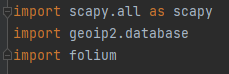
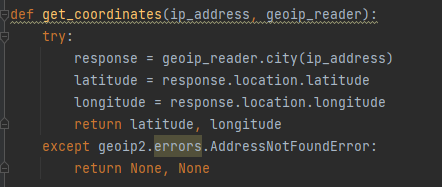
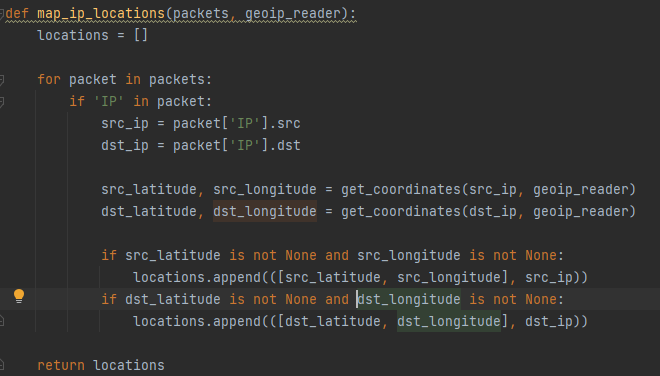
**CODE EXPLAINED**



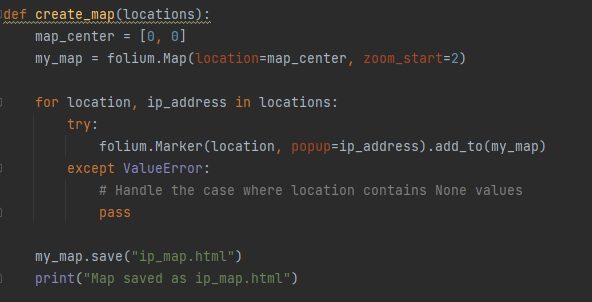
* **scapy.all**: Imports the Scapy library, a powerful interactive packet manipulation program.
* **geoip2.database**: Imports the GeoIP2 database module, which is used for IP geolocation.
* **folium**: Imports Folium, a Python wrapper for Leaflet.js, which is used for creating interactive maps.



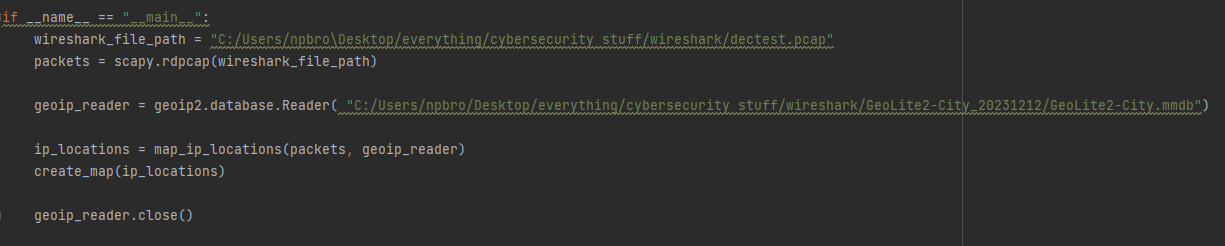
* This function takes an ip\_address and a geoip\_reader as parameters.
* It attempts to retrieve the geographical information (latitude and longitude) for the given ip\_address using the geoip\_reader.
* If successful, it returns a tuple containing the latitude and longitude.
* If the address is not found in the GeoIP database, it catches the AddressNotFoundError exception and returns None for both latitude and longitude.



* This function takes a list of **packets** and a **geoip\_reader** as parameters.
* It initializes an empty list called **locations** to store tuples containing coordinates and IP addresses.
* It iterates over each packet in the provided list of packets.
* For each packet, it checks if it contains an 'IP' layer.
* If the packet has an 'IP' layer, it extracts the source (**src\_ip**) and destination (**dst\_ip**) IP addresses.
* It calls the **get\_coordinates** function for both source and destination IPs to get their geographical coordinates.
* If coordinates are found for the source IP, it appends a tuple containing the coordinates and source IP to the **locations** list.
* If coordinates are found for the destination IP, it appends a similar tuple for the destination IP.
* Finally, it returns the list of locations.



* This function takes a list of **locations** as a parameter.
* It initializes a variable **map\_center** with the coordinates **[0, 0]** and creates a Folium map (**my\_map**) at that center with an initial zoom level of 2.
* It then iterates over each location in the **locations** list, where each location is a tuple containing coordinates and an IP address.
* For each location, it attempts to create a Folium marker at that location on the map, with a popup displaying the corresponding IP address.
* It catches the **ValueError** exception, which may occur if the location contains **None** values (latitude or longitude). It ignores such cases using the **pass** statement.
* After processing all locations, it saves the map to an HTML file named "ip\_map.html" using **my\_map.save("ip\_map.html")**.
* It prints a message indicating that the map has been saved.



* This block is the main entry point of the script and executes only if the script is run directly (not imported as a module).
* It defines the path to the Wireshark file (**wireshark\_file\_path**) and uses Scapy's **rdpcap** function to read the packets from the specified file.
* It creates a GeoIP reader (**geoip\_reader**) using the GeoLite2 database file.
* It calls the **map\_ip\_locations** function to obtain a list of IP locations from the packets and the GeoIP reader.
* Finally, it calls the **create\_map** function with the obtained locations and closes the GeoIP reader.