Ansible

Lazy Engineer's Best Friend!

Marco Lampis
Presents

How many coffees have you wasted setting up your dev environment?

How many coffees have you wasted setting up your dev environment?

Installing tools

Configuring Shell

Multiple coffe breaks

Time wasted

Environment setup

Traditional way (4h)

- Install tools manually
- Configure shell
- Wait... and wait...

Ansible way (5m)

```
ansible-playbook setup.yml
```

- Enjoy your coffee
- Feel at home

Meet

Minimal

Just SSH and Python
No agents, no complications

Consistent

Same result, every time, everywhere

Safe

Agentless architecture with OpenSSH security

Reliable

Idempotent operations prevent unexpected changes

Use cases

The route for a boring-repeatable life

Configuration Management

Keep your systems in the desired state

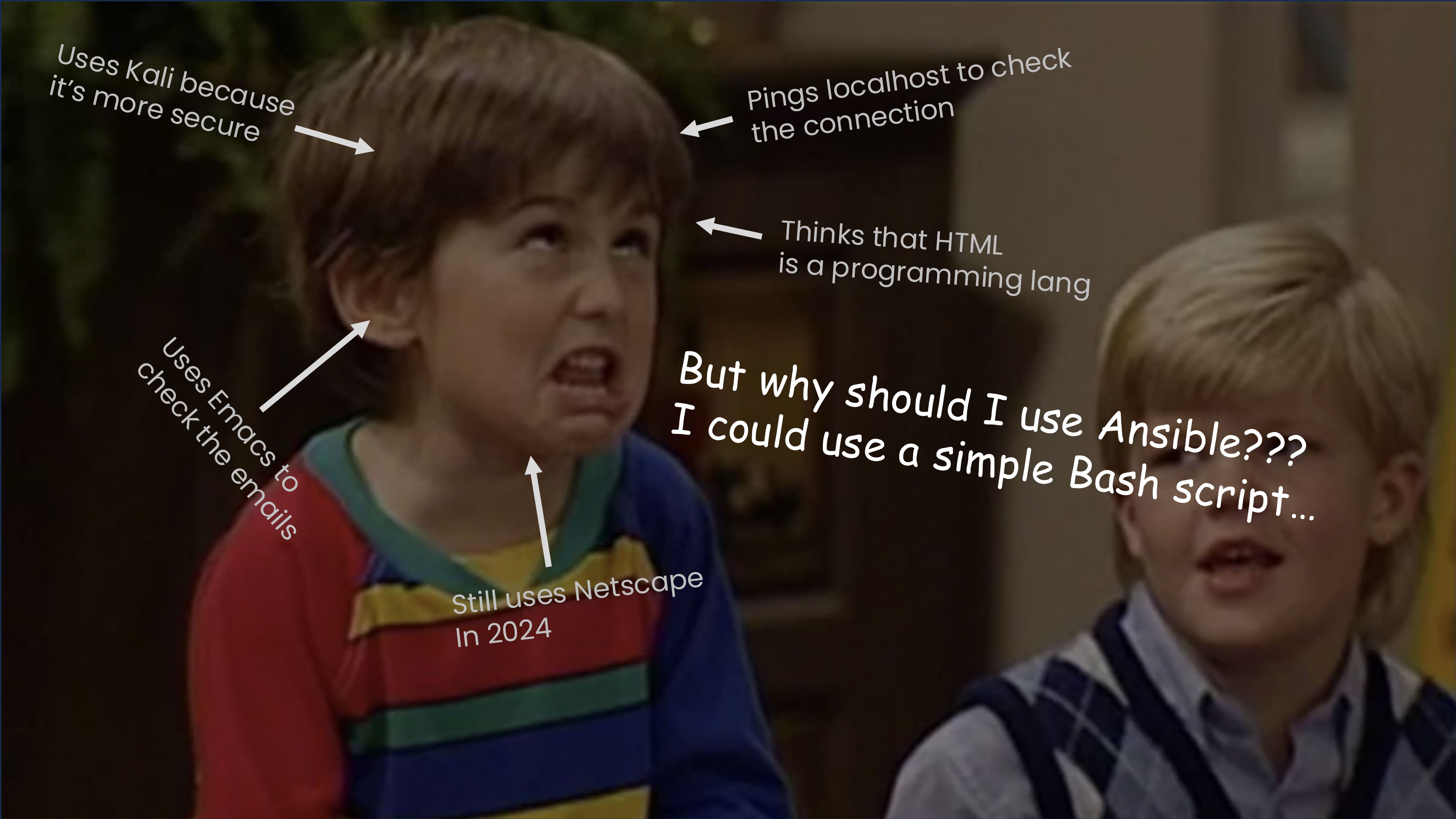
Application Deployment

Deploy software, databases and manage services

Infrastructure Automation

Set up new servers, configure networks, and manage cloud resources

But why should I use Ansible???
I could use a simple Bash script…



Uses Kali because
it’s more secure

Pings localhost to check
the connection

Thinks that HTML
is a programming lang

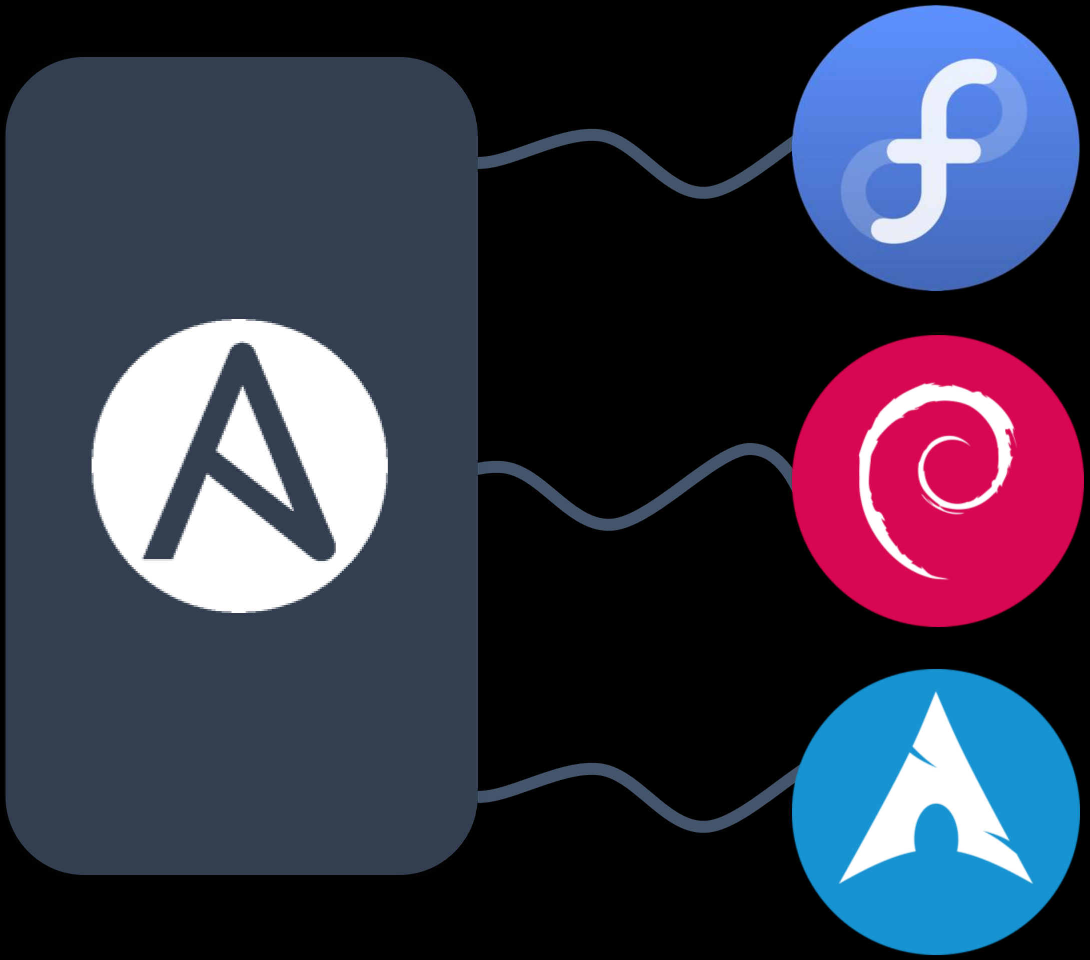
Uses Emacs to
check the emails

But why should I use Ansible???
I could use a simple Bash script…

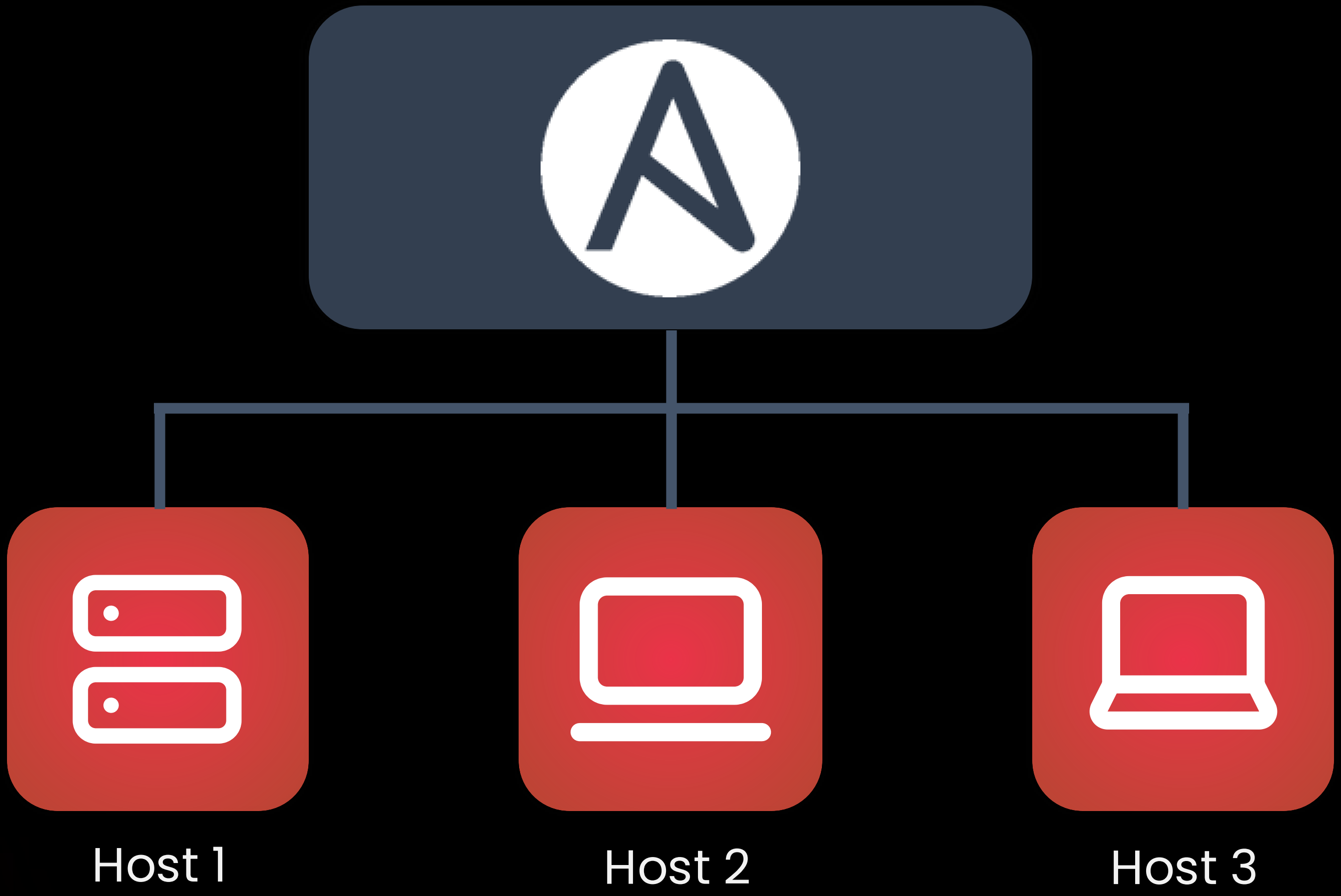
Still uses Netscape
In 2024

1) Every dev hates Bash

2) It works on multiple OS



2) It works on multiple nodes



3) You all want to be **LAZY**

Let

Ansible

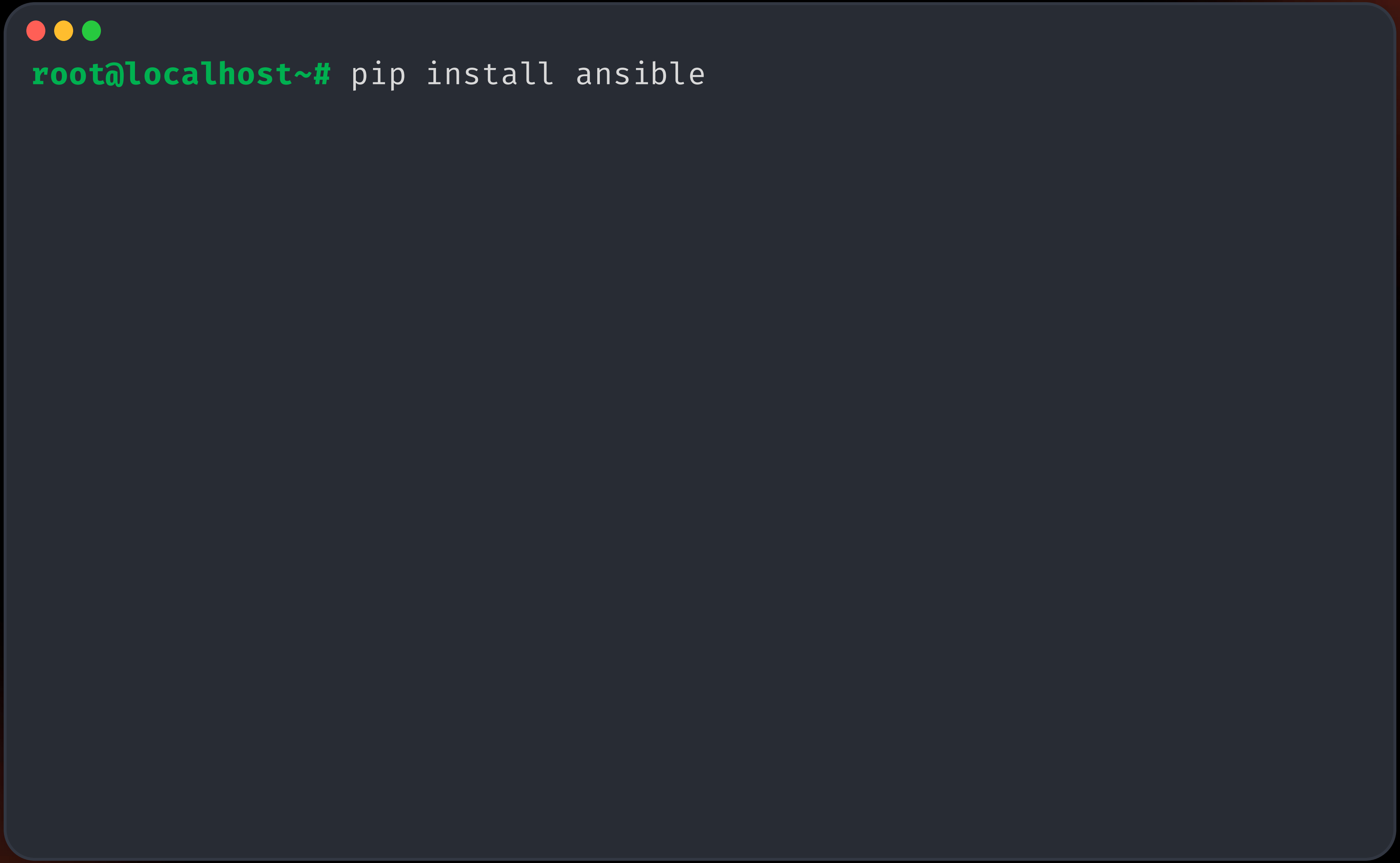
automatiZe

Your work

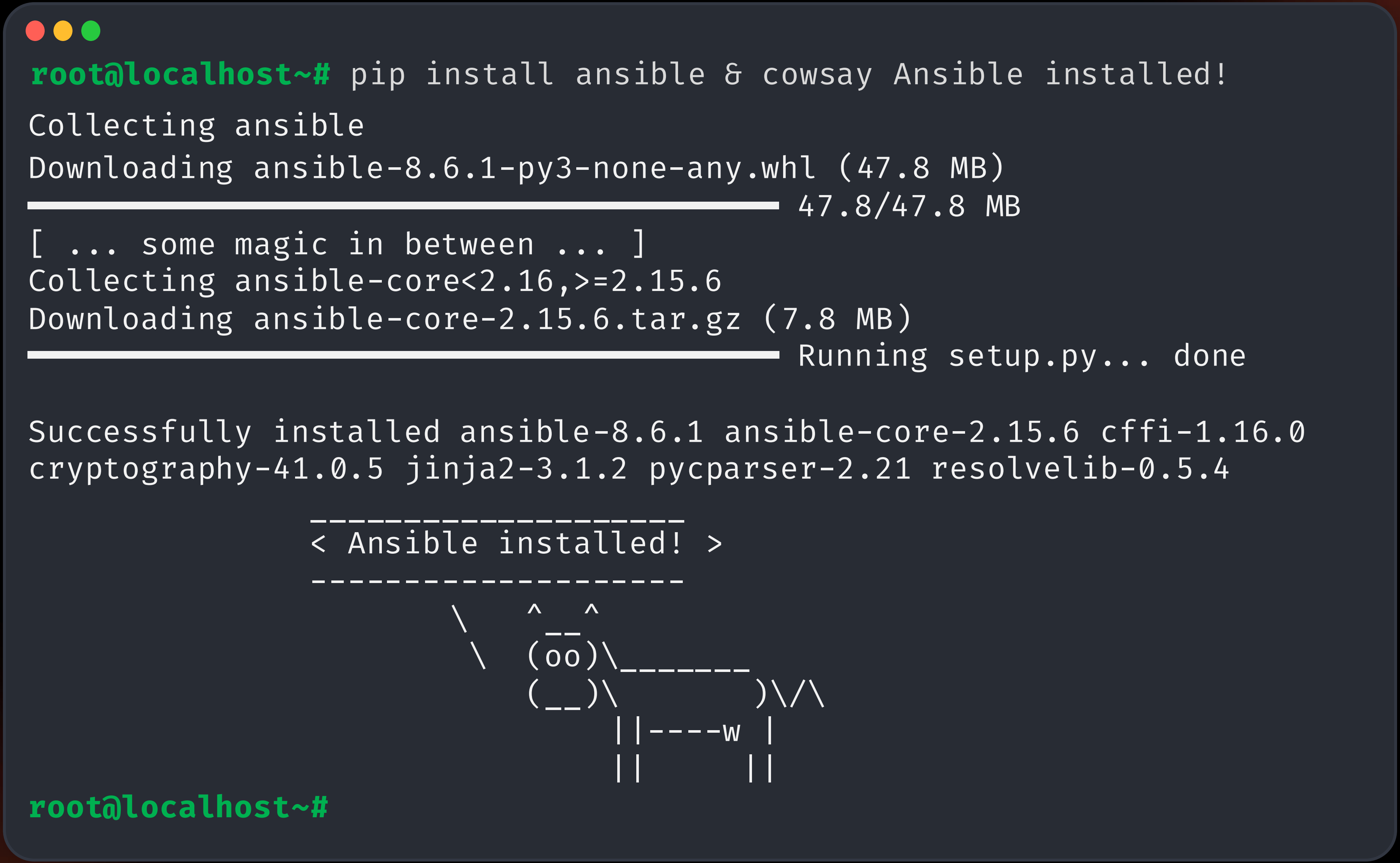
I choose a lazy person to do a hard job. Because a lazy person will find an easy way to do it.

Bill Gates (the guy from Microsoft)

ANSIBLE 101



```
root@localhost~# pip install ansible
```



```
root@localhost~# pip install ansible & cowsay Ansible installed!
Collecting ansible
Downloading ansible-8.6.1-py3-none-any.whl (47.8 MB)
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 47.8/47.8 MB
[ ... some magic in between ... ]
Collecting ansible-core<2.16,>=2.15.6
Downloading ansible-core-2.15.6.tar.gz (7.8 MB)
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ Running setup.py... done

Successfully installed ansible-8.6.1 ansible-core-2.15.6 cffi-1.16.0
cryptography-41.0.5 jinja2-3.1.2 pycparser-2.21 resolvelib-0.5.4

               ____________________
               < Ansible installed! >
               --------------------
                       \   ^__^
                        \  (oo)\_______
                           (__)\       )\/\
                               ||----w |
                               ||     ||
root@localhost~#
```

Playbooks

A YAML recipe that describes how to automate your work

Hosts

The devices which will execute the configuration

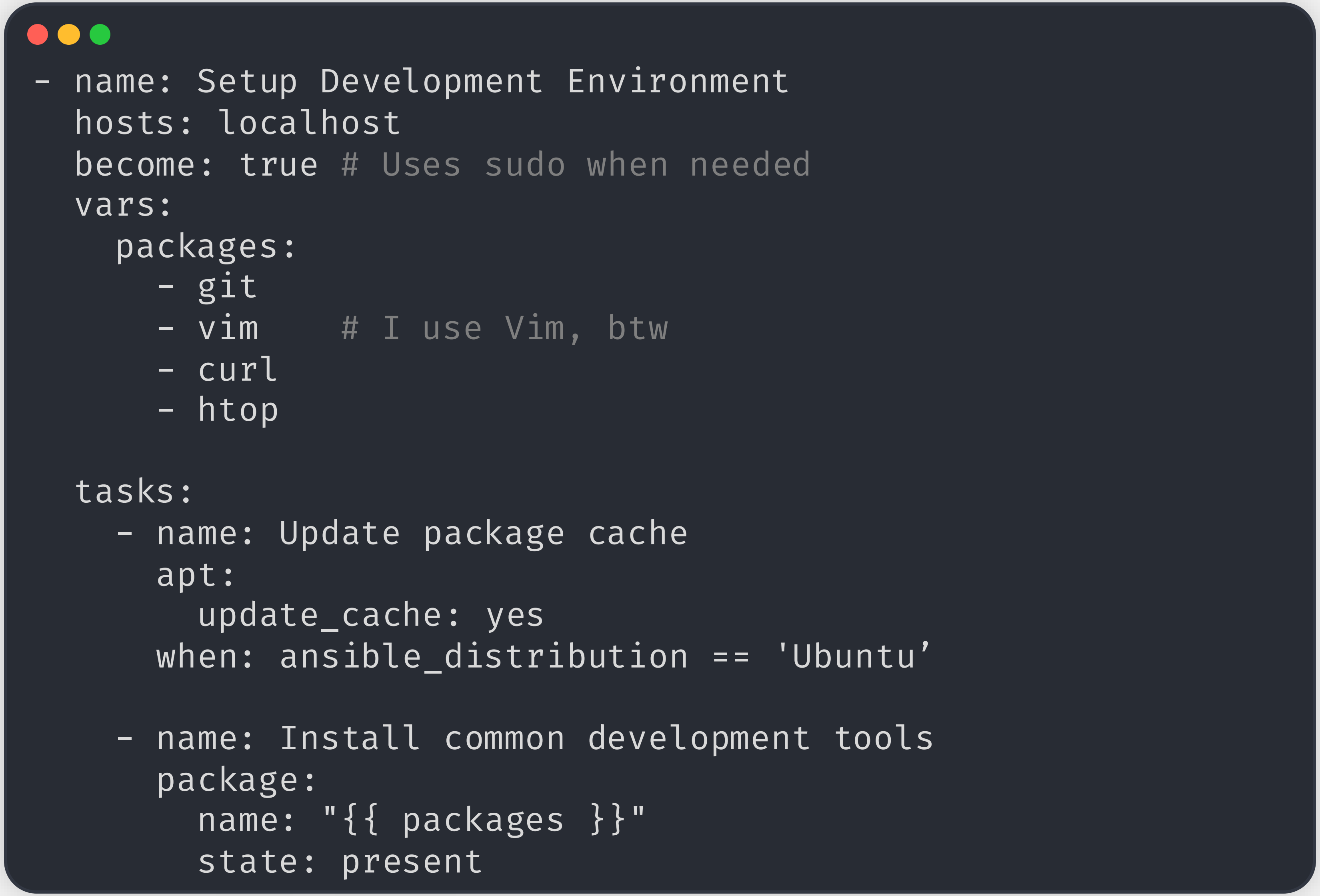
Tasks

The instructions to perform

Variables

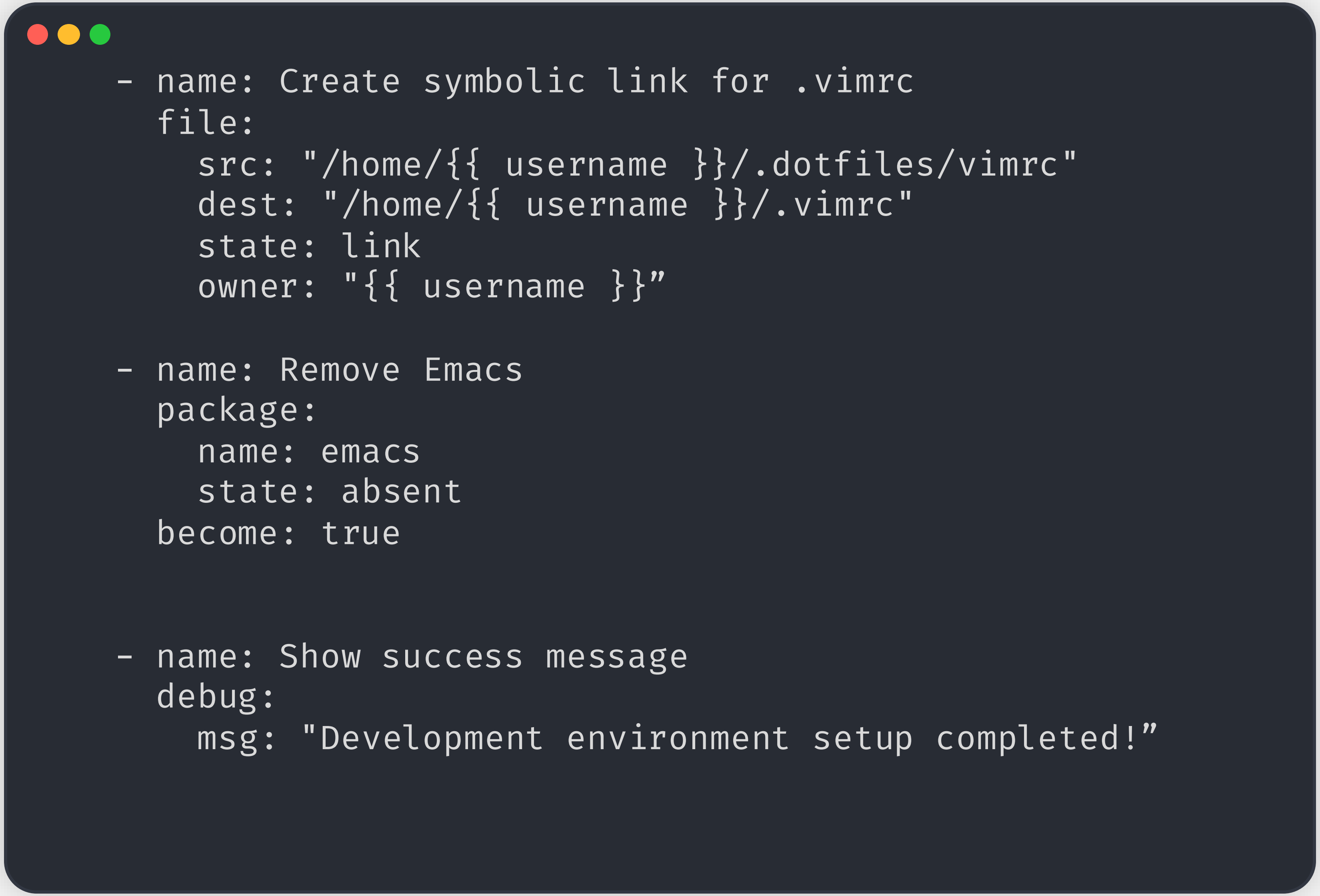
Values that will be replaced with your settings

Playbooks

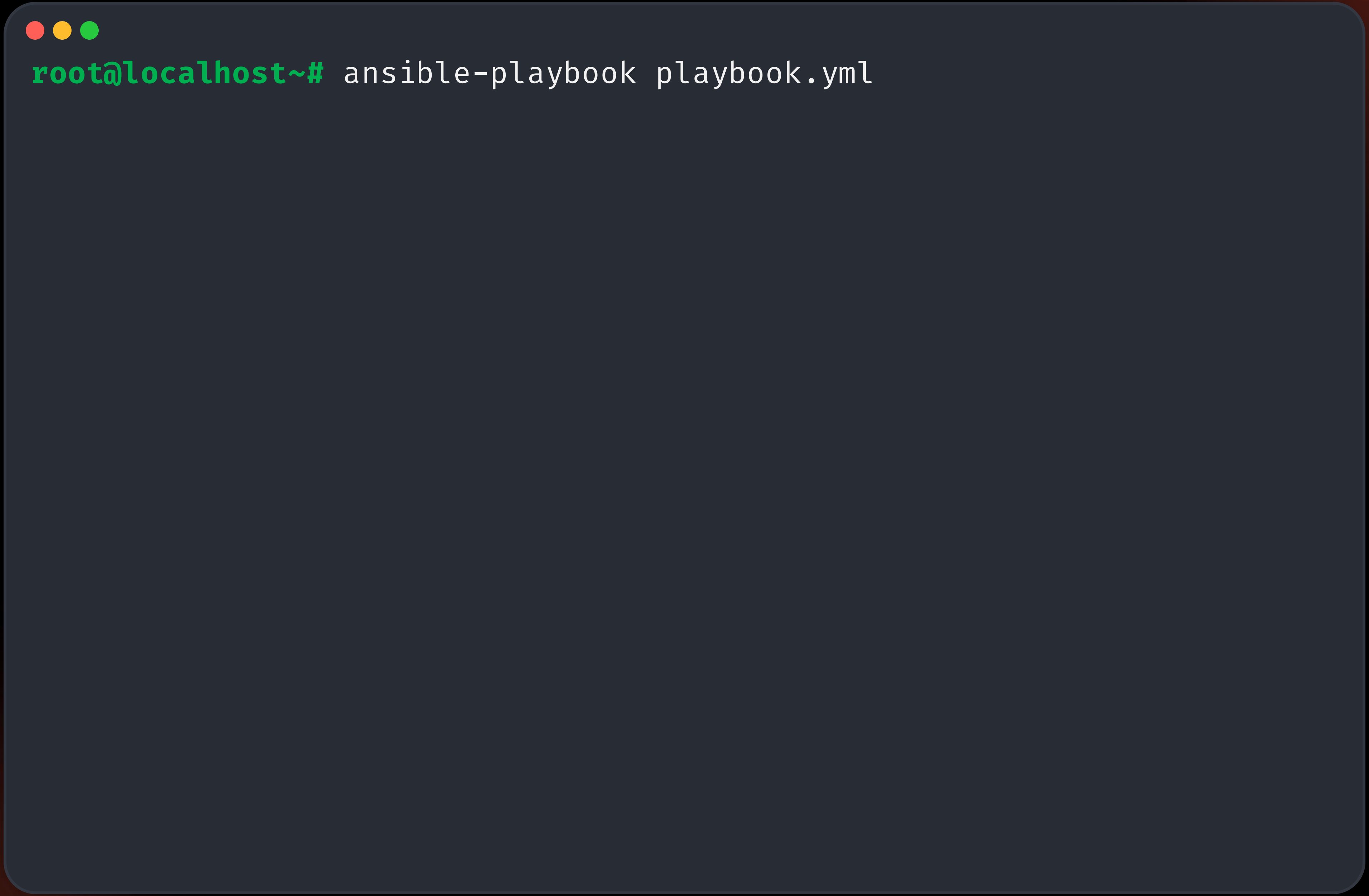


```yaml
- name: Setup Development Environment
  hosts: localhost
  become: true # Uses sudo when needed
  vars:
    packages:
      - git
      - vim    # I use Vim, btw
      - curl
      - htop

  tasks:
    - name: Update package cache
      apt:
        update_cache: yes
      when: ansible_distribution == 'Ubuntu’

    - name: Install common development tools
      package:
        name: "{{ packages }}"
        state: present
```

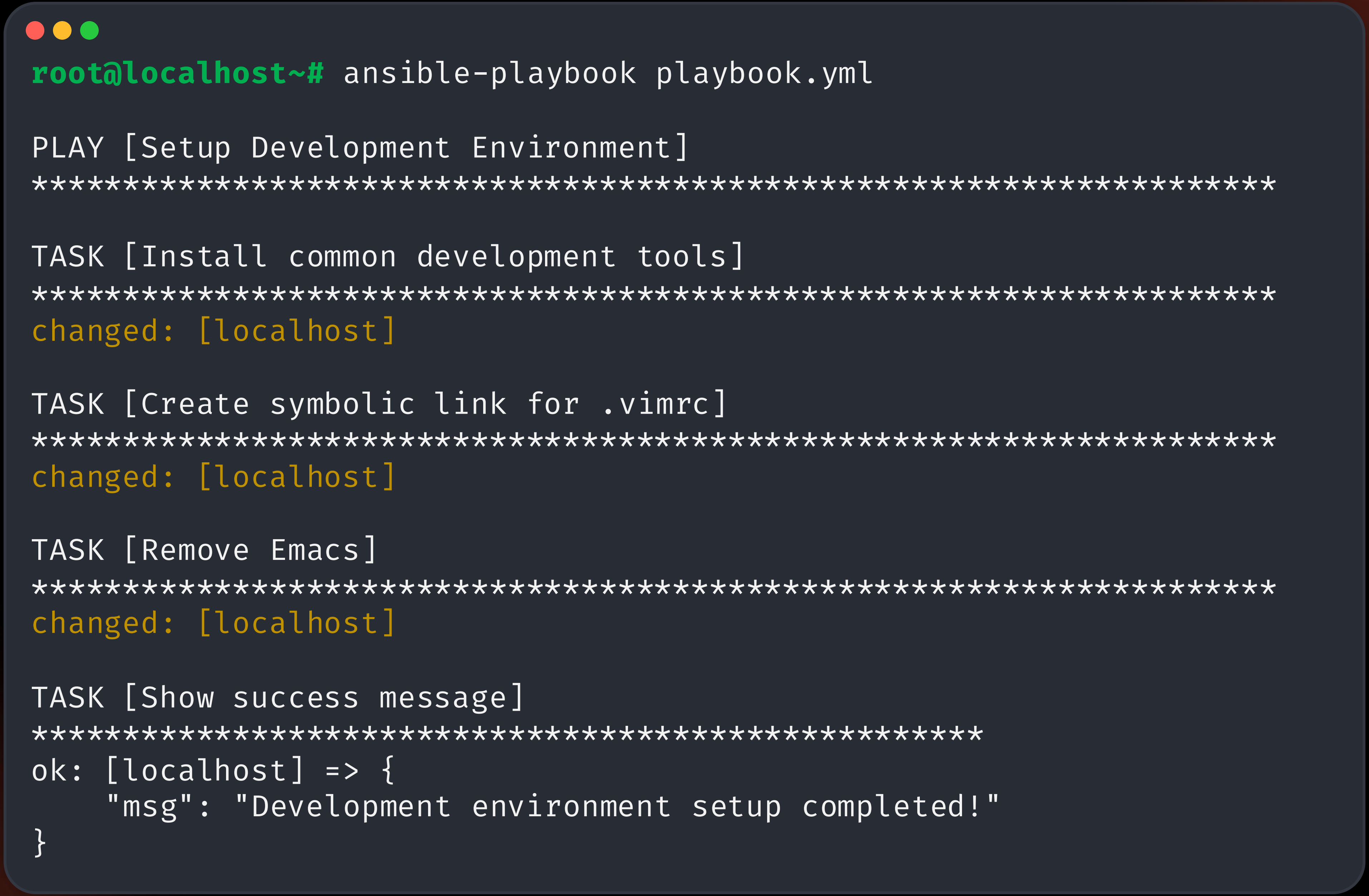
Playbooks



```
- name: Create symbolic link for .vimrc
  file:
    src: "/home/{{ username }}/.dotfiles/vimrc"
    dest: "/home/{{ username }}/.vimrc"
    state: link
    owner: "{{ username }}”

- name: Remove Emacs
  package:
    name: emacs
    state: absent
  become: true


- name: Show success message
  debug:
    msg: "Development environment setup completed!”
```



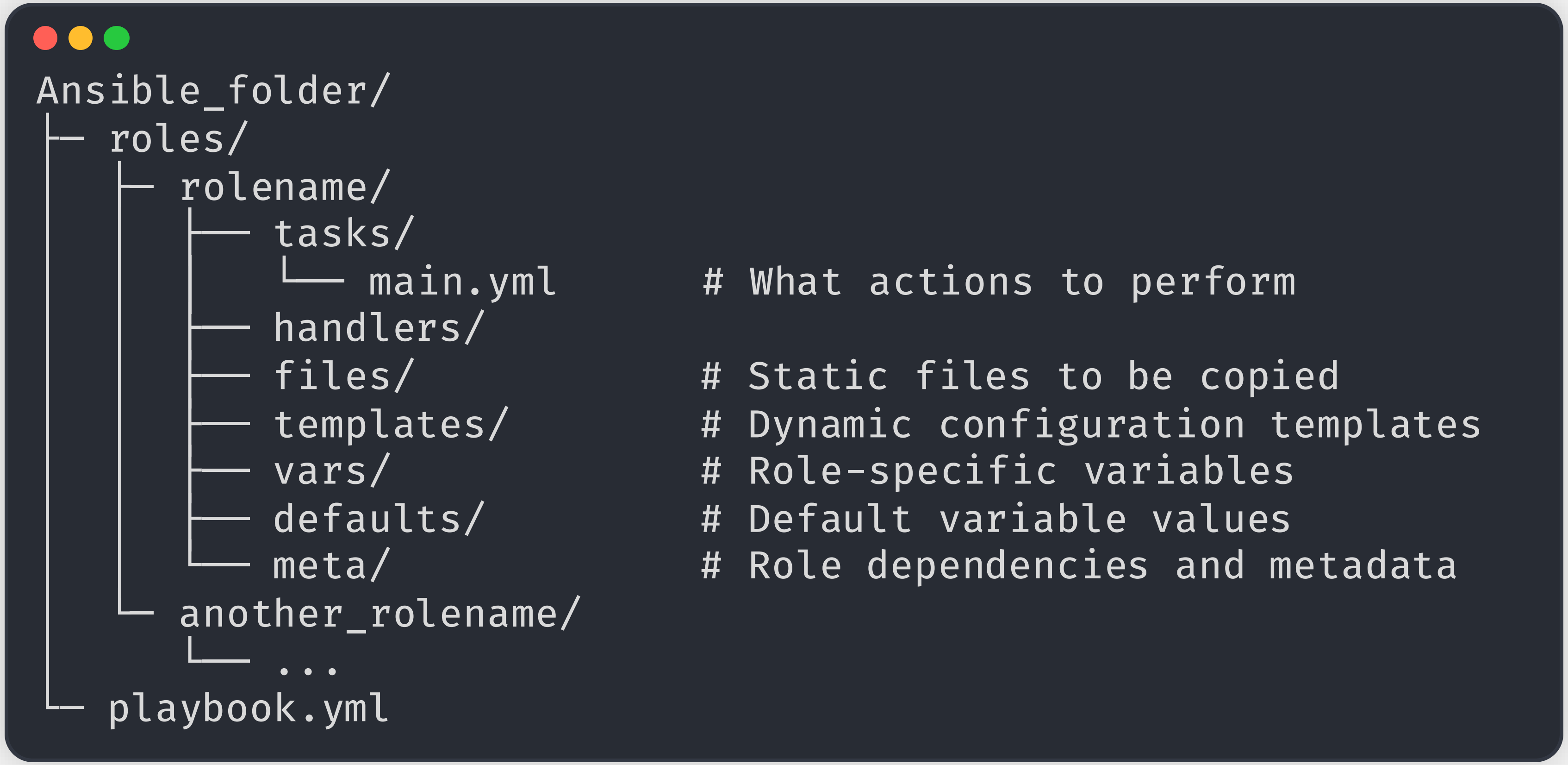
```
root@localhost~# ansible-playbook playbook.yml
```



```
root@localhost~# ansible-playbook playbook.yml

PLAY [Setup Development Environment]
********************************************************************

TASK [Install common development tools]
********************************************************************
changed: [localhost]

TASK [Create symbolic link for .vimrc]
********************************************************************
changed: [localhost]

TASK [Remove Emacs]
********************************************************************
changed: [localhost]

TASK [Show success message]
****************************************************
ok: [localhost] => {
    "msg": "Development environment setup completed!"
}
```

Roles

A modular and reusable set of tasks, files and configurations



```
Ansible_folder/
├─ roles/
│  ├─ rolename/
│  │  ├── tasks/
│  │  │   └── main.yml      # What actions to perform
│  │  ├── handlers/
│  │  ├── files/            # Static files to be copied
│  │  ├── templates/        # Dynamic configuration templates
│  │  ├── vars/             # Role-specific variables
│  │  ├── defaults/         # Default variable values
│  │  └── meta/             # Role dependencies and metadata
│  └─ another_rolename/
│     └── ...
└─ playbook.yml
```

GALAXY

Ansible Galaxy

The PIP repository for Ansible roles

Brew

Automatically install MacOS packages using the Homebrew repositories

Mac OS

Setup the Mac OS system settings such as autocompletion and keyboard

NPM

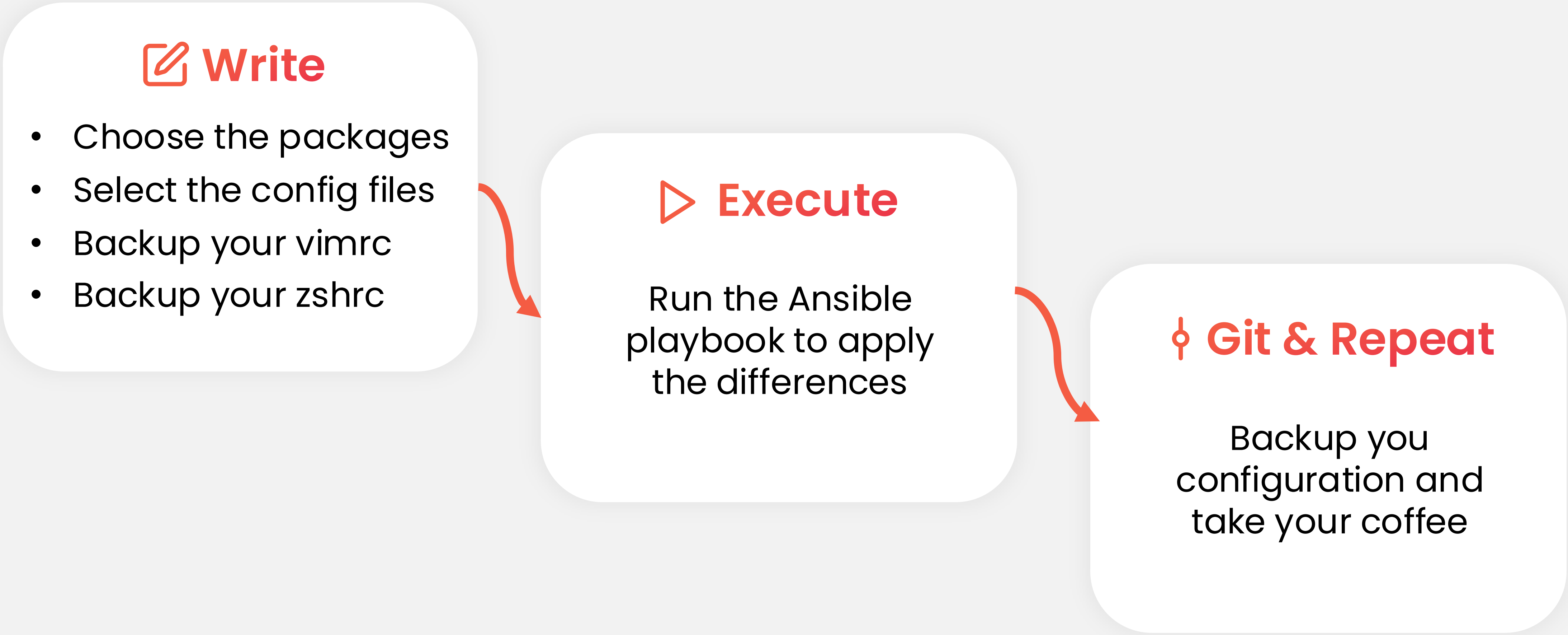
Install Node packages from NPM

PIP

Install Python packages from PIP

A dotfiles use case

Configuration management for your computer



Write

- Choose the packages
- Select the config files
- Backup your vimrc
- Backup your zshrc

Execute

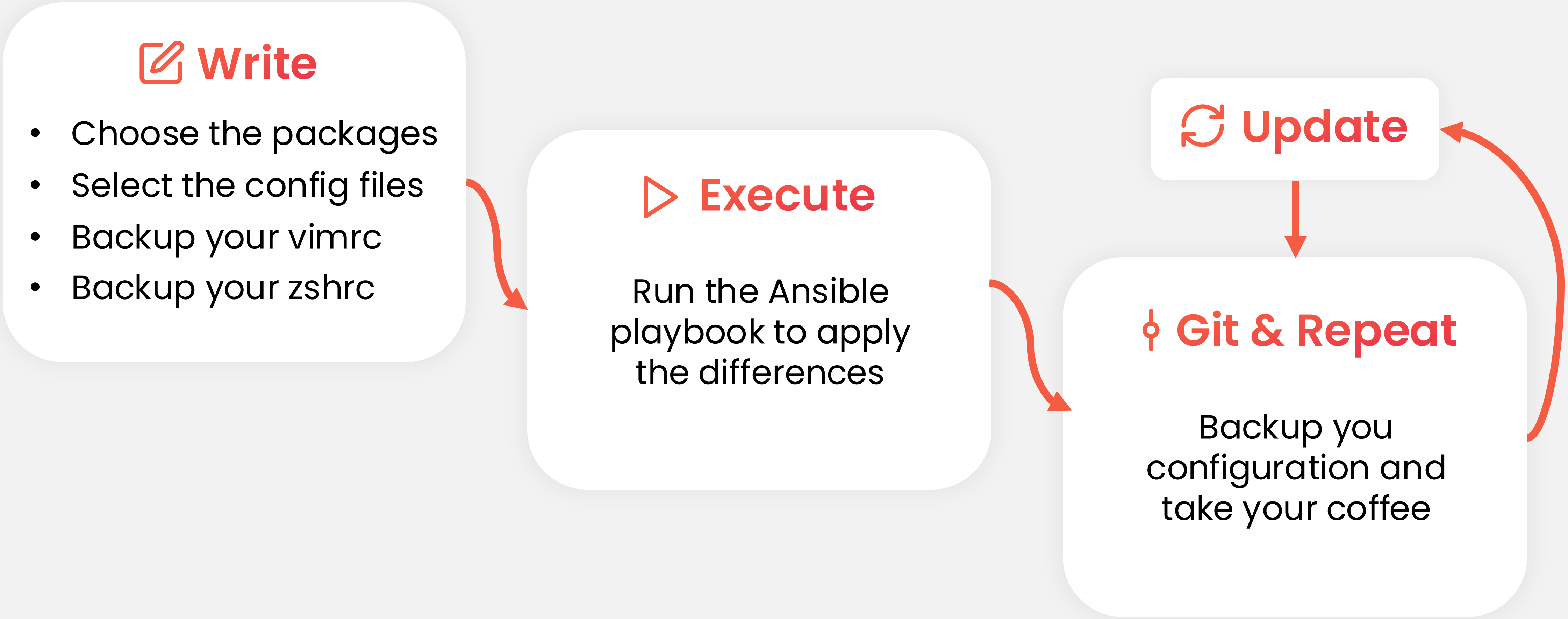
Run the Ansible playbook to apply the differences

Git & Repeat

Backup you configuration and take your coffee

A dotfiles use case

Configuration management for your computer



Write

- Choose the packages
- Select the config files
- Backup your vimrc
- Backup your zshrc

Execute

Run the Ansible playbook to apply the differences

Git & Repeat

Backup you configuration and take your coffee

Update

The moral?

Save time in the long run

Make fewer mistakes

Create reusable solutions

The moral?

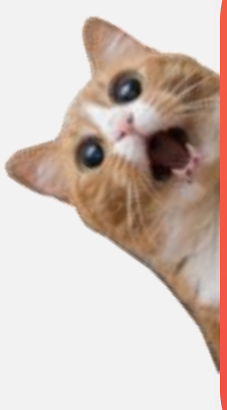
More time for coffee

Save time in the long run

M

Create reusable solutions

The moral?



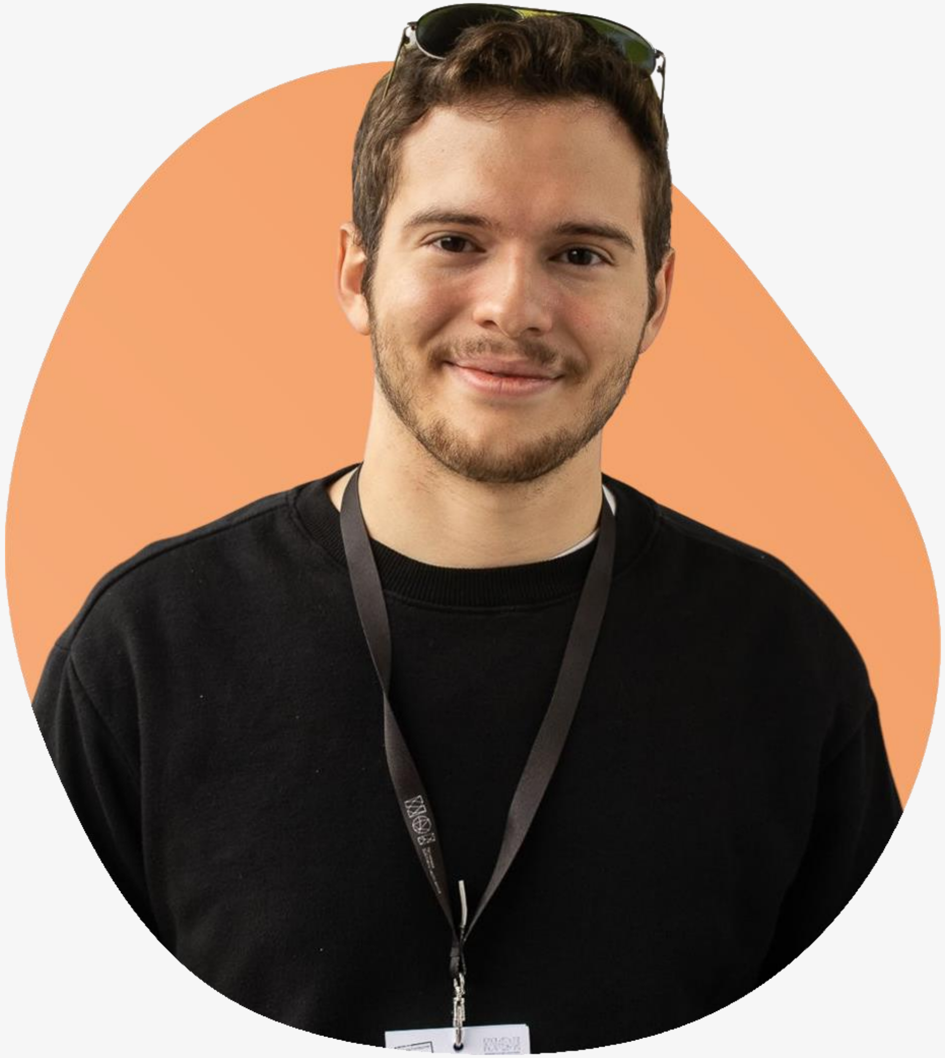
More time for coffee

Save **time** in the long run

Create **reusable** solutions

Marco Lampis

- Ex **Software Engineer Intern** @Amazon
- MSc student @Politecnico di Torino
- **Tutor** of Computer Science



Give a feedback!

Contacts

https://mlampis.dev

lampis.marco2000@gmail.com

mlampis

Guray00

m.lampis00