## **TD 1: Install kubernetes using minikube**

## **Install minikube**

Minikube is a great tool to create local kubernetes cluster. It is able to run docker containers within a local virtual machine behind the scenes for you, relatively the same way it would work in a production environment.

### **Installation requirements**

Here is the specs you'll need to meet to be able to install minikube:

* 2 CPUs or more
* 2GB of free memory
* 20GB of free disk space
* Internet connection
* Container or virtual machine manager, such as: [Docker](https://minikube.sigs.k8s.io/docs/drivers/docker/), [Hyperkit](https://minikube.sigs.k8s.io/docs/drivers/hyperkit/), [Hyper-V](https://minikube.sigs.k8s.io/docs/drivers/hyperv/), [KVM](https://minikube.sigs.k8s.io/docs/drivers/kvm2/),%C2%A0%5BParallels%5D(https://minikube.sigs.k8s.io/docs/drivers/parallels/), [Podman](https://minikube.sigs.k8s.io/docs/drivers/podman/), [VirtualBox](https://minikube.sigs.k8s.io/docs/drivers/virtualbox/), or [VMWare](https://minikube.sigs.k8s.io/docs/drivers/vmware/)

If your local machine meets all of the above requirements 👆 then you may move on to the next section 👇

### **Installation guide**

1. For MacOS users: [Homebrew](https://brew.sh/) is the recommended method  
   **brew install minikube**

You will need to install Homebrew first. If you haven't, simply [go on this page](https://docs.brew.sh/Installation) and copy/paste the code snippet in your terminal.   
  
It should look something like this: **/bin/bash -c "$(curl -fsSL** [**https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh**](https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)**)**".

You can verify that it's installed by typing:   
  
brew --version

1. For Windows users: we recommend using [Chocolatey](https://chocolatey.org/)  
   **choco install minikube**

👋 *You will need to install chocolatey first. If you haven't, simply* [*go on this page*](https://chocolatey.org/install) *and copy/paste the code snippet in Powershell. The code should look something like this: Set-ExecutionPolicy Bypass -Scope Process -Force; [System.Net.ServicePointManager]::SecurityProtocol = [System.Net.ServicePointManager]::SecurityProtocol -bor 3072; iex ((New-Object System.Net.WebClient).DownloadString('https://community.chocolatey.org/install.ps1')). Once it's done, verify that it's correctly installed by typing: choco*

1. **For Linux users**:[**binary download**](https://minikube.sigs.k8s.io/docs/start/#binary-download)

**curl -LO** [**https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64**](https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64) **sudo install minikube-linux-amd64 /usr/local/bin/minikube**  
  
iIf none of the methods work for you, check out all the different possibilities on the [official documentation](https://minikube.sigs.k8s.io/docs/start/)  
  
To verify that minikube is installed, simply type: minikube on your terminal and you should see a help menu!

## **Use minikube**

Here are some basic commands for launching and stopping minikube:

* Start minikube: **minikube start and a cluster will be created.**
* Once you are done: **minikube stop**
* Delete cluster: **minikube delet**e

Also there are interesting flags that you can add to minikube start:

* Choose driver - **minikube start --driver=hyperkit**
  + This command specifies the virtual machine manager that minikube is supposed to use to build a cluster
  + It is useful to know how to set it up manually as some applications won't work with certain Virtual machine managers
  + Default is **--driver=docker**
  + Also, you will need to have the virtual manager installed first, to be able to use it as driver. We advise you to have two virtual machine managers like docker + hyperkit for Mac. Here is the full list:
    - [Docker](https://minikube.sigs.k8s.io/docs/drivers/docker/)
    - [Hyperkit](https://minikube.sigs.k8s.io/docs/drivers/hyperkit/)
    - [Hyper-V](https://minikube.sigs.k8s.io/docs/drivers/hyperv/)
    - [KVM](https://minikube.sigs.k8s.io/docs/drivers/kvm2/),%C2%A0%5BParallels%5D(https://minikube.sigs.k8s.io/docs/drivers/parallels/)
    - [Podman](https://minikube.sigs.k8s.io/docs/drivers/podman/)
    - [VirtualBox](https://minikube.sigs.k8s.io/docs/drivers/virtualbox/)
    - [VMWare](https://minikube.sigs.k8s.io/docs/drivers/vmware/)
* Choose CPU & memory - **minikube start --cpus 2 --memory 8192**
  + In the above example minikube will allocate 2 CPUs and 8Go of RAM
  + Make sure you have the according ressources on your local machine 😉

## **Install kubectl**

kubectl is the command line tool to interact with your Kubernetes cluster. Here are two simple ways to install it on your computer:

1. For MACOS users: **brew install kubect**l
   * 👋 *You will need to install Homebrew first. If you haven't, simply* [*go on this page*](https://docs.brew.sh/Installation) *and copy/paste the code snippet in your terminal. It should look something like this: /bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)". You can verify that it's installed by typing: brew --version*
2. For Windows users: **choco install kubernetes-cli**
   * *👋 You will need to install chocolatey first. If you haven't, simply* [*go on this page*](https://chocolatey.org/install) *and copy/paste the code snippet in Powershell. The code should look something like this: Set-ExecutionPolicy Bypass -Scope Process -Force; [System.Net.ServicePointManager]::SecurityProtocol = [System.Net.ServicePointManager]::SecurityProtocol -bor 3072; iex ((New-Object System.Net.WebClient).DownloadString('https://community.chocolatey.org/install.*ps1')). Once it's done, verify that it's correctly installed by typing: choco
3. For Linux users: brew install kubectl works if you have Homebrew installed. Otherwise:
   * Update the apt package index and install packages needed to use the Kubernetes apt repository:

sudo apt-get update

sudo apt-get install -y apt-transport-https ca-certificates curl

* + Download the Google Cloud public signing key:

**sudo curl -fsSLo /usr/share/keyrings/kubernetes-archive-keyring.gpg** https://packages.cloud.google.com/apt/doc/apt-key.gpg

Add the Kubernetes apt repository:

**echo "deb [signed-by=/usr/share/keyrings/kubernetes-archive-keyring.gpg] https://apt.kubernetes.io/ kubernetes-xenial main" | sudo tee /etc/apt/sources.list.d/kubernetes.list**

* + Update apt package index with the new repository and install kubectl:

**sudo apt-get update**

**sudo apt-get install -y kubectl**

Once you are done with installation, you can verify that everything works correctly by running:

**kubectl version --client**

You should get the following output (possibly with different versions):

Client Version: version.Info{Major:"1", Minor:"22", GitVersion:"v1.22.5", GitCommit:"5c99e2ac2ff9a3c549d9ca665e7bc05a3e18f07e", GitTreeState:"clean", BuildDate:"2021-12-16T08:38:33Z", GoVersion:"go1.16.12", Compiler:"gc", Platform:"darwin/amd64"}