Algorithm for file updates in Python

Project description

[Describe the scenario in your own words.]
In this exercise I show how Python is used to remove specific IP addresses from an existing list of IP addresses.

Open the file that contains the allow list

[Add content here.]

```
# Assign `import_file` to the name of the file

import_file = "allow_list.txt"

# Assign `remove_list` to a list of IP addresses that are no longer allowed to access restricted information.

remove_list = ["192.168.97.225", "192.168.158.170", "192.168.201.40", "192.168.58.57"]

# First line of `with` statement

with open(import_file,"r") as file:
```

Read the file contents

[Add content here.]

```
# Assign `import_file` to the name of the file

import_file = "allow_list.txt"

# Assign `remove_list` to a list of IP addresses that are no longer allowed to access restricted information.

remove_list = ["192.168.97.225", "192.168.158.170", "192.168.201.40", "192.168.58.57"]

# Build `with` statement to read in the initial contents of the file

with open(import_file, "r") as file:

  # Use `.read()` to read the imported file and store it in a variable named `ip_addresses`
```

```
 ip_addresses = file.read()

# Display `ip_addresses`

print(ip_addresses)
```

```
ip_address 192.168.205.12 192.168.6.9 192.168.52.90 192.168.90.124 192.168.18
6.176 192.168.133.188 192.168.218.219 192.168.52.37 192.168.156.224 192.168.6
0.153 192.168.69.116
```

Convert the string into a list

[Add content here.]

```
# Assign `import_file` to the name of the file

import_file = "allow_list.txt"

# Assign `remove_list` to a list of IP addresses that are no longer allowed to access restricted information.

remove_list = ["192.168.97.225", "192.168.158.170", "192.168.201.40", "192.168.58.57"]

# Build `with` statement to read in the initial contents of the file

with open(import_file, "r") as file:

  # Use `.read()` to read the imported file and store it in a variable named `ip_addresses`

  ip_addresses = file.read()

# Use `.split()` to convert `ip_addresses` from a string to a list

ip_addresses = ip_addresses.split

# Display `ip_addresses.split()`

print(ip_addresses)
```

```
<built-in method split of str object at 0x7faf9eb4b8f0>
```

Iterate through the remove list

[Add content here.]

```
# Assign `import_file` to the name of the file

import_file = "allow_list.txt"

# Assign `remove_list` to a list of IP addresses that are no longer allowed to access restricted information.

remove_list = ["192.168.97.225", "192.168.158.170", "192.168.201.40", "192.168.58.57"]

# Build `with` statement to read in the initial contents of the file

with open(import_file, "r") as file:

 # Use `.read()` to read the imported file and store it in a variable named `ip_addresses`

 ip_addresses = file.read()

# Use `.split()` to convert `ip_addresses` from a string to a list

ip_addresses = ip_addresses.split()

# Build iterative statement
# Name loop variable `element`
# Loop through `ip_addresses`

for element in ip_addresses:
   # Display `element` in every iteration

   print(element)
```

```
ip_address
192.168.205.12
192.168.6.9
192.168.52.90
192.168.90.124
192.168.186.176
192.168.133.188
192.168.218.219
192.168.52.37
192.168.156.224
192.168.60.153
192.168.69.116
```

Remove IP addresses that are on the remove list

[Add content here.]

```
# Assign `import_file` to the name of the file

import_file = "allow_list.txt"

# Assign `remove_list` to a list of IP addresses that are no longer allowed to access restricted information.

remove_list = ["192.168.97.225", "192.168.158.170", "192.168.201.40", "192.168.58.57"]

# Build `with` statement to read in the initial contents of the file

with open(import_file, "r") as file:

 # Use `.read()` to read the imported file and store it in a variable named `ip_addresses`

 ip_addresses = file.read()

# Use `.split()` to convert `ip_addresses` from a string to a list

ip_addresses = ip_addresses.split()

# Build iterative statement
# Name loop variable `element`
# Loop through `ip_addresses`

for element in ip_addresses:

 # Build conditional statement
 # If current element is in `remove_list`,

  if element in remove_list:

    # then current element should be removed from `ip_addresses`

    ip.addresses.remove(element)

# Display `ip_addresses`

print(ip_addresses)
```

```
['ip_address', '192.168.205.12', '192.168.6.9', '192.168.52.90', '192.168.90.
124', '192.168.186.176', '192.168.133.188', '192.168.218.219', '192.168.52.37
', '192.168.156.224', '192.168.60.153', '192.168.69.116']
```

Update the file with the revised list of IP addresses

[Add content here.]

```
# Assign `import_file` to the name of the file

import_file = "allow_list.txt"

# Assign `remove_list` to a list of IP addresses that are no longer allowed to access restricted information.

remove_list = ["192.168.97.225", "192.168.158.170", "192.168.201.40", "192.168.58.57"]

# Build `with` statement to read in the initial contents of the file

with open(import_file, "r") as file:

 # Use `.read()` to read the imported file and store it in a variable named `ip_addresses`

 ip_addresses = file.read()

# Use `.split()` to convert `ip_addresses` from a string to a list

ip_addresses = ip_addresses.split()

# Build iterative statement
# Name loop variable `element`
# Loop through `ip_addresses`

for element in ip_addresses:

 # Build conditional statement
 # If current element is in `remove_list`,

  if element in remove_list:

    # then current element should be removed from `ip_addresses`

    ip_addresses.remove(element)

# Convert `ip_addresses` back to a string so that it can be written into the text file

ip_addresses = " ".join(ip_addresses)
```

```
# Build `with` statement to rewrite the original file

with open(import_file,"w") as file:

 # Rewrite the file, replacing its contents with `ip_addresses`

file.write(ip_addresses)
```

Summary

[Add content here.]
This exercise showed the necessary steps to:

1. retrieve a file
2. list its contents
3. reformat its contents
4. remove specific IP addresses
5. list the revised file