* 1. **Add Flow, Update Flow**
     + **URI : http://{controller-ip}:8181/restconf/config/opendaylight-inventory:nodes/node/{node-name}/table/{table-id}/flow/{flow-id}**
     + **method : put**
     + **Request body**

{

""flow-node-inventory:flow": [

{

"table\_id":"integer",

"idle-timeout":"integer",

"out\_port":"integer",

"match":{

"layer-4-match":"object",

"tcp-flag-match":{

"tcp-flag":"integer"

}

"icmpv6-match":{

"icmpv6-type":"integer", # ICMP type

"icmpv6-code":"integer"

}

"flow-statistics":{

"duration":{

"second":"integer:,

"nanosecond":"integer"

}

"byte-count":"integer",

"packet-count":"integer"

}

"ethernet-match":{

"ethernet-type":{

"type":"integer" # Ethernet frame type

}

"ethernet-source":{

"mask":"string",

"address":"string" # Ethernet source address

}

"ethernet-destination":{

"mask":"string",

"address":"string" # Ethernet destination address

}

}

"tunnel":"{

"tunnel-mask":"integer",

"tunnel-id":""integer", # Metadata associated in the logical port

}

"ip-match":{

"ip-protocol":"integer", # IP protocol

"ip-ecn":"integer", # IP ECN (2 bits in ToS field)

"ip-dscp","integer", # IP DSCP (6 bits in ToS field).,

"ip-proto":"enum" # IP Proto (IPv4 or IPv6 Protocol Number)

}

"icmpv4-match":{

"icmpv4-type":integer", # ICMP type

"icmpv4-code":"integer"

}

"metadata":{

"metadata-mask":"integer",

"metadata":"integer"

}

"protocol-match-fields":{

"mpls-bos":"integer", # BoS bit in the first MPLS shim header

"mpls-tc":"integer", # TC in the first MPLS shim header

"mpls-label":"integer", # Label in the first MPLS shim header

"pbb":{

"pbb-mask":"integer",

"pbb-isid":"integer" # I-SID in the first PBB service instance tag

}

}

"vlan-match":"{

"vlan-pcp"integer", # VLAN priority.,

"vlan-id":{

"vlan-id-present":"boolean",

"vlan-id":"integer",

}

}

"extension-list":[

{

"extension":{ # A vendor has to augment this container

"nxm-nx-nsp":{

"value":"integer"

}

"nxm-nx-nsi":{

"nsi":"integer"

}

"nxm-nx-reg":{

"value":"integer",

"reg":"nxm-nx-reg"

}

"nxm-nx-arp-tha (object[(config)nxm-nx-arp-tha], optional),

(config)nxm-nx-arp-tha {

"mac-address (string, optional)

}

"nxm-of-eth-src (object[(config)nxm-of-eth-src], optional),

(config)nxm-of-eth-src {

"mac-address (string, optional)

}

"nxm-of-arp-tpa":"{ #  Prereqs: NXM\_OF\_ETH\_TYPE must match either 0x0806 or 0x8035

"ipv4-address ":"string"

}

"nxm-nx-arp-sha":{

"mac-address":"string"

}

"nxm-nx-tun-ipv4-dst":{

"ipv4-address":"string"

}

"nxm-of-eth-type":{

"value":"integer"

}

"nxm-of-arp-spa":{ # Prereqs: NXM\_OF\_ETH\_TYPE must match either 0x0806 or 0x8035.,

"ipv4-address (string, optional)

}

"nxm-of-eth-dst": {

"mac-address":"string"

}

"nxm-nx-tun-ipv4-src":"{

"ipv4-address":"string"

}

"nxm-of-arp-op":{ # Prereqs: NXM\_OF\_ETH\_TYPE must match either 0x0806 or 0x8035.,

"value":"integer"

}

"nxm-nx-tun-id":{

"value":"integer"

}

"dos-ekis":"string"

}

"extension-key":"extension-key"

}

]

"in-phy-port":"string",

"in-port":"string",

"layer-3-match":"object",

}

"out\_group":"integer",

"container-name":"string",

"flow-name":"string",

"strict":"boolean",

"cookie\_mask":"integer",

"barrier":"boolean",

"id":"string",

"installHw":"boolean",

"priority":"integer",

"flags":"undefined",

"hard-timeout":"integer",

"buffer\_id":"integer",

"cookie":"integer",

"instructions":[

{

"instruction":{

"order":"integer",

"instruction":"object"

}

}

]

}

]

}

* 1. **DELETE Flow**
     + **URI : http://{controller-ip}:8181/restconf/config/opendaylight-inventory:nodes/node/{node-name}/table/{table-id}/flow/{flow-id}**
     + **method : DELETE**
     + **body : 필요 없음**

* 1. **GET Flow**
     + **URI : http://{controller-ip}:8181/restconf/config/opendaylight-inventory:nodes/node/{node-name}/table/{table-id}/flow/{flow-id}**
     + **method : GET**
     + **Response : Add Flow와 같은 형식.**

* 1. **GET All Flow**
     + **URI : http://{controller-ip}:8181/restconf/config/opendaylight-inventory:nodes/node/{node-name}/table/{table-id}/**
     + **method : GET**
     + **Response : Table ID로 묶인 ADD Flow와 같은 형식의 배열**
     + **example Response Body :**

{

"flow-node-inventory:table": [

{

"id": 0,

"flow": [

{

"id": "1",

"table\_id": 0,

"instructions": {

"instruction": [

{

"order": 0,

"apply-actions": {

"action": [

{

"order": 0,

"dec-nw-ttl": {}

}

]

}

}

]

},

"match": {

"ipv4-destination": "10.0.155.2/24",

"ethernet-match": {

"ethernet-type": {

"type": 2048

}

}

},

"priority": 2,

"flow-name": "Foo"

}

]

}

]

}