МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ФАКУЛЬТЕТ ПРИКЛАДНОЙ МАТЕМАТИКИ И ИНФОРМАТИКИ

СЕРГИЕНКО ЛЕВ ЭДУАРДОВИЧ

Отчет по
ЛАБОРАТОРНАЯ РАБОТА 5
ПО ДИСЦИПЛИНЕ
«Непрерывное интегрирование и сборка программного обеспечения»

Проектирование модели управления версиями ПО, конфигурациями и непрерывной поставкой

Преподаватель

Давидовская М.И.
Филиппов М.А.

2025

1. Цель работы.

Изучение решений для автоматизации управления инфраструктурой и установки приложений.

2. Вариант задания.

3. Код приложений, конфигурационных файлов.

Задание 1. Основы управления конфигурацией инфраструктуры с Terraform

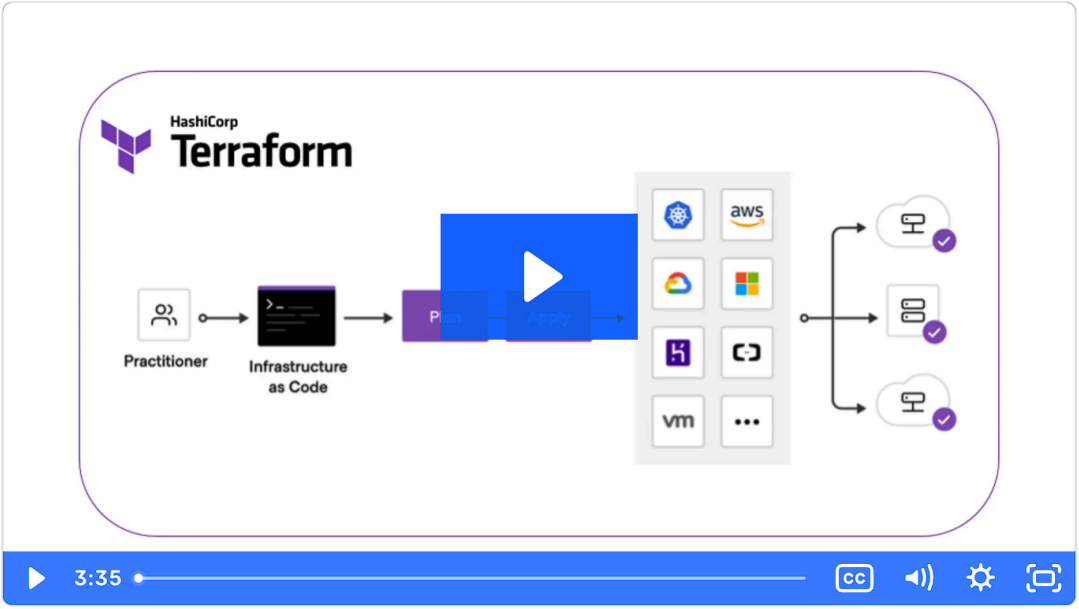
Руководство 1

Developer / Terraform / Tutorials / Docker / Infrastructure as Code

What is Infrastructure as Code with Terraform?

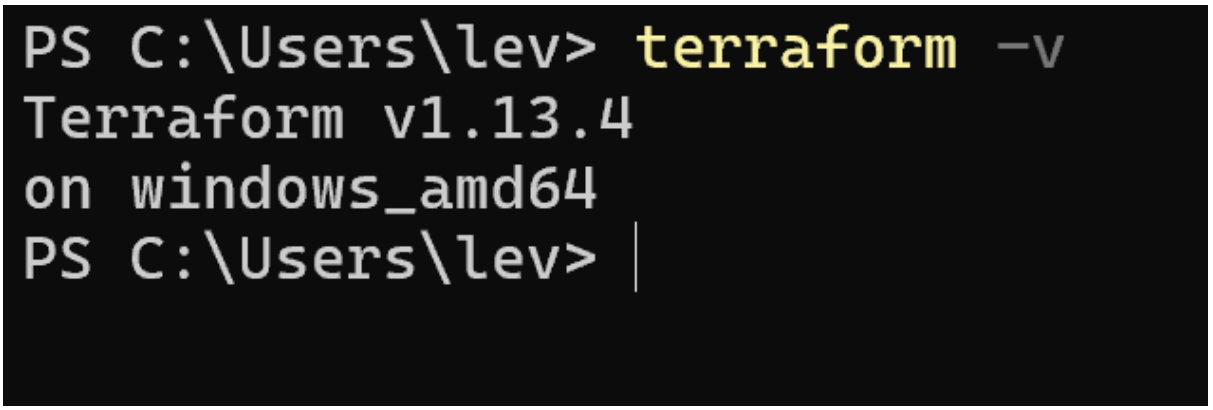
3min | Terraform Video Interactive

Reference this often? Create an account to bookmark tutorials.



Просмотрел

Руководство 2



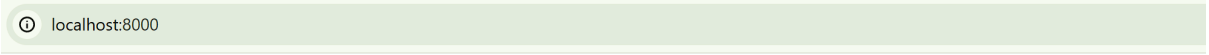
установил!

Руководство 3

```
PS C:\Users\lev\Desktop\devops\devops-gr12b-lab5-foreverNP\1\learn-terraform-docker-container> terraform apply

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
  + create

Terraform will perform the following actions:

  # docker_container.nginx will be created
  + resource "docker_container" "nginx" {
      + attach                                      = false
      + bridge                                      = (known after apply)
      + command                                     = (known after apply)
      + container_logs                              = (known after apply)
      + container_read_refresh_timeout_milliseconds = 15000
      + entrypoint                                  = (known after apply)
      + env                                         = (known after apply)
      + exit_code                                   = (known after apply)
      + hostname                                    = (known after apply)
      + id                                          = (known after apply)
      + image                                       = (known after apply)
      + init                                        = (known after apply)
      + ipc_mode                                    = (known after apply)
      + log_driver                                  = (known after apply)
      + logs                                        = false
      + must_run                                    = true
      + name                                        = "tutorial"
      + network_data                                = (known after apply)
      + read_only                                   = false
      + remove_volumes                              = true
      + restart                                     = "no"
      + rm                                          = false
      + runtime                                     = (known after apply)
      + security_opts                               = (known after apply)
      + shm_size                                    = (known after apply)
      + start                                       = true
      + stdin_open                                  = false
      + stop_signal                                 = (known after apply)
      + stop_timeout                                = (known after apply)
      + tty                                         = false
      + wait                                        = false
      + wait_timeout                                = 60

      + healthcheck (known after apply)

      + labels (known after apply)

      + ports {
          + external = 8000
          + internal = 80
          + ip       = "0.0.0.0"
          + protocol = "tcp"
        }
    }

  # docker_image.nginx will be created
  + resource "docker_image" "nginx" {
      + id           = (known after apply)
      + image_id     = (known after apply)
      + keep_locally = false
      + name         = "nginx:latest"
      + repo_digest  = (known after apply)
    }

Plan: 2 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.
```

```
  Enter a value: yes

docker_image.nginx: Creating...
docker_image.nginx: Creation complete after 7s [id=sha256:3b7732505933ca591ce4a6d860cb713ad96a3176b82f7979a8dfa9973486a0d6nginx:latest]
docker_container.nginx: Creating...
docker_container.nginx: Creation complete after 1s [id=e707d6c3b1046310412a756925c6e95d3babd24d9a326ed0c21aa5d1e1c7c46f]

Apply complete! Resources: 2 added, 0 changed, 0 destroyed.
PS C:\Users\lev\Desktop\devops\devops-gr12b-lab5-foreverNP\1\learn-terraform-docker-container>
```

```
PS C:\Users\lev\Desktop\devops\devops-gr12b-lab5-foreverNP\1\learn-terraform-docker-container> terraform show
# docker_container.nginx:
resource "docker_container" "nginx" {
    attach                                      = false
    bridge                                      = null
    command                                     = [
        "nginx",
        "-g",
        "daemon off;",
    ]
    container_read_refresh_timeout_milliseconds = 15000
    cpu_set                                     = null
    cpu_shares                                  = 0
    domainname                                  = null
    entrypoint                                  = [
        "/docker-entrypoint.sh",
    ]
    env                                         = []
    hostname                                    = "e707d6c3b104"
    id                                          = "e707d6c3b1046310412a756925c6e95d3babd24d9a326ed0c21aa5d1e1c7c46f"
    image                                       = "sha256:3b7732505933ca591ce4a6d860cb713ad96a3176b82f7979a8dfa9973486a0d6"
    init                                        = false
    ipc_mode                                    = "private"
    log_driver                                  = "json-file"
    logs                                        = false
    max_retry_count                             = 0
    memory                                      = 0
    memory_swap                                 = 0
    must_run                                    = true
    name                                        = "tutorial"
    network_data                                = [
        {
            gateway                 = "172.17.0.1"
            global_ipv6_address     = null
            global_ipv6_prefix_length = 0
            ip_address              = "172.17.0.2"
            ip_prefix_length        = 16
            ipv6_gateway            = null
            mac_address             = "02:42:ac:11:00:02"
            network_name            = "bridge"
        },
    ]
    network_mode                                = "bridge"
    pid_mode                                    = null
    privileged                                  = false
    publish_all_ports                           = false
    read_only                                   = false
    remove_volumes                              = true
    restart                                     = "no"
    rm                                          = false
    runtime                                     = "runc"
    security_opts                               = []
    shm_size                                    = 64
    start                                       = true
    stdin_open                                  = false
    stop_signal                                 = "SIGQUIT"
    stop_timeout                                = 0
    tty                                         = false
    user                                        = null
    userns_mode                                 = null
    wait                                        = false
    wait_timeout                                = 60
    working_dir                                 = null

    ports {
        external = 8000
        internal = 80
        ip       = "0.0.0.0"
        protocol = "tcp"
    }
}

# docker_image.nginx:
resource "docker_image" "nginx" {
    id           = "sha256:3b7732505933ca591ce4a6d860cb713ad96a3176b82f7979a8dfa9973486a0d6nginx:latest"
    image_id     = "sha256:3b7732505933ca591ce4a6d860cb713ad96a3176b82f7979a8dfa9973486a0d6"
    keep_locally = false
    name         = "nginx:latest"
    repo_digest  = "nginx@sha256:3b7732505933ca591ce4a6d860cb713ad96a3176b82f7979a8dfa9973486a0d6"
}
```



Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

Руководство 4

```
PS C:\Users\lev\Desktop\devops\devops-gr12b-lab5-foreverNP\1\learn-terraform-docker-container> terraform apply
docker_image.nginx: Refreshing state... [id=sha256:3b7732505933ca591ce4a6d860cb713ad96a3176b82f7979a8dfa9973486a0d6nginx:latest]
docker_container.nginx: Refreshing state... [id=e3defbf32fb53e94aa0d65a464a15c67ec63993a86c5871b6647896cae7c8be7]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
-/+ destroy and then create replacement

Terraform will perform the following actions:

  # docker_container.nginx must be replaced
-/+ resource "docker_container" "nginx" {
      + bridge                                      = (known after apply)
      ~ command                                     = [
          - "nginx",
          - "-g",
          - "daemon off;",
        ] -> (known after apply)
      + container_logs                              = (known after apply)
      - cpu_shares                                  = 0 -> null
      - dns                                         = [] -> null
      - dns_opts                                    = [] -> null
      - dns_search                                  = [] -> null
      ~ entrypoint                                  = [
          - "/docker-entrypoint.sh",
        ] -> (known after apply)
      ~ env                                         = [] -> (known after apply)
      + exit_code                                   = (known after apply)
      - group_add                                   = [] -> null
      ~ hostname                                    = "e3defbf32fb5" -> (known after apply)
      ~ id                                          = "e3defbf32fb53e94aa0d65a464a15c67ec63993a86c5871b6647896cae7c8be7" -> (known after apply)
      ~ init                                        = false -> (known after apply)
      ~ ipc_mode                                    = "private" -> (known after apply)
      ~ log_driver                                  = "json-file" -> (known after apply)
      - log_opts                                    = {} -> null
      - max_retry_count                             = 0 -> null
      - memory                                      = 0 -> null
      - memory_swap                                 = 0 -> null
        name                                        = "tutorial"
      ~ network_data                                = [
          - {
              - gateway                   = "172.17.0.1"
              - global_ipv6_prefix_length = 0
              - ip_address                = "172.17.0.2"
              - ip_prefix_length          = 16
              - mac_address               = "02:42:ac:11:00:02"
              - network_name              = "bridge"
                # (2 unchanged attributes hidden)
            },
        ] -> (known after apply)
      - network_mode                                = "bridge" -> null # forces replacement
      - privileged                                  = false -> null
      - publish_all_ports                           = false -> null
      ~ runtime                                     = "runc" -> (known after apply)
      ~ security_opts                               = [] -> (known after apply)
      ~ shm_size                                    = 64 -> (known after apply)
      ~ stop_signal                                 = "SIGQUIT" -> (known after apply)
      ~ stop_timeout                                = 0 -> (known after apply)
      - storage_opts                                = {} -> null
      - sysctls                                     = {} -> null
      - tmpfs                                       = {} -> null
        # (20 unchanged attributes hidden)

      ~ healthcheck (known after apply)

      ~ labels (known after apply)

      ~ ports {
          ~ external = 8000 -> 8080 # forces replacement
            # (3 unchanged attributes hidden)
        }
    }

Plan: 1 to add, 0 to change, 1 to destroy.

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

docker_container.nginx: Destroying... [id=e3defbf32fb53e94aa0d65a464a15c67ec63993a86c5871b6647896cae7c8be7]
docker_container.nginx: Destruction complete after 1s
docker_container.nginx: Creating...
docker_container.nginx: Creation complete after 0s [id=2586b724f81238fd1da0d4daed2f9ba3477bf2b61474a74522a8bd9e1a916cd5]

Apply complete! Resources: 1 added, 0 changed, 1 destroyed.
PS C:\Users\lev\Desktop\devops\devops-gr12b-lab5-foreverNP\1\learn-terraform-docker-container>
```

Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

Руководство 5

```
PS C:\Users\lev\Desktop\devops\devops-gr12b-lab5-foreverNP\1\learn-terraform-docker-container> terraform destroy
docker_image.nginx: Refreshing state... [id=sha256:3b7732505933ca591ce4a6d860cb713ad96a3176b82f7979a8dfa9973486a0d6nginx:latest]
docker_container.nginx: Refreshing state... [id=2586b724f81238fd1da0d4daed2f9ba3477bf2b61474a74522a8bd9e1a916cd5]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
  - destroy

Terraform will perform the following actions:

  # docker_container.nginx will be destroyed
  - resource "docker_container" "nginx" {
      - attach                                      = false -> null
      - command                                     = [
          - "nginx",
          - "-g",
          - "daemon off;",
        ] -> null
      - container_read_refresh_timeout_milliseconds = 15000 -> null
      - cpu_shares                                  = 0 -> null
      - dns                                         = [] -> null
      - dns_opts                                    = [] -> null
      - dns_search                                  = [] -> null
      - entrypoint                                  = [
          - "/docker-entrypoint.sh",
        ] -> null
      - env                                         = [] -> null
      - group_add                                   = [] -> null
      - hostname                                    = "2586b724f812" -> null
      - id                                          = "2586b724f81238fd1da0d4daed2f9ba3477bf2b61474a74522a8bd9e1a916cd5" -> null
      - image                                       = "sha256:3b7732505933ca591ce4a6d860cb713ad96a3176b82f7979a8dfa9973486a0d6" -> null
      - init                                        = false -> null
      - ipc_mode                                    = "private" -> null
      - log_driver                                  = "json-file" -> null
      - log_opts                                    = {} -> null
      - logs                                        = false -> null
      - max_retry_count                             = 0 -> null
      - memory                                      = 0 -> null
      - memory_swap                                 = 0 -> null
      - must_run                                    = true -> null
      - name                                        = "tutorial" -> null
      - network_data                                = [
          - {
              - gateway                   = "172.17.0.1"
              - global_ipv6_prefix_length = 0
              - ip_address                = "172.17.0.2"
              - ip_prefix_length          = 16
              - mac_address               = "02:42:ac:11:00:02"
              - network_name              = "bridge"
                # (2 unchanged attributes hidden)
            },
        ] -> null
      - network_mode                                = "bridge" -> null
      - privileged                                  = false -> null
      - publish_all_ports                           = false -> null
      - read_only                                   = false -> null
      - remove_volumes                              = true -> null
      - restart                                     = "no" -> null
      - rm                                          = false -> null
      - runtime                                     = "runc" -> null
      - security_opts                               = [] -> null
      - shm_size                                    = 64 -> null
      - start                                       = true -> null
      - stdin_open                                  = false -> null
      - stop_signal                                 = "SIGQUIT" -> null
      - stop_timeout                                = 0 -> null
      - storage_opts                                = {} -> null
      - sysctls                                     = {} -> null
      - tmpfs                                       = {} -> null
      - tty                                         = false -> null
      - wait                                        = false -> null
      - wait_timeout                                = 60 -> null
        # (7 unchanged attributes hidden)

      - ports {
          - external = 8080 -> null
          - internal = 80 -> null
          - ip       = "0.0.0.0" -> null
          - protocol = "tcp" -> null
        }
    }

  # docker_image.nginx will be destroyed
  - resource "docker_image" "nginx" {
      - id           = "sha256:3b7732505933ca591ce4a6d860cb713ad96a3176b82f7979a8dfa9973486a0d6nginx:latest" -> null
      - image_id     = "sha256:3b7732505933ca591ce4a6d860cb713ad96a3176b82f7979a8dfa9973486a0d6" -> null
      - keep_locally = false -> null
      - name         = "nginx:latest" -> null
      - repo_digest  = "nginx@sha256:3b7732505933ca591ce4a6d860cb713ad96a3176b82f7979a8dfa9973486a0d6" -> null
    }

Plan: 0 to add, 0 to change, 2 to destroy.

Do you really want to destroy all resources?
  Terraform will destroy all your managed infrastructure, as shown above.
  There is no undo. Only 'yes' will be accepted to confirm.

  Enter a value: yes

docker_container.nginx: Destroying... [id=2586b724f81238fd1da0d4daed2f9ba3477bf2b61474a74522a8bd9e1a916cd5]
docker_container.nginx: Destruction complete after 0s
docker_image.nginx: Destroying... [id=sha256:3b7732505933ca591ce4a6d860cb713ad96a3176b82f7979a8dfa9973486a0d6nginx:latest]
docker_image.nginx: Destruction complete after 1s

Destroy complete! Resources: 2 destroyed.
```

удалил

Руководство 6

```
PS C:\Users\lev\Desktop\devops\devops-gr12b-lab5-foreverNP\1\learn-terraform-docker-container> terraform apply

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
  + create

Terraform will perform the following actions:

  # docker_container.nginx will be created
  + resource "docker_container" "nginx" {
      + attach                                      = false
      + bridge                                      = (known after apply)
      + command                                     = (known after apply)
      + container_logs                              = (known after apply)
      + container_read_refresh_timeout_milliseconds = 15000
      + entrypoint                                  = (known after apply)
      + env                                         = (known after apply)
      + exit_code                                   = (known after apply)
      + hostname                                    = (known after apply)
      + id                                          = (known after apply)
      + image                                       = (known after apply)
      + init                                        = (known after apply)
      + ipc_mode                                    = (known after apply)
      + log_driver                                  = (known after apply)
      + logs                                        = false
      + must_run                                    = true
      + name                                        = "ExampleNginxContainer"
      + network_data                                = (known after apply)
```

```
PS C:\Users\lev\Desktop\devops\devops-gr12b-lab5-foreverNP\1\learn-terraform-docker-container> terraform apply -var "container_name=YetAnotherName"

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
  + create

Terraform will perform the following actions:

  # docker_container.nginx will be created
  + resource "docker_container" "nginx" {
      + attach                                      = false
      + bridge                                      = (known after apply)
      + command                                     = (known after apply)
      + container_logs                              = (known after apply)
      + container_read_refresh_timeout_milliseconds = 15000
      + entrypoint                                  = (known after apply)
      + env                                         = (known after apply)
      + exit_code                                   = (known after apply)
      + hostname                                    = (known after apply)
      + id                                          = (known after apply)
      + image                                       = (known after apply)
      + init                                        = (known after apply)
      + ipc_mode                                    = (known after apply)
      + log_driver                                  = (known after apply)
      + logs                                        = false
      + must_run                                    = true
      + name                                        = "YetAnotherName"
      + network_data                                = (known after apply)
      + read_only                                   = false
```

Руководство 7

```
PS C:\Users\lev\Desktop\devops\devops-gr12b-lab5-foreverNP\1\learn-terraform-docker-container> terraform apply
docker_image.nginx: Refreshing state... [id=sha256:3b7732505933ca591ce4a6d860cb713ad96a3176b82f7979a8dfa9973486a0d6nginx:latest]
docker_container.nginx: Refreshing state... [id=4bae93e5f8f603d581ea43a88a51e9a21a72eb25c705a9b9db2736fc84002448]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
-/+ destroy and then create replacement

Terraform will perform the following actions:

  # docker_container.nginx must be replaced
-/+ resource "docker_container" "nginx" {
      + bridge                                      = (known after apply)
      ~ command                                     = [
          - "nginx",
          - "-g",
          - "daemon off;",
        ] -> (known after apply)
      + container_logs                              = (known after apply)
      - cpu_shares                                  = 0 -> null
      - dns                                         = [] -> null
      - dns_opts                                    = [] -> null
      - dns_search                                  = [] -> null
      ~ entrypoint                                  = [
          - "/docker-entrypoint.sh",
        ] -> (known after apply)
      ~ env                                         = [] -> (known after apply)
      + exit_code                                   = (known after apply)
      - group_add                                   = [] -> null
      ~ hostname                                    = "4bae93e5f8f6" -> (known after apply)
      ~ id                                          = "4bae93e5f8f603d581ea43a88a51e9a21a72eb25c705a9b9db2736fc84002448" -> (known after apply)
      ~ init                                        = false -> (known after apply)
      ~ ipc_mode                                    = "private" -> (known after apply)
      ~ log_driver                                  = "json-file" -> (known after apply)
      - log_opts                                    = {} -> null
      - max_retry_count                             = 0 -> null
      - memory                                      = 0 -> null
      - memory_swap                                 = 0 -> null
        name                                        = "ExampleNginxContainer"
      ~ network_data                                = [
          - {
              - gateway                   = "172.17.0.1"
              - global_ipv6_prefix_length = 0
              - ip_address                = "172.17.0.2"
              - ip_prefix_length          = 16
              - mac_address               = "02:42:ac:11:00:02"
              - network_name              = "bridge"
                # (2 unchanged attributes hidden)
            },
        ] -> (known after apply)
      - network_mode                                = "bridge" -> null # forces replacement
      - privileged                                  = false -> null
      - publish_all_ports                           = false -> null
      ~ runtime                                     = "runc" -> (known after apply)
      ~ security_opts                               = [] -> (known after apply)
      ~ shm_size                                    = 64 -> (known after apply)
      ~ stop_signal                                 = "SIGQUIT" -> (known after apply)
      ~ stop_timeout                                = 0 -> (known after apply)
      - storage_opts                                = {} -> null
      - sysctls                                     = {} -> null
      - tmpfs                                       = {} -> null
        # (20 unchanged attributes hidden)

      ~ healthcheck (known after apply)

      ~ labels (known after apply)

        # (1 unchanged block hidden)
    }

Plan: 1 to add, 0 to change, 1 to destroy.

Changes to Outputs:
  + container_id = (known after apply)
  + image_id     = "sha256:3b7732505933ca591ce4a6d860cb713ad96a3176b82f7979a8dfa9973486a0d6nginx:latest"

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

docker_container.nginx: Destroying... [id=4bae93e5f8f603d581ea43a88a51e9a21a72eb25c705a9b9db2736fc84002448]
docker_container.nginx: Destruction complete after 1s
docker_container.nginx: Creating...
docker_container.nginx: Creation complete after 0s [id=0614f3c459fc9403a6e7a22219bd68231ee651cc97a88ae9c153c152286da2fa]

Apply complete! Resources: 1 added, 0 changed, 1 destroyed.
```

```
Outputs:

container_id = "0614f3c459fc9403a6e7a22219bd68231ee651cc97a88ae9c153c152286da2fa"
image_id = "sha256:3b7732505933ca591ce4a6d860cb713ad96a3176b82f7979a8dfa9973486a0d6nginx:latest"
```

Задание 2. Управление конфигурацией серверов с помощью Ansible

```
lev@redmibook-15 /m/c/U/l/D/d/devops-gr12b-lab5-foreverNP (main)> ansible --version
                                                                  ansible-inventory --version

ansible 2.10.8
  config file = None
  configured module search path = ['/home/lev/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python3/dist-packages/ansible
  executable location = /usr/bin/ansible
  python version = 3.10.12 (main, Aug 15 2025, 14:32:43) [GCC 11.4.0]
ansible-inventory 2.10.8
  config file = None
  configured module search path = ['/home/lev/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python3/dist-packages/ansible
  executable location = /usr/bin/ansible-inventory
  python version = 3.10.12 (main, Aug 15 2025, 14:32:43) [GCC 11.4.0]
```

Продемонстрировать подключение к виртуальным машинам, созданным для кластера Kubernetes в лабораторной работе 3 и установкой любого пакета, например zsh, с помощью команд.

```
PLAY RECAP *********************************************************************************************************
localhost                  : ok=4    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

```
lev@redmibook-15 /m/c/U/l/D/d/d/2/1 (main)> ansible-playbook -i localhost, install_zsh.yml


PLAY [Install zsh on minikube nodes via minikube ssh] **************************************************************

TASK [Get minikube node names] *************************************************************************************
ok: [localhost]

TASK [Debug - show found nodes] ************************************************************************************
ok: [localhost] => {
    "msg": [
        "minikube",
        "minikube-m02",
        "minikube-m03"
    ]
}

TASK [Install zsh on each node via minikube ssh] *******************************************************************
changed: [localhost] => (item=minikube)
changed: [localhost] => (item=minikube-m02)
changed: [localhost] => (item=minikube-m03)

TASK [Show installation results] ***********************************************************************************
ok: [localhost] => {
    "install_results.results": [
        {
            "ansible_loop_var": "item",
            "changed": true,
            "cmd": "echo \"Installing zsh on minikube\"\nminikube ssh -n \"minikube\" -- 'sudo apt update && sudo apt install -y zsh || sudo yum install -y zsh || true'
\n",
            "delta": "0:00:01.475585",
            "end": "2025-10-19 17:21:06.728616",
            "failed": false,
            "invocation": {
                "module_args": {
                    "_raw_params": "echo \"Installing zsh on minikube\"\nminikube ssh -n \"minikube\" -- 'sudo apt update && sudo apt install -y zsh || sudo yum install
 -y zsh || true'\n",
                    "_uses_shell": true,
```

```
lev@redmibook-15 /m/c/U/l/D/d/d/2/1 (main)> minikube ssh -n minikube -- 'zsh --version'
zsh 5.8.1 (x86_64-ubuntu-linux-gnu)
lev@redmibook-15 /m/c/U/l/D/d/d/2/1 (main)> minikube ssh -n minikube-m02 -- 'zsh --version'
zsh 5.8.1 (x86_64-ubuntu-linux-gnu)
lev@redmibook-15 /m/c/U/l/D/d/d/2/1 (main)> minikube ssh -n minikube-m03 -- 'zsh --version'
zsh 5.8.1 (x86_64-ubuntu-linux-gnu)
lev@redmibook-15 /m/c/U/l/D/d/d/2/1 (main)>
```

Настроить конфигурации в виде файлов playbook на основе примеров из статьи в п. 4 задания и продемонстрировать их.

```
PLAY [Gather minikube nodes] ***************************************************************************************

TASK [Get minikube node names] *************************************************************************************
ok: [localhost]

TASK [Set fact with node list] *************************************************************************************
ok: [localhost]

TASK [Show nodes] **************************************************************************************************
ok: [localhost] => {
    "minikube_nodes": [
        "minikube",
        "minikube-m02",
        "minikube-m03"
    ]
}

PLAY [Apply 'common' config to all minikube nodes] *****************************************************************

TASK [Ensure basic packages + motd on each node] *******************************************************************
changed: [localhost] => (item=minikube)
changed: [localhost] => (item=minikube-m02)
changed: [localhost] => (item=minikube-m03)

PLAY [Deploy web (apache) on web nodes] ****************************************************************************

TASK [Install Apache and deploy demo index.html] *******************************************************************
changed: [localhost] => (item=minikube)
changed: [localhost] => (item=minikube-m02)
changed: [localhost] => (item=minikube-m03)

PLAY RECAP *********************************************************************************************************
localhost                  : ok=5    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

```
PLAY [Get minikube nodes list] ********************************************************************************************

TASK [Get nodes] **********************************************************************************************************
ok: [localhost]

TASK [Set nodes fact] *****************************************************************************************************
ok: [localhost]

TASK [Pre-task: disable monitoring/put out of rotation (simulated)] *******************************************************
ok: [localhost] => (item=minikube) => {
    "msg": "Would disable monitoring / remove minikube from LB pool (simulate)"
}
ok: [localhost] => (item=minikube-m02) => {
    "msg": "Would disable monitoring / remove minikube-m02 from LB pool (simulate)"
}
ok: [localhost] => (item=minikube-m03) => {
    "msg": "Would disable monitoring / remove minikube-m03 from LB pool (simulate)"
}

TASK [Update app on the node (install packages + replace index)] **********************************************************
changed: [localhost] => (item=minikube)
changed: [localhost] => (item=minikube-m02)
changed: [localhost] => (item=minikube-m03)

TASK [Post-task: re-enable monitoring/put back into rotation (simulated)] *************************************************
ok: [localhost] => (item=minikube) => {
    "msg": "Would re-enable monitoring / add minikube back to LB pool (simulate)"
}
ok: [localhost] => (item=minikube-m02) => {
    "msg": "Would re-enable monitoring / add minikube-m02 back to LB pool (simulate)"
}
ok: [localhost] => (item=minikube-m03) => {
    "msg": "Would re-enable monitoring / add minikube-m03 back to LB pool (simulate)"
}

TASK [Show update result for node] ****************************************************************************************
ok: [localhost] => {
    "update_result.stdout_lines": "VARIABLE IS NOT DEFINED!"
}

PLAY RECAP ****************************************************************************************************************
localhost                  : ok=6    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

Выполнить примеры из руководства Ansible, Playbook и продемонстрировать результаты.

```
lev@redmibook-15 /m/c/U/l/D/d/d/2/ansible (main)> cd  lab6-agent

                                                    # собираем образ базового контейнера для лабы
                                                    ./build.sh
[+] Building 70.8s (8/8) FINISHED                                                                                                                    docker:default
 => [internal] load build definition from Dockerfile                                                                                                            0.0s
 => => transferring dockerfile: 412B                                                                                                                            0.0s
 => [internal] load metadata for docker.io/library/ubuntu:20.04                                                                                                 1.0s
 => [internal] load .dockerignore                                                                                                                               0.0s
 => => transferring context: 2B                                                                                                                                 0.0s
 => [1/4] FROM docker.io/library/ubuntu:20.04@sha256:8feb4d8ca5354def3d8fce243717141ce31e2c428701f6682bd2fafe15388214                                           3.2s
 => => resolve docker.io/library/ubuntu:20.04@sha256:8feb4d8ca5354def3d8fce243717141ce31e2c428701f6682bd2fafe15388214                                           0.0s
 => => sha256:13b7e930469f6d3575a320709035c6acf6f5485a76abcf03d1b92a64c09c2476 27.51MB / 27.51MB                                                                2.0s
 => => extracting sha256:13b7e930469f6d3575a320709035c6acf6f5485a76abcf03d1b92a64c09c2476                                                                       1.1s
 => [2/4] RUN apt-get update && apt-get install -y openssh-server python3-pip inetutils-ping net-tools curl                                                     42.6s
 => [3/4] RUN sed -i 's/#PermitRootLogin prohibit-password/PermitRootLogin yes/' /etc/ssh/sshd_config                                                           0.5s
 => [4/4] RUN echo "root:root" | chpasswd                                                                                                                       0.5s
 => exporting to image                                                                                                                                         22.9s
 => => exporting layers                                                                                                                                        17.5s
 => => exporting manifest sha256:4c2eaf12ea75a569283e8032ac25c59a9871d6f385377268556bf10c2e7060ee                                                              0.0s
 => => exporting config sha256:9e2b91f1446697497dd9bbf5d701f5926e162c3c721ae6d1072e95c0553c2884                                                                 0.0s
 => => exporting attestation manifest sha256:e8432e828fcef24ab01d19ff70252ee1d2c858bf659d9f938af36186292f18b8                                                   0.0s
 => => exporting manifest list sha256:9b1ba98f2dd863c0695a20a390b023efb87ec6aae0de1b67d11bffad887c30a2                                                          0.0s
 => => naming to docker.io/library/lab6-agent:latest                                                                                                            0.0s
 => => unpacking to docker.io/library/lab6-agent:latest                                                                                                         5.3s

 1 warning found (use docker --debug to expand):
 - JSONArgsRecommended: JSON arguments recommended for ENTRYPOINT to prevent unintended behavior related to OS signals (line 10)
lev@redmibook-15 /m/c/U/l/D/d/d/2/a/lab6-agent (main)> cd ../
                                                       docker-compose up -d
WARN[0000] /mnt/c/Users/lev/Desktop/devops/devops-gr12b-lab5-foreverNP/2/ansible/docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion
[+] Building 8.5s (4/5)                                                                                                                              docker:default
 => [ansible internal] load build definition from Dockerfile                                                                                                    0.0s
 => => transferring dockerfile: 106B                                                                                                                            0.0s
 => [ansible internal] load metadata for docker.io/library/lab6-agent:latest                                                                                    0.0s
 => [ansible internal] load .dockerignore                                                                                                                       0.0s
 => => transferring context: 2B                                                                                                                                 0.0s
 => [ansible 1/2] FROM docker.io/library/lab6-agent:latest@sha256:9b1ba98f2dd863c0695a20a390b023efb87ec6aae0de1b67d11bffad887c30a2                              1.9s
 => => resolve docker.io/library/lab6-agent:latest@sha256:9b1ba98f2dd863c0695a20a390b023efb87ec6aae0de1b67d11bffad887c30a2                                      1.8s
 => [ansible 2/2] RUN apt update && apt install -y ansible sshpass                                                                                              6.5s
 => => # Get:21 http://archive.ubuntu.com/ubuntu focal/universe amd64 python3-requests-ntlm all 1.1.0-1 [6004 B]
 => => # Get:22 http://archive.ubuntu.com/ubuntu focal/universe amd64 python3-selinux amd64 3.0-1build2 [139 kB]
 => => # Get:23 http://archive.ubuntu.com/ubuntu focal/universe amd64 python3-xmltodict all 0.12.0-1 [12.6 kB]
 => => # Get:24 http://archive.ubuntu.com/ubuntu focal/universe amd64 python3-winrm all 0.3.0-2 [21.7 kB]
 => => # Get:25 http://archive.ubuntu.com/ubuntu focal/universe amd64 sshpass amd64 1.06-1 [10.5 kB]
 => => # debconf: delaying package configuration, since apt-utils is not installed
[+] Building 8.6s (4/5)                                                                                                                              docker:default
 => [ansible internal] load build definition from Dockerfile                                                                                                    0.0s
 => => transferring dockerfile: 106B                                                                                                                            0.0s
 => [ansible internal] load metadata for docker.io/library/lab6-agent:latest                                                                                    0.0s
[+] Building 8.7s (4/5)                                                                                                                              docker:default
 => [ansible internal] load build definition from Dockerfile                                                                                                    0.0s
 => => transferring dockerfile: 106B                                                                                                                            0.0s
 => [ansible internal] load metadata for docker.io/library/lab6-agent:latest                                                                                    0.0s
[+] Building 27.0s (7/7) FINISHED                                                                                                                    docker:default
 => [ansible internal] load build definition from Dockerfile                                                                                                    0.0s
 => => transferring dockerfile: 106B                                                                                                                            0.0s
 => [ansible internal] load metadata for docker.io/library/lab6-agent:latest                                                                                    0.0s
 => [ansible internal] load .dockerignore                                                                                                                       0.0s
 => => transferring context: 2B                                                                                                                                 0.0s
 => [ansible 1/2] FROM docker.io/library/lab6-agent:latest@sha256:9b1ba98f2dd863c0695a20a390b023efb87ec6aae0de1b67d11bffad887c30a2                              1.9s
 => => resolve docker.io/library/lab6-agent:latest@sha256:9b1ba98f2dd863c0695a20a390b023efb87ec6aae0de1b67d11bffad887c30a2                                      1.8s
 => [ansible 2/2] RUN apt update && apt install -y ansible sshpass                                                                                             19.2s
 => [ansible] exporting to image                                                                                                                                5.7s
 => => exporting layers                                                                                                                                         2.5s
 => => exporting manifest sha256:94ff6fb460dcd8b4bba59954f32771f0d55aa931dc57d3a45e55c6b51cb9a634                                                              0.0s
 => => exporting config sha256:5e5dc6a47c6e6e63ef255ac80c968e0bfc1a82b37099e98fe8456ebce0e3a8ea                                                                 0.0s
 => => exporting attestation manifest sha256:6451224d44a16be7055105a7d16956afb433a45fb2ab2080c10fb6eb2785b882                                                   0.0s
 => => exporting manifest list sha256:9f7006548098a658635e0b763f8b4860f34b41e7f57f8969acd5808cd5200630                                                          0.0s
 => => naming to docker.io/library/ansible-ansible:latest                                                                                                       0.0s
 => => unpacking to docker.io/library/ansible-ansible:latest                                                                                                    3.1s
 => [ansible] resolving provenance for metadata file                                                                                                            0.0s
[+] Running 3/3
 ✓ Network ansible_default      Created                                                                                                                         0.1s
 ✓ Container ansible-app-1      Started                                                                                                                         1.1s
 ✓ Container ansible-ansible-1  Started                                                                                                                         1.1s
lev@redmibook-15 /m/c/U/l/D/d/d/2/ansible (main)> docker-compose down
WARN[0000] /mnt/c/Users/lev/Desktop/devops/devops-gr12b-lab5-foreverNP/2/ansible/docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion
[+] Running 3/3
 ✓ Container ansible-ansible-1  Removed
 ✓ Container ansible-app-1      Removed
 ✓ Network ansible_default      Removed
lev@redmibook-15 /m/c/U/l/D/d/d/2/ansible (main)>
```

1. **Поднять кластер и установить тестовую базу данных**

```
lev@redmibook-15 /m/c/U/l/D/d/d/2/ansible (main)> docker-compose up -d --scale app=3
WARN[0000] /mnt/c/Users/lev/Desktop/devops/devops-gr12b-lab5-foreverNP/2/ansible/docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion
[+] Running 5/5
 ✓ Network ansible_default      Created
                                                                         0.1s
 ✓ Container ansible-app-3      Started
                                                                         1.8s
 ✓ Container ansible-app-1      Started
                                                                         0.9s
 ✓ Container ansible-app-2      Started
                                                                         1.4s
 ✓ Container ansible-ansible-1  Started
                                                                         2.0s
lev@redmibook-15 /m/c/U/l/D/d/d/2/ansible (main)> docker ps
CONTAINER ID   IMAGE                                 COMMAND                  CREATED          STATUS          PORTS                                                                                                                                  NAMES
aad0348094d1   ansible-ansible                       "/bin/sh -c 'service…"   25 seconds ago   Up 23 seconds   3306/tcp, 5000/tcp, 0.0.0.0:8080->8080/tcp                                                                                             ansible-ansible-1
9e9e26e435d9   lab6-agent                            "/bin/sh -c 'service…"   25 seconds ago   Up 24 seconds   0.0.0.0:3301->3306/tcp, 0.0.0.0:8082->5000/tcp                                                                                         ansible-app-1
6592d5982f50   lab6-agent                            "/bin/sh -c 'service…"   25 seconds ago   Up 23 seconds   0.0.0.0:3302->3306/tcp, 0.0.0.0:8083->5000/tcp                                                                                         ansible-app-2
9c5822f5b219   lab6-agent                            "/bin/sh -c 'service…"   25 seconds ago   Up 23 seconds   0.0.0.0:3303->3306/tcp, 0.0.0.0:8084->5000/tcp                                                                                         ansible-app-3
d5ec7ecc2f13   gcr.io/k8s-minikube/kicbase:v0.0.48   "/usr/local/bin/entr…"   45 minutes ago   Up 45 minutes   127.0.0.1:51402->22/tcp, 127.0.0.1:51403->2376/tcp, 127.0.0.1:51405->5000/tcp, 127.0.0.1:51406->8443/tcp, 127.0.0.1:51404->32443/tcp   minikube-m03
87e4cf0713a4   gcr.io/k8s-minikube/kicbase:v0.0.48   "/usr/local/bin/entr…"   45 minutes ago   Up 45 minutes   127.0.0.1:51368->22/tcp, 127.0.0.1:51369->2376/tcp, 127.0.0.1:51371->5000/tcp, 127.0.0.1:51367->8443/tcp, 127.0.0.1:51370->32443/tcp   minikube-m02
29bfa6369e1a   gcr.io/k8s-minikube/kicbase:v0.0.48   "/usr/local/bin/entr…"   45 minutes ago   Up 45 minutes   127.0.0.1:59047->22/tcp, 127.0.0.1:59043->2376/tcp, 127.0.0.1:59045->5000/tcp, 127.0.0.1:59046->8443/tcp, 127.0.0.1:59044->32443/tcp   minikube
lev@redmibook-15 /m/c/U/l/D/d/d/2/ansible (main)> docker exec -it `docker ps -aqf "name=ansible-ansible-1"` bash
Error response from daemon: No such container: `docker
lev@redmibook-15 /m/c/U/l/D/d/d/2/ansible (main) [1]> docker ps -aqf "name=ansible-ansible-1"
aad0348094d1
lev@redmibook-15 /m/c/U/l/D/d/d/2/ansible (main)> docker exec -it aad034 bash
root@aad0348094d1:/# ansible all -m ping
[WARNING]:  * Failed to parse /inventory/application_vars with script plugin: problem running /inventory/application_vars --list ([Errno 8] Exec format error: '/inventory/application_vars')
[WARNING]:  * Failed to parse /inventory/application_vars with yaml plugin: Syntax Error while loading YAML.   did not find expected <document start>  The error appears to be in '/inventory/application_vars': line 3, column 1, but may be elsewhere in the file depending
on the exact syntax problem.  The offending line appears to be:   mysql_host=ansible-app-3 ^ here
[WARNING]:  * Failed to parse /inventory/application_vars with ini plugin: /inventory/application_vars:1: Section [application:vars] not valid for undefined group: application
[WARNING]: Unable to parse /inventory/application_vars as an inventory source
[WARNING]:  * Failed to parse /inventory/database_vars with script plugin: problem running /inventory/database_vars --list ([Errno 8] Exec format error: '/inventory/database_vars')
[WARNING]:  * Failed to parse /inventory/database_vars with yaml plugin: Syntax Error while loading YAML.   did not find expected <document start>  The error appears to be in '/inventory/database_vars': line 3, column 1, but may be elsewhere in the file depending on the
exact syntax problem.  The offending line appears to be:   mysql_root_password=root ^ here
[WARNING]:  * Failed to parse /inventory/database_vars with ini plugin: /inventory/database_vars:1: Section [database:vars] not valid for undefined group: database
[WARNING]: Unable to parse /inventory/database_vars as an inventory source
ansible-app-2 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
ansible-ansible-1 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
ansible-app-1 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
ansible-app-3 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
root@aad0348094d1:/#
```

```
root@aad0348094d1:/# ansible all -m command -a 'uptime'
[WARNING]:  * Failed to parse /inventory/application_vars with script plugin: problem running /inventory/application_vars --list ([Errno 8] Exec format error: '/inventory/application_vars')
[WARNING]:  * Failed to parse /inventory/application_vars with yaml plugin: Syntax Error while loading YAML.   did not find expected <document start>  The error appears to be in '/inventory/application_vars': line 3,
column 1, but may be elsewhere in the file depending on the exact syntax problem.  The offending line appears to be:   mysql_host=ansible-app-3 ^ here
[WARNING]:  * Failed to parse /inventory/application_vars with ini plugin: /inventory/application_vars:1: Section [application:vars] not valid for undefined group: application
[WARNING]: Unable to parse /inventory/application_vars as an inventory source
[WARNING]:  * Failed to parse /inventory/database_vars with script plugin: problem running /inventory/database_vars --list ([Errno 8] Exec format error: '/inventory/database_vars')
[WARNING]:  * Failed to parse /inventory/database_vars with yaml plugin: Syntax Error while loading YAML.   did not find expected <document start>  The error appears to be in '/inventory/database_vars': line 3, column
1, but may be elsewhere in the file depending on the exact syntax problem.  The offending line appears to be:   mysql_root_password=root ^ here
[WARNING]:  * Failed to parse /inventory/database_vars with ini plugin: /inventory/database_vars:1: Section [database:vars] not valid for undefined group: database
[WARNING]: Unable to parse /inventory/database_vars as an inventory source
ansible-ansible-1 | CHANGED | rc=0 >>
 15:01:26 up  3:31,  1 user,  load average: 0.37, 0.45, 0.48
ansible-app-2 | CHANGED | rc=0 >>
 15:01:26 up  3:31,  1 user,  load average: 0.37, 0.45, 0.48
ansible-app-1 | CHANGED | rc=0 >>
 15:01:26 up  3:31,  1 user,  load average: 0.37, 0.45, 0.48
ansible-app-3 | CHANGED | rc=0 >>
 15:01:26 up  3:31,  1 user,  load average: 0.37, 0.45, 0.48
root@aad0348094d1:/#
```

```
root@aad0348094d1:/# ansible-playbook /playbooks/playbook-database.yml
[WARNING]:  * Failed to parse /inventory/application_vars with script plugin: problem running /inventory/application_vars --list ([Errno 8] Exec format error: '/inventory/application_vars')
[WARNING]:  * Failed to parse /inventory/application_vars with yaml plugin: Syntax Error while loading YAML.   did not find expected <document start>  The error appears to be in '/inventory/application_vars': line 3,
column 1, but may be elsewhere in the file depending on the exact syntax problem.  The offending line appears to be:   mysql_host=ansible-app-3 ^ here
[WARNING]:  * Failed to parse /inventory/application_vars with ini plugin: /inventory/application_vars:1: Section [application:vars] not valid for undefined group: application
[WARNING]: Unable to parse /inventory/application_vars as an inventory source
[WARNING]:  * Failed to parse /inventory/database_vars with script plugin: problem running /inventory/database_vars --list ([Errno 8] Exec format error: '/inventory/database_vars')
[WARNING]:  * Failed to parse /inventory/database_vars with yaml plugin: Syntax Error while loading YAML.   did not find expected <document start>  The error appears to be in '/inventory/database_vars': line 3, column
1, but may be elsewhere in the file depending on the exact syntax problem.  The offending line appears to be:   mysql_root_password=root ^ here
[WARNING]:  * Failed to parse /inventory/database_vars with ini plugin: /inventory/database_vars:1: Section [database:vars] not valid for undefined group: database
[WARNING]: Unable to parse /inventory/database_vars as an inventory source

PLAY [database] ****************************************************************************************************************************************************************************************

TASK [Gathering Facts] *********************************************************************************************************************************************************************************
ok: [ansible-app-3]

TASK [Install MySQL and other required packages] *******************************************************************************************************************************************************
[DEPRECATION WARNING]: Invoking "apt" only once while using a loop via squash_actions is deprecated. Instead of using a loop to supply multiple items and specifying `name: "{{ item }}"`, please use `name: ['mysql-
server']` and remove the loop. This feature will be removed in version 2.11. Deprecation warnings can be disabled by setting deprecation_warnings=False in ansible.cfg.
        changed: [ansible-app-3] => (item=['mysql-server'])
[WARNING]: Updating cache and auto-installing missing dependency: python3-apt

TASK [Save root password in .my.cnf - to access locally from the host] *********************************************************************************************************************************
changed: [ansible-app-3]

TASK [Make sure pymysql is present] ********************************************************************************************************************************************************************
changed: [ansible-app-3]

TASK [Start the MySQL service] *************************************************************************************************************************************************************************
changed: [ansible-app-3]

TASK [Set password for root user] **********************************************************************************************************************************************************************
[WARNING]: Module did not set no_log for update_password
changed: [ansible-app-3]

TASK [Stop the MySQL service(if was running)] **********************************************************************************************************************************************************
changed: [ansible-app-3]

TASK [Expose mysql for other hosts] ********************************************************************************************************************************************************************
[WARNING]: Consider using the replace, lineinfile or template module rather than running 'sed'.  If you need to use command because replace, lineinfile or template is insufficient you can add 'warn: false' to this
command task or set 'command_warnings=False' in ansible.cfg to get rid of this message.
changed: [ansible-app-3]

TASK [copy permissions] ********************************************************************************************************************************************************************************
changed: [ansible-app-3]

TASK [add sample data to database host] ****************************************************************************************************************************************************************
changed: [ansible-app-3]

TASK [Start the MySQL service] *************************************************************************************************************************************************************************
changed: [ansible-app-3]

TASK [apply permissions] *******************************************************************************************************************************************************************************
changed: [ansible-app-3]

TASK [insert sample data into database] ****************************************************************************************************************************************************************
changed: [ansible-app-3]

PLAY RECAP *********************************************************************************************************************************************************************************************
ansible-app-3              : ok=13   changed=12   unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

```
root@aad0348094d1:/# ssh ansible_app_3
ssh: Could not resolve hostname ansible_app_3: Temporary failure in name resolution
root@aad0348094d1:/# ssh ansible-app-3
root@ansible-app-3's password:
Permission denied, please try again.
root@ansible-app-3's password:
Welcome to Ubuntu 20.04.6 LTS (GNU/Linux 5.15.167.4-microsoft-standard-WSL2 x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.
Last login: Sun Oct 19 15:02:51 2025 from 172.19.0.5
root@9c5822f5b219:~#
```

```
root@9c5822f5b219:~# mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 14
Server version: 8.0.42-0ubuntu0.20.04.1 (Ubuntu)

Copyright (c) 2000, 2025, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

```
mysql> use testdb
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> select * from test;
+--------------------+
| message            |
+--------------------+
| Ansible To Do List |
| Get ready          |
| Ansible is fun     |
+--------------------+
3 rows in set (0.00 sec)

mysql>
```

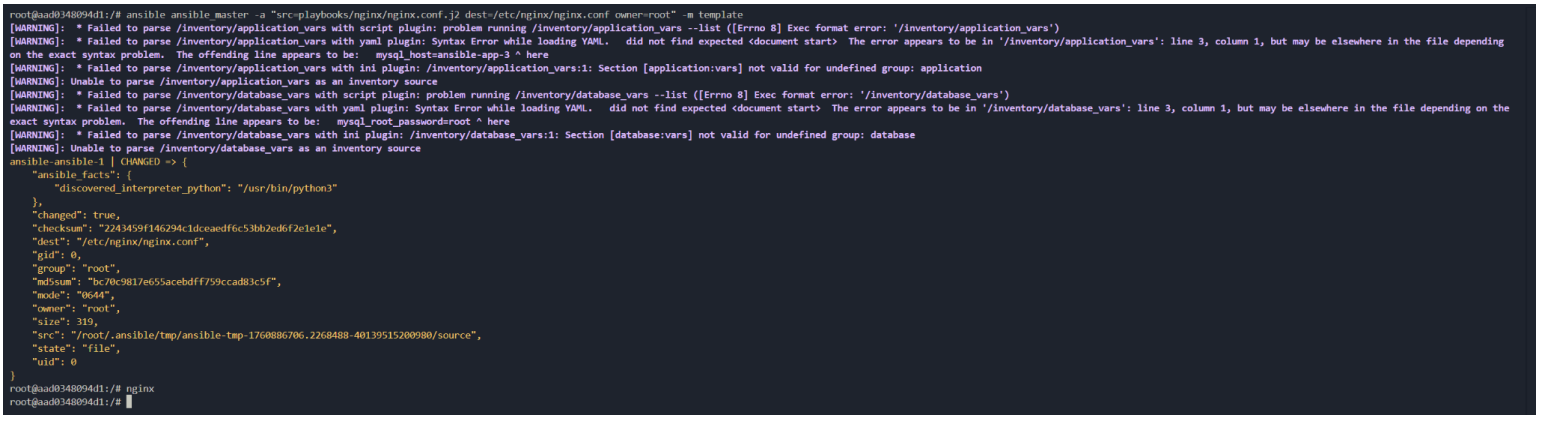
Задание 2: Подготовить работу приложений

```
lev@redmibook-15 /m/c/U/l/D/d/d/2/ansible (main)> docker exec -it aad034 bash
root@aad0348094d1:/# ansible-playbook /playbooks/playbook-application.yml
[WARNING]:  * Failed to parse /inventory/application_vars with script plugin: problem running /inventory/application_vars --list ([Errno 8] Exec format error: '/inventory/application_vars')
[WARNING]:  * Failed to parse /inventory/application_vars with yaml plugin: Syntax Error while loading YAML.   did not find expected <document start>  The error appears to be in '/inventory/application_vars': line 3, column 1, but may be elsewhere in the file depending
on the exact syntax problem.  The offending line appears to be:   mysql_host=ansible-app-3 ^ here
[WARNING]:  * Failed to parse /inventory/application_vars with ini plugin: /inventory/application_vars:1: Section [application:vars] not valid for undefined group: application
[WARNING]: Unable to parse /inventory/application_vars as an inventory source
[WARNING]:  * Failed to parse /inventory/database_vars with script plugin: problem running /inventory/database_vars --list ([Errno 8] Exec format error: '/inventory/database_vars')
[WARNING]:  * Failed to parse /inventory/database_vars with yaml plugin: Syntax Error while loading YAML.   did not find expected <document start>  The error appears to be in '/inventory/database_vars': line 3, column 1, but may be elsewhere in the file depending on the
exact syntax problem.  The offending line appears to be:   mysql_root_password=root ^ here
[WARNING]:  * Failed to parse /inventory/database_vars with ini plugin: /inventory/database_vars:1: Section [database:vars] not valid for undefined group: database
[WARNING]: Unable to parse /inventory/database_vars as an inventory source

PLAY [application] ***********************************************************************************************************************************************************************************************************************

TASK [Gathering Facts] *******************************************************************************************************************************************************************************************************************
ok: [ansible-app-1]
ok: [ansible-app-2]

TASK [Install MySQL client and other required packages] **********************************************************************************************************************************************************************************
[DEPRECATION WARNING]: Invoking "apt" only once while using a loop via squash_actions is deprecated. Instead of using a loop to supply multiple items and specifying `name: "{{ item }}"`, please use `name: ['mysql-client']` and remove the loop. This feature will be
removed in version 2.11. Deprecation warnings can be disabled by setting deprecation_warnings=False in ansible.cfg.
[DEPRECATION WARNING]: Invoking "apt" only once while using a loop via squash_actions is deprecated. Instead of using a loop to supply multiple items and specifying `name: "{{ item }}"`, please use `name: ['mysql-client']` and remove the loop. This feature will be
removed in version 2.11. Deprecation warnings can be disabled by setting deprecation_warnings=False in ansible.cfg.
changed: [ansible-app-1] => (item=['mysql-client'])
[WARNING]: Updating cache and auto-installing missing dependency: python3-apt
changed: [ansible-app-2] => (item=['mysql-client'])

TASK [Prepare app] ***********************************************************************************************************************************************************************************************************************
changed: [ansible-app-2]
changed: [ansible-app-1]

TASK [Installing python requirements for app] ********************************************************************************************************************************************************************************************
changed: [ansible-app-2]
changed: [ansible-app-1]

TASK [Prepare env for launch] ************************************************************************************************************************************************************************************************************
changed: [ansible-app-1]
changed: [ansible-app-2]

TASK [Killing previously started pythons (if any)] ***************************************************************************************************************************************************************************************
fatal: [ansible-app-2]: FAILED! => {"changed": true, "cmd": "ps aux | grep -i app.py | awk {'print $2'} | xargs kill -9", "delta": "0:00:00.009065", "end": "2025-10-19 15:07:03.809705", "msg": "non-zero return code", "rc": -9, "start": "2025-10-19 15:07:03.800640", "stder
r": "kill: (661): No such process", "stderr_lines": ["kill: (661): No such process"], "stdout": "", "stdout_lines": []}
...ignoring
fatal: [ansible-app-1]: FAILED! => {"changed": true, "cmd": "ps aux | grep -i app.py | awk {'print $2'} | xargs kill -9", "delta": "0:00:00.007713", "end": "2025-10-19 15:07:03.815152", "msg": "non-zero return code", "rc": -9, "start": "2025-10-19 15:07:03.807439", "stder
r": "kill: (661): No such process", "stderr_lines": ["kill: (661): No such process"], "stdout": "", "stdout_lines": []}
...ignoring

TASK [Start the app] *********************************************************************************************************************************************************************************************************************
changed: [ansible-app-1]
changed: [ansible-app-2]

PLAY RECAP *******************************************************************************************************************************************************************************************************************************
ansible-app-1              : ok=7    changed=6    unreachable=0    failed=0    skipped=0    rescued=0    ignored=1
ansible-app-2              : ok=7    changed=6    unreachable=0    failed=0    skipped=0    rescued=0    ignored=1
```

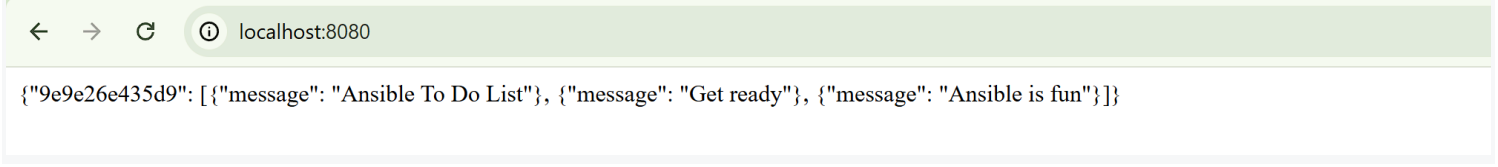
```
root@aad0348094d1:/# ssh ansible-app-1
root@ansible-app-1's password:
Welcome to Ubuntu 20.04.6 LTS (GNU/Linux 5.15.167.4-microsoft-standard-WSL2 x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.
Last login: Sun Oct 19 15:07:03 2025 from 172.19.0.5
root@9e9e26e435d9:~# ps ax | grep python
    683 ?        S      0:00 python3 app.py
    706 pts/0    S+     0:00 grep --color=auto python
root@9e9e26e435d9:~# netstat -nlp | grep 5000
tcp        0      0 0.0.0.0:5000            0.0.0.0:*               LISTEN      683/python3
root@9e9e26e435d9:~# exit
logout
Connection to ansible-app-1 closed.
root@aad0348094d1:/# ansible application -a "curl localhost:5000" -m shell
[WARNING]:  * Failed to parse /inventory/application_vars with script plugin: problem running /inventory/application_vars --list ([Errno 8] Exec format error: '/inventory/application_vars')
[WARNING]:  * Failed to parse /inventory/application_vars with yaml plugin: Syntax Error while loading YAML.   did not find expected <document start>  The error appears to be in '/inventory/application_vars': line 3, column 1, but may be elsewhere in the file depending
on the exact syntax problem.  The offending line appears to be:   mysql_host=ansible-app-3 ^ here
[WARNING]:  * Failed to parse /inventory/application_vars with ini plugin: /inventory/application_vars:1: Section [application:vars] not valid for undefined group: application
[WARNING]: Unable to parse /inventory/application_vars as an inventory source
[WARNING]:  * Failed to parse /inventory/database_vars with script plugin: problem running /inventory/database_vars --list ([Errno 8] Exec format error: '/inventory/database_vars')
[WARNING]:  * Failed to parse /inventory/database_vars with yaml plugin: Syntax Error while loading YAML.   did not find expected <document start>  The error appears to be in '/inventory/database_vars': line 3, column 1, but may be elsewhere in the file depending on the
exact syntax problem.  The offending line appears to be:   mysql_root_password=root ^ here
[WARNING]:  * Failed to parse /inventory/database_vars with ini plugin: /inventory/database_vars:1: Section [database:vars] not valid for undefined group: database
[WARNING]: Unable to parse /inventory/database_vars as an inventory source
[WARNING]: Consider using the get_url or uri module rather than running 'curl'.  If you need to use command because get_url or uri is insufficient you can add 'warn: false' to this command task or set 'command_warnings=False' in ansible.cfg to get rid of this message.
ansible-app-2 | CHANGED | rc=0 >>
{"6592d5982f50": [{"message": "Ansible To Do List"}, {"message": "Get ready"}, {"message": "Ansible is fun"}]}  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   110  100   110    0     0   4782      0 --:--:-- --:--:-- --:--:--  4782
ansible-app-1 | CHANGED | rc=0 >>
{"9e9e26e435d9": [{"message": "Ansible To Do List"}, {"message": "Get ready"}, {"message": "Ansible is fun"}]}  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   110  100   110    0     0   4583      0 --:--:-- --:--:-- --:--:--  4583
```

Задание 3: Установить балансировщик нагрузки nginx используя ad-hoc команды

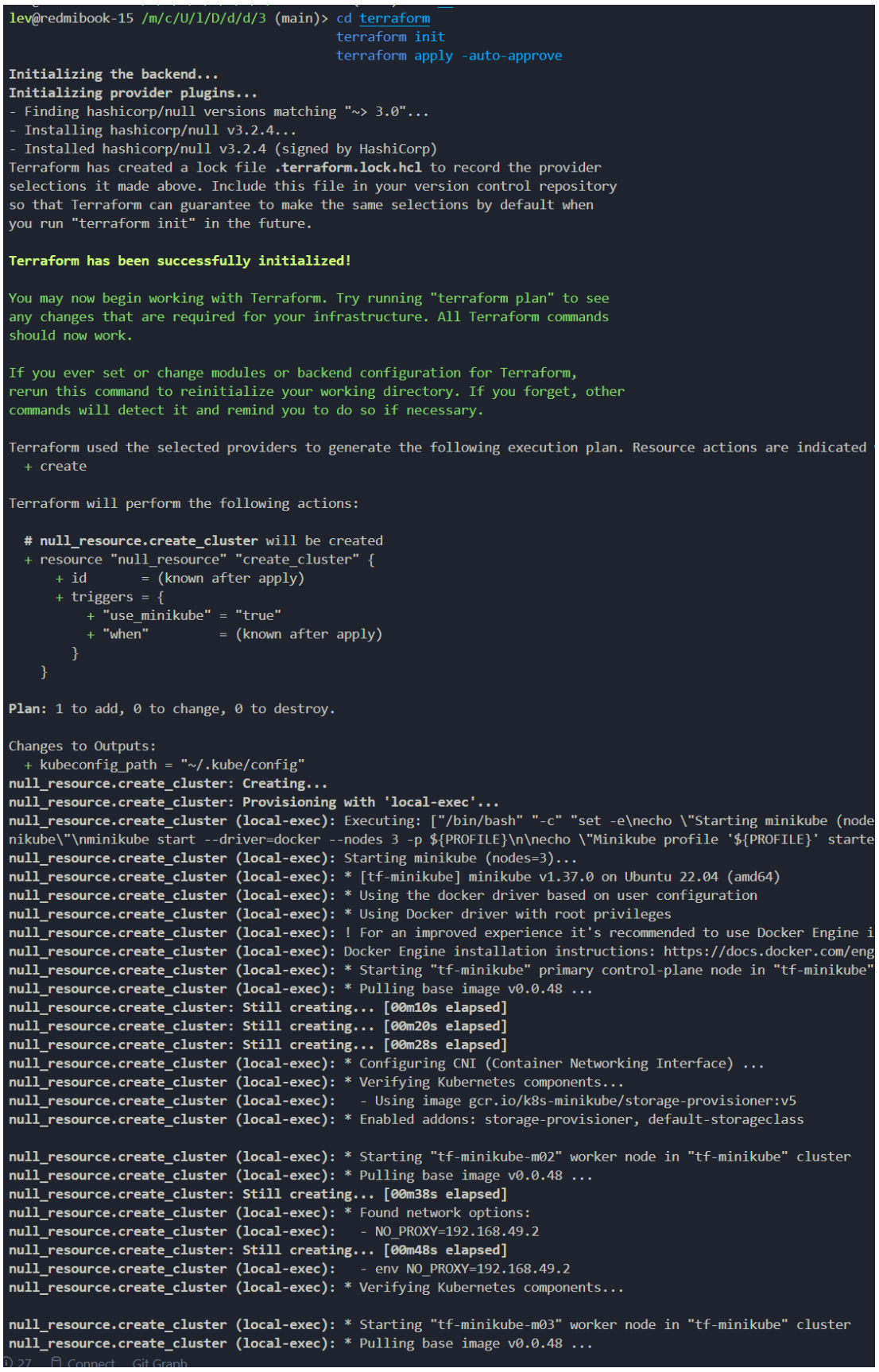
Поставим необходимый пакет nginx, давайте установим его без плейбука



Скопируем шаблон конфигурационного файла nginx, поставив в него хосты наших приложений, используется шаблизатор [jinja]
запустим `nginx` из консоли на хосте `ansible_ansible_1`



Задание 3. Создание кластера с помощью Terraform и Ansible локально



```
null_resource.create_cluster: Still creating... [00m10s elapsed]
null_resource.create_cluster: Still creating... [00m20s elapsed]
null_resource.create_cluster: Still creating... [00m28s elapsed]
null_resource.create_cluster (local-exec): * Configuring CNI (Container Networking Interface) ...
null_resource.create_cluster (local-exec): * Verifying Kubernetes components...
null_resource.create_cluster (local-exec):   - Using image gcr.io/k8s-minikube/storage-provisioner:v5
null_resource.create_cluster (local-exec): * Enabled addons: storage-provisioner, default-storageclass

null_resource.create_cluster (local-exec): * Starting "tf-minikube-m02" worker node in "tf-minikube" cluster
null_resource.create_cluster (local-exec): * Pulling base image v0.0.48 ...
null_resource.create_cluster: Still creating... [00m38s elapsed]
null_resource.create_cluster (local-exec): * Found network options:
null_resource.create_cluster (local-exec):   - NO_PROXY=192.168.49.2
null_resource.create_cluster: Still creating... [00m48s elapsed]
null_resource.create_cluster (local-exec):   - env NO_PROXY=192.168.49.2
null_resource.create_cluster (local-exec): * Verifying Kubernetes components...

null_resource.create_cluster (local-exec): * Starting "tf-minikube-m03" worker node in "tf-minikube" cluster
null_resource.create_cluster (local-exec): * Pulling base image v0.0.48 ...
null_resource.create_cluster: Still creating... [00m55s elapsed]
null_resource.create_cluster (local-exec): * Found network options:
null_resource.create_cluster (local-exec):   - NO_PROXY=192.168.49.2,192.168.49.3
null_resource.create_cluster (local-exec):   - env NO_PROXY=192.168.49.2
null_resource.create_cluster (local-exec):   - env NO_PROXY=192.168.49.2,192.168.49.3
null_resource.create_cluster: Still creating... [01m05s elapsed]
null_resource.create_cluster (local-exec): * Verifying Kubernetes components...

null_resource.create_cluster (local-exec): ! /usr/local/bin/kubectl is version 1.30.2, which may have incompatibilities with Kubernetes 1.34.0.
null_resource.create_cluster (local-exec):   - Want kubectl v1.34.0? Try 'minikube kubectl -- get pods -A'
null_resource.create_cluster (local-exec): * Done! kubectl is now configured to use "tf-minikube" cluster and "default" namespace by default
null_resource.create_cluster (local-exec): Minikube profile 'tf-minikube' started with 3 nodes.
null_resource.create_cluster (local-exec): error: no context exists with the name: "minikube"
null_resource.create_cluster (local-exec): NAME            STATUS   ROLES           AGE   VERSION   INTERNAL-IP    EXTERNAL-IP   OS-IMAGE             KERNEL-VERSION                     CONTAINER-RUNTIME
null_resource.create_cluster (local-exec): tf-minikube     Ready    control-plane   40s   v1.34.0   192.168.49.2   <none>        Ubuntu 22.04.5 LTS   5.15.167.4-microsoft-standard-WSL2   docker://28.4.0
null_resource.create_cluster (local-exec): tf-minikube-m02   Ready    <none>          16s   v1.34.0   192.168.49.3   <none>        Ubuntu 22.04.5 LTS   5.15.167.4-microsoft-standard-WSL2   docker://28.4.0
null_resource.create_cluster (local-exec): tf-minikube-m03   Ready    <none>          1s    v1.34.0   192.168.49.4   <none>        Ubuntu 22.04.5 LTS   5.15.167.4-microsoft-standard-WSL2   docker://28.4.0
null_resource.create_cluster: Creation complete after 1m6s [id=6064720969169904451]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.

Outputs:

kubeconfig_path = "~/.kube/config"
```

```
lev@redmibook-15 /m/c/U/l/D/d/d/3/terraform (main)> kubectl get nodes -o wide
NAME              STATUS   ROLES           AGE    VERSION   INTERNAL-IP    EXTERNAL-IP   OS-IMAGE             KERNEL-VERSION                     CONTAINER-RUNTIME
tf-minikube       Ready    control-plane   116s   v1.34.0   192.168.49.2   <none>        Ubuntu 22.04.5 LTS   5.15.167.4-microsoft-standard-WSL2   docker://28.4.0
tf-minikube-m02   Ready    <none>          92s    v1.34.0   192.168.49.3   <none>        Ubuntu 22.04.5 LTS   5.15.167.4-microsoft-standard-WSL2   docker://28.4.0
tf-minikube-m03   Ready    <none>          77s    v1.34.0   192.168.49.4   <none>        Ubuntu 22.04.5 LTS   5.15.167.4-microsoft-standard-WSL2   docker://28.4.0
lev@redmibook-15 /m/c/U/l/D/d/d/3/terraform (main)>
```

```
lev@redmibook-15 /m/c/U/l/D/d/d/3 (main)> cd ansible
                                          ansible-playbook -i inventory.ini deploy-nginx.yml

PLAY [Deploy nginx DaemonSet to all nodes] *************************************************************************************************************************

TASK [Wait for kubeconfig and API to be ready] *********************************************************************************************************************
ok: [localhost]

TASK [Apply nginx DaemonSet (kubectl apply)] ***********************************************************************************************************************
changed: [localhost]

TASK [Show kubectl apply output] ***********************************************************************************************************************************
ok: [localhost] => {
    "apply_out.stdout_lines": [
        "daemonset.apps/nginx-daemonset created"
    ]
}

TASK [Wait for DaemonSet pods to be ready (max 120s)] **************************************************************************************************************
changed: [localhost]

TASK [Show pods (wide)] ********************************************************************************************************************************************
changed: [localhost]

TASK [debug] *******************************************************************************************************************************************************
ok: [localhost] => {
    "pods.stdout_lines": [
        "NAME                    READY   STATUS    RESTARTS   AGE   IP           NODE              NOMINATED NODE   READINESS GATES",
        "nginx-daemonset-4gs8c   1/1     Running   0          22s   10.244.2.2   tf-minikube-m03   <none>           <none>",
        "nginx-daemonset-6f81r   1/1     Running   0          22s   10.244.0.3   tf-minikube       <none>           <none>",
        "nginx-daemonset-zd5dd   1/1     Running   0          22s   10.244.1.2   tf-minikube-m02   <none>           <none>"
    ]
}

PLAY RECAP *********************************************************************************************************************************************************
localhost                  : ok=6    changed=3    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

```
lev@redmibook-15 /m/c/U/l/D/d/d/3/ansible (main)> kubectl get daemonset -n default
                                                  kubectl get pods -o wide -n default
                                                  kubectl get pods -o wide -n default --selector=app=nginx
NAME              DESIRED   CURRENT   READY   UP-TO-DATE   AVAILABLE   NODE SELECTOR   AGE
nginx-daemonset   3         3         3       3            3           <none>          67s
NAME                    READY   STATUS    RESTARTS   AGE   IP           NODE              NOMINATED NODE   READINESS GATES
nginx-daemonset-4gs8c   1/1     Running   0          67s   10.244.2.2   tf-minikube-m03   <none>           <none>
nginx-daemonset-6f8lr   1/1     Running   0          67s   10.244.0.3   tf-minikube       <none>           <none>
nginx-daemonset-zd5dd   1/1     Running   0          67s   10.244.1.2   tf-minikube-m02   <none>           <none>
NAME                    READY   STATUS    RESTARTS   AGE   IP           NODE              NOMINATED NODE   READINESS GATES
nginx-daemonset-4gs8c   1/1     Running   0          67s   10.244.2.2   tf-minikube-m03   <none>           <none>
nginx-daemonset-6f8lr   1/1     Running   0          67s   10.244.0.3   tf-minikube       <none>           <none>
nginx-daemonset-zd5dd   1/1     Running   0          67s   10.244.1.2   tf-minikube-m02   <none>           <none>
lev@redmibook-15 /m/c/U/l/D/d/d/3/ansible (main)>
```

4. Ответы на контрольные вопросы.

1. **Для каких задач применяется Ansible?**

Автоматизация конфигурации, деплоя приложений, управления состоянием серверов и оркестрации задач (безагентный подход).

2. **Какие способы управления серверами применяются в Ansible?**

Подключения: SSH (по умолчанию), Paramiko, WinRM (Windows), local, docker, lxc и др. (модули/плагины соединения).

3. **Что входит в состав файла ansible playbook?**

Плей(ы) с полями: hosts, tasks, handlers, vars, roles, pre_tasks, post_tasks, become, vars_files и пр.

4. **Что такое группы узлов (хостов) в Ansible?**

Логические объединения хостов в инвентаре для удобного таргетинга и задания общих переменных (group_vars).

5. **Как одной командой выполнить ping группы узлов (хостов)?**

ansible <group> -m ping

6. **Что такое роль Ansible?**

Модульная, реиспользуемая структура (tasks, handlers, defaults, vars, files, templates, meta) для организации и переиспользования конфигураций.

7. **Что такое файл инвентаризации Ansible и как он используется?**

Файл (INI/YAML или динамический) с перечнем хостов и групп; Ansible использует его для определения целевых машин и переменных.

8. **Что такое специальная команда Ansible?**

(Ad-hoc) одноразовая CLI-команда для быстрого выполнения модуля без playbook, например ansible <host> -m shell -a "uptime".

9. **Как перезапустить службу с помощью Ansible?**

Через модуль service/systemd. Пример ad-hoc:

ansible <group> -m service -a "name=httpd state=restarted"

В playbook - использовать service/systemd с state: restarted.

10. **Что такое HashiCorp Terraform?**

Инструмент для декларативного создания и управления инфраструктурой (IaC) через провайдеры.

11. **Что такое инфраструктура как код (IaC)?**

Подход - описывать и управлять инфраструктурой в виде читаемого кода, что дает версионирование, повторяемость и автоматизацию.

12. **Как установить Terraform?**

Скачать бинарь с сайта HashiCorp и поместить в PATH, либо через пакетные менеджеры (brew/apt/yum/choco).

13. **Что такое провайдер (поставщик) в Terraform?**

Плагин, который управляет ресурсами конкретной платформы (AWS, Azure, GCP, vSphere и т.д.).

14. **Объясните базовую структуру файла конфигурации Terraform.**

Блоки: provider (настройка провайдера), resource (описывает объект), variable (входные параметры), output (выходные значения), module (повторно используемые компоненты).

15. Что такое файл состояния Terraform?

terraform.tfstate - JSON-файл, где хранится текущее известное состояние инфраструктуры и соответствие ресурсов к конфигурации.

16. Для чего нужна terraform destroy команда?

Удаляет ресурсы, созданные Terraform, - полное уничтожение управляемой инфраструктуры.

17. Что такое состояние в terraform.

Модель текущих ресурсов и их атрибутов, используемая для планирования и применения изменений; может храниться локально или в удалённом бэкенде.

18. Как определить зависимости в Terraform?

Зависимости выводятся автоматически через ссылки на атрибуты других ресурсов. Для явных зависимостей используется depends_on.