

< English consonants & vowels >

★ 모든소리는voiced sound 와voiceless sound로나뉜다. 모음은전부다voiced sound이다.

★ Monophthongs-단모음/ Diphthongs-장모음

★ Phonology (좀더cognitive 하다) vs. Phonetics (좀더물리적이고physical 하다)

< Phonetics >

★ 성대는파도와같다. 열렸다닫혔다한다.

진동이생기는건막힌성대가강한바람에의해펄럭펄럭움직여서진동처럼느끼는것이다.

남자는1초에100번, 여자는200~250번왔다갔다한다.

★ How speech is described ∽Articulatory phonetics (mouth) / Acoustic phonetics (air) / Auditory phonetics (ear)

< Articulation >

★ The vocal tract ∽nose, pharynx, larynx, ear

★ 침을삼킬때epiglottis가기도를막는다.

★ Nasal tract, Oral tract.

★ Velum은lower되거나올라가거나둘중하나. Raised 되면nasal tract이막힌다.

★ 5 speech organs = constrictors = articulators / Oro-nasal process, articulatory process, phonation process

< Phonation process in larynx >

★ Voiced 와voiceless 구분.

★ Larynx = voicebox

< Oro-nasal process in velum >

★ Nasals : m, n, ng

< Articulatory process in lips / tongue tip / tongue body >

<Control of constrictors (articulators) >

★ CL (location) / CD (degree)

코딩은 자동화다.

C 언어, R, 파이썬 등이 언어...

모든 lng는 단어가 있다. 단어를 어떻게 combine 하면 커뮤니케이션이 되나.

컴퓨터 lng와 사람의 lng는 똑같다.

단어는 뭘까? 단어는 그 속에 의미를 포함하고 있다. 의미는 정보가 들어있는 거다.

정보를 담는 그릇 같은걸 생각하면 단어가 정보를 담는 그릇.

단어라는 그릇이 하나 있다.

정보로서의 사과를 담으면 그 단어는 사과가 되고 물을 담으면 물이 된다.

computer language 에 단어에 해당하는 부분이 변수 variable이다.

정보를 담는 그릇이 변수로서 필요한거고 숫자도, 문자도 담을 수 있다.

(사람과 기계 간의 커뮤니케이션) 정보를 가지고 기계한테 커뮤니케이션을 하는데 문법이 필요하다.

기계와의 문법은 생각보다 그렇게 어렵지 않다.

변수라는 그 그릇에 정보를 넣는 것. 정보를 assign하는 것. variable assignment. (1)

conditioning에 대한 문법이 필요하다. (~할 때 이렇게 해주세요..) If conditioning 문법.

(2)

자동화의 가장 중요한 것 중 하나가 여러 번 반복하는 것. 그것은 for 라는 것을 쓴다.

for loop (3)

가장 중요한 컴퓨터 프로그래밍의 공통점.

어떤 입력을 넣었을 때 내가 원하는 출력이 나오게 함수. (1,2,3을 이용해서 입력과 출력으로 packaging 하는게 함수)

함수 속에도 함수가 들어갈 수 있다.

재사용이 가능, 반복적 사용 가능.

<주피터 노트북>

In [] 부분 셀렉 해서 파란색 되면

a - 위에 셀 만들어짐

b - 아래에 셀 만들어짐

x - 셀 삭제

셀 바이 셀로 실행이 된다.

print(a) 해서 1이 나오는건 cell 이 아니다. 이건 출력 결과일뿐.

셀 올라가서 다른 입력값 넣었던 셀 run 하고 print 하면 결과값 또 바뀜.

실행하는 단축키- shift + enter

제일 마지막에 변수명 하나를 치면 out 값으로 변수 값을 보여준다. 하나는 보여준다.

print 안해도..

싱글 quote, 더블 quote 상관 없다.

한 줄에 다 쓰고 싶으면 ; 이거 이용해서 한줄로 쓸 수도 있다.

대괄호 대신에 괄호를 써도 됨.

리스트랑 tuple은 이름만 다르다.

괄호 쓰면 tuple,

대괄호는 리스트.

tuple이 조금 더 보안에 강하다.

```
a = {'a': 'apple', 'b': 'banana'}
```

딕셔너리.

표제어.

coma보면 리스트에 몇 개 들어있는지 알 수 있다.

리스트랑 달리 curly bracket을 썼다.

중괄호를 써야지 dictionary이고 몇 개 들어갈지 부분은 콤마로.

표제어랑 설명.

function - print, len(), type() 등

앞에다가 #을 붙여서 노트를 남겨라.

code -> markdown 해도 메모가 된다. (#붙이면 크기 커짐)

모든 lng 는 for 과 if 를 쓴다.

for 함수:

in 뒤에 있는 것을 하나하나 가져와서 i 에 넣어서 실행한다.

range 라는건 뒤 괄호에 만약 4 가 들어가면 0 부터 3 까지의 리스트를 만들어준다. 4

라는건 4 개의 인덱스를 만들어줘라 라는 뜻.

```
>>
```

```
a = [1,2,3,4]
```

```
for i in range(4):
```

```
    print(a[i])
```

```
>>
```

1

```
2
3
4
```

```
>>
a = [1,2,3,4]
for i in range(len(a)):
    print(a[i])
```

```
>>
1
2
3
4
```

enumerate 번호를 매기는 것.

```
>>>
a = ['red', 'green', 'blue', 'purple']
b = [0.2, 0.3, 0.1, 0.4]
```

```
for i, s in enumerate(a):
    print(i,s)
```

```
>>>
0 red
1 green
2 blue
3 purple
```

어떤 format 으로 하고 싶을 때, 내가 넣고 싶은 변수 자리에 중괄호{} 넣어서 만든다.
format 뒤 괄호 안에 들어 있는 것들이 중괄호 안으로 들어간다.

```
>>>
a = ['red', 'green', 'blue', 'purple']
b = [0.2, 0.3, 0.1, 0.4]
```

```
for i, s in enumerate(a):
    print("{}: {}".format(s, b[i]*100))
```

```
>>>
red: 20.0%
green: 30.0%
blue: 10.0%
purple: 40.0%
```

zip 은 그 자체나 print 함수로 출력되진 않고 list(zip()) 해야한다.

zip 함수 : 동일한 개수로 이루어진 자료형을 묶어주는 함수라고
점프투파이썬에 나온다.

조금 더 내 방식으로 설명하면 **각기 다른 개수로 되어있는 자료형일지라도
동일한 개수로 쌍을 지어 묶어준다** 라고 설명하고 싶다.

If 함수

if 에서 통과가 안되면 아예 결과값 안나옴.