350-901 Q&As

Developing Applications Using Cisco Core Platforms and APIs (DEVCOR)

Pass Cisco 350-901 Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

https://www.leads4pass.com/350-901.html

100% Passing Guarantee
100% Money Back Assurance

Following Questions and Answers are all new published by Cisco Official Exam Center

QUESTION 1

Meraki Dashboard API Response

Response Status Code200

Response Link Header

; rel-first,

; rel-next,

; rel-last Response Body

{

\\' name": \\II \\'

"serial": "Q2CV-V49B -RCMZ",

"mac": \\'\\'0c:8d:db:95:aa:39",

"networkid": "L-566327653141846927",

"model11 : "MV71",

"address": "430 E Cactus Ave .\nLas Vegas, NV 89183",

"lat": 36.00017,

"lng": -115.15302,

"notes": \\'\\'",

"tagsn: \\'",,

"lanip: "192.168.0.25",

\\' configurationUpdatedAt": "2019-08-08T02:15:36Z", \\' firmware11 : "ca.rnera-3-3011

},

{

"ncune": "Alex\\'s MR84 - 1"1

"serial": "Q2EK-2LYB-PCZP",

"mac": \\'eO: 55:3d:10:56:8a", "networkid": "L 566327653141846927",

"model": "MR84",

"address": \\'\\'11 ,

"lat": 39.9482993357826,

"lng": -82.9895675461739,

"notes": "",

"tags": \\' ",

"lanip: null,

"configurationVpdatedAt": "2018-02-03Tll:02:37Z",

"firmware11 : "Not running configured version\\'\\'

},

{

"na.rne11 : "Vegas Living Room MR84 11 ,

"serial": "Q2EK-3UBE-RRUY",

"mac": "e0:55:3d:10:5a:ca", "networkid": "L_566327653141846927" 1

"model": "MR84",

"address": "430 E Cactus Ave.\nLas vegas, NV 89183", "lat": 36.00015,

"lng": -115.15308,

"notes": \\' ", "tags": "11 "lanip: "192.168.0 .20",

"configurationVpdatedAt": "2018-09-29Tl2:23:21Z",

\\' firmware": "Not running configured version"

Refer to the exhibit.

```
import request
import json

meraki_api_key = "<api key>"
url =
"https://api.meraki.com/api/v0/organizations/12345567890/devices"
headers = {
          "X-Cisco-Meraki-API-Key": meraki_api_key,
     }
params = {
     "perPage": 3
}
res = requests.get(url, headers=headers, params=params)
formatted_message = """
Meraki Dasboard API Response
-----------------------------
Response Status Code  : {}
Response Link Header  : {}
Response Body         : {}
-----------------------------
""".format(res.status_code, res.headers.get('Link'),
json.dumps(res.json(), indent=4))
print(formatted_message)
```

```
<https://n6.meraki.com/api/v0/organizations/1234567890/devices?perPage=
3&startingAfter=0000-0000-0000>; rel=first,
<https://n6.meraki.com/api/v0/organizations/1234567890/devices?perPage=
3&startingAfter=Q2EK-3UBE-RRUY>; rel=next,
<https://n6.meraki.com/api/v0/organizations/1234567890/devices?
endingBefore=zzzz-zzzz-zzzz&perPage=3>; rel=last
```

Which line of code must be added to this code snippet to allow an application to pull the next set of paginated items?

A. requests.get(url, links=[`next\\'][`url\\'])

B. requests.get(url, headers=links[`next\\'][`url\\'])

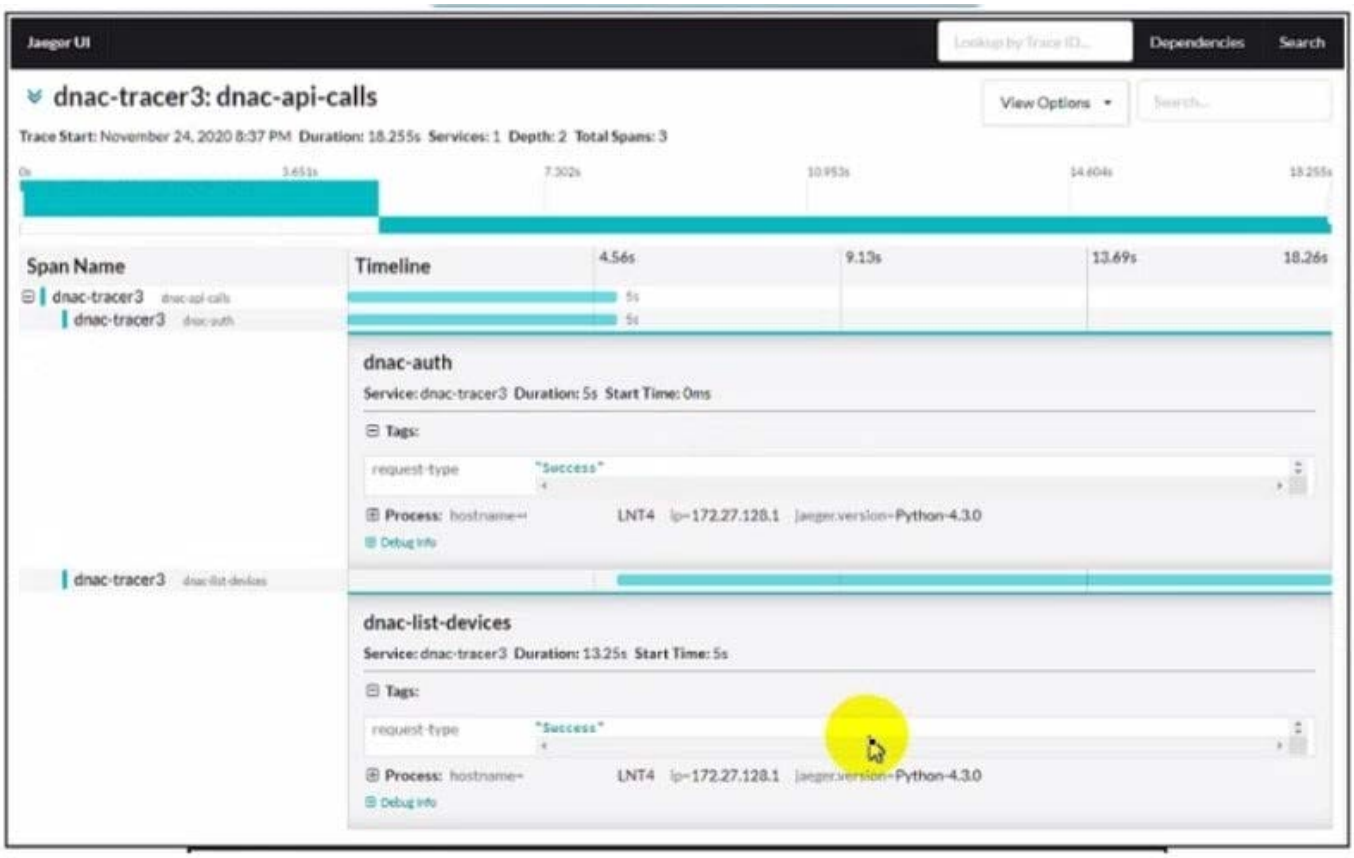
C. requests.get(res.links[`next\\'][`url\\'], headers=headers)

D. requests.get(res.headers.get(`Link")[`next\\'][`url\\'], headers=headers)

Correct Answer: C

QUESTION 2

Refer to the exhibit An application is developed to perform multiple API calls. The calls will be performed on the infrastructure devices Delays in the information transfer occur when the application is executed. What are two reasons for the issue? (Choose two)




```
def init_tracer(service):
    logging.getLogger('').handlers = []
    logging.basicConfig(format='%(message)s', level=logging.DEBUG)
    config = Config(
        config={'sampler': {'type': 'const', 'param': 1,},'logging': True,},
        service_name=service,)
    return config.initialize_tracer()

tracer = init_tracer('dnac-tracer')
base_url = 'https://sandboxdnac.cisco.com/'

with tracer.start_span('dnac-api-calls') as span:
    with tracer.start_span('dnac-auth', child_of=span) as site_span:
        try:
            dnac = DNACenterAPI(username='devnetuser', password='Cisco123!',
                                base_url=base_url, version='1.3.3',
                                verify=False)
            print('auth passed')
            site_span.set_tag('request-type', 'Success')
        except Exception as e:
            print('failed')
            site_span.set_tag('request-type', e)

with tracer.start_span('dnac-list-devices', child_of=span) as site_span:
    try:
        devices = [dnac.devices.get_device_list() for device in devices]
        print(devices)
        site_span.set_tag('request-type', 'Success')
    except Exception as e:
        print('Failed to list devices')
        site_span.set_tag('request-type', e)
```

A. The list devices API call is failing and does not return a result

B. Listing devices lakes longer than usual due to high network latency

C. One of the API calls takes roughly three times as long to complete

D. The list devices API call is inefficient and should be refactored

E. The requests are being rate limited to prevent multiple calls causing the excessive load

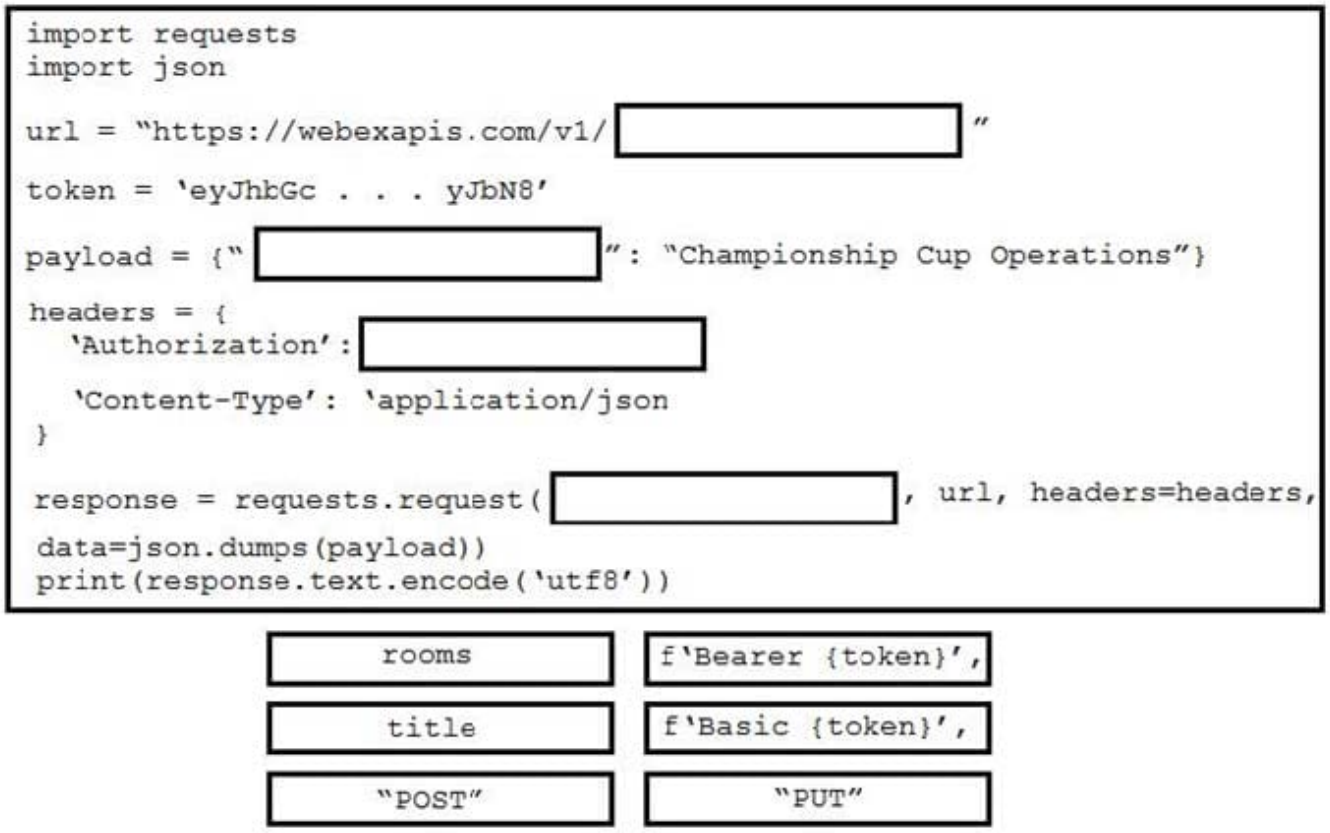
Correct Answer: BC

QUESTION 3

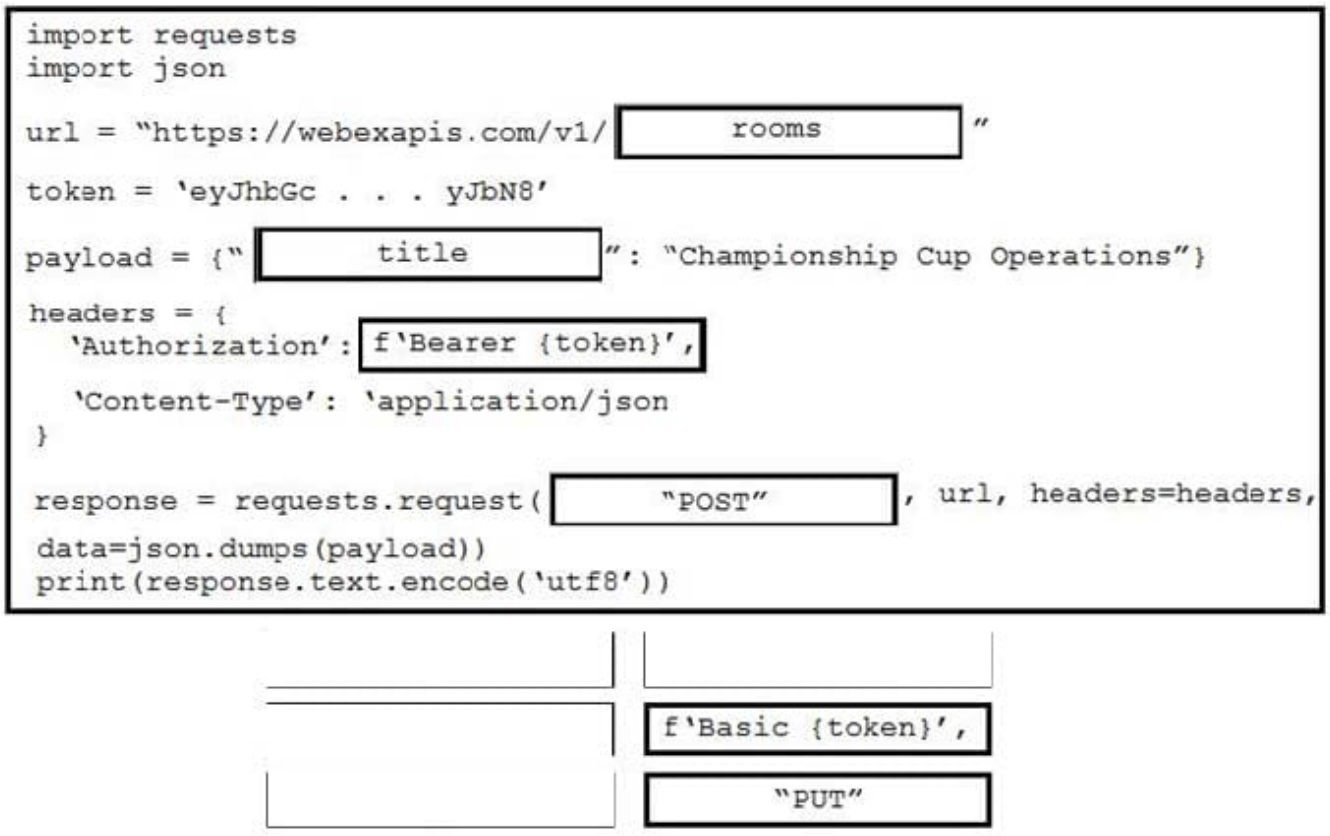
DRAG DROP

Drag and drop the code from the bottom onto the box where the code is missing to create a Webex space. Not all options are used.

Select and Place:



Correct Answer:



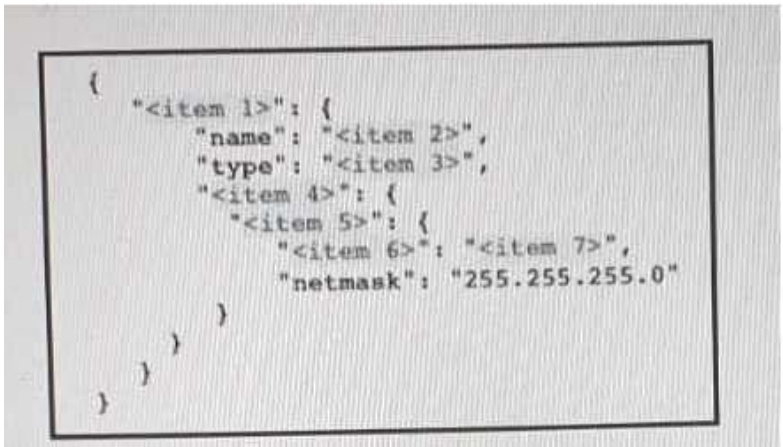
QUESTION 4

DRAG DROP

Refer to the exhibit above and click on the resource tabs in the top left corner to view an IETF TANG MODEL and a Python file that changes the configuration via RESTCONF. Drag and drop the code snippets from the left onto the item numbers on the right that match the missing sections in the exhibit to complete the JSON file that changes configuration of interface GigabitEthernet1 to have an IPv4 configuration of 10.10.0.1/24. Not all options are used.

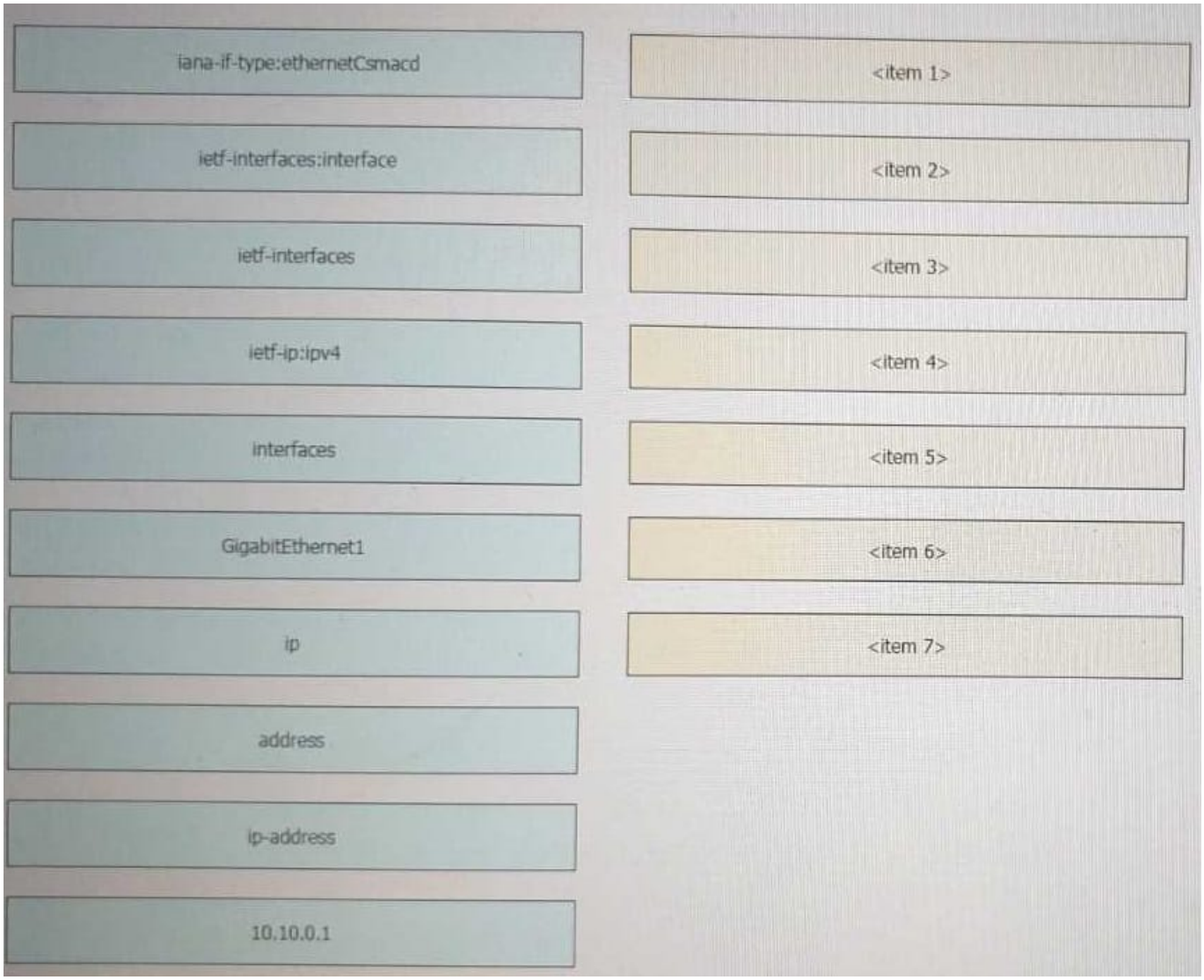
```
HOST = 'xehost1.exam.local'
PORT = '443'
USER = 'admin'
PASS = 'samplepassword'
INTERFACE = 'GigabitEthernet1'

url_base = "https://{h}:{p}/restconf".format(h=HOST, p=PORT)

url = url_base + '/data/ietf-
interfaces:interfaces/interface={i}".format(i=INTERFACE)

with open('./interface.json') as f:
  data = json.load(f)

response = requests.put(url,
                        auth=(USER, PASS),
                        headers=headers,
                        verify=False,
                        json=data
                        )
print(response.text)
```

```
RFC 8344                    YANG IP Management                  March 2018
2.  IP Data Model
This document defines the YANG module "ietf-ip", which augments the
"interface" lists defined in the "ietf-interfaces" module [RFC8343]
with IP-specific data nodes.

The data model has the following structure for IP data nodes per
interface, excluding the deprecated data nodes:

module: ietf-ip
  augment /if:interfaces/if:interface:
    +--rw ipv4!
    |  +--rw enabled?       boolean
    |  +--rw forwarding?    boolean
    |  +--rw mtu?           uint16
    |  +--rw address* [ip]
    |  |  +--rw ip                  inet:ipv4-address-no-zone
    |  |  +--rw (subnet)
    |  |  |  +--:(prefix-length)
    |  |  |  |  +--rw prefix-length?   uint8
    |  |  |  +--:(netmask)
    |  |  |     +--rw netmask?          yang:dotted-quad
    |  |  |                {ipv4-non-contiguous-netmasks}?
    |  |  +--ro origin?             ip-address-origin
    |  +--rw neighbor* [ip]
    |     +--rw ip                     inet:ipv4-address-no-zone
    |     +--rw link-layer-address     yang:phys-address
    |     +--ro origin?                neighbor-origin
    +--rw ipv6!
       +--rw enabled?                        boolean
       +--rw forwarding?                     boolean
       +--rw mtu?                            uint32
       +--rw address* [ip]
       |  +--rw ip                  inet:ipv6-address-no-zone
       |  +--rw prefix-length       uint8
       |  +--ro origin?             ip-address-origin
       |  +--ro status?             enumeration
       +--rw neighbor* [ip]
       |  +--rw ip                     inet:ipv6-address-no-zone
       |  +--rw link-layer-address     yang:phys-address
       |  +--ro origin?                neighbor-origin
       |  +--ro is-router?             empty
       |  +--ro state?                 enumeration
       +--rw dup-addr-detect-transmits?   uint32
```

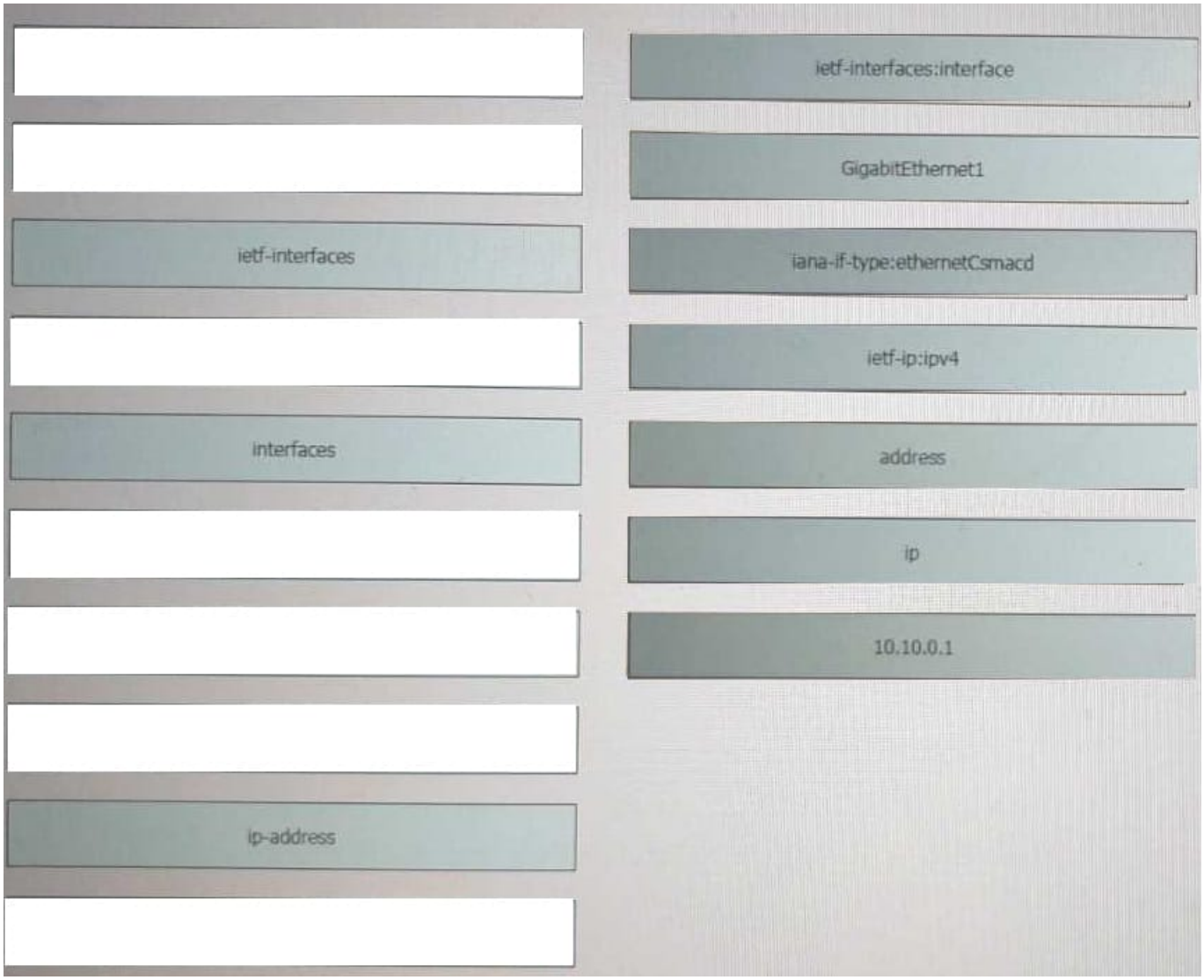


```
{
  "<item 1>": {
    "name": "<item 2>",
    "type": "<item 3>",
    "<item 4>": {
      "<item 5>": {
        "<item 6>": "<item 7>",
        "netmask": "255.255.255.0"
      }
    }
  }
}
```

Select and Place:



Correct Answer:



QUESTION 5

Which command is used 10 enable application hosting on a Cisco IOS XE device?

A. iox

B. iox-service

C. application -honing

D. app- hosting

Correct Answer: A