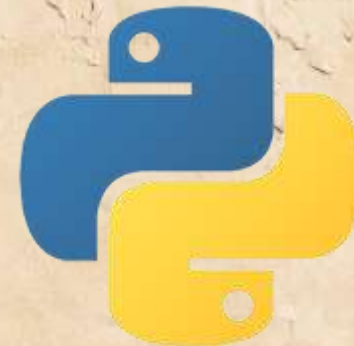


RosettaMk2in Python

Kosta 242기, node.js과정
2022년 05월, 홍승진 제작



순서

- 1 서론, 제작의의와 유례
- 2 본론, 기능 구현과 방법에 대해서
- 3 결론, 소감과 생각



Part 1

서론, 프로그램을 만든 이유, 프로그램 명칭의 유래

Part 1, 이견 대체 어느 나라 말인가....

무언가 익숙하면서 읽히지는 않는 이 글귀는 대체 무엇이란 말인가? 오호 통제라

코딩

네가ㅏㅏ(" tjfaud : “)

검색

Rnrmf, dbxnqm, spdlqj

게임

Tlqkf, snffjTekrh rodjrdnfgksp

Part 1, 이건 대체 어느 나라 말인가....

무언가 익숙하면서 읽히지는 않는 이 글귀는 대체 무엇이란 말인가? 오호 통제라

코딩

Print(“설명 : “)

검색

구글, 유튜브, 네이버

게임

(--자체검열--), 놀렀다고 개억울하네

Part 1, 이견 대체 어느 나라 말인가....

무언가 익숙하면서 읽히지는 않는 이 글귀는 대체 무엇이란 말인가? 오호 통제라

~ 、	! 1	@ 2	# 3	\$ 4	% 5	^ 6	& 7	* 8	(9) 0	- _	+ =	 ₩	←
Tab ↔	Q ㅅ ㅊ	W ㅅ ㅈ	E ㅅ ㅊ	R ㅅ ㅈ	T ㅅ ㅈ	Y ㅅ ㅈ	U ㅅ ㅈ	I ㅅ ㅈ	O ㅅ ㅈ	P ㅅ ㅈ	{ [}]	
Caps Lock ⬆	A ㅅ ㅊ	S ㅅ ㅈ	D ㅅ ㅈ	F ㅅ ㅈ	G ㅅ ㅈ	H ㅅ ㅈ	J ㅅ ㅈ	K ㅅ ㅈ	L ㅅ ㅈ	:	" '	↵ Enter		
Shift ⬆	Z ㅅ ㅈ	X ㅅ ㅈ	C ㅅ ㅈ	V ㅅ ㅈ	B ㅅ ㅈ	N ㅅ ㅈ	M ㅅ ㅈ	< ,	> .	? /	Shift ⬆			
Ctrl	Win Key	Alt	한 자						한 / 영	Alt	Win Key	Menu	Ctrl	

한국의 2벌 표준 자판 (KS X 5002)



1

배우기 편하고,
쓰기 편하다.

2

다양한 기능을
import 하기 편하다.

3

과제 제출로 쓸 수 있다. (중요)

Part 1, 이집트 상형문자 번역의 시작, 로제타 스톤

로제타석은 196년에 만들어지고 1799년에 발견되어 19세기에 이집트 상형문자의 번역을 가능하게 해주었습니다.

BC.196



이집트 프톨레마이오스 5세 때,
로제타석이 제작이 됨.

로제타석 만들어짐

AD.1799



나폴레옹의 프랑스육군과
이집트 원정

로제타석 발견

AD.1822



토마스영, 장프랑수아
샹폴리옹에게 해석됨

로제타석 해석됨

AD.2022



대영제국 박물관에서
로제타석이 현재까지 전시

로제타석 보관

Part 1, 로제타 스톤 in 대영제국 박물관!

현재 로제타석은 대영제국박물관에서 보관 중 입니다.



1

프랑스가 찾았으나, 현재는
대영제국 박물관에서 보관중

2

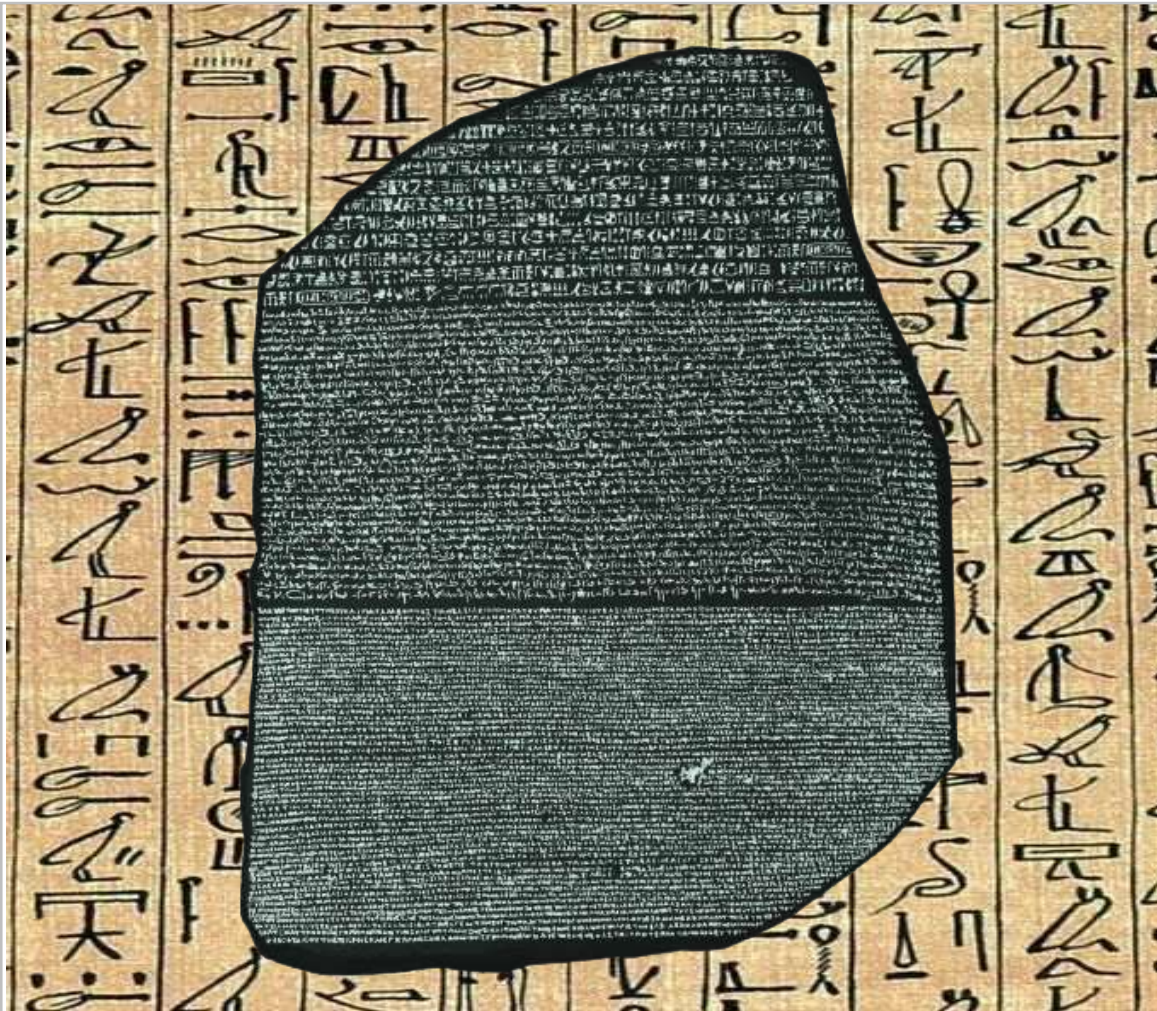
내용은 이집트의 파라오
프톨레마이오스 5세가 사제들
에게 베푼 은혜 찬양

3

아직도 일부 내용은 해석중,
완전번역은 되지 않음.

Part 1, 로제타스톤 and 구성

로제타스톤에 대한 해석과 내용



상

히어로글리프

-> 이집트 신성문자, 이집트 귀족 언어, 그림같이 생김.

중

민중문자

-> 이집트 일반인들의 언어

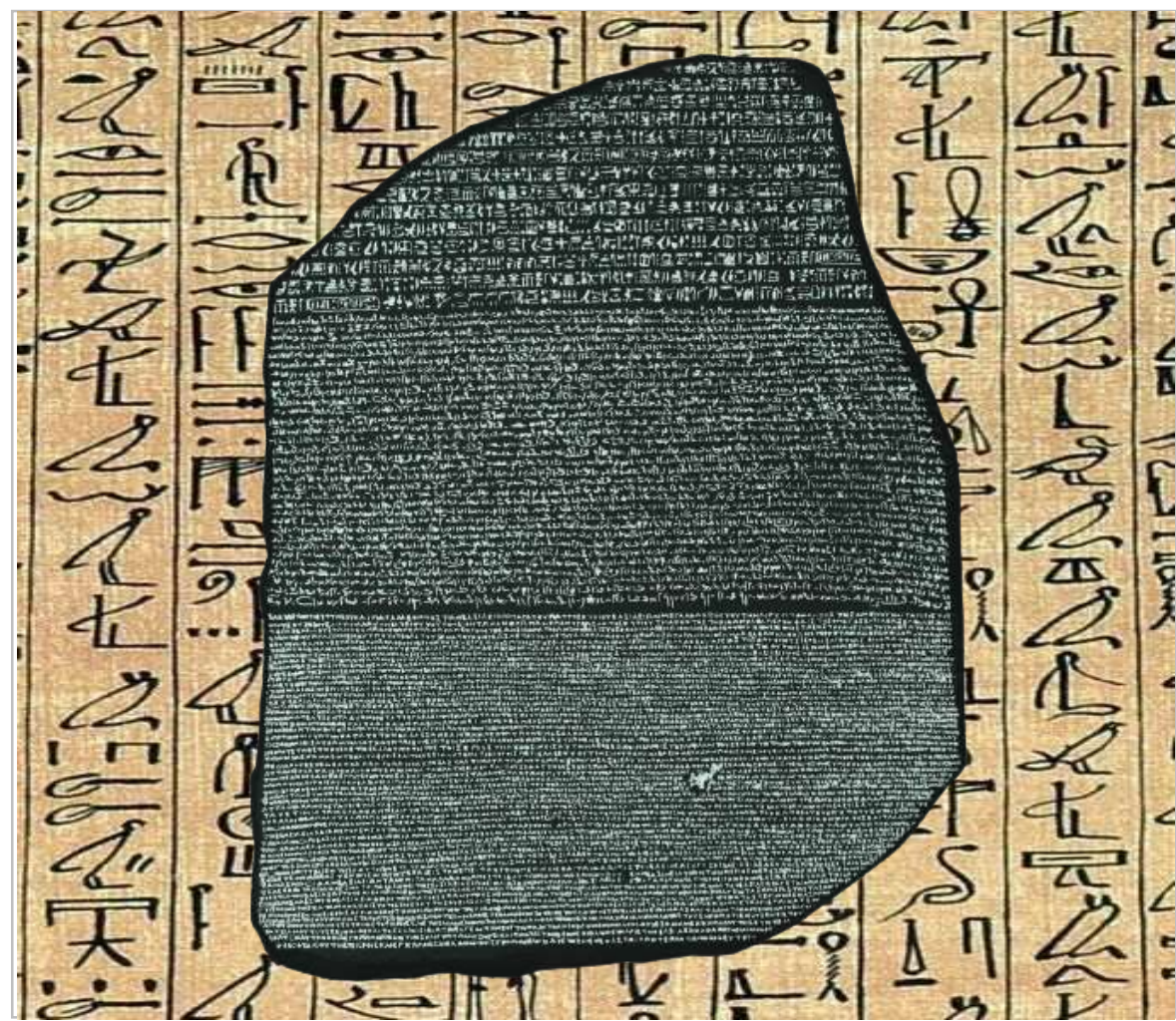
하

그리스 문자

-> 당대에 콥트문자로 보급됨,
그리스어로 전체 해석 가능!

Part 1, 본질적으로는 상당히 비슷한 이번 주제 && 로제타석

무언가 잘 안읽히는 것을 볼 수 있게 해주는 시작점이 되다.



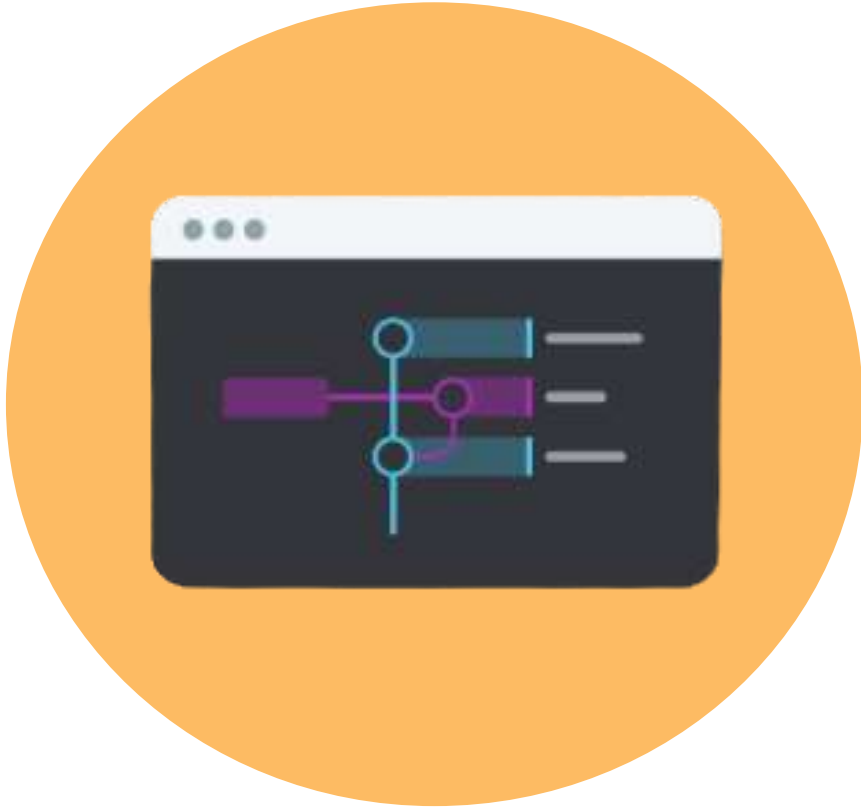
Part 2

기능과 방법,

프로그램에서 제공하는 기능,
기능을 위하여 사용한 방법

Part 2, 활용한 중대한 import 2건!

Hangul_Utils, tkinter



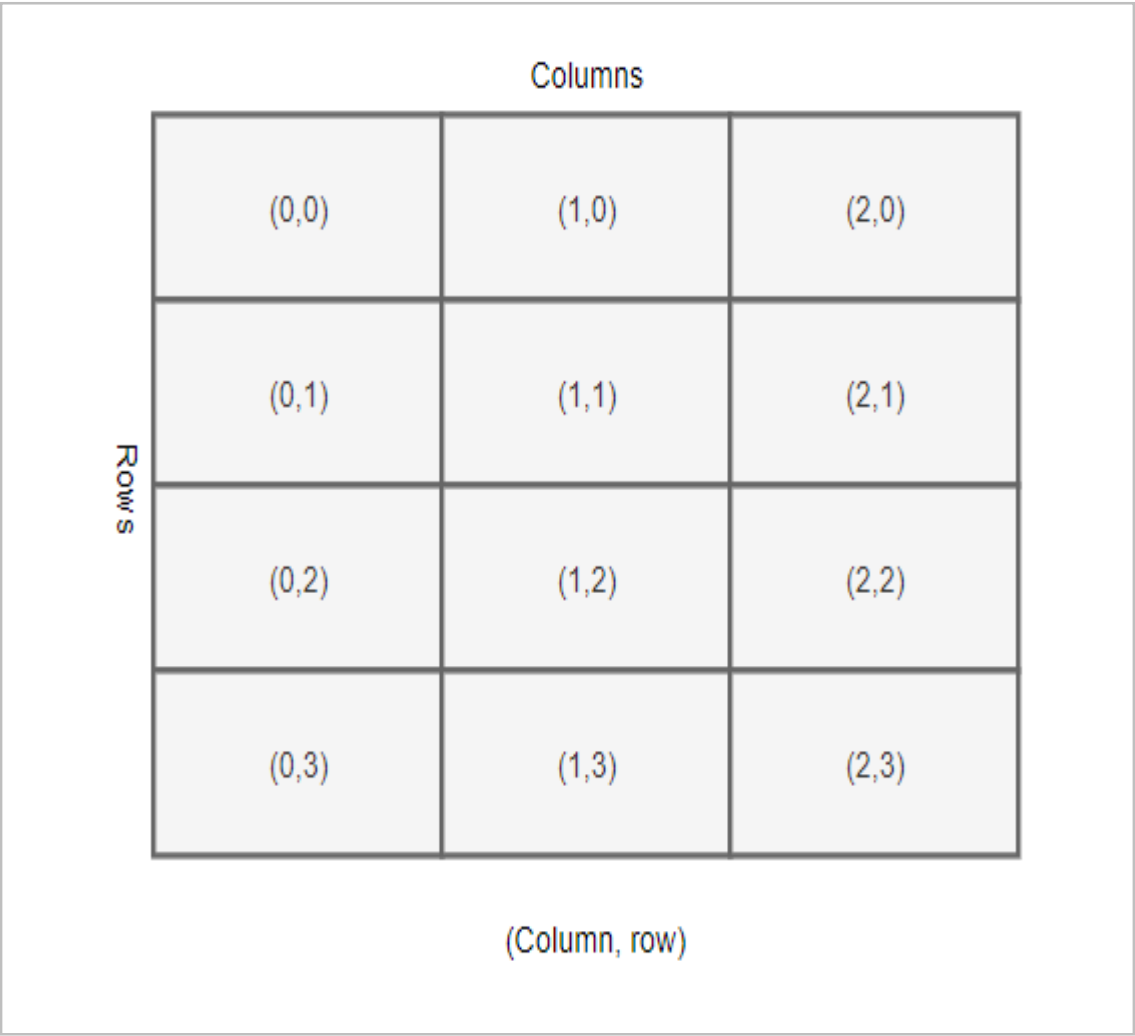
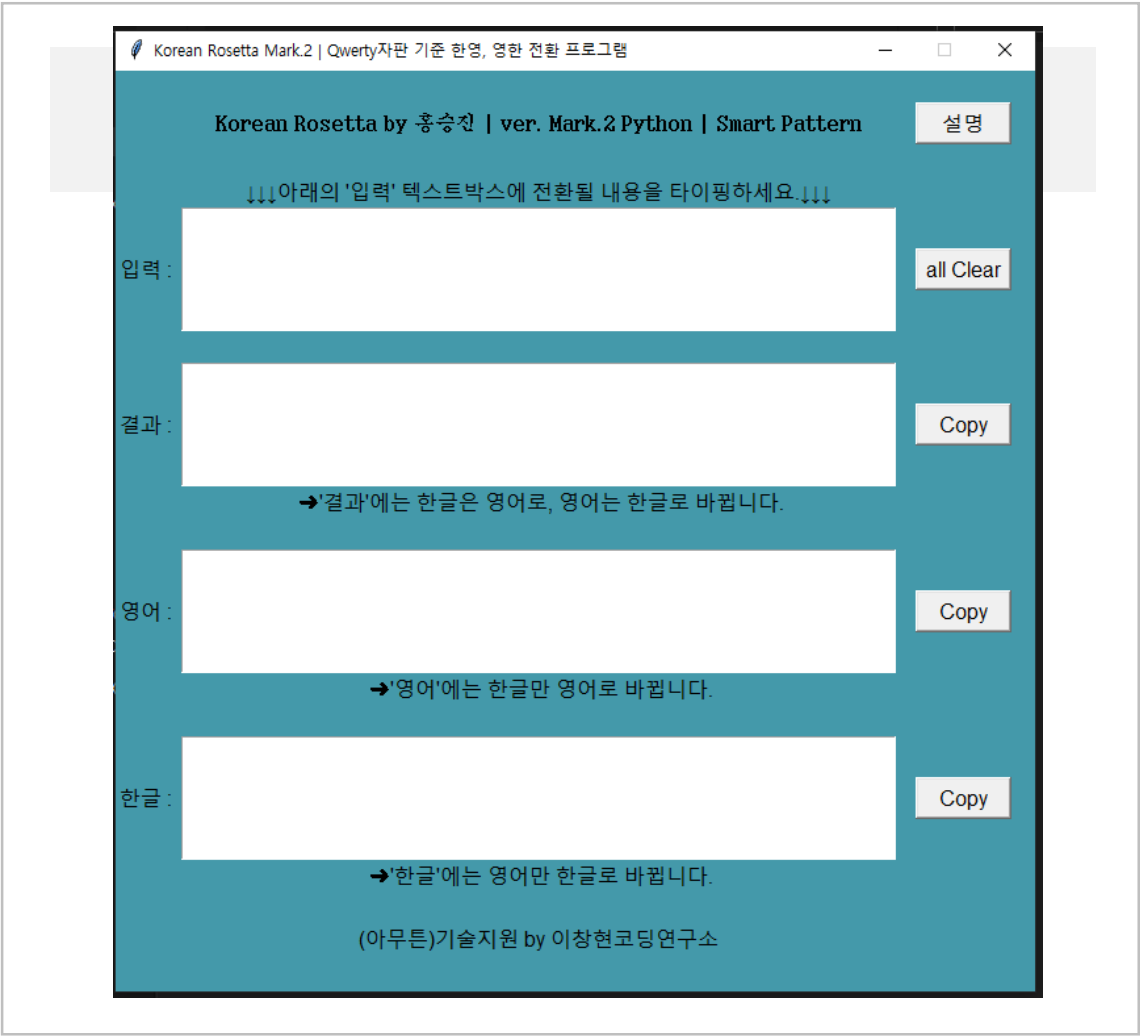
파이썬을 인터페이스화 시켜주는
Python의 tkinter



한글문장을 자음과 모음별로 나누고 합쳐주는
Hangul_Utils

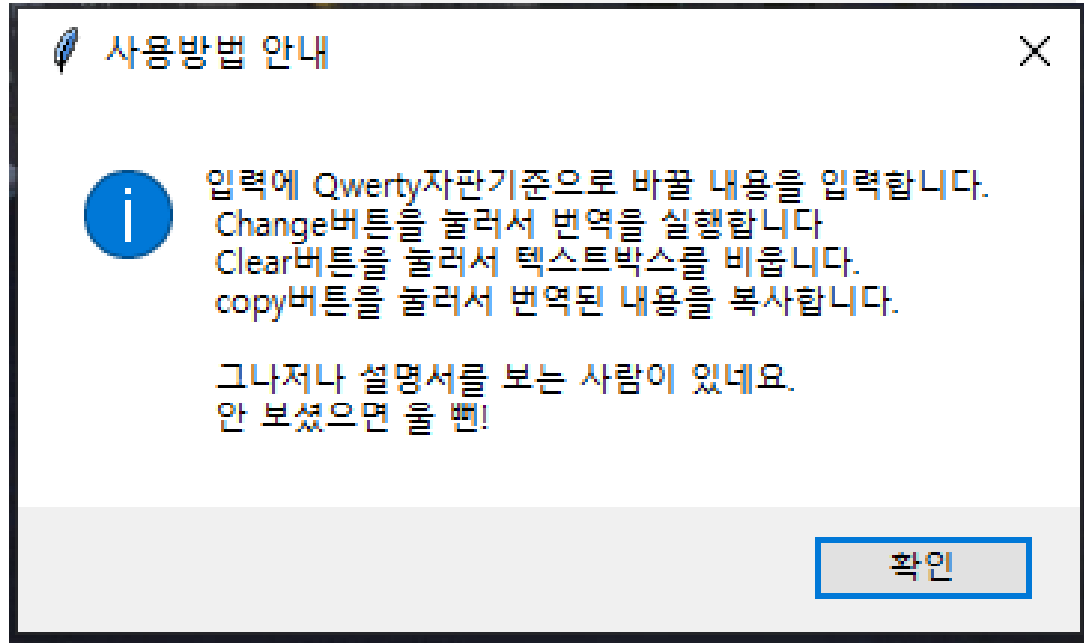
Part 2, Tkinter의 본연적 활용: GUI 구현과 Grid 구성

파이썬을 인터페이스를 갖추게하고 Grid를 통하여 어디에 위치할지를 결정합니다.



Part 2, Tkinter의 부가적 활용: 알림창 띄우기, 키보드 입력받기

설명서를 띄우기 위하여 알림창을 활용하며, 키보드가 입력이 감지되면 자동으로 입력창의 내용을 번역하도록 설정합니다.



```
#기능: any keyboard pressed되면 번역작동, 입력감지
def keyPressed(event):
    # 키보드 문자하나 출력
    DoChangeLanguge()
    print(event.char)
window.bind('<Key>', keyPressed)
window.focus_set()# 키보드 포커를 갖게 한다
```

Part 2, Split_syllables && join_jamos

```
from hangul_utils import split_syllables, join_jamos
```

Split_syllables

입력된 문장을 분할해주는 기능

다람쥐 헌 쳇바퀴에 타고파



다 람 쥐 헌 쳇 바퀴 에 타고 파

join_jamos

입력된 문장을 합쳐주는 기능

다 람 쥐 헌 쳇 바퀴 에 타고 파



다람쥐 헌 쳇바퀴에 타고파

Part 2, Dictionary 구조를 사용하기 위한 설정 (한글to영어)

```
from hangul_utils import split_syllables, join_jamos
```

```
# 자음-초성/중성(영어to한글)
cons = {'r':'ㄱ', 'R':'ㄲ', 's':'ㄴ', 'e':'ㄷ', 'E':'ㄸ', 'f':'ㄹ', 'a':'ㅁ', 'q':'ㅂ', 'Q':'ㅃ', 't':'ㅅ', 'T':'ㅆ',
        'd':'ㅇ', 'w':'ㅈ', 'W':'ㅊ', 'c':'ㅊ', 'z':'ㅋ', 'x':'ㅌ', 'v':'ㅍ', 'g':'ㅎ',
        #불필요한 shift 눌린 입력대비(자음)
        'C':'ㅊ', 'V':'ㅍ', 'A':'ㅁ', 'S':'ㄴ', 'D':'ㅇ', 'F':'ㄹ', 'G':'ㅎ', 'Z':'ㅋ'
    }

# 모음-중성(영어to한글)
vowels = {'k':'ㅏ', 'o':'ㅗ', 'i':'ㅑ', 'O':'ㅓ', 'j':'ㅕ', 'p':'ㅗ', 'u':'ㅜ', 'P':'ㅜ', 'h':'ㅛ', 'hk':'ㅛ', 'ho':'ㅛ', 'hl':'ㅛ',
          'y':'ㅟ', 'n':'ㅜ', 'nj':'ㅟ', 'np':'ㅟ', 'nl':'ㅟ', 'b':'ㅛ', 'm':'ㅛ', 'ml':'ㅛ', 'l':'ㅛ',
          #불필요한 shift 눌린 입력대비(모음)
          'H':'ㅛ', 'J':'ㅕ', 'I':'ㅑ', 'K':'ㅏ', 'L':'ㅛ', 'Y':'ㅟ', 'U':'ㅜ', 'I':'ㅑ', 'B':'ㅛ', 'N':'ㅜ', 'M':'ㅛ',
    }

# 자음-중성(영어to한글)
cons_double = {'rt':'ㄱ', 'sw':'ㅗ', 'sg':'ㄴ', 'fr':'ㄹ', 'fa':'ㅁ', 'fq':'ㅂ', 'ft':'ㅅ', 'fx':'ㅆ', 'fv':'ㅍ', 'fg':'ㅎ', 'qt':'ㅊ'}
```

한글to영어

```
KorToEng = {'ㅂ':'q', 'ㅃ':'Q', 'ㅈ':'w', 'ㅊ':'W', 'ㄷ':'e',
            'ㄸ':'E', 'ㄱ':'r', 'ㄲ':'R', 'ㅅ':'t', 'ㅆ':'T',
            'ㅟ':'y', 'ㅕ':'u', 'ㅑ':'i', 'ㅁ':'a', 'ㄴ':'s',
            'ㅇ':'d', 'ㄹ':'f', 'ㅎ':'g', 'ㅏ':'h', 'ㅓ':'j',
            'ㅏ':'k', 'ㅛ':'l', 'ㅋ':'z', 'ㅌ':'x', 'ㅊ':'c',
            'ㅍ':'v', 'ㅛ':'b', 'ㅜ':'n', 'ㅛ':'m', 'ㄱ':'rt',
            'ㅗ':'wq', 'ㄴ':'sg', 'ㄹ':'fr', 'ㅁ':'fa', 'ㅂ':'fq',
            'ㅅ':'ft', 'ㅆ':'fx', 'ㅍ':'fv', 'ㅎ':'fg',
            'ㅊ':'qt', 'ㅏ':'o', 'ㅓ':'O', 'ㅗ':'p', 'ㅜ':'P',
            'ㅛ':'hl', 'ㅛ':'ho', 'ㅛ':'hk', 'ㅟ':'nj',
            'ㅟ':'np', 'ㅟ':'nl', 'ㅛ':'ml',
    }
```


Part 2, Dictionary 구조를 사용하기 위한 설정 (영어to한글)

```
from hangul_utils import split_syllables, join_jamos
```

```
# 자음-초성/종성(영어to한글)
```

```
cons = {'r': 'ㄱ', 'R': 'ㄲ', 's': 'ㄴ', 'e': 'ㄷ', 'E': 'ㄸ', 'f': 'ㄹ', 'a': 'ㄴ', 'q': 'ㅁ', 'Q': 'ㅂ', 't': 'ㄴ', 'T': 'ㄸ',  
        'd': 'ㅇ', 'w': 'ㅈ', 'W': 'ㅊ', 'c': 'ㅊ', 'z': 'ㅋ', 'x': 'ㅌ', 'v': 'ㅍ', 'g': 'ㅎ',  
        #불필요한 shift 눌린 입력대비(자음)  
        'C': 'ㅊ', 'V': 'ㅍ', 'A': 'ㄴ', 'S': 'ㄴ', 'D': 'ㅇ', 'F': 'ㄹ', 'G': 'ㅎ', 'Z': 'ㅋ'  
    }
```

```
# 모음-중성(영어to한글)
```

```
vowels = {'k': 'ㅏ', 'o': 'ㅗ', 'i': 'ㅑ', 'O': 'ㅓ', 'j': 'ㅕ', 'p': 'ㅑ', 'u': 'ㅜ', 'P': 'ㅜ', 'h': 'ㅓ', 'hk': 'ㅑ', 'ho': 'ㅓ', 'hl': 'ㅑ',  
          'y': 'ㅛ', 'n': 'ㅓ', 'nj': 'ㅑ', 'np': 'ㅑ', 'nl': 'ㅑ', 'b': 'ㅓ', 'm': 'ㅡ', 'ml': 'ㅡ', 'l': 'ㅣ',  
          #불필요한 shift 눌린 입력대비(모음)  
          'H': 'ㅓ', 'J': 'ㅕ', 'I': 'ㅑ', 'K': 'ㅏ', 'L': 'ㅣ', 'Y': 'ㅛ', 'U': 'ㅜ', 'I': 'ㅑ', 'B': 'ㅓ', 'N': 'ㅓ', 'M': 'ㅡ',  
    }
```

```
# 자음-중성(영어to한글)
```

```
cons_double = {'rt': 'ㄲ', 'sw': 'ㅌ', 'sg': 'ㄴ', 'fr': 'ㄹ', 'fa': 'ㄴ', 'fq': 'ㄹ', 'ft': 'ㄹ', 'fx': 'ㄹ', 'fv': 'ㅍ', 'fg': 'ㄹ', 'qt': 'ㅂ'}
```

```
# 한글to영어
```

```
KorToEng = {'ㅁ': 'q', 'ㅂ': 'Q', 'ㅈ': 'w', 'ㅊ': 'W', 'ㄷ': 'e',  
            'ㄸ': 'E', 'ㄱ': 'r', 'ㄲ': 'R', 'ㅓ': 't', 'ㅌ': 'T',  
            'ㅛ': 'y', 'ㅜ': 'u', 'ㅑ': 'i', 'ㅓ': 'a', 'ㄴ': 's',  
            'ㅇ': 'd', 'ㄹ': 'f', 'ㅎ': 'g', 'ㅓ': 'h', 'ㅕ': 'j',  
            'ㅏ': 'k', 'ㅣ': 'l', 'ㅋ': 'z', 'ㅌ': 'x', 'ㅊ': 'c',  
            'ㅍ': 'v', 'ㅓ': 'b', 'ㅓ': 'n', 'ㅡ': 'm', 'ㄲ': 'rt',  
            'ㅌ': 'wq', 'ㄴ': 'sg', 'ㄹ': 'fr', 'ㄴ': 'fa', 'ㄹ': 'fq',  
            'ㄹ': 'ft', 'ㄹ': 'fx', 'ㅍ': 'fv', 'ㄹ': 'fg',  
            'ㅂ': 'qt', 'ㅗ': 'o', 'ㅓ': 'O', 'ㅑ': 'p', 'ㅜ': 'P',  
            'ㅑ': 'hl', 'ㅓ': 'ho', 'ㅑ': 'hk', 'ㅑ': 'nj',  
            'ㅑ': 'np', 'ㅑ': 'nl', 'ㅡ': 'ml',  
    }
```

Part 2, 한글을 영어로 만들어 주기

from hangul_utils import split_syllables, join_jamos #상대적으로 쉬움.

```
text = Textbox_Input.get("1.0", "end")
FinalResult = '' #변환 결과
vc = '' #바꿔야할 글자를 처음 저장하는 변수
print("입력받은 값text : " + text) #유지보수용
# 1. 한글을 분해하여 나열한뒤 되돌려 놓기, 영어는 그대로 pass
text = split_syllables(text)

i = 0 #반복용 i
while i < len(text): #텍스트의 길이가 0보다 크다면!
    v = vc[i] #fv!의 조합에 따른 배열 결과는 v[0]이다.
    t = text[i]
    j = 1 #배열의 순서 +1씩 해주는 장치
    try:
        if v == 'k': # to영어
            FinalResult+=KorToEng[t]
        else: # 기타
            FinalResult+=t
    except: # 혹시 번역이 안된 내용이 있다면!
        if v in KorToEng:
            FinalResult+=KorToEng[t]
        else:
            FinalResult+=t
    i += j
print("패턴파악용 vc 값확인 : " + vc) #유지보수용
Textbox_Output_hanEng.delete("1.0", "end") #결과 출력전 결과창 비움
Textbox_Output_hanEng.insert(INSERT, join_jamos(FinalResult))
```

한글to영어

```
KorToEng = {'ㅂ': 'q', 'ㅃ': 'Q', 'ㅈ': 'w', 'ㅉ': 'W', 'ㄷ': 'e',
            'ㄸ': 'E', 'ㄱ': 'r', 'ㄲ': 'R', 'ㅅ': 't', 'ㅆ': 'T',
            'ㅇ': 'y', 'ㅋ': 'u', 'ㆁ': 'i', 'ㅓ': 'a', 'ㅕ': 's',
            'ㅗ': 'd', 'ㅛ': 'f', 'ㅎ': 'g', 'ㅗ': 'h', 'ㅛ': 'j',
            'ㅌ': 'k', 'ㅣ': 'l', 'ㅋ': 'z', 'ㅌ': 'x', 'ㅊ': 'c',
            'ㅍ': 'v', 'ㅑ': 'b', 'ㅓ': 'n', 'ㅡ': 'm', 'ㅑ': 'rt',
            'ㅗ': 'wq', 'ㅛ': 'sg', 'ㅓ': 'fr', 'ㅑ': 'fa', 'ㅑ': 'fq',
            'ㅓ': 'ft', 'ㅑ': 'fx', 'ㅓ': 'fv', 'ㅑ': 'fg',
            'ㅑ': 'qt', 'ㅓ': 'o', 'ㅓ': 'O', 'ㅑ': 'p', 'ㅑ': 'P',
            'ㅓ': 'h1', 'ㅑ': 'ho', 'ㅓ': 'hk', 'ㅑ': 'nj',
            'ㅑ': 'np', 'ㅓ': 'nl', 'ㅑ': 'ml', }
```

상대적으로 심플함!

한글은 영어로 만들기 쉬움

영어는 한글로 만들기 어려움! ㅏㅏ

Part 2, 영어를 한글로 만들어 주기

from hangul_utils import split_syllables, join_jamos #상대적으로 어려움.

```
# 1. 한글을 분해하여 나열한뒤 되돌려 놓기, 영어는 그대로
text = split_syllables(text)
print("처리하는 값SampleText : " + text) #유지보수
# 2. 해당 글자가 자음인지 모음인지 확인
for t in text:
    if t in cons : #자음-초성/종성
        vc+='c'
    elif t in vowels: #모음-중성
        vc+='v'
    elif t in KorToEng: #영어로 바꿀 한글일 경우
        vc+='k'
    else: #테이블에 포함안되는 예외
        vc+='!'
# cvv → fVW / cv → fv / cc → dd
vc = vc.replace('cvv', 'fVW').replace('cv', 'fv')
#자음 + 모음(중성) + 모음(중성) -> fVW / 자음(초성/
#위 vc를 replace하는 것은 글자 별로 조합을 파악하기
# 3. 자음 / 모음 / 두글자 자음 에서 검색
```

```
# 자음-초성/종성(영어to한글)
cons = {'r':'ㄹ', 'R':'ㄹ', 's':'ㄴ', 'e':'ㄷ', 'E':'ㄷ', 'f':'ㄴ',
'a':'ㅁ', 'q':'ㅍ', 'Q':'ㅍ', 't':'ㅅ', 'T':'ㅅ',
'd':'ㅇ', 'w':'ㅈ', 'W':'ㅈ', 'c':'ㅊ', 'z':'ㅋ', 'x':'ㅌ',
'v':'ㅍ', 'g':'ㅎ',
'C':'ㅊ', 'V':'ㅍ', 'A':'ㅁ', 'S':'ㄴ', 'D':'ㅇ', 'F':'ㄴ',
',','G':'ㅎ', 'Z':'ㅋ'}

# 모음-중성(영어to한글)
vowels = {'k':'ㅏ', 'o':'ㅗ', 'i':'ㅑ', 'O':'ㅗ', 'j':'ㅓ', 'p':'ㅓ',
'u':'ㅕ', 'P':'ㅓ', 'h':'ㅛ', 'hk':'ㅛ', 'ho':'ㅛ', 'hl':'ㅛ',
'y':'ㅛ', 'n':'ㅓ', 'nj':'ㅓ', 'np':'ㅓ', 'nl':'ㅓ', 'b':'
ㅓ', 'm':'ㅓ', 'ml':'ㅓ', 'l':'ㅓ',
'H':'ㅓ', 'J':'ㅓ', 'I':'ㅓ', 'K':'ㅓ', 'L':'ㅓ', 'Y':'ㅓ',
',','U':'ㅓ', 'I':'ㅓ', 'B':'ㅓ', 'N':'ㅓ', 'M':'ㅓ',}
```

```
# 자음-중성(영어to한글)
cons_double = {'rt':'ㄹ', 'sw':'ㄴ', 'sg':'ㄴ', 'fr':'ㄴ', 'fa':'ㄴ',
'fq':'ㄴ', 'ft':'ㄴ', 'fx':'ㄴ', 'fv':'ㄴ', 'fg':'ㄴ', 'qt':'ㅅ'}
```

Part 2, 영어를 한글로 만들어 주기

from hangul_utils import split_syllables, join_jamos #상대적으로 어려움.

```
vc = vc.replace('cvv', 'fVW').replace('cv', 'fv').replace('cc', 'f')
#자음 + 모음(중성) + 모음(중성) -> fVW / 자음(초성/중성) + 모음 -> fv
#위 vc를 replace하는 것은 글자 별로 조합을 파악하기 위해서 사용!
# 3. 자음 / 모음 / 두글자 자음 에서 검색
i = 0 #반복용 i
while i < len(text): #텍스트의 길이가 0보다 크다면!
    v = vc[i] #fv!의 조합에 따른 배열 결과는 v[0]이다.
    t = text[i]
    j = 1 #배열의 순서 +1씩 해주는 장치
    try:
        if v == 'f' or v == 'c': # 초성(f) & 자음(c) = 자음
            FinalResult+=cons[t]
        elif v == 'V': # 더블 모음
            FinalResult+=vowels[text[i:i+2]]
            j+=1
        elif v == 'v': # 모음
            FinalResult+=vowels[t]
        elif v == 'd': # 더블 자음
            FinalResult+=cons_double[text[i:i+2]]
            j+=1
        else: # 기타
            FinalResult+=t
    except: # 혹시 번역이 안된 내용이 있다면!
        if v in cons:
            FinalResult+=cons[t]
        elif v in vowels:
            FinalResult+=vowels[t]
        else:
            FinalResult+=t
    i += j
```

자음-초성/중성(영어to한글)

```
cons = {'r': 'ㄹ', 'R': 'ㄴ', 's': 'ㄷ', 'e': 'ㄱ', 'E': 'ㄷ', 'f': 'ㄴ',
        'a': 'ㅁ', 'q': 'ㅂ', 'Q': 'ㅅ', 't': 'ㅈ', 'T': 'ㅊ',
        'd': 'ㅇ', 'w': 'ㅈ', 'W': 'ㅊ', 'c': 'ㅈ', 'z': 'ㅈ', 'x': 'ㅈ',
        'v': 'ㅈ', 'g': 'ㅈ',
        'C': 'ㅈ', 'V': 'ㅈ', 'A': 'ㅁ', 'S': 'ㄷ', 'D': 'ㅇ', 'F': 'ㄴ',
        'G': 'ㅈ', 'Z': 'ㅈ'}
```

모음-중성(영어to한글)

```
vowels = {'k': 'ㅏ', 'o': 'ㅗ', 'i': 'ㅑ', 'O': 'ㅗ', 'j': 'ㅑ', 'p': 'ㅑ',
        'u': 'ㅜ', 'P': 'ㅑ', 'h': 'ㅓ', 'hk': 'ㅑ', 'ho': 'ㅑ', 'hl': 'ㅑ',
        'y': 'ㅑ', 'n': 'ㅓ', 'nj': 'ㅑ', 'np': 'ㅑ', 'nl': 'ㅑ', 'b': 'ㅑ',
        'm': 'ㅓ', 'ml': 'ㅓ', 'l': 'ㅓ',
        'H': 'ㅓ', 'J': 'ㅑ', 'I': 'ㅑ', 'K': 'ㅑ', 'L': 'ㅓ', 'Y': 'ㅑ',
        'U': 'ㅜ', 'I': 'ㅑ', 'B': 'ㅓ', 'N': 'ㅓ', 'M': 'ㅓ'}
```

자음-중성(영어to한글)

```
cons_double = {'rt': 'ㄹ', 'sw': 'ㄴ', 'sg': 'ㄷ', 'fr': 'ㄴ', 'fa': 'ㄴ',
        'fq': 'ㄴ', 'ft': 'ㄴ', 'fx': 'ㄴ', 'fv': 'ㄴ', 'fg': 'ㄴ', 'qt': 'ㅅ'}
```


Part 2,

한글과 영어 간 전환 순서

from hangul_utils import split_syllables, join_jamos

	한글to영어	영어to한글
1	뵈트 ㅡㄷ 손 ㅡㅍ ㅌ도	ekfkawnl gjs cptqkznldp xkrhvk
2	ㄴㅅㅐㅈ ㅡㄷ ㅅㅏㄷ ㅡㅍ ㅌㄷㅑ	(패턴파악) fvfvcfVV!fvc!fvcfvfVVfv!fvfvfv!
3	(패턴파악) kkk!kk!kkk!kkkkk!	ㄷ ㅌ ㄹ ㅌ ㅓㅈㅌ ㅎ ㅌ ㄴㅈㅑ ㅅㅅ ㅌ ㅋㅌ ㅇㅑ ㅌ ㅌ ㅌㅑㅑ ㅌ
4	show me the money	다람쥐 헌 쳇바퀴에 타고파

Part 3

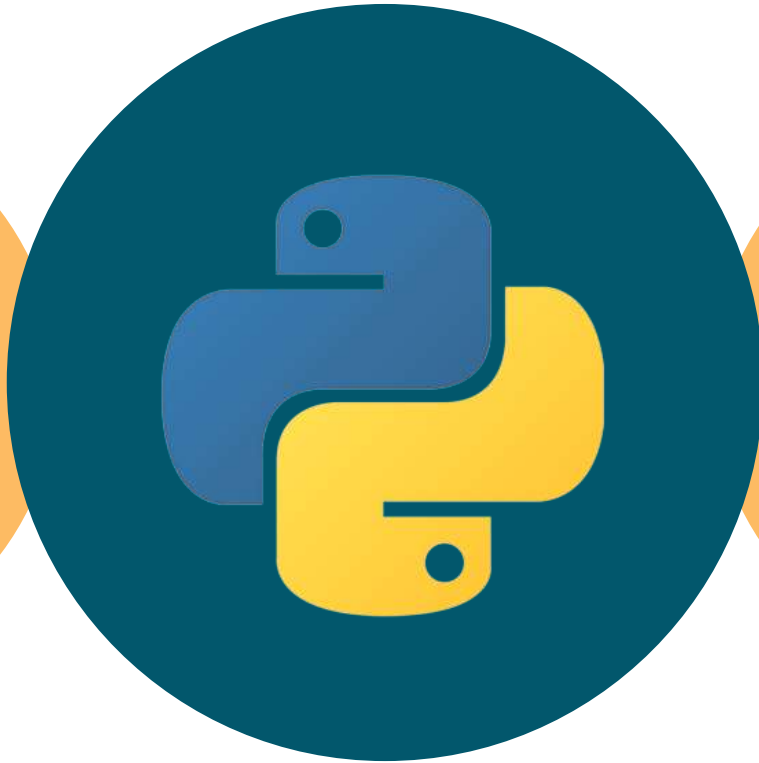
결론, Python을 사용한 소감, 본연의 생각 및 시연

Part 3, Python으로 프로그램을 만들면서 느낀 점, 소감

```
if DoPython { "쓸만한 것 같다. "} elif CodePython { "유틸기 많다. "} else { "약간 난해하다." }
```



배우기 쉽다.
쓰기 쉽다.



Python !



Import가 쉽고,
무궁무진하다.

Project Presentation !
한글과 English를
Qwerty기준에서
바꿔해주는 Python