```
                                #============================#
                                #    Helm command Cheat Sheet #
                                #============================#

#-----------------------------------------
kUBERNETES:  Environment setting ~/.bashrc
#-----------------------------------------
alias k='kubectl'
alias kdes='kubectl describe'
alias ke='kubectl explain --recursive'
alias kn='kubectl config set-context --current --namespace'
alias kg='kubectl config get-contexts'
alias kuc='kubectl config use-context'
alias d='docker'
export do='--dry-run=client -o yaml'
export ws='-o wide --show-labels'
export now='--grace-period 0 --force'
source <(kubectl completion bash|sed s/kubectl/k/g)

#Switch to a namespace e.g ravi-prac
kn ravi-prac

Note: All the below helm command will run in the namespace ravi-prac.
      To run the helm command in some differnt namespace use helm -n <namespace>

#-----------------------
 Helm: repo and chart
#-----------------------
1. Add a repo
$helm repo add bitnami https://charts.bitnami.com/bitnami
$helm repo add nginx https://helm.nginx.com/stable

2. List all the repo
[root@centos8-Master: ~]# helm repo list
NAME            URL
bitnami         https://charts.bitnami.com/bitnami
nginx-stable    https://helm.nginx.com/stable

3. update the repo to get the latest charts.
$ helm repo update

4. List all the charts in a repo
[root@centos8-Master: ~]# helm search repo nginx-stable
NAME                                            CHART VERSION   APP VERSION
nginx-stable/nginx-appprotect-dos-arbitrator    0.1.0           1.1.0
nginx-stable/nginx-devportal                    1.5.0           1.5.0
nginx-stable/nginx-ingress                      0.17.0          3.1.0
nginx-stable/nginx-service-mesh                 0.7.0           1.7.0
nginx-stable/nms                                1.3.1           NIM 2.9.0|ACM 1.5.0
nginx-stable/nms-acm                            1.5.0           1.5.0
nginx-stable/nms-hybrid                         2.9.0           2.9.0

5. Search a particular chart in a helm repo
[root@centos8-Master: ~]# helm search repo nms [Note: This will search the nms in all the added
  repo]
NAME                    CHART VERSION   APP VERSION             DESCRIPTION
nginx-stable/nms        1.3.1           NIM 2.9.0|ACM 1.5.0     A chart for installing the NGINX
  Management Suite
nginx-stable/nms-acm    1.5.0           1.5.0                   A Helm chart for Kubernetes
nginx-stable/nms-hybrid 2.9.0           2.9.0                   A Helm chart for Kubernetes
[root@centos8-Master: ~]# helm search repo nginx-stable/nms [Note: This will search the nms only
  in nginx-stable repo]
NAME                    CHART VERSION   APP VERSION             DESCRIPTION
nginx-stable/nms        1.3.1           NIM 2.9.0|ACM 1.5.0     A chart for installing the NGINX
  Management Suite
nginx-stable/nms-acm    1.5.0           1.5.0                   A Helm chart for Kubernetes
nginx-stable/nms-hybrid 2.9.0           2.9.0                   A Helm chart for Kubernetes
[root@centos8-Master: ~]#
```

6. To get all the versions of a chart in a repo
[root@centos8-Master: ~]# helm search repo -l nginx-stable/nms

#---------------------------------
Helm: Dry Run & Template Debug
#---------------------------------
$ helm install <chartName> <repoName>/<chartPath> --dry-run --debug
- example -
$ helm install nginx-ingress nginx-stable/nginx-ingress --dry-run -debug
$ helm install nginx-nms nginx-stable/nms --dry-run --debug

To debug the Helm chart templates without sending them to a Kubernetes API server:
you can download the chart and render it locally using the following command:
$ helm pull --untar bitnami/nginx-ingress-controller
$ helm template <chartFolder|chartArchive>
- examples -
$ helm template ./nginx-ingress-controller
$ helm template nginx-ingress-0.15.2.tgz
You can also lint the Helm chart to verify its validity and that it follows the best practices:
$ helm lint <chartFolder|chartArchive>
- examples -
$ helm lint ./nginx-ingress
$ helm lint nginx-ingress-0.15.2.tgz

#---------------------------------
Helm: Chart values update
#---------------------------------
$ helm list -a [Namespace: current]
$ helm list -A [Namespace: all]
- sample output -
```
NAME     NAMESPACE  REVISION  UPDATED       STATUS    CHART           APP VERSION
grafana  default    1         2023-01-12... deployed  grafana-6.12.1  8.0.1
```
To get custom values, used for the Helm chart release, execute:
$ helm get values <releaseName>
- example -
$ helm get values grafana
- sample output -
```
USER-SUPPLIED VALUES:
service:
  port: 8080
```

To get the values from a specified revision of a Helm chart release, use:
$ helm get values <releaseName> --revision <revisionNumber>

To get all the values:
$ helm get values <releaseName> -a
$ helm get values nginx-ingress-controller -a -o yaml > nginx-ingress-controller-values.yaml

To update the Helm chart values, create a new-values.yml file, for example:
```
# new-values.yml
replicas: 2
service:
  port: 8080
```
Upgrade the Helm chart by applying the new-values.yml file:
$ helm upgrade -f <VALUES_FILE> <CHART_NAME> <REPO_NAME>/<PATH_TO_CHART> \
               --version <CHART_VERSION>
- example -
$ helm upgrade -f new-values.yml grafana grafana/grafana --version grafana-6.12.1

Upgrade the Helm chart by changing some values:
helm upgrade mysql-release bitnami/mysql --set auth.rootPassword=$ROOT_PASSWORD

To verify the Helm chart upgrade, check the new revision number:
$ helm list
- sample output -
```
NAME     NAMESPACE  REVISION  UPDATED       STATUS    CHART           APP VERSION
grafana  default    2         2023-01-12... deployed  grafana-6.12.1  8.0.1
```
To check the new custom values, execute:

```
$ helm get values <releaseName>
- example -
$ helm get values grafana
- sample output -
USER-SUPPLIED VALUES:
replicas: 2
service:
  port: 8080

Uninstall a helm chart:
$helm uninstall mywebapp

Install a chart with new values.:
$helm pull --untar bitnami/apache
$cd apache
Inspect the file values.yaml and make changes:
so that 2 replicas of the webserver are running and the http is exposed on nodeport 30080:

vim values.yaml
service:
  nodePorts:
    http: 30080
    https: ""

$helm install mywebapp -f values.yaml bitnami/apache
#-----------------------------------
Helm: Release History and rollback
#-----------------------------------
[root@centos8-Master: ~]# helm list -a
NAME                            NAMESPACE       REVISION
  UPDATED                                   STATUS          CHART                                APP
  VERSION
apache-local                    ravi-prac       1               2023-04-02 20:30:13.564669167
  +0530 IST deployed        apache-9.2.11                   2.4.55
nginx-ingress-controller        ravi-prac       3               2023-04-03 08:24:56.256420286
  +0530 IST deployed        nginx-ingress-controller-9.5.1  1.7.0


$ helm get values nginx-ingress-controller -a -o yaml > nginx-ingress-controller-values.yaml

Make some changes in the nginx-ingress-controller-values.yaml e.g maxreplicas : 15


# helm upgrade -f nginx-ingress-controller-values.yaml nginx-ingress-controller
  bitnami/nginx-ingress-controller

# helm history nginx-ingress-controller
REVISION        UPDATED                         STATUS          CHART
  APP VERSION     DESCRIPTION
1               Mon Apr  3 08:10:13 2023        superseded      nginx-ingress-controller-9.5.1
  1.7.0           Install complete
2               Mon Apr  3 08:23:30 2023        deployed        nginx-ingress-controller-9.5.1
  1.7.0           Upgrade complete

# helm rollback nginx-ingress-controller 1
Rollback was a success! Happy Helming!

# helm history nginx-ingress-controller
REVISION        UPDATED                         STATUS          CHART
  APP VERSION     DESCRIPTION
1               Mon Apr  3 08:10:13 2023        superseded      nginx-ingress-controller-9.5.1
  1.7.0           Install complete
2               Mon Apr  3 08:23:30 2023        superseded      nginx-ingress-controller-9.5.1
  1.7.0           Upgrade complete
3               Mon Apr  3 08:24:56 2023        deployed        nginx-ingress-controller-9.5.1
  1.7.0           Rollback to 1

#-----------------------------------------------------
Helm: Getter and setter for the helm chart template
```

```
#-------------------------------------------------
To render/see the Helm chart template, execute:
$ helm template <CHART_NAME> <REPO_NAME>/<PATH_TO_CHART>
- example -
$ helm template nginx-ingress nginx/nginx-ingress

To render a specific version of the Helm chart:
$ helm template <CHART_NAME> <REPO_NAME>/<PATH_TO_CHART> --version=<CHART_VERSION>
- example -
$ helm template nginx-ingress nginx/nginx-ingress --version=0.15.1

While rendering the Helm chart template you can override the default chart values, by using a
  --set flag:
$ helm template <CHART_NAME> <REPO_NAME>/<PATH_TO_CHART> --set <VAR>=<VAL>
- example -
$ helm template nginx-ingress nginx/nginx-ingress --set replicaCount=1 \
                                                  --set service.create=false

You can also pass in a new value file (or multiplie) using a --values flag:
$ helm template <CHART_NAME> <REPO_NAME>/<PATH_TO_CHART> --values=<FILE_NAME>
- example -
$ helm template nginx-ingress nginx/nginx-ingress --values ./baseSettings.yaml \
                                                  --values ./customSettings.yaml

#-----------------------------------------
Helm: Download the helm chart tarball
#-----------------------------------------
To download a Helm chart tarball, execute:
$ helm pull <REPO_NAME>/<PATH_TO_CHART>
- example -
$ helm pull nginx-stable/nginx-ingress

You can optionally unpack the tarball on the fly by adding the --untar option:
$ helm pull <REPO_NAME>/<PATH_TO_CHART> --untar

To download a specific version of the Helm chart:
$ helm pull <REPO_NAME>/<PATH_TO_CHART> --version <CHART_VERSION>
- example -
$ helm pull nginx-stable/nginx-ingress --version 0.10.5
```