PyDidYouMean Documentation

Release 0.10

Al Sweigart

November 07, 2015

Contents

1 Installation 1

2 Typical Usage for “File not found” Errors 3

3 Typical Usage for “Commmand not found” Errors 5

4 Indices and tables 7

CHAPTER 1

Installation

PyDidYouMean can be installed using *pip*:

```
pip install pydidyoumean
```

The module is available on PyPI and GitHub

The documentation is available on ReadTheDocs

CHAPTER 2

Typical Usage for "File not found" Errors

If your code is unable to find a filename in the current working directory, you can print a "Did you mean %s?n" message by calling:

```
>>> import pydidyoumean
>>> pydidyoumean.printFileSuggestion('typo_filename.txt')
Did you mean typo_filename.exe?
```

The default arguments for *printFileSuggestion()* are as follows:

```
>>> import pydidyoumean
>>> pydidyoumean.printFileSuggestion('typo_filename.txt', message='Did you mean %s?\n', folder='.', t
```

The *%s* in the *message* will be replaced by the suggested file. The *threshold* is the max Levenshtein edit distance (the number of character additions, removals, or replacements needed to convert one string into another). The *printFileSuggestion()* function compares the typo filename against all files and folders in *folder*. If there are multiple filenames tied for the lowest edit distance, it is undefined which one is returned.

You can get the string that *printFileSuggestion()* prints by calling *formatFileSuggestion()*, which has all the same arguments:

```
>>> import pydidyoumean
>>> pydidyoumean.formatFileSuggestion('typo_filename.txt')
'Did you mean typo_filename.exe?\n'
```

(This naming is similar to the *pprint* module's *pprint()* and *pformat()* conventions.)

suggestFile() will return a string of the filename that *formatFileSuggestion()* would use, rather than an error message string. It does not have a *message* parameter.

suggestAllFiles() will return a generator object of all filenames that are within the threshold, sorted by distance. It does not have a *message* parameter. Pass this generator object to the *list()* function to get this as a list.

CHAPTER 3

Typical Usage for "Commmand not found" Errors

Rather than files, if your user has entered a typo'd command (or some other plain text), you can use the *printSuggestion()* function (analogous to the *printFileSuggestion()*). Instead of a *folder* parameter, *printSuggestion()* requires you pass a sequence of strings of all the possible commands that the user might have meant. The rest of the keywords are the same as *printFileSuggestion()*.

Example:

```
>>> import pydidyoumean
>>> pydidyoumean.printSuggestion('checkoout', ['add', 'bisect', 'branch', 'checkout', 'clone', 'comm
Did you mean checkout?
```

There are also functions *formatSuggestion()* (like *formatFileSuggestion()*), *suggest()* (like *suggestFile()*), and *suggestAll()* (like *suggestAllFiles()*)

Contents:

CHAPTER 4

Indices and tables

- genindex
- modindex
- search