

Algorithm for file updates in Python

Project description

A file must be regularly updated to identify the employees who can access restricted content. This file contains information based on who is working with personal patient records. Restricted access is based on IP addresses. There are two lists: one is an allow list for IP addresses permitted to access the restricted subnetwork, and the other is a remove list that identifies IP addresses of employees who are not permitted. The algorithm will use Python to check IP addresses within the `allow_list.txt` and remove those identified in the remove list.

Open the file that contains the allow list

First, assign the file name as a string to a variable:

```
# assign 'import_file' to the name of the file
import_file = "allow_list.txt"
```

Then, use a `with` statement to open the file:

```
# Build 'with' statement to read in the initial contents of the file
with open(import_file, "r") as file:
```

Read the file contents

In order to read the contents of the file, the `.read()` method is used to convert it into a string.

```
# use '.read()' to read the imported file and store it in a variable
ip_addresses = file.read()
```

Convert the string into a list

Before removing individual IP addresses, the string needs to be converted into a list.

```
# use -split() to convert 'ip_addresses' from a string to a list
ip_addresses.split()
```

Iterate through the remove list

A for loop is used to iterate through the IP addresses that are elements in the `remove_list`.

```
# Build iterative statement
# Loop through 'remove_list'
```

```
for element in remove_list:
```

Remove IP addresses that are on the remove list

The algorithm will then remove any IP address from the `allow_list` that is also contained in the `remove_list`.

```
# Create a conditional statement to evaluate if 'element' is in
'ip_address'
    If element in ip_addresses:
        # use the '.remove()' to remove elements from 'ip_addresses'
        ip_addresses.remove(element)
```

Update the file with the revised list of IP addresses

Lastly, the algorithm will update the `allow_list` file with the revised list of IP addresses.

```
# Convert 'ip_addresses' back to a string
ip_addresses.join("\n")

# Build 'with' statement to rewrite the original file
with open(import_file, "w") as file:
    # rewrite the file
    file.write()
```

Summary

The algorithm uses `with` statements, `open()` functions, `read()` and `write()` methods, `split()` methods, `for` loops, and the `remove()` method to identify and remove IP addresses identified in a `remove_list` variable from the `allow_list.txt` file. These components provide the ability to work with files.