# Algorithm for file updates in Python

## Project description

A health care organization needs me to regularly update a file of employees who can access restricted access level content. The file includes employee IP addresses that currently have access to personal patient records, and I have been given a list of IP addresses to remove. In this project, I create a Python algorithm that checks whether the allow list contains any of the IP addresses to remove, and remove them.

## Open the file that contains the allow list

I first start by storing the allow\_list.txt file path to a variable that I can use to open it.

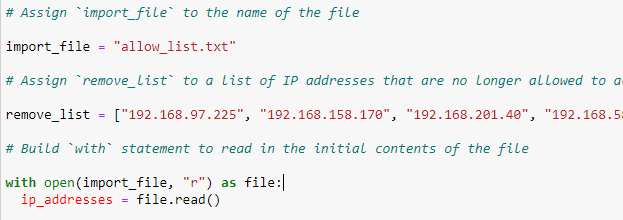


Then using the *with open() as file* syntax I open the file to read the currently allowed IP addresses listed in the file.



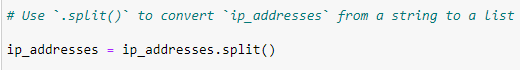
## Read the file contents

With the file open, I can now read the contents with the *.read()* method which converts the contents of the file into a python string.



## Convert the string into a list

Since I will need to iterate through the list of IP addresses to remove any unallowed IP addresses, I convert the file contents string into a Python list with *.split().*



## Iterate through the remove list

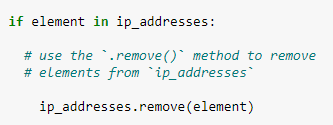
Now that all of the allowed IP addresses were easily available in a list, I iterated through all of the IP addresses I needed to remove so that I could check if they existed in the currently allowed IP addresses.





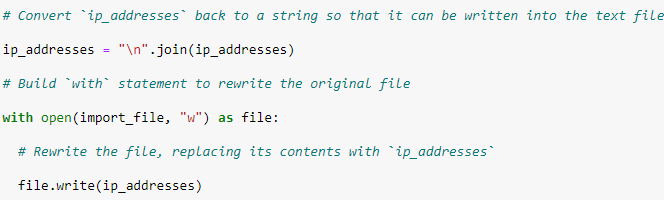
## Remove IP addresses that are on the remove list

While iterating through the IP addresses to remove, I created a conditional to remove an IP address from the list of allowed IP addresses if it was currently allowed but should be removed.



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

Now that I had the updated list of IP addresses, I needed to update the allow\_list.txt file with the updated list. I first converted the Python list back to a string where each IP address is on its own line. Then I opened the allow\_list.txt file with the *“w”* parameter to completely overwrite the file to avoid duplicate information. Finally, I wrote the updated string of allowed IP addresses to the file.



## Summary

For this task, I created a Python algorithm that takes in a file of IP addresses and uses a list of IP addresses to remove to update the file. Using Python functionalities like *with open*, *.read()*, and *.remove()* I convert the file to a Python list and iterate through the list of IP addresses to remove and compare them to the IP addresses currently listed in the file. After removing the invalid IP addresses, I convert the list back to a string to then update the valid IP addresses file with the new list.