# Algorithm for file updates in Python

## Project description

In my organization, we manage access to restricted content with an allow list of IP addresses, detailed in the allow\_list.txt file. We also maintain a separate remove list to identify IP addresses that should no longer have access. I developed an algorithm to automate the process of updating the allow\_list.txt file and removing those IP addresses.

## Open the file that contains the allow list

The open() function in Python lets you access a file. First, you provide the file name (or in this case a variable holding the name). Then, you supply a string to specify how you want to handle the file. For reading the file, pass in "r" as the second parameter.



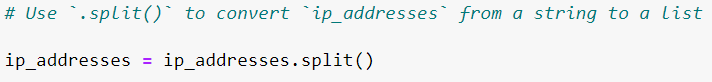
## Read the file contents

The .read() method in Python allows you to read a file. To do so, call file.read() on the opened file. Then, to display the contents, pass the variable holding the file’s contents as an argument to the print() function.



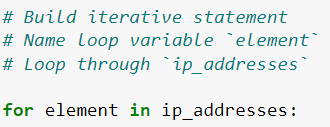
## Convert the string into a list

The .split() method in Python converts a string to a list. By default, it splits on whitespace, but you can specify a different character. In this case, the default behavior works well since each IP address in allow\_list.txt is on a new line, and .split() will separate them into a list.



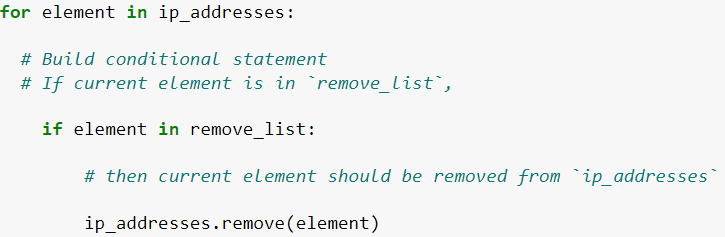
## Iterate through the remove list

I constructed a for loop to go through ip\_addresses, starting with the for keyword. I then use element as the loop variable and in for the loop condition.



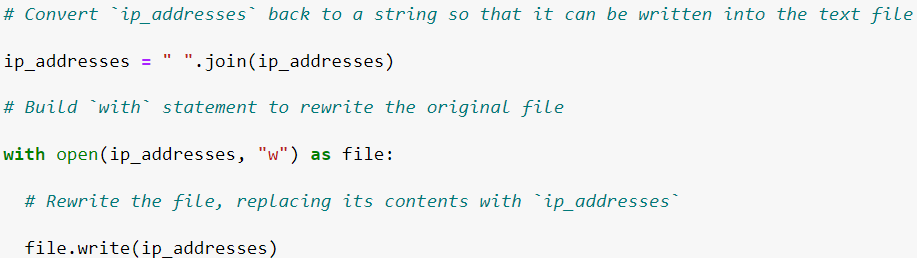
## Remove IP addresses that are on the remove list

To build the conditional statement, I use the in operator to check if element is in remove\_list. Then to remove element from ip\_addresses, I call the .remove() method on ip\_addresses and pass in element, ip\_addresses.remove(element).



## Update the file with the revised list of IP addresses

I start the with statement by calling open(), passing the name of the file and "w" to indicate writing. Inside the with statement, I'll use .write() to replace the file's contents with ip\_addresses, file.write(ip\_addresses).



## Summary

I developed an algorithm that removes IP addresses listed in the remove\_list variable from the allow\_list.txt file. The process involves opening the file, converting its contents to a string, and then transforming that string into a list stored in the ip\_addresses variable. I then iterate through each IP address in the remove\_list. For each address, I check if it exists in the ip\_addresses list. If it does, I use the .remove() method to delete it from ip\_addresses. Finally, I use the .join() method to convert ip\_addresses back into a string and overwrite the contents of allow\_list.txt with the updated list.