Automation with Python

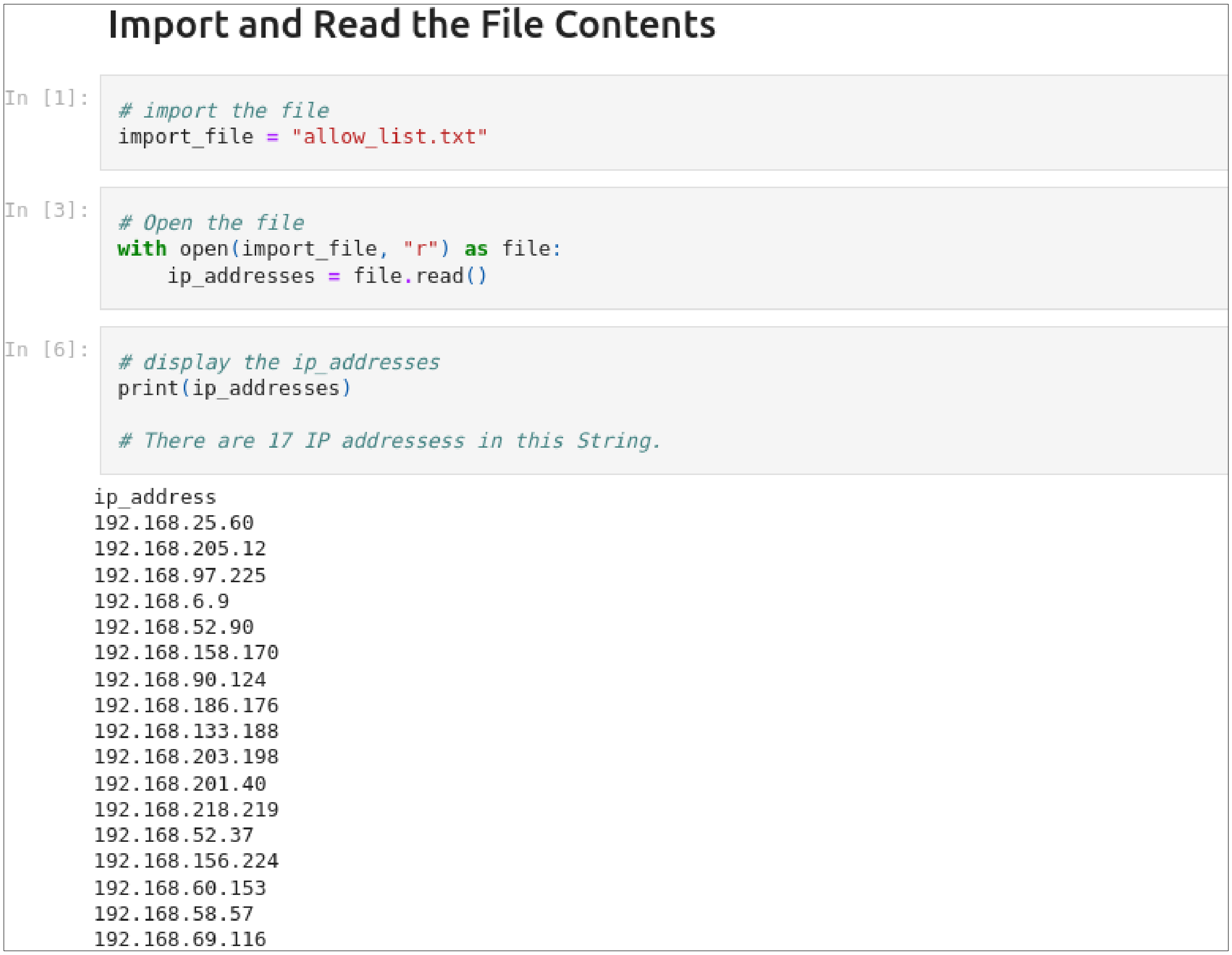
What is Python?

Python is that is used for many different tasks, including web development, software development, data science, and machine learning, using libraries such as numpy, pandas, matplotlib, seaborn, scikit-learn, scipy, and statsmodels.

Scenario

Regularly update a file that identifies the employees who can access restricted content. Employees are restricted access to files based on their IP address. There is an allow list for IP addresses permitted to sign into the restricted sub-network. There's also a remove list that identifies which employees you must remove from this allow list.

Question: Create an algorithm that checks whether the allow list contains any IP addresses identified on the remove list.



Import and Read the File Contents

In [1]:

```
# import the file
import_file = "allow_list.txt"
```

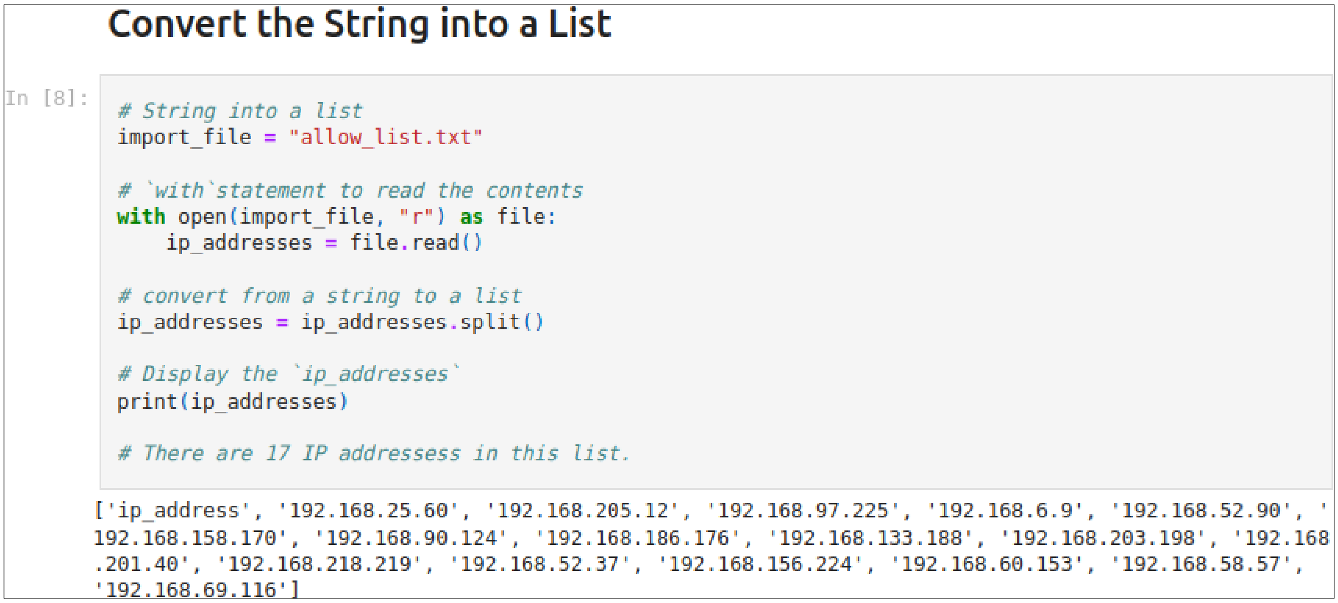
In [3]:

```
# Open the file
with open(import_file, "r") as file:
    ip_addresses = file.read()
```

In [6]:

```
# display the ip_addresses
print(ip_addresses)

# There are 17 IP addressess in this String.
```

```
ip_address
192.168.25.60
192.168.205.12
192.168.97.225
192.168.6.9
192.168.52.90
192.168.158.170
192.168.90.124
192.168.186.176
192.168.133.188
192.168.203.198
192.168.201.40
192.168.218.219
192.168.52.37
192.168.156.224
192.168.60.153
192.168.58.57
192.168.69.116
```



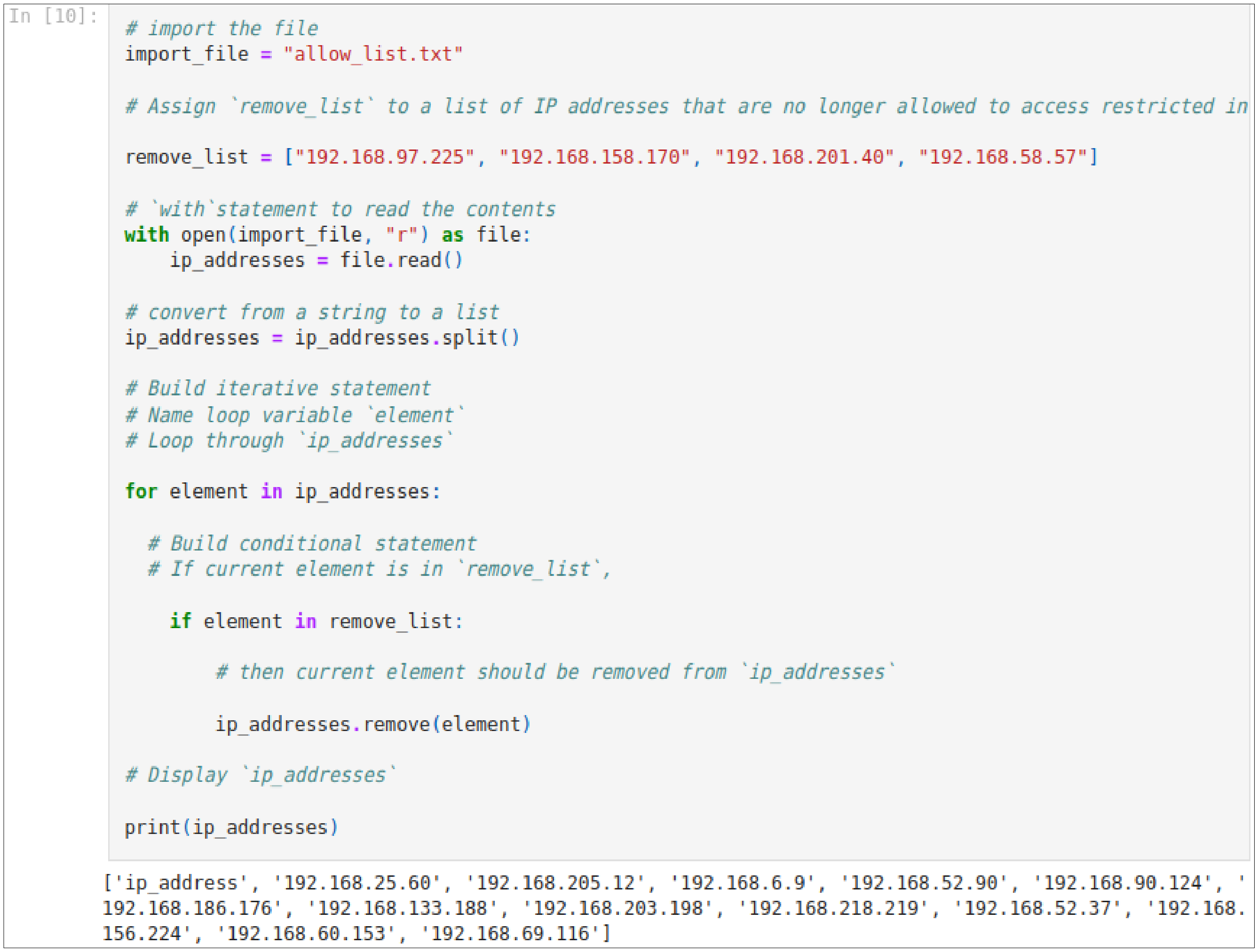
Convert the String into a List

In [8]:

```
# String into a list
import_file = "allow_list.txt"

# `with`statement to read the contents
with open(import_file, "r") as file:
    ip_addresses = file.read()

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

# Display the `ip_addresses`
print(ip_addresses)

# There are 17 IP addressess in this list.
```

```
['ip_address', '192.168.25.60', '192.168.205.12', '192.168.97.225', '192.168.6.9', '192.168.52.90', '
192.168.158.170', '192.168.90.124', '192.168.186.176', '192.168.133.188', '192.168.203.198', '192.168
.201.40', '192.168.218.219', '192.168.52.37', '192.168.156.224', '192.168.60.153', '192.168.58.57',
'192.168.69.116']
```

Remove IP Addresses That Are on the Remove List



In [10]:

```
# import the file
import_file = "allow_list.txt"

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

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

# `with`statement to read the contents
with open(import_file, "r") as file:
    ip_addresses = file.read()

# convert 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', '192.168.25.60', '192.168.205.12', '192.168.6.9', '192.168.52.90', '192.168.90.124', '
192.168.186.176', '192.168.133.188', '192.168.203.198', '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

In [11]:

```
import_file = "allow_list.txt"
remove_list = ["192.168.97.225", "192.168.158.170", "192.168.201.40", "192.168.58.57"]
with open(import_file, "r") as file:
  ip_addresses = file.read()
ip_addresses = ip_addresses.split()
for element in ip_addresses:
    if element in remove_list:
        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)
```

```
# Build `with` statement to read in the updated file
with open(import_file, "r") as file:
    # Read in the updated file and store the contents in `text`
    text = file.read()
# Display the contents of `text`
print(text)
```

```
ip_address 192.168.25.60 192.168.205.12 192.168.6.9 192.168.52.90 192.168.90.124 192.168.186.176 192.
168.133.188 192.168.203.198 192.168.218.219 192.168.52.37 192.168.156.224 192.168.60.153 192.168.69.1
16
```

In [ ]: