

Задания

Все задания и вспомогательные файлы можно скачать в [репозитории](#).

Предупреждение: Начиная с раздела «4. Типы данных в Python» для проверки заданий есть автоматические тесты. Они помогают проверить все ли соответствует поставленной задаче, а также дают обратный отклик по тому, что не соответствует задаче. Как правило, после первого периода адаптации к тестам, становится проще делать задания с тестами. Проверка заданий выполняется с помощью утилиты `runeng`. Подробнее [о том как работать с утилитой `runeng`](#).

Задание 5.1

В задании создан словарь, с информацией о разных устройствах.

Необходимо запросить у пользователя ввод имени устройства (`r1`, `r2` или `sw1`). И вывести информацию о соответствующем устройстве на стандартный поток вывода (информация будет в виде словаря).

Пример выполнения скрипта:

```
$ python task_5_1.py
Введите имя устройства: r1
{'location': '21 New Globe Walk', 'vendor': 'Cisco', 'model': '4451', 'ios': '15.4', 'ip':
 ↪': '10.255.0.1'}
```

Ограничение: нельзя изменять словарь `london_co`.

Все задания надо выполнять используя только пройденные темы. То есть эту задачу можно решить без использования условия `if`.

```
london_co = {
    "r1": {
        "location": "21 New Globe Walk",
        "vendor": "Cisco",
        "model": "4451",
        "ios": "15.4",
        "ip": "10.255.0.1"
    },
    "r2": {
        "location": "21 New Globe Walk",
        "vendor": "Cisco",
        "model": "4451",
        "ios": "15.4",
        "ip": "10.255.0.2"
    }
}
```

(continues on next page)

(продолжение с предыдущей страницы)

```

    },
    "sw1": {
        "location": "21 New Globe Walk",
        "vendor": "Cisco",
        "model": "3850",
        "ios": "3.6.XE",
        "ip": "10.255.0.101",
        "vlans": "10,20,30",
        "routing": True
    }
}

```

Задание 5.1а

Переделать скрипт из задания 5.1 таким образом, чтобы, кроме имени устройства, запрашивался также параметр устройства, который нужно отобразить.

Вывести информацию о соответствующем параметре, указанного устройства.

Пример выполнения скрипта:

```
$ python task_5_1a.py
Введите имя устройства: r1
Введите имя параметра: ios
15.4
```

Ограничение: нельзя изменять словарь london_co.

Все задания надо выполнять используя только пройденные темы. То есть эту задачу можно решить без использования условия if.

```

london_co = {
    "r1": {
        "location": "21 New Globe Walk",
        "vendor": "Cisco",
        "model": "4451",
        "ios": "15.4",
        "ip": "10.255.0.1"
    },
    "r2": {
        "location": "21 New Globe Walk",
        "vendor": "Cisco",
        "model": "4451",
        "ios": "15.4",
        "ip": "10.255.0.2"
    }
}
```

(continues on next page)

(продолжение с предыдущей страницы)

```
},
"sw1": {
    "location": "21 New Globe Walk",
    "vendor": "Cisco",
    "model": "3850",
    "ios": "3.6.XE",
    "ip": "10.255.0.101",
    "vlans": "10,20,30",
    "routing": True
}
}
```

Задание 5.1b

Переделать скрипт из задания 5.1а таким образом, чтобы, при запросе параметра, отображался список возможных параметров. Список параметров надо получить из словаря, а не прописывать вручную.

Вывести информацию о соответствующем параметре, указанного устройства.

Пример выполнения скрипта:

```
$ python task_5_1b.py
Введите имя устройства: r1
Введите имя параметра (location, vendor, model, ios, ip): ip
10.255.0.1

$ python task_5_1b.py
Введите имя устройства: sw1
Введите имя параметра (location, vendor, model, ios, ip, vlans, routing): ip
10.255.0.101
```

Ограничение: нельзя изменять словарь london_co.

Все задания надо выполнять используя только пройденные темы. То есть эту задачу можно решить без использования условия if.

```
london_co = {
    "r1": {
        "location": "21 New Globe Walk",
        "vendor": "Cisco",
        "model": "4451",
        "ios": "15.4",
        "ip": "10.255.0.1"
    },
}
```

(continues on next page)

(продолжение с предыдущей страницы)

```

"r2": {
    "location": "21 New Globe Walk",
    "vendor": "Cisco",
    "model": "4451",
    "ios": "15.4",
    "ip": "10.255.0.2"
},
"sw1": {
    "location": "21 New Globe Walk",
    "vendor": "Cisco",
    "model": "3850",
    "ios": "3.6.XE",
    "ip": "10.255.0.101",
    "vlans": "10,20,30",
    "routing": True
}
}

```

Задание 5.1с

Переделать скрипт из задания 5.1b таким образом, чтобы, при запросе параметра, которого нет в словаре устройства, отображалось сообщение „Такого параметра нет“. Задание относится только к параметрам устройств, не к самим устройствам.

Примечание: Попробуйте набрать неправильное имя параметра или несуществующий параметр, чтобы увидеть какой будет результат. А затем выполняйте задание.

Если выбран существующий параметр, вывести информацию о соответствующем параметре, указанного устройства.

Пример выполнения скрипта:

```

$ python task_5_1c.py
Введите имя устройства: r1
Введите имя параметра (ios, model, vendor, location, ip): ips
Такого параметра нет

```

Ограничение: нельзя изменять словарь london_co.

Все задания надо выполнять используя только пройденные темы. То есть эту задачу можно решить без использования условия if.

```
london_co = {
    "r1": {
        "location": "21 New Globe Walk",
        "vendor": "Cisco",
        "model": "4451",
        "ios": "15.4",
        "ip": "10.255.0.1"
    },
    "r2": {
        "location": "21 New Globe Walk",
        "vendor": "Cisco",
        "model": "4451",
        "ios": "15.4",
        "ip": "10.255.0.2"
    },
    "sw1": {
        "location": "21 New Globe Walk",
        "vendor": "Cisco",
        "model": "3850",
        "ios": "3.6.XE",
        "ip": "10.255.0.101",
        "vlans": "10,20,30",
        "routing": True
    }
}
```

Задание 5.1d

Переделать скрипт из задания 5.1с таким образом, чтобы, при запросе параметра, пользователь мог вводить название параметра в любом регистре.

Пример выполнения скрипта:

```
$ python task_5_1d.py
Введите имя устройства: r1
Введите имя параметра (ios, model, vendor, location, ip): IOS
15.4
```

Ограничение: нельзя изменять словарь london_co.

Все задания надо выполнять используя только пройденные темы. То есть эту задачу можно решить без использования условия if.

```
london_co = {
    "r1": {
        "location": "21 New Globe Walk",
```

(continues on next page)

(продолжение с предыдущей страницы)

```

    "vendor": "Cisco",
    "model": "4451",
    "ios": "15.4",
    "ip": "10.255.0.1"
},
"r2": {
    "location": "21 New Globe Walk",
    "vendor": "Cisco",
    "model": "4451",
    "ios": "15.4",
    "ip": "10.255.0.2"
},
"sw1": {
    "location": "21 New Globe Walk",
    "vendor": "Cisco",
    "model": "3850",
    "ios": "3.6.XE",
    "ip": "10.255.0.101",
    "vlans": "10,20,30",
    "routing": True
}
}

```

Задание 5.2

Запросить у пользователя ввод IP-сети в формате: 10.1.1.0/24

Затем вывести информацию о сети и маске в таком формате:

```

Network:
10      1      1      0
00001010  00000001  00000001  00000000

Mask:
/24
255      255      255      0
11111111  11111111  11111111  00000000

```

Проверить работу скрипта на разных комбинациях сеть/маска.

Подсказка: Получить маску в двоичном формате можно так:

```

In [1]: "1" * 28 + "0" * 4
Out[1]: "111111111111111111111111110000"

```

Ограничение: Все задания надо выполнять используя только пройденные темы.

Задание 5.2а

Всё, как в задании 5.2, но, если пользователь ввел адрес хоста, а не адрес сети, надо преобразовать адрес хоста в адрес сети и вывести адрес сети и маску, как в задании 5.2.

Пример адреса сети (все биты хостовой части равны нулю):

- 10.0.1.0/24
- 190.1.0.0/16

Пример адреса хоста:

- 10.0.1.1/24 - хост из сети 10.0.1.0/24
- 10.0.5.195/28 - хост из сети 10.0.5.192/28

Если пользователь ввел адрес 10.0.1.1/24, вывод должен быть таким:

```
Network:  
10      0      1      0  
00001010  00000000  00000001  00000000  
  
Mask:  
/24  
255      255      255      0  
11111111  11111111  11111111  00000000
```

Проверить работу скрипта на разных комбинациях хост/маска, например: 10.0.5.195/28, 10.0.1.1/24

Подсказка:

Есть адрес хоста в двоичном формате и маска сети 28. Адрес сети это первые 28 бит адреса хоста + 4 ноля. То есть, например, адрес хоста 10.1.1.195/28 в двоичном формате будет `bin_ip = "00001010000000010000000111000011"`.

А адрес сети будет первых 28 символов из `bin_ip + 0000` (4 потому что всего в адресе может быть 32 бита, а $32 - 28 = 4$): `00001010000000010000000111000000`

Ограничение: Все задания надо выполнять используя только пройденные темы.

Задание 5.3

Скрипт должен запрашивать у пользователя:

- информацию о режиме интерфейса (access/trunk)
- номере интерфейса (тип и номер, вида Gi0/3)
- номер VLANа (для режима trunk будет вводиться список VLANов)

В зависимости от выбранного режима, на стандартный поток вывода, должна возвращаться соответствующая конфигурация access или trunk (шаблоны команд находятся в списках `access_template` и `trunk_template`).

При этом, сначала должна идти строка `interface` и подставлен номер интерфейса, а затем соответствующий шаблон, в который подставлен номер VLANa (или список VLANов).

Ограничение: Все задания надо выполнять используя только пройденные темы. То есть эту задачу можно решить без использования условия `if` и циклов `for/while`.

Подсказка: Подводящим к этому заданию было задание 5.1. Чтобы было легче решить это задание, можно посмотреть на задание 5.1 и разобраться как там получилось вывести разную информацию в зависимости от ввода пользователя.

Ниже примеры выполнения скрипта, чтобы было проще понять задачу.

Пример выполнения скрипта, при выборе режима `access`:

```
$ python task_5_3.py
Введите режим работы интерфейса (access/trunk): access
Введите тип и номер интерфейса: Fa0/6
Введите номер влан(ов): 3

interface Fa0/6
switchport mode access
switchport access vlan 3
switchport nonegotiate
spanning-tree portfast
spanning-tree bpduguard enable
```

Пример выполнения скрипта, при выборе режима `trunk`:

```
$ python task_5_3.py
Введите режим работы интерфейса (access/trunk): trunk
Введите тип и номер интерфейса: Fa0/7
Введите номер влан(ов): 2,3,4,5

interface Fa0/7
switchport trunk encapsulation dot1q
switchport mode trunk
switchport trunk allowed vlan 2,3,4,5
```

```
access_template = [
    "switchport mode access", "switchport access vlan {}",
    "switchport nonegotiate", "spanning-tree portfast",
    "spanning-tree bpduguard enable"
]
```

(continues on next page)

(продолжение с предыдущей страницы)

```
trunk_template = [
    "switchport trunk encapsulation dot1q", "switchport mode trunk",
    "switchport trunk allowed vlan {}"
]
```

Задание 5.3а

Дополнить скрипт из задания 5.3 таким образом, чтобы, в зависимости от выбранного режима, задавались разные вопросы в запросе о номере VLANа или списка VLANов:

- для access: «Введите номер VLAN:»
- для trunk: «Введите разрешенные VLANы:»

Ограничение: Все задания надо выполнять используя только пройденные темы. То есть эту задачу можно решить без использования условия if и циклов for/while.

```
access_template = [
    "switchport mode access", "switchport access vlan {}",
    "switchport nonegotiate", "spanning-tree portfast",
    "spanning-tree bpduguard enable"
]

trunk_template = [
    "switchport trunk encapsulation dot1q", "switchport mode trunk",
    "switchport trunk allowed vlan {}"
]
```