

 **blissray** Update lab_8. Resolves #5

8a1a674 on 5 Dec 2017

1 contributor

233 lines (186 sloc)13.9 KB

Lab - Morse code (morsecode)

Copyright 2017 © document created by teamLab.gachon@gmail.com

Introduction

[PDF 파일 다운로드](#)

이번 Lab은 파이썬이 제공하는 데이터 구조중 가장 활용성이 높은 Dict Type의 데이터를 활용하여 모스부호(Morse Code)를 번역하는 프로그램을 개발합니다. 이미 우리가 잘 알고 있듯이 모스 부호는 일종의 신호로 빛이나 소리를 이용하여 메시지를 전달하는 방식을 의미합니다¹.

실제 모스 부호는 정해진 규칙이 있는데, 일반적인 규칙은 아래와 같습니다.

A●-
B-●●●
C-●-●
D-●●
E●
F●●-●
G--●
H●●●●
I●●

J●---
K-●-
L●-●●
M--
N-●
O---
P●--●
Q--●-
R●-●

S●●●
T-
U●●-
V●●●-
W●--
X-●●-
Y-●--
Z--●●

이러한 규칙을 컴퓨터에서 사용하는 문자중 "-", "."을 사용해서 나타내면 아래와 같이 표현할 수 있습니다.

문자	부호	문자	부호	문자	부호	문자	부호
A	.-	B	-...	C	-.-.	D	-..
E	.	F	..-.	G	--.	H
I	..	J	.----	K	-.-	L	.-..
M	--	N	-.	O	---	P	.-.-.
Q	---.-	R	.-.	S	...	T	-
U	..-	V	...-	W	.-.-	X	-.-.
Y	-.--	Z	--..				

이번 Lab은 사용자가 알파벳 문자를 입력하면 모스 부호로 모스 부호를 입력하면 알파벳으로 바뀌주는 모스부호 변환기 프로그램을 만드는 것을 목적으로 합니다. 예를 들면 아래와 같은 형태입니다.

사용자의 문자열 입력

사용자 입력: SOS 출력값: ... --- ...

사용자의 모스부호 입력

사용자 입력: ... --- ... 출력값: SOS

물론 변환이 불가능한 입력에 한해서는 여러 메시지를 출력해야 합니다. 지금까지처럼요. 이미 Baseball을 통해 다양한 코딩 방식에 대해 경험한 여러분들에게는 그리 어렵지 않은 숙제일 거라고 믿습니다. 예! 물론 제 희망사항입니다. 즐거운 마음으로 도전해 봅시다.

숙제 파일(lab_8.zip) 다운로드

먼저 해야 할 일은 숙제 파일을 다운로드 받는 것 입니다. Chrome 또는 익스플로러와 같은 웹 브라우저 주소창에 아래 주소를 입력합니다.

https://github.com/TeamLab/introduction_to_python_TEAMLAB_MOOC/blob/master/lab_assignment/lab_8/lab_8.zip

다운로드를 위해 View Raw 또는 Download 버튼을 클릭합니다. 또는 아래 다운로드 링크를 클릭하면 자동으로 다운로드가 됩니다. [Lab morsecode - 다운로드](#) 다운로드 된 lab_8.zip 파일을 작업 폴더로 이동한 후 압축해제 후 작업하시길 바랍니다. 압축해제 하면 폴더가 linux_mac 과 windows 로 나뉘져 있습니다. 자신의 OS에 맞는 폴더로 이동해서 코드를 수정해 주시기 바랍니다.

수정전 알아두어야 할 파이썬 함수 join

이번 Lab을 진행하기전에 알아두고 시작하면 좋은 파이썬 함수가 있습니다. 바로 List에 있는 값들을 String type의 값으로 변환 시켜줄때 사용하는 join 함수이다. 함수의 사용법은 아래와 같습니다.

```
>>> test_list = ["a","b","c","d","e"]
>>> "-".join(test_list)
'a-b-c-d-e'
>>> ".".join(test_list)
'a.b.c.d.e'
>>> " ".join(test_list)
'a b c d e'
```

위 코드에서 보듯이 join 함수를 사용법 "연결부호".join(연결할 리스트 변수) 로 이루어 져있다. 부호가 변환에 따라 연결된 List 값도 각각 변환되어 출력됩니다. join 함수는 변환된 모스부호를 연결해서 표현할때 유용하게 사용될 수 있는 함수이므로 참고하기 바랍니다.

morsecode.py 파일 Overview

atom 으로는 morsecode.py 을 열어 전체적인 개요를 살펴봅시다. atom morsecode.py 명령으로 파일을 열어보면 main 함수와 여러개의 함수들이 존재할 것 입니다. 각 함수들은 여러분께서 직접 작성해서 제출해야 하는 함수이고, main 함수는 실제 morsecode를 실행하는 함수입니다. 먼저 본 Lab에서 제공하는 두개의 Helper 함수를 살펴봅시다.

첫 번째 Helper 함수는 get_morse_code_dict 함수입니다. 본 함수를 사용자가 호출 할 경우, 모스부호를 호출할 수 있는 dict type의 데이터를 반환해준다. 함수는 아래와 같습니다.

```
def get_morse_code_dict():
    morse_code = {
        "A": ".-", "N": "-.", "B": "-...", "O": "---", "C": "-.-.",
        "P": ".-..", "D": "-..", "Q": "--.", "E": ".",
        "R": ".-.", "F": "..-.", "S": "...", "G": "--.",
        "T": "-", "H": "....", "U": "..-", "I": "..", "V": "...-",
        "K": "-.-", "X": "-.-.", "J": "-.-.-", "W": "-.-",
        "L": "-.-.", "Y": "-.-.", "M": "--", "Z": "--.."
    }
    return morse_code
```

위 함수는 아래와 같이 활용 가능합니다.

```
>>> morse_code_dict = get_morse_code_dict()
>>> morse_code_dict["A"]
'.-'
```

두 번째 Helper 함수는 `get_help_message` 함수입니다. 본 함수를 사용자가 호출 할 경우, 아래와 같이 모스부호에 대한 설명이 출력됩니다.
`main` 함수에서 사용할 함수이므로 당장 신경쓸 필요는 없습니다.

```
>>> morsecode.get_help_message()
'HELP - International Morse Code List\nA: .-\\tB: -...\\tC: -.\\tD: -.\\tE: .\\t\n
F: ..\\tG: --\\tH: ....\\tI: ..\\tJ: .---\\tK: -.\\tL: -.\\tM: --\\tN: -.\\tO: ---\\t\n
P: .---\\tQ: --.\\tR: .-\\tS: ...\\tT: -\\t\\nU: ..\\tV: ...\\tW: .---\\tX: -.\\t\n
Y: -.---\\t\\nZ: --.\\t'
```

위의 결과값중 `\\t` 는 TAB Size로 간격을 벌리는 특수 기호입니다. 실제로 Console창에서 실행시킬때는 8칸씩 글자가 떨어져서 나옵니다.
이제 수정해야 할 함수 리스트를 살펴 봅시다.

함수	설명
<code>is_help_command</code>	문자열값을 입력받아 입력된 값이 대소문자 구분없이 "H" 또는 "HELP"일 경우 True, 그렇지 않을 경우 False를 반환함
<code>is_validated_english_sentence</code>	문자열값을 입력받아 입력된 값이 모스부호로 변환이 가능하면 True를 불가능하면 False를 반환함. 모스 부호로 변환이 불가능한 경우는 아래 세 가지에 해당됨 1) 숫자가 포함되어 있음 2) <code>_@#%\$^&*()-+=[]{}"; '`~</code> 와 같은 특수문자가 포함되어 있음, 단 문장부호(.,!?)는 제외됨 3) 문장부호(.,!?)를 제외하면 입력값이 없거나 빈칸 만 입력했을 경우 예를 들면, "Hello, !" 이나 "Hello, !" 혹은 " Hello, !" 같은 경우에도 문장부호 , , ! 그리고 빈칸을 제외하면 문제가 없을 경우는 True 로 반환되어야함
<code>is_validated_morse_code</code>	모스부호 형태의 무자열 값을 입력받아, 알파벳으로 변환이 가능하면 True, 불가능하면 False를 반환함. 알파벳으로 변환되는 경우는 아래 두 가지에 해당되지 않는 경우임 1) "-,;,;" 외 다른 글자가 포함되어 있는 경우 2) <code>get_morse_code_dict</code> 함수에 정의된 Morse Code 부호외 다른 코드가 입력된 경우 ex).....
<code>get_cleaned_english_sentence</code>	문자열값으로 Morse Code로 변환 가능한 영어 문장을 입력받아, 입력된 영어문장에서 4개의 문장부호를 ",.,!?" 삭제하고, 양쪽끝 여백을 제거한 문자열 값 반환
<code>decoding_character</code>	문자열값으로 <code>get_morse_code_dict</code> 함수로 알파벳으로 치환이 가능한 값을 입력받아 Morse Code를 알파벳으로 변환한 값을 반환함
<code>encoding_character</code>	문자열값으로 알파벳 한 글자의 입력받아 <code>get_morse_code_dict</code> 함수의 반환 값으로 인해 변환된 모스부호 문자열값을 반환함
<code>decoding_sentence</code>	문자열 값으로 모스 부호로 이루어진 문자열을 입력받아 모스부호를 알파벳으로 변환한 문자열을 반환함, 단 이때 모스부호의 띄어쓰기는 고려되어야함, 예를 들면 -. -. -. --- 는 NNN 0 로 변환되어야 함. 즉 알파벳 변환시 띄어쓰기가 있는 경우 모스부호는 두칸의 띄어쓰기가 되어야함
<code>encoding_sentence</code>	문자열 값으로 모스 부호로 변환이 가능한 영어문장을 입력받아, 입력된 영어문장 문자열 값을 모스 부호로 변환된 알파벳으로 변환한 문자열을 반환하되, 양쪽 끝에 빈칸은 삭제함 이때 중요한 것은 모든 모스부호는 한칸씩 띄어쓰기를 해서 표현되어야 하며, 만약 입력된 영어 문장에서 단어와 단어사이에에는 두 칸의 띄어쓰기를 표시해야함

마지막 `encoding_sentence` 함수가 조금 이해하기 어려울 건데 다음 그림과 같다고 생각하면 됩니다. 즉 ! 와 같은 문장부호는 삭제하고, 글자와 글자사이에는 한 칸의 여백을 단어와 단어 사이에는 두 칸의 여백을 줘서 모스부호를 출력하면 됩니다. 이 규칙은 `decoding_sentence` 에도 동일하게 적용됩니다.

Lab morsecode에서 encoding_sentence만 pass되지 않는다는 분들이 많으셔서 전체 공지합니다! encoding_sentence 함수에서는 영어 단어 사이에 띄어쓰기가 두 개 이상일 경우 한 개로 처리해준 후 모스부호로 변환해 주어야 합니다!



main 함수 수정하기

이번엔 메인함수이다. 이번 Lab의 메인함수는 비교적 간단합니다. 다음이 기본 Template입니다.

```
def main():
    print("Morse Code Program!!")
    # ===Modify codes below=====

    # =====
    print("Good Bye")
    print("Morse Code Program Finished!!")
```

main 함수는 다음과 같은 규칙을 가진다.

1. 사용자가 0을 입력하면 종료된다.
2. 사용자가 대소문자에 상관없이 "h"또는 "help"를 입력하면 get_help_message 함수를 호출하여 아래와 같이 출력된다.

```
Input your message(H - Help, 0 - Exit): H
HELP - International Morse Code List
A: .-   B: -... C: -.-. D: -..  E: .
F: ..-. G: --   H: .... I: ..   J: .---
K: -.-  L: .-.  M: --   N: -.-  O: ---
P: .--- Q: --.- R: .-   S: ...  T: -
U: ..-  V: ...- W: --   X: -.-  Y: -.-
Z: --..
```

3. 모스부호로 변환이 가능한 알파벳 문장이 입력되면 모스부호로 변환된 값이 출력된다.

```
Input your message(H - Help, 0 - Exit): SOS
... --- ...
Input your message(H - Help, 0 - Exit): Hello!
.... . .-.. .-. ---
Input your message(H - Help, 0 - Exit): This is Gachon
- . .... . .-.. .-. --- .- .-. .-.. -.- .-
```

4. 알파벳으로 변환이 가능한 모스부호가 입력되면 알파벳으로 변환해준다.

```
Input your message(H - Help, 0 - Exit): ... --- ...
SOS
Input your message(H - Help, 0 - Exit): ... . ...
SES
Input your message(H - Help, 0 - Exit): . -.- ---
EX0
Input your message(H - Help, 0 - Exit): .... --- -
HOT
```

```
Input your message(H - Help, 0 - Exit): . -.-. . . -..
EXID
```

5. 그외 변환이 불가능한 입력일 경우 에러 메시지를 출력한다.

```
Input your message(H - Help, 0 - Exit): I'm Gachon.
Wrong Input
Input your message(H - Help, 0 - Exit): This is "CS50".
Wrong Input
Input your message(H - Help, 0 - Exit): Hello 123!
Wrong Input
```

실제 실행된 프로그램의 예제화면은 아래와 같습니다.

```
estella.gachon.ac.kr - PuTTY
_morsecode$ python3.4 morsecode_answer.py
Morse Code Program!!
Input your message(H - Help, 0 - Exit): 123
Wrong Input
Input your message(H - Help, 0 - Exit): Hello
.... . -.-. . . -.. ---
Input your message(H - Help, 0 - Exit): This is CS50.
Wrong Input
Input your message(H - Help, 0 - Exit): You are not alone.
--- ..- .- .-. . -.- -.- .- ..- --- -.
Input your message(H - Help, 0 - Exit): We are here to be good programmer.
--- . -.- .- .-. . .-. . - --- -.- . -.- --- --- -.- .-. --- ---
Input your message(H - Help, 0 - Exit): Hi!! Hi!! We Are Gachon.
--- . -.- .- .-. . -.- .- --- ..- --- -.
Input your message(H - Help, 0 - Exit): My name is Sungchul Choi
--- ..- .- .-. . .-. . - --- -.- .- ..- --- -.- ..- ..- --- ..-
Input your message(H - Help, 0 - Exit): -. -. -. ---
NNN O
Input your message(H - Help, 0 - Exit): --- --- --- ... ..
OOSSH
Input your message(H - Help, 0 - Exit): ----
Wrong Input
Input your message(H - Help, 0 - Exit): -----
Wrong Input
Input your message(H - Help, 0 - Exit): -. -.- -.- -.- -.-
AAGDM
Input your message(H - Help, 0 - Exit): 0
Good Bye
Morse Code Program Finished!!
```

###숙제 template 파일 제출하기 (윈도우의 경우)

1. `windows` + `r`를 누르고 cmd 입력 후 확인을 클릭합니다.
2. 작업을 수행한 폴더로 이동 합니다.
3. 밑에 명령어를 cmd창에 입력합니다.

```
install.bat
submit.bat [YOUR_HASH_KEY]
```

숙제 template 파일 제출하기 (Mac or Linux)

1. 터미널을 구동합니다.
2. 작업을 수행한 디렉토리로 이동 합니다.
3. 밑에 bash창을 입력합니다.

```
bash install.sh
bash submit.sh [YOUR_HASH_KEY]
```

backend.ai 서비스의 업데이트에 의해 실행전 반드시 `bash install.sh` 또는 `install.bat` 수행을 바랍니다.

제대로 작성했다면 아래와 같은 메시지가 뜰 것이다.

Function Name	Passed?	Feedback
is_validated_morse_code	PASS	Good Job
encoding_character	PASS	Good Job
is_help_command	PASS	Good Job
decoding_sentence	PASS	Good Job
decoding_character	PASS	Good Job
is_validated_english_sentence	PASS	Good Job
get_cleaned_english_sentence	PASS	Good Job
main	PASS	Good Job
encoding_sentence	PASS	Good Job

Next Work

꽤 많은 프로그램을 짜다보면 이 정도 Lab은 그다지 어렵지 않게 느껴질 수도 있습니다. 점점 Lab의 수준이 낮아지므로 이제 조금 긴장을 풀고 한개씩 해결해 나가는 것을 추천합니다. 다음 시간에는 File Handling에 대한 Lab을 수행하게 됩니다. 앞으로는 훨씬 더 다양한 문제를 해결할 수 있을 것 입니다.

Human knowledge belongs to the world - from movie 'Password' -

Footnotes

1: [Morse Code Wiki](#) 페이지로 이동하면 화단에 Morse Code를 들을 수 있다. 