Hacking The Fender

`The Fender`, a notorious computer hacker and general villain of the people, has compromised several top-secret passwords including your own. Your mission, should you choose to accept it, is threefold. You must acquire access to `The Fender`'s systems, you must update his `"passwords.txt"` file to scramble the secret data. The last thing you need to do is add the signature of `Slash Null`, a different hacker whose nefarious deeds could be very conveniently halted by `The Fender` if they viewed `Slash Null` as a threat.

Use your knowledge of working with Python files to retrieve, manipulate, obscure, and create data in your quest for justice. Work with CSV files and other text files in this exploration of the strength of Python file programming.

If you get stuck during this project, check out the **project walkthrough video** which can be found in the help menu.

Tasks

19/19 Complete
Mark the tasks as complete by checking them off

Reading In The Passwords

1.
Are you there? We've opened up a communications link to `The Fender`'s secret computer. We need you to write a program that will read in the compromised usernames and passwords that are stored in a file called `"passwords.csv"`.

First import the CSV module, since we'll be needing it to parse the data.
Hint
You can import the CSV module using the following syntax:

```
import csv
```

2.
We need to create a list of users whose passwords have been compromised, create a new list and save it to the variable `compromised_users`.
Hint
Creating a new list for a variable called `new_list` with the following syntax:

```
new_list = []
```

3.

Next we'll need you to open up the file itself. Store it in a file object called `password_file`.

Hint

Use `with...as` syntax to open a file, remember it begins an indented block:

```
with open('filename.txt') as file_object:
    # pass
```

4.

Pass the `password_file` object holder to our CSV reader for parsing. Save the parsed `csv.DictReader` object as `password_csv`.

Hint

Use `csv.DictReader` to parse the file like so:

```
parsed_csv = csv.DictReader(file_object)
```

5.

Now we'll want to iterate through each of the lines in the CSV.

Create a for loop and save each row of the CSV into the temporary variable `password_row`.

6.

Inside your `for` loop, print out `password_row['Username']`. This is the username of the person whose password was compromised.

Run your code, do you see a list of usernames?

7.

Remove the `print` statement. We want to add each username to the list of `compromised_users`. Use the list's `.append()` method to add the username to `compromised_users` instead of printing them.

Hint

A list's `.append()` method adds it to the list.

```
my_list.append(my_dict['Key'])
```

8.

Exit out of your `with` block for `"passwords.csv"`. We have all the data we need from that file.

Start a new `with` block, opening a file called `compromised_users.txt`. Open this file in write-mode, saving the file object as `compromised_user_file`.

Hint

You can open a file in write-mode by passing `"w"` as an argument to `open()`.

```
with open('new_file.txt', 'w') as new_file:
    # write to file here
```

9.

Inside the new context-managed block opened by the `with` statement start a new `for` loop.

Iterate over each of your `compromised_users`.

10.

Write the username of each `compromised_user` in `compromised_users` to `compromised_user_file`.

Hint

Use the file object's `.write()` method:

```
write_me = "write me to file"
with open('file.txt', 'w') as file_obj:
  file_obj.write(write_me)
```

11.

Exit out of that `with` block. You're doing great so far! We've got the data we need to employ as insurance against `The Fender`.

Notifying the Boss

12.

Your boss needs to know that you were successful in retrieving that compromised data. We'll need to send him an encoded message over the internet. Let's use JSON to do that.

First we'll need to import the `json` module.

13.

Open a new JSON file in write-mode called `boss_message.json`. Save the file object to the variable `boss_message`.

14.

Create a Python dictionary object within your `with` statement that relays a boss message. Call this `boss_message_dict`.

Give it a `"recipient"` key with a value `"The Boss"`.

Also give it a `"message"` key with the value `"Mission Success"`.

Hint

Define a Python dictionary using the curly braces `{` and `}`:

```
new_dict = {
  'key1': 'val1',
  'key2': 'val2'
}
```

15.

Write out `boss_message_dict` to `boss_message` using `json.dump()`.
Hint

`json.dump()` takes two arguments: a Python data object (like a dictionary or an array) and a file object to write the resulting JSON to.

Pass the two parameters to `json.dump()` like so:

```
json.dump(dict_obj, file_obj)
```
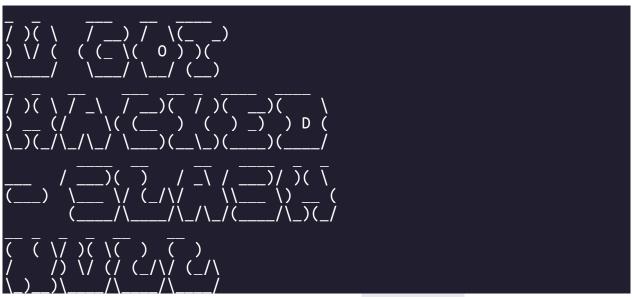
Scrambling the Password

16.

Now that we've safely recovered the compromised users we'll want to remove the `"passwords.csv"` file completely.

Create a new `with` block and open `"new_passwords.csv"` in write-mode. Save the file object to a variable called `new_passwords_obj`.

17.

Enemy of the people, `Slash Null`, is who we want `The Fender` to think was behind this attack. He has a signature, whenever he hacks someone he adds this signature to one of the files he touches. Here is the signature:

```
 __    __   __   ___
/ )( \  / _)/ \ (  _)
) \/ (  ( (_\( o ) )(
\____/   \__/ \_/ (__)

 __    __ __   __ __    ___   ___   ___
/ )( \ / _\ /  __)( / )( __)(    \
) __ (/    \( (__  ) (  ) _)  ) D (
\_)(_/\_/\_/ \___)(__\)(___)(____/

        __   __   __   __   __   __
 __    / _)( _) / _\ / _)/ )( \
(__)  \ __ \/ (_/\/ __/\(__)__(
       (___/\____/\_/\_/(____/\_)(_/

 __  __  __   __   __   __
(  (  \/ )( \(  ) (  )
/ __/ )  /) \/ (/ (_/\) (_/\
\__)_)\____/\____/\____/
```

Save that as a multiline string to the variable `slash_null_sig`.
Hint
Remember you can start a multiline string with three quotes:

```
multiline_str = """
 __   __   __   __
/ _)/  \ /  \ (  )
( (_( o )( o )/ (_/\
\___)\_/ \_/ \____/
"""
```

18.

Write `slash_null_sig` to `new_passwords_obj`. Now we have the file to replace `passwords.csv` with!

19.

What an incredible success! We'll take care of moving the new passwords file over the old one in case you want to practice hacking `The Fender` in the future.

Thank you for your service, programmer.