되는 부분은 python의 다른 모듈들을 이용하여 사용자가 처리하도록 하였다. 가장 중요한 읽고 쓰는 기능만 있어도, 엑셀의 자동화는 가능하다
- ④ 자주 사용하는 기능은 만들어서 넣었으며
- ⑤ 차후 값들을 변경하거나 sqlite등을 이용하여 개인적으로 사용하던 database를 별도의 함수나 파일로 만들어 다룰 예정이다

2. X, Y좌표에 대하여

	A	B	C	D	E	F	G
1							
2							
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							
15							



	A	B	C	D	E	F	G
1							
2							
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							



- ① 기본적인 x축은 가로의 영역이다
- ② 그런데 왜그런지 엑셀에서의 좌표를 표기할때
- ③ 가로방향이 x로 보이는 것이 더 옳다는 것이다
- ④ 그래서 x는 더 가까운 것이다
- ⑤ 대부분의 자료는 세로 방향으로 많이 적으므로 x는 x방향으로 정의한다

	A	B	C	D	E	F	G
1							
2							
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							

X 방향

Y방향

3. 함수의 이름 및 기타 이름 규칙

▣ 함수 이름

실행동사_적용대상_변화내용

(언더바로 3개를 연결한다)

* 실행동사

read, write, Insert, delete, change, set, copy, move 등

- database에서 사용하는 용어사용
- Set : 변경할 내용이 값이 아닌경우

* 적용대상

range, sheet, x, y, line 등

* 변화내용

name, color, value, end 등

▣ 주소사용방법

- 가로와 세로 표기 : x = 가로, row, y = 세로, column
- 한개의 셀 : cell사용
- 여러개의 셀 : range사용

예제)

B1 => [1,2]

B1D3 => [1,2,3,4]

```
# 셀에 값을 쓰고 싶을 때 : write_cell_value (시트이름, 셀주소, 값)
# 영역에 값을 쓰고 싶을 때 : write_range_value (시트이름, range주소, 값)
# 현재 셀의 위치를 알고 싶을 때 : read_cell_address()
# 셀을 값이 있는 오른쪽 끝까지 이동시키고 싶을때 : move_rignt_end()
# 셀에 색깔을 넣고 싶을 때 : set_cell_color(시트이름, 셀주소, 값)
# 값을 복사하고 싶을때 : 값을 읽어온 후 -> 값을 쓰기
# 어디선가 읽어온 주소값을 사용하고 싶을때 : check_address_value(주소값)
```

4. 나의 코드를 만들어 사용하는 방법

■ 자신의 코드 작성

- ① 자신의 코드를 작성한다
- ② 이름은 차후에 모든 사람에게 공유가 가능하면서 고유의 이름을 위하여 "user_고유이름_버전_간단한설명.txt"으로 만들어서 사용하기를 권장한다
- ③ 만든 코드를 사용자 폴더(user_code)에 복사한다
- ④ 복사한 코드가 메뉴에 나올 수 있도록 메뉴화일에 만든 코드이름을 넣는다

	이름	수정한 날짜
pyezxl		
__pycache__	__pycache__	2021-04-15 오후 7:35
excel_addin	excel_addin	2021-04-15 오후 7:35
pyezxl_code	pyezxl_code	2021-04-15 오후 7:35
pyezxl_manual	pyezxl_manual	2021-04-15 오후 7:35
pyezxl_menu	pyezxl_menu	2021-04-15 오후 7:35
user_code	user_code	2021-04-15 오후 7:35
user_menu	user_menu	2021-04-15 오후 7:35
pyezxl-0.0.6.dist-info	__init__	2021-04-15 오후 7:35
	pyezxl	2021-04-15 오후 7:35

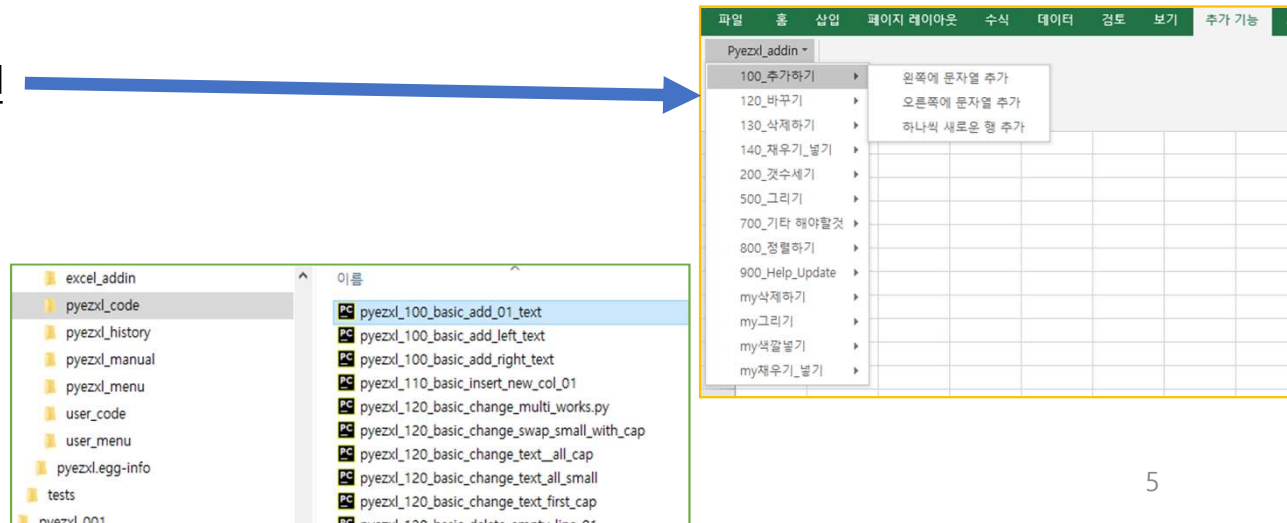
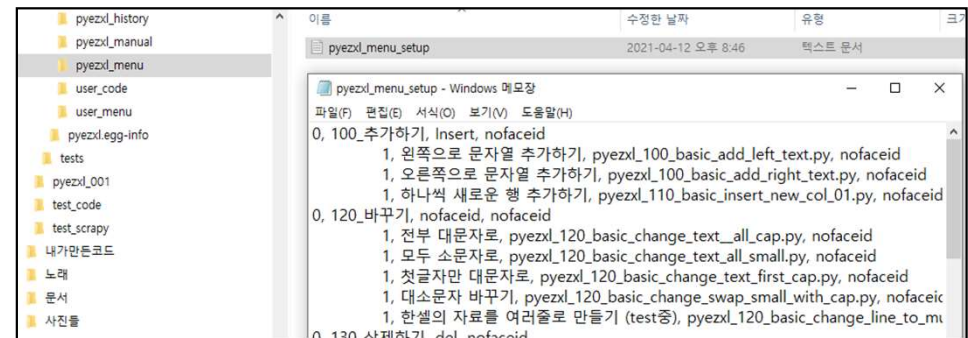
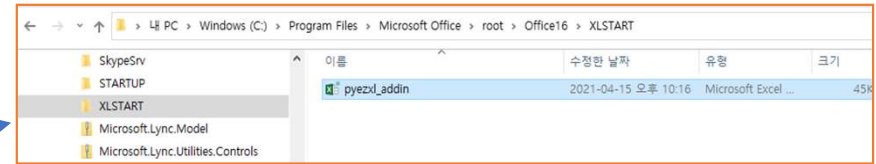
■ 자신의 코드 작성 (예)

- ① user_psj_01_sort_columns.py으로 코드 작성
- ② 만든 코드를 사용자 폴더(user_code)에 복사한다
- ③ 복사한 코드가 메뉴에 나올 수 있도록 user_menu부분을 수정한다

```
1 0,user_추가기능_1,del,nofaceid
2 →1,유저_테스트_11,user_test_11.py,nofaceid
3 →1,유저_테스트_12,user_test_12.py,nofaceid
4 0,user_추가기능_2,del,nofaceid
5 →1,유저_테스트_21,user_test_21.py,nofaceid
6 →1,유저_테스트_22,user_psj_01_sort_columns.py,nofaceid
```

5. 실행되는 순서

- ① 엑셀을 실행한다
- ② Xlstart에 들어있는 pyezxl_addin파일
이 실행된다
- ③ Pyezxl이 설치된 경로에서
pyezxl_menu와 user_menu를 읽어와
서 엑셀의 추가메뉴를 만든다
- ④ 사용자가 엑셀의 추가메뉴를 누르면
- ⑤ 추가 메뉴가 만들어질때 연결이된
Python코드를 Pyezxl이 설치된 곳에
서 찾아서 실행한다



6. 만든 이유

- 엑셀의 VBA는 사용하기 어려워 Python을 이용하여 쉽게 엑셀의 기능을 사용하도록 만들고 싶었다
- 이런 파이썬의 새로운 모듈을 만들어서 여러사람들이 쉽게 만들 수 있도록 하고 싶었다
- 또한 엑셀의 메뉴바를 만들어 기능을 쉽게 만들고 싶었다
- 누군가 만들어 놓은 좋은 코드를 쉽게 공유가 가능하도록 하고싶었다
- 공유는 버튼을 누르면 자동으로 실행되도록 하고싶었다(아직...)
- 자신만이 사용하고 싶은 코드는 따로 사용이 가능하도록 하고싶었다
- 다른나라의 언어로 편하게 사용가능하도록 변경 예정

7. 저작권 / 가격정책

* 저작권

이화일의 소스 코드 및 아이디어는 박상진과 그 가족들이 공통으로 만들어낸 창작프로그램으로 저작권이 보호된다
자신의 코드를 제외하고, 공식적으로 배포한 모든 코드와 프로그램은 저작자의 권한이다.
개인적인 용도로의 사용은 가능하며,
상업적용도로는 저작권자의 동의없이 일체의 사용을 금한다.

* 가격정책

등록용과 비등록용의 사용상 차이는 없으며,
등록용은 차후 개별 serial no를 별도로 송부 예정이며
등록비용은 향후 다른 프로젝트의 일부로 사용될 예정입니다
등록비용 : www.halmoney.com 방문 요망