



	dsouzarc Merge branch 'master' of https://github.com/dsouzarc/dotfiles	Latest commit 23c14bb Mar 30, 2017
..		
	NoteTaker	Randomly generated notes Mar 30, 2017
	User-Agent	On '/' or '.', adds all files in directory and all subdirectories. Go.. Dec 5, 2015
	.DS_Store	Randomly generated notes Mar 30, 2017
	CommVaultUpload.py	Bug fixes + enhancement Dec 5, 2015
	CredentialManager.py	Added logical set_value May 14, 2015
	CredentialManagerExample.py	Changed get_key to get_value May 6, 2015
	Dictionary.py	Prints definition of a word and examples of the word in a sentence May 24, 2015
	EmailAttachment.py	Enhanced to send file from current working directory May 24, 2015
	EmailMe.py	Removed dependency on SendGrid API - sends POST requests insteac May 10, 2015
	GitHubImageRenamer.py	Fixed bug that deleted previously renamed screenshots + made it easie.. Nov 18, 2015
	MacAppIconResizer.py	Added file names adjacent to image sizes Jan 15, 2016
	MusicCleaner.py	Simple script that removes duplicate songs from my roommate's music l.. Nov 17, 2015
	PasswordGenerator.py	Added comments May 28, 2015
	README.md	Updated instructions Mar 30, 2017
	TextAttachment.py	Texts attachment to my phone Jun 18, 2015
	TextLink.py	Removed dependency on SendGrid API - sends POST requests insteac May 10, 2015
	TextPDF.py	Gets a PDF view of a link and texts the PDF to mysel Mar 30, 2017
	TextPeople.py	Removed dependency on SendGrid API - sends POST requests insteac May 10, 2015
	TextPhone.py	Added comments May 10, 2015
	TextPhoneP.py	TextPhone but without CredentialManager to store infc May 10, 2015
	TextScreenshot.py	Gets a screenshot of a link's contents and texts the imageto mysel Mar 30, 2017
	iOSAppIconResizer.py	Cleaned up by moving files to respective directories Feb 15, 2015

README.md

Written by Ryan D'souza

Various Python files I run from the commandline to help me with miscellaneous tasks

CommVault Uploader

Super lightweight and quick way to upload files of any size to the cloud. It can also add all files in a directory and the directory's subdiretories. It's quicker than logging into Dropbox/Box/Copy/Mega/Gmail etc. to save files to the cloud. Plus, it adds files from subdirectories as well.

The best part is that the app is only one file and runs on standard Python - there's no need to install 3rd party libraries. The only thing that's needed is your password.

Downloading it is as simple as

```
curl -L tiny.cc/dsouzarcUploader > CommVaultUpload.py
```

And using it is as basic as

```
python CommVaultUpload.py fileOne.mp3
```

or

```
python CommVaultUpload.py fileOne.docx fileTwo.docx movie2.docx
```

or

```
python CommVaultUpload.py /
```

To upload all files in that directory and its subdirectories

Credential Manager

Super lightweight, secure (base64encode), quick, and easy to use credential manager to save and retrieve passwords. Stores the encrypted passwords in a secret json file in a secret folder in the home directory, enabling the script to be run from anywhere on the computer.

There's only one file needed (quick for copying/pasting), and all imports are from the standard python library so downloading it is as simple as

```
curl -L tiny.cc/credentialManager.py > CredentialManager.py
```

and using it is as simple as

```
python CredentialManager.py
```

I mainly built this to use on my school's computer where installing commandline tools like 'pip' is not allowed. Because of that, I can't install python's 'keyring' library to securely store my usernames/passwords. So, I made this script which can be run from anywhere

How to use

Installation & Setup

```
#Sets up directory and json file, and also prompts for quickly saving a username and password (password entering is hidden)
python CredentialManager.py
```

Accessing stored data

```
import CredentialManager;

username = CredentialManager.get_value("sendgridUsername");
password = CredentialManager.get_value("sendgridPassword");
```

Quickly modifying data (Option 1) --> Example

```
import CredentialManager;

CredentialManager.save_credentials("sendgridUsername", "someusername", sendgridPassword", "somePassword");
CredentialManager.save_key("PromMeParseAPIKey", "someapikey");

CredentialManager.delete_key("PromMeParseAPIKey");
```

Quickly modifying data (Option 2) --> Run from the terminal. Each term (ie. 'save_key') is a string param and causes the program to prompt for additional input

```
python CredentialManager save_key
python CredentialManager set_value
python CredentialManager get_value
python CredentialManager does_key_exist
python CredentialManager delete_key
```

Quickly modifying data (Option 3) --> Run from the terminal

```
python -c 'import CredentialManager; CredentialManager.save_key("myKey", "myValue")'
python -c 'import CredentialManager; CredentialManager.delete_key("gmailUsername")'
```

Note Taker

Takes random notes on a book. For more details, see [README.md](#)

Dictionary

I wrote this script because using a web browser to look up the definition of a word takes way too long on my computer (mid-2009 Macbook Pro)

This script prints out the definition of a word and usage of the word in a sentence

Dependencies: None

Usage:

```
python Dictionary.py perfunctory
```

Or, to prompt:

```
python Dictionary.py
```

iOS App Icon Resizer

Quickly resizes an image of any extension to 5 png images according to Apple's iOS icon requirements. Creates a directory in the directory it is run from called 'AppIcon' and saves each file there with the appropriate title ie. [iPhone6@3x.png](#) will be 180 x 180

```
#Can be run from any directory
python iOSAppIconResizer.py imageName.jpg
```

Text Phone

Quickly sends a text to myself from the command line

Dependencies: [CredentialManager.py](#)

Run Instructions:

```
python TextPhone.py message
```

Or, as a prompt:

```
python TextPhone.py
```

Text Phone Plain

Also quickly sends a text to myself from the command line. The main difference between this and 'TextPhone.py' is that it has no dependencies (it prompts for password via a secure input formatter)

Run Instructions (prompts for info):

```
python TextPhoneP.py
```

Text Link

Quickly sends a link to myself from the commandline If the link is too long (over 150 characters), shortens the link by using [po.st](#)

Dependencies: [CredentialManager.py](#)

Run Instructions:

```
python TextLink.py linkhere
```

Or, as a prompt:

```
python TextLink.py
```

Text People

Quickly sends a text to anyone from anyone from the commandline

Dependencies: [CredentialManager.py](#)

Run Instructions (prompts for info):

```
python TextPeople.py
```

Email Attachment

I wrote this Python script to quickly email files to myself because it takes way too long to do something that simple on my mid-2009 Macbook Pro. Gmail, when accessed via a web browser, uses way too much CPU processing power and lags for the first 20 seconds because of all the JavaScript it has to load. The Mail app uses way too much RAM to always be open, and when I do need to open it to quickly send myself something, the app opening/loading takes about 20 seconds. Those timings only include opening the app; going to 'Create new message', setting the 'To' to myself, clicking 'Attach File' and navigating to the file, uploading and then sending the file brings the total time to email myself a small attachment to about two minutes.

This Python script sends an attachment to myself in about 2 seconds and uses almost no additional RAM or CPU processing power. The script can be run from any directory and gets the file name (optional command line parameter) from the current working directory

Dependencies: [CredentialManager.py](#)

Run Instructions:

```
python EmailAttachment.py fileName
```

Or, as a prompt:

```
python EmailAttachment.py
```

