Assignment-1: Creating topology and Flow Rules

- Create a custom topology using python scripting in Mininet (Any custom topology of your choice with minimum 4 switching devices and 11 hosts).
- Specify the link rate and link delay of each link randomly between 0 to 5 Mbps and 2 ms - 30 ms respectively.
- Assign IPs to each device and hosts so that they are all in same subnet.
- Don't use built-in commands in Mininet such as Tree, Linear, etc. to create standard topologies.
- Use ODL as the controller to connect your switches in mininet.

Output Expected:

1. All devices and hosts should ping each other.
2. You will be required to submit your python script for generating the topology, a snap shot of the topology from ODL and output of following commands from the Mininet console.
   - pingall
   - nodes
   - links
   - net
   - <host> ifconfig (for any one host in your network)
   - <switching device> ifconfig (for any one switching device in your network)
3. Write flow rule on a switch (say s2) using ODL to drop incoming packets from a particular host (say host h1) so that any host connected with s2 will not be able to ping h1 and vice versa.
   a) Achieve the same using the IP address of h1.
   b) Achieve the same using MAC address of h1.

Also familiarize yourself with commands like dpctl and ovs-ofctl. You can experiment with viewing/modifying flows in your devices etc. This will be useful in upcoming assignments.

References:

1. How to install mininet and ODL[1]: https://www.youtube.com/watch?v=K5E6_eik23k
2. How to add Flow Entry using ODL[2]: https://www.youtube.com/watch?v=RwJZXDpeL6o
3. Flow Examples ODL: https://docs.opendaylight.org/projects/openflowplugin/en/latest/users/flow-examples.html
4. Quick Mininet Walkthrough: http://mininet.org/walkthrough/
5. Mininet API Documentation: http://mininet.org/api/annotated.html

Finally, Practice Searching to satisfy your quest for knowledge!

[1] Pls do take a note of features to be installed in ODL!
[2] Flow entries can be added by **ReST APIs**, and also through **dpctl** & **ovs-ofctl** directly to the SDN devices!