JSON

PYTHON

```
"nodes": [
    {
        "role": "ACCESS",
        "name": "esp-e-2",
        "id": "f214463a-57ee-4782-a77d-d7896189c4af",
        "description": null,
        "deviceType": "Cisco Catalyst 9300 Switch",
        "platformId": "C9300-24T",
        …
    },
    …
]
```

```
for node in neighbors["nodes"]:
    my_nodes[node["id"]] = node["platformId"]
```

```
"links": [
    {
        "source": "f214463a-57ee-4782-a77d-d7896189c4af",
        "linkStatus": null,
        "sourceLinkStatus": "UP",
        "targetLinkStatus": "UP",
        "target": "ea62d532-cd20-426e-b308-5bb3b5fad545",
        "id": "10.2.253.1-GigabitEthernet2/0/7",
        "portUtilization": null,
        "sourceInterfaceName": "GigabitEthernet2/0/7",
        "targetInterfaceName": "GigabitEthernet1/0/7",
        …
    },
    …
]
```

```
for link in neighbors["links"]:
    my_link = {
        "source": link["source"],
        "source_name": link["sourceInterfaceName"],
        "target": link["target"],
        "target_name": link["targetInterfaceName"]
    }
    my_links.append(my_link)
```

JSON

```
"nodes": [
        {
                "role": "ACCESS",
                "name": "esp-e-2",
                "id": "f214463a-57ee-4782-a77d-d7896189c4af",
                "description": null,
                "deviceType": "Cisco Catalyst 9300 Switch",
                "platformId": "C9300-24T",
                ...
        },
        ...
]
```

```
"links": [
        {
                "source": "f214463a-57ee-4782-a77d-d7896189c4af",
                "linkStatus": null,
                "sourceLinkStatus": "UP",
                "targetLinkStatus": "UP",
                "target": "ea62d532-cd20-426e-b308-5bb3b5fad545",
                "id": "10.2.253.1-GigabitEthernet2/0/7",
                "portUtilization": null,
                "sourceInterfaceName": "GigabitEthernet2/0/7",
                "targetInterfaceName": "GigabitEthernet1/0/7",
                ...
        },
        ...
]
```

➔ The node ID is used as a key in "my_node" dictionary. By using the link's source and target ID as a key-value in "my_node", we can get the value (platform) that matches to that ID.

PYTHON

```python
for node in neighbors["nodes"]:
    my_nodes[node["id"]] = node["platformId"]


for link in neighbors["links"]:
    my_link = {
        "source": link["source"],
        "source_name": link["sourceInterfaceName"],
        "target": link["target"],
        "target_name": link["targetInterfaceName"]
    }
    my_links.append(my_link)


for link in my_links:
    print("{} {} --> {} {}".format(
        my_nodes[link["source"]],
        link["source_name"],
        my_nodes[link["target"]],
        link["target_name"]
    ))
```