



English Words

12.06.2021

Team 4_OILC-302

Srinitha Chinthala

Shiva kumar Reddy Bagareddy

Sai kumar Manikonda

Sunitha Kotta

Sreekanth Thaduri

Varsha Rokkam

Overview

Read a word from the user, Then for meaningful English words using the same input word , as many combinations of words could be made. And Print all English Words.

Required Softwares

- Python IDLE
- Python Version 3.9.5
- Modules - itertools, enchant -3.2.0.

Modules

I. `itertools`

This module implements a number of iterator building blocks inspired by constructs from APL, Haskell, and SML. Each has been recast in a form suitable for Python.

The module standardizes a core set of fast, memory efficient tools that are useful by themselves or in combination. Together, they form an “iterator algebra” making it possible to construct specialized tools succinctly and efficiently in pure Python.

`itertools.permutations(iterable, r=None)`

Return successive *r* length permutations of elements in the *iterable*.

If *r* is not specified or is None, then *r* defaults to the length of the *iterable* and all possible full-length permutations are generated.

The permutation tuples are emitted in lexicographic ordering according to the order of the input *iterable*. So, if the input *iterable* is sorted, the combination tuples will be produced in sorted order.

Elements are treated as unique based on their position, not on their value. So if the input elements are unique, there will be no repeat values in each permutation.

II. Enchant

To use this module we need to install it first. For that we use the following command **pip install pyenchant**

Enchant is used to check the spelling of words and suggest corrections for words that are miss-spelled. It can use many popular spell checking packages to perform this task, including ispell, aspell and MySpell. It is quite flexible at handling multiple dictionaries and multiple languages.

class Dict(_EnchantObject)

Dict(tag=None, broker=None)

Dictionary object for the Enchant spellchecker.

Dictionary objects are responsible for checking the spelling of words and suggesting possible corrections. Each dictionary is owned by a Broker object, but unless a new Broker has explicitly been created then this will be the 'enchant' module default Broker and is of little interest.

The important methods of this class include:

- * check(): check whether a word is spelled correctly
- * suggest(): suggest correct spellings for a word
- * add(): add a word to the user's personal dictionary
- * remove(): add a word to the user's personal exclude list
- * add_to_session(): add a word to the current spellcheck session
- * store_replacement(): indicate a replacement for a given word

check(self, word)

Check the spelling of a word.

This method takes a word in the dictionary language and returns True if it is correctly spelled, and false otherwise.


Code

```
from itertools import permutations
import enchant
while True:
    input_str= input("Please Enter a String : ").upper()
    input_string = ""
    for i in input_str:
        if i.isalpha():
            input_string= input_string + i
    special_chr = "!@#$%^&*()_+ -={}[]\|;: '/?.,><"
    if input_str.isnumeric():
        print("Input containing only numbers, give at-least 2 characters")
    elif input_str in special_chr:
        print("Input containing only empty or special char, give at-least 2 characters")
    else:
        count = 0
        for i in input_string:
            if i.isalpha():
                count += 1
        if count < 2:
            print("Please Enter more than 2 Alphabets, There is no Alphabetsin Input ")
        else:
            all_permutation = {''.join(permutation) for i in range(len(input_string))
                               for permutation in permutations(input_string, i + 2)}
            Dictionary_object = enchant.Dict("en_IN")
            count = 0
            for i in all_permutation:
                if Dictionary_object.check(i):
                    print(i)
                    count += 1
            print(count,"Words from the",input_str)
            break
```

Input and OutPut

```
Please Enter a String : saikumar
SUI
RS
MUIR
MARAS
UKR
RMS
SA
AURA
SKI
MARISA
SI
SKA
ARUM
SIAM
UM
KAURIS
AMU
KS
MARA
RU
MAURA
ARKS
MUSK
MAIA
ARIUS
MRI
AIR
KRA
UK
ARAS
KARA
MURKS
AKA
SAAR
AKSUM
SAKAI
USM
```

Reference Websites and Doc

- 
1. [itertools — Functions creating iterators for efficient looping — Python 3.9.5 documentation](#)
 2. [PyEnchant — PyEnchant 3.2.0 documentation](#)
 3. [Welcome to Python.org](#)

Thank You