**IOS XE NETCONF YANG 2**

Directions: Complete each of the scripts below and submit them by the date specified in the assignment sheet and Blackboard. Submit **copies of your code** and **screenshots of the code running with each task**. Also, be sure to use the document, **Script Requirements as a guide** to writing good code. **Full credit will not be earned if you do not meet these script requirements.** **10 points**

1. From the turnipTheBeet repository, open the file, NETCONF-EDIT-RUN-INT-py and note the following:

Text

Description automatically generated

We are going to use a new interface model. The Cisco-IOS-XE-native model is an augmented model supported by Cisco. Note the highlighted attributes, <primary>. This allows us to enter an address as a primary address. We would have run into a problem with the ietf-interfaces model where each new address was added as a primary address, but former addresses were set as secondary addresses. If we tried to update an address on the same subnet, but different host address, it would fail because the interface can only have one address per subnet in it. The Cisco-IOS-XE-native model resolves this issue. The xmlInt string defined on lines 11 – 29 are in XML format.

Within the string are some variable values, %addr%, %intName%, %intNum%, and %mask%, all underlined or circled in red. Note that lines 36 – 39 simply use a string.replace() method to substitute values in the original string when we make updates. The above code uses the xmlInt string template to change a hard-coded address of 172.16.1.2 /24 on interface GigabitEthernet2. Note the string is passed on line 46 to the device, and the edit\_config method is called from the connection (m).

1. Be sure to save this code to your working folder and modify it to:
   1. Ask a user for an address and subnet mask to change GigabitEthernet2 to:
   2. Check the address and mask for validity in separate functions.
   3. Change the address in a third function
      1. Move the NETCONF commands to a new function
      2. Pass the device dictionary and the new xmlInt string to the function