CREATE CUSTOM CHART



1 helm create mychart



```
mychart
   Chart.yaml
    charts
    templates
        NOTES.txt
        _helpers.tpl
        deployment.yaml
        hpa.yaml
        ingress.yaml
       service.yaml
        serviceaccount.yaml
        tests
        test-connection.yaml
    values.yaml
```

CREATE CUSTOM CHART

- Chart.yaml: Contains metadata about the chart, such as its name, version, description, and keywords.
- values.yaml: A configuration file that holds default values for your chart's parameters. These values can be overwritten during installation.
- **charts**: A folder where you can put other Helm charts that your chart depends on. These are called subcharts.
- **templates**: A folder containing Kubernetes manifest templates that Helm uses to generate the final manifests for deployment.
 - helpers.tpl: This file contains template helpers that are used across other template files.
 - **tests**: This folder contains test files that can be executed to validate the chart's functionality.
- .helmignore: This file lists patterns for files and directories that should be ignored when packaging the chart. It works similarly to a .gitignore file.

INSTALL CUSTOM CHART



1 helm install myapp mychart

```
NAME: myapp
LAST DEPLOYED: Tue Apr 25 22:01:07 2023

NAMESPACE: default

STATUS: deployed

REVISION: 1

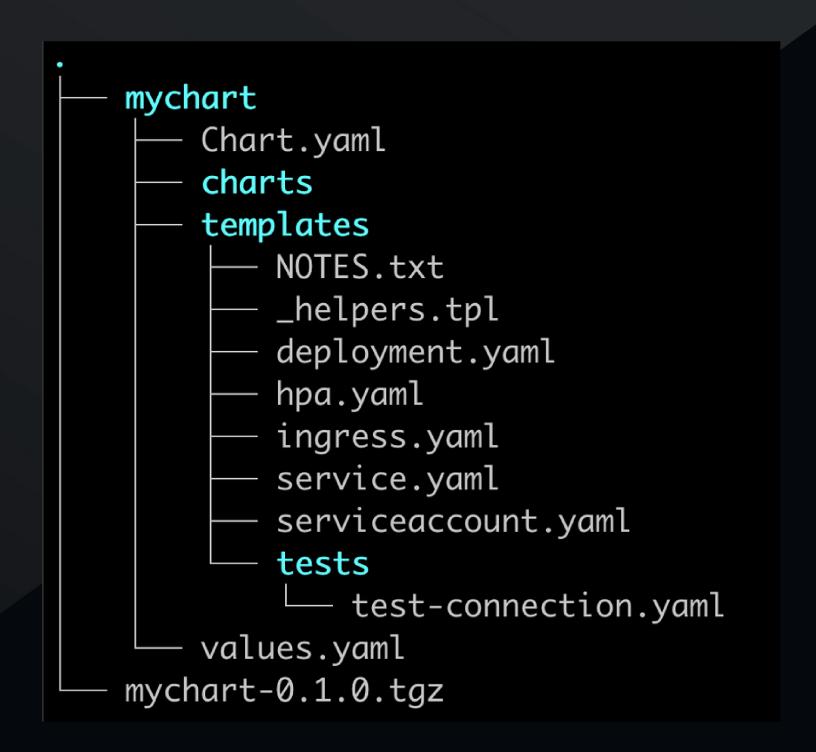
NOTES:

1. Get the application URL by running these commands:
    export POD_NAME=$(kubectl get pods --namespace default -l "app.kubernetes.io/name=mychart,app.kubernet
es.io/instance=myapp" -o jsonpath="{.items[0].metadata.name}")
    export CONTAINER_PORT=$(kubectl get pod --namespace default $POD_NAME -o jsonpath="{.spec.containers[0].ports[0].containerPort}")
    echo "Visit http://127.0.0.1:8080 to use your application"
    kubectl --namespace default port-forward $POD_NAME 8080:$CONTAINER_PORT
```

PACKAGE THE CHART



1 helm package mychart



LINTING THE CHART

```
helm lint
    ==> Linting .
    [INF0] Chart.yaml: icon is recommended
4
5 1 chart(s) linted, 0 chart(s) failed
```