curl -LO <https://get.helm.sh/helm-v3.8.0-linux-amd64.tar.gz>

tar -xvf helm-v3.8.0-linux-amd64.tar.gz

mv linux-amd64/helm /usr/local/bin/

Các repo của Helm được lưu trữ ở đây: <https://artifacthub.io/packages/search?kind=0&sort=relevance&page=1>

Tải repo về

helm repo add bitnami <https://charts.bitnami.com/bitnami>

helm pull bitnami/nginx

<https://medium.com/nontechcompany/useful-helm-commands-in-daily-life-5079d6731d15>

Just getting the definition straight before we start, Chart means the template and Release is what is deployed into the cluster

**Use official charts**

From time to time, you may want to use some of the official Helm chart to deploy the official application, and maybe modify some of the values

Add/Update Repository

helm repo add nvdp <https://nvidia.github.io/k8s-device-plugin>  
helm repo update

Search for chart and all versions available

For searching for a chart from official repository, there might be multiple versions of the chart. you might want to check what do they provide

helm search repo bitnami/mysql --versions

Download official chart to see implementation and values

Once you have your eyes on a given version. you can use this command to download the whole chart into your computer. whether to inspect its content, values or even use it to modify and release

helm pull bitnami/mysql --version 8.5.8 . --untar

View official chart info

If you don’t want to download the content and open them via your favorite IDE, you can just use these commands to view the detail

helm show chart bitnami/redis

helm show values bitnami/redis

Install release from official chart with version

helm upgrade mysql bitnami/mysql --version 8.5.8 -f values-uat01.yaml --install

this will create a release called mysql from official bitnami mysql chart with version 8.5.8

**-f values-uat01.yaml** assume you have modify some of the values and supply it with this file. you can also use --set name=value to supply just individual value as opposed to the whole file if you want

Install or upgrade a chart (run in chart directory)

in case you have downloaded the chart to modify its content, or it’s your original chart

use this command to either install or upgrade

helm upgrade test . -f values-uat01.yaml -n namespace --install

Install or upgrade a chart with dry run (run in chart directory)

Practically, you want to always dry run your release before the actual release to early detect some compilation error of your chart

helm upgrade test . -f values-uat01.yaml -n namespace --install --dry-run

Render the template locallly

If you want to just render the template quickly to see what will be created, the most used command is the template command

helm template test . -f values-uat01.yaml

will output what will actually be rendered from your char

Upgrade Existing Release with new values

This command comes in handy when you just want to change few values of existing release. you can use the --reuse-values to use the whole values of previous release and supply new value

helm upgrade --reuse-values --set image.registry=docker.io/bitnami/redis --set image.tag=6.0.7-debian-10-r0 redis stable/redis --version 3.2.5 --dry-run

Uninstall a Release

helm uninstall test

Rollback to previous release

helm rollback test

List release

this is the most used command that you will use to list all the releases in the context namespace

helm ls  
helm ls -n namespace

List all release

By default, helm ls lists only releases that are deployed or failed

use -a to list all of them

helm ls -a

Show information about current release

//values  
helm get values test  
// all k8s objects  
helm get manifest test  
// all info, chart name, manifest, values and notes  
helm get all test

Hope you guys enjoy it.