

-음운론

목이 떨리는지 유무에 따른 분류

유성음 - voiced sound (모든 모음, 일부 자음) 허파에서 기압을 만들어서 성대에서 막히면 진동

무성음 - voiceless sound (일부 자음) 성대에서 안막히고 통과시키면 진동x

Phonation: 이러한 유성음과 무성음을 구별하는 것

코로 나오는 소리

m n n(eng)

monophthongs (단모음)

diphthongs (이중모음)

Phonology(음운론): 우리 머릿속에서 돌아가는 인지적인 과정, 절차

phonetics(음성학): 더 물리적이고 항상 차이가 있음 '가'를 10번 말하면 항상 다름

study on speech: 사람이 하는 말에 대한 모든 연구

성대: 피치, 음의 높낮이를 결정

입모양 : 아 에 이 오 우 차이를 만들

acoustic phonetics: 사람의 입을 떠나서 공기를 타고 가는 것에 대한 것

alveolar, palate(velum) uvula~larynx : pharynx 용어들 기본적으로 암기

oral tract 와 nasal tract

velum raised - nasal tract는 막힘 이때 나는 소리 - 모든 모음+ 비음을 뺀 모든 자음

velum lowered - nasal tract는 열림 이때 나는 소리 - 비음

코로 숨을 쉴때: velum lowered

lips/tongue tip/tongue body - 3가지 constrictor(협착을 만드는 주체)

degree(상하)와 location(앞뒤)을 control

lips, tongue body, tongue tip 에서 구체적인 location이 여러가지가 있다

ex) '드'라는 것을 constrictor, location(CL), degree(CD) 측면에서 봤을 때 constrictor는 tongue tip, location은 alveolar, degree는 stop이다

stops 폐쇄음, fricatives 마찰음, approximants 접근음?

m,n ng은 stops이다 tongue body가 완전히 막았으니까

Velum raised, larynx epiglottis opened, constrictors tongue tip, CD는alveolar, CL은 stop - T소리

모든 소리는 이러한 5가지요조로 정해진다

모든 모음은 constrictor로서 tongue body 만 쓴다
모음과 같은 constrictor을 쓰는 자음의 예를 드시오 K...등
모음과 같은 constrictor를 쓰면서 velum이 lower되면 - ng

phoneme psycho에서 p는 발음이 되지 않음 - p는 phoneme이 아니다 철자일뿐

소리를 녹음했을 때 measure할수있는 것
duration, pitch, intensity(pitch와 상관없이 측정 가능), formant(f1,f2... f의 갯수에 따라 모음이 무엇인지 결정)
formant의 갯수, 그 차이에 따라 모음을 결정할 수 있다

pitch를 측정할 때 여자/남자 구분하기 위해서는 setting에 range를 설정해줘야함

vowel acoustics(이번학기에 배울 가장 기본적인theme)

시험문제 예) P를 3가지 CONstrictor로 정의하시오

1. lips
tongue tip - CL(BY LABIAL) CD(STOP)
tongue body
2. velum - raised
3. larynx - opened

B의 경우에는 P와 모두 같고 3만 closed
D의 경우에는 TT에서 alviolar, stop
Z의 경우에는 TT에서 ALV, FRI
N의 경우에는 TT에서 ALV, STOP 2.Lowered
N의 경우와 같지만 3. opened되면 숨쉬는 것

Praat 시험@

기본이 되는 사인웨이브의 특징은
1.1초동안 몇번 반복되는지(frequency)에 의해
2.사인웨이브의 크기에 의해
정의된다

사인웨이브 가장 기본적인 형태
이 세상에 존재하는 모든 sound를 비롯한 sign은 여러 다르게 생긴 사인웨이브의 결합으로 표현될 수 있다 -
세상의 모든 복잡한 것들을 단순한 사인곡선을 통해 표현할 수 있다는 정의가 있음
magnitude(amplitude)x - 위아래 폭
frequency - hz

사인곡선의 합으로 나타낸다는 것은 그 구간에 있는 합들을 더한 것
simplex tone 이 모여서 (syntasis? 합성)
complex tone 을 형성
이 complex tone을 simple tone으로 쪼개서 분석하는 것 (spectral analysis 분해)

x축 시간 y축 value(voltage값)

각각의 사인웨이브들을 spectrum(그래프)로 바꿀 수 있어야함(x축 frequency, y축 amplitude)

spectrogram spectrum은 시간 개념이 없음 어떤 frequency성질을 가지고 있는지 보여줄 뿐, spectrum을
반복해서 보여주는게 spectrogram

complex tone의 주기는 simplex tone들중에서 제일 낮은 헤르츠(frequency) 를 가진 o tone의 주기를
따라간다(주기=진동수)

1초에 진동이 몇번되는가 = 아'라는 소리를 확대했을때 반복되는 주기가 1초에 몇번되는가

성대에서 떨릴 때 나는 순수한 소리(laryx로부터 나는 소리) - source
튜브가 어떻게 달라지는가(입모양에 따라 달라지는 것) - filter

첫번째 보이는 주파수가 우리의 피치(음의 높낮이)(F0) - Fundamental frequency
우리의 보통의 source는 decrease되는 모양임 the number of frequency by second
x2 x3 x4 ... - harmonic

시험문제)
여성이라면 피치가 더 높기때문에 첫 시작 피치가 높고 그 피치의 높이만큼 주기의 간격이 되기 때문에 주기는
더 적음

우리가 입모양을 바꿔서 head라는 소리를 냈을 때)
배음의 구조는 안깨졌지만 amplitude의 pattern이 깨졌음(원래 순수 source는 decrease하는 pattern)

웨이브 그래프 가로 시간 세로 value

스펙트로그램에서 가로 시간 세로 frequency

f0가(fundamental frequency - pitch) 정해지고 배음의 합으로 인간의 목소리의 source가 된다. 단위는 Hz
인간의 목소리의 소스는 대체로 harmonics가 decreasing

이러한 source가 우리들의 vocal tract를 만나서 filter하는 과정을 거침

formants 중요 스펙트로그램에서 검은색이 짙은 곳이 peak와 같고 흰색이 valley와 같은 모습임 이러한 산맥을 formant라고 함 첫번째 나오는 산맥 f1, f2 ...

사물에서 나는 소리는 harmonics가 없음 기타에서 나는 소리 역시 complex tone이기 때문에 harmonic가 있음

pulse train

아 에 이 오 우 각각 빨간줄 두줄을 가지고 구분 가능 (f1과 f2 산맥으로) f1, f2를 x축 y축으로 놓고 좌표를 놓고 찍었을 때 우리의 입안에서의 위치와 같음 f1은 모음의 높낮이(위아래, height) f2는 앞뒤(front, back)

coding

자동화: 똑같은 것이 반복될 때 사용

컴퓨터, 핸드폰에 있는 모든 것, 모든 프로그램은 자동화가 되어있는 것

컴퓨터에 다양한 언어가 존재(java, c파이썬..)

모든 language는 단어가 있고 이 단어들을 이용해서 combine을 한다(문법)

단어에는 의미가 들어있다(정보)

이 것을 컴퓨터에 대입하면 단어:변수(variable), 문법

컴퓨터 program문법

1.빈그릇에 정보를 assign하는 것

2.자동화할때 conditioning 조건에 따라 실행(if)

3.여러번 반복하는 것(for)

4.제일 중요한 것 함수를 배워야 함 내가 어떤 것을 넣으면 원하는 결과가 나오는 것 위의 3가지를 입력으로 packaging했을 때 어떠한 출력이 나오는 것 함수 속에 함수존재 가능(재사용)

variables

정보의 종류는 2가지 - 숫자, 글자

기호=는 절대 같다는 뜻이 아니라 오른쪽에 있는 정보를 왼쪽에 있는 variables로 assign한다 (a=1이면 1이라는 정보를 a에 assign)

print(a)하면 1이 나옴 어떤 변수를 넣으면 결과가 나옴

python에서 유용한 함수들의 패키지

anconda가 좋은 함수들을 모아 놓은 것 추가

입력이라는 것 ()

셀의 왼쪽 누르고 b-아래에 셀을 만들어줌

a-위에 셀을 만들어줌

x-지워짐

셀by셀끼리 실행 가능

a라는 변수는 하나밖에 없기 때문에 a=2라고 assign을 하면 2라고 나옴(덮어써짐)

run 단축키 -shift+enter

quote가 안되어 있으면 각각 변수 처리됨 문자를 입력할때는"를 넣어줘야 한다

"를 안하면 love가 변수처리된다 따라서 love를 정의해줘야함(x같은 느낌)

```
a=1
```

```
b=2
```

```
c=3
```

```
c
```

이런식으로 마지막에 변수를 하나 치면 print를 치지 않아도 하나정도는 결과값을 구해줌

가로로 쓰고싶으면 세미콜론; 사용 (a=1; b=2; c=3)

print도 마찬가지로 세미콜론 사용하면 결과값을 여러개 보여줌

list:한꺼번에 모아서 넣는 것 대괄호를 이용하면 됨(a=[1, 2, 3, 4, 5])

type(a) a가 가지고 있는 성질 출력

a가 숫자를 여러개 가지고 있으면 list, 나머지 경우 float, int 등

str(ing) 문자

float 실수(1.2)

int 정수? 한개(1)

list (1,2,3,4,5)여러개(문자든 숫자든 여러개 들어가있으면 집합으로 되어있으면)

a=['love','bye'] 이런 것도 list 아이템은 3개

대괄호 대신 괄호를 써도 똑같음

대신 결과 값이 tuple이라고 나옴 (tuple=list tuple이 보안에 좀 강하다 바꾸기 힘들다)

dictionary: a={'a':'apple','b':'banana'} 표제어와 같은 이런식 대괄호가 아니고 중괄호를 써야함 안에는 콤마, 콜론으로 표시

정보를 가져오는 것 retrieve

[1,2]에서

나열되어있는 것중에 가장 앞에 것은 첫번째가 아니라 영번째

영번째 - 1

첫번째 - 2

두줄적이고싶은거 한줄로 - 세미콜론

대괄호 - 정보를 가져온다는 의미

딕셔너리에서

페어세트에서 앞부분을 인덱스로 쓴다 (꺼낼때 앞부분을 쳐야 정보가 나온다)

s='abcdef' -string

n=[100,200,300] - list

list에서 -1번째는 맨뒤에서 첫번째것

print(s[1:3]) 에서 :은 더하기 - 첫번째것부터 네번째것 전까지 더하기

len() 함수(length) - variable안에 있는 정보의 갯수(길이)

어떠한 variable 다음에 .upper() - 대문자로 만들어줌

파이선에서는 함수라고 해서 항상 괄호열고 해야하는 것이 아니고 위에서 처럼 점찍고 하는 함수도 많음

find() 는 무엇인가 찾아주는 함수이면서 첫번째 것만 찾아준다 this가 많아도 첫번째 this만 (0부터 시작)

rindex

오른쪽에 있는 index를 찾아주는 기능

strip()함수

순수한 text만 남겨주는 함수

split(' ')

긴 문장에 있는 단어들을 list로 만들어주는 함수

원래문장 단어사이에 스페이스면 저 "사이에 스페이스 혹시 문장이 콤마로 이루어져있으면 "사이에 콤마

join함수

s=' '.join(tokens)

스페이스를 이용해서 토큰에 들어있는 리스트들을 붙여라

s=','.join(tokens)

콤마를 이용해서 토큰에 들어있는 리스트들을 붙여라

s=s.replace('this','that')

모든 this를 that으로 바꿔라

string과 list 비슷

variable 중요

function은 미리 만들어 놓은것
print len 등..

'for' loop 'if' conditioning

#을 붙이고 그 뒤에 노트를 하면 그 내용은 실행이 안됨

for ()in (): in뒤에 있는 것들을 하나하나씩 돌려서 i가 그 하나하나를 받아서
print () 무엇인가를 해라

in뒤에 range함수 사용 가능 range 뒤에 숫자가 나오면 list를 만들어줌 4 가 들어가면 4개의 index를
만들어줘라(0,1,2,3)

for loop 다음칸에는 indent? 들여쓰기가 있어야 한다

어떤 list를 그 output값이 변수가 두개로 받아질 수 있다

i는 0 s 는 red...

zip은 딕셔너리처럼 페어로 존재하게 해줌

==가 우리가 아는 = 의 기능

!= 는 =이 아니면 not equal

```
for i in range(1,3):  
    for j in range(3,5):  
        print(i*j)
```

두개의 for loop함수 무조건 시험

```
for i in range(1,3):  
    면 1부터 3전까지 range : 1,2
```

range함수에 숫자 두개를 넣으면