

# 4 Kubernetes Manifest 작성을 위한 Helm 활용

## 05 Helm Values 및 Template 소개

# Helm Chart 생성

## Helm Chart 생성 명령어

\$ **helm create** [Helm Chart명]

## Helm Chart create후 디렉토리 구조

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

## Helm Chart의 파일 구조

파일명 (디렉토리명/)	설명
Chart.yaml	Helm Chart에 대한 이름, 버전, 설명이 설정된 파일
charts/	Helm Chart에서 참조해서 사용할 특정 Helm Charts 패키지를 저장하는 디렉토리
values.yaml	Helm Chart의 변수값을 설정된 파일
templates/	Template Manifest 파일 저장 디렉토리
_helpers.tpl	Template Manifest에서 공용으로 사용하는 변수값 정의 및 구현하는 파일
NOTES.txt	Helm install, upgrade, status시 출력되는 도움말 혹은 정보를 조합해 출력하는 파일
*.yaml	Helm Charts로 릴리즈 배포할 Kubernetes Object의 Template Manifest 파일

## Chart.yaml의 적용 항목

항목	설명
apiVersion	<ul style="list-style-type: none"> <li>Helm 자체의 API Version으로 Helm3부터는 v2 지원 (Required)</li> </ul>
name	<ul style="list-style-type: none"> <li>Helm Chart명 (Required)</li> </ul>
description	<ul style="list-style-type: none"> <li>Helm Chart의 설명 (Optional)</li> </ul>
type	<ul style="list-style-type: none"> <li>Helm Chart의 유형. 유형은 application 및 library중 택1 (Required)</li> <li>application은 버전이 지정된 아카이브로 패키징 할 수 있는 템플릿 모음, 배포 O</li> <li>library는 chart 개발시 필요한 유틸리티 혹은 기능을 제공, 템플릿이 없으므로 배포 X</li> </ul>
version	<ul style="list-style-type: none"> <li>Helm Chart버전으로 SemVer(Semantic Versioning)규칙을 준수해야함 (Required)</li> <li>버전 형태는 숫자로 된 x.y.z 형식이어야함 (<a href="https://semver.org/lang/ko/">https://semver.org/lang/ko/</a>)</li> </ul>
appVersion	<ul style="list-style-type: none"> <li>Helm Charts에서 제공되는 App 버전이며, SemVer형식을 따르지 않아도 됨(Optional)</li> <li>보통은 배포될 App Container Image의 버전(Tag)를 명시</li> </ul>

# values.yaml 적용 항목

## 05. Helm Values 및 Template 소개

```
# Default values for test.
# This is a YAML-formatted file.
# Declare variables to be passed into your templates.

replicaCount: 1

image:
  repository: nginx
  pullPolicy: IfNotPresent
  # Overrides the image tag whose default is the chart appVersion.
  tag: "1.16.0"

imagePullSecrets: []
nameOverride: ""
fullnameOverride: ""

serviceAccount:
  # Specifies whether a service account should be created
  create: true
  # Annotations to add to the service account
  annotations: {}
  # The name of the service account to use.
  # If not set and create is true, a name is generated using the fullname template
  name: ""

podAnnotations: {}

podSecurityContext: {}
  # fsGroup: 2000

securityContext: {}
  # capabilities:
  #   drop:
  #     - ALL
```

## template/\*.yaml 내 적용 항목

항목	설명
.values	values.yaml 파일에서 설정된 변수값 구현
.Charts	Chart.yaml 파일에서 설정된 변수값 구현
.Release	배포할 때 할당된 정보를 변수로 구현 (ex. Helm Charts 릴리즈명을 nginx로 할 경우 .Release.Name에 nginx로 구현)
include	_helpers.tpl에서 설정된 변수값 구현
if ~ end	if문(statement) 구현 (구문안에 else 적용 가능, if not 형태로 적용 가능)
with ~ end	변수에 대한 범위(scope)를 지정, 해당 범위는 .아래 설정된 변수값을 구현
nindent	들여쓰기 적용 공백수(char 단위)
toYaml	구현된 변수를 yaml형식으로 변경
default	기본적으로 구현되는 변수
quote	변수값이 string type으로 구현



# Helm Chart 코드 문법 스캔

## Helm Chart 코드 문법 스캔 명령어

\$ **helm lint** [Helm Chart 디렉토리 경로]

## Helm Chart 코드 문법 정상 결과

```
==> Linting .  
[INFO] Chart.yaml: icon is recommended  
  
1 chart(s) linted, 0 chart(s) failed
```

## Helm Chart 코드 문법 비정상 결과(예)

```
==> Linting .  
[INFO] Chart.yaml: icon is recommended  
[ERROR] values.yaml: unable to parse YAML: error converting YAML to JSON: yaml: line 46: did not find expected key  
[ERROR] templates/: cannot load values.yaml: error converting YAML to JSON: yaml: line 46: did not find expected key  
[ERROR] : unable to load chart  
        cannot load values.yaml: error converting YAML to JSON: yaml: line 46: did not find expected key  
  
Error: 1 chart(s) linted, 1 chart(s) failed
```

# Helm Charts 렌더링(Dry-Run)

## 05. Helm Values 및 Template 소개

### Helm Chart 렌더링(Dry-Run) 명령어

\$ **helm template** [Helm Chart 디렉토리 경로]

```

-----
# Source: test/templates/serviceaccount.yaml
apiVersion: v1
kind: ServiceAccount
metadata:
  name: RELEASE-NAME-test
  labels:
    helm.sh/chart: test-0.1.0
    app.kubernetes.io/name: test
    app.kubernetes.io/instance: RELEASE-NAME
    app.kubernetes.io/version: "1.16.0"
    app.kubernetes.io/managed-by: Helm
-----
# Source: test/templates/service.yaml
apiVersion: v1
kind: Service
metadata:
  name: RELEASE-NAME-test
  labels:
    helm.sh/chart: test-0.1.0
    app.kubernetes.io/name: test
    app.kubernetes.io/instance: RELEASE-NAME
    app.kubernetes.io/version: "1.16.0"
    app.kubernetes.io/managed-by: Helm
spec:
  type: ClusterIP
  ports:
    - port: 80
      targetPort: http
      protocol: TCP
      name: http
  selector:
    app.kubernetes.io/name: test
    app.kubernetes.io/instance: RELEASE-NAME
-----
# Source: test/templates/deployment.yaml
apiVersion: apps/v1

```



# Helm Charts 패키징

## 05. Helm Values 및 Template 소개

### Helm Chart 패키징 명령어

\$ **helm package** [Helm Chart 디렉토리 경로]

```
test@najeonghoui-MacBookPro ~/temp ls
test
test@najeonghoui-MacBookPro ~/temp ls test/
Chart.yaml templates values.yaml
test@najeonghoui-MacBookPro ~/temp helm package test/
Successfully packaged chart and saved it to: /Users/test/temp/test-0.1.0.tgz
test@najeonghoui-MacBookPro ~/temp ls
test test-0.1.0.tgz
test@najeonghoui-MacBookPro ~/temp
```

# Helm Chart Test 수행

## Helm Chart Test 수행 명령어

### \$ helm test [Helm Chart 릴리즈명]

```
test@najeonghoui-MacBookPro ~/temp$ helm test nginx
NAME: nginx
LAST DEPLOYED: Thu Mar 3 01:45:32 2022
NAMESPACE: default
STATUS: deployed
REVISION: 3
TEST SUITE: nginx-test-test-connection
Last Started: Thu Mar 3 04:24:03 2022
Last Completed: Thu Mar 3 04:24:08 2022
Phase: Succeeded
NOTES:
1. Get the application URL by running these commands:
  http://chart-example.local/
test@najeonghoui-MacBookPro ~/temp$ kubectl get po
NAME                                READY   STATUS    RESTARTS   AGE
nginx-test-749944c746-552gw         1/1     Running   0           162m
nginx-test-749944c746-f4k5k         1/1     Running   0           3h2m
nginx-test-test-connection          0/1     Completed 0           12s
test@najeonghoui-MacBookPro ~/temp$ kubectl logs nginx-test-test-connection
Connecting to nginx-test:80 (10.102.71.24:80)
saving to 'index.html'
index.html      100% |*****| 612 0:00:00 ETA
'index.html' saved
```

## Helm Chart 내 Chart.yaml에 설정된 정보 확인

Helm Chart 내 Chart.yaml에 설정된 정보 확인 명령어

\$ **helm show chart** [Helm Chart 디렉토리 경로]

```
test@najeonghoui-MacBookPro ~/temp ➔ helm show chart test/  
apiVersion: v2  
appVersion: 1.16.0  
description: A Helm chart for Kubernetes  
name: test  
type: application  
version: 0.1.0  
  
test@najeonghoui-MacBookPro ~/temp ➔ ls test/  
Chart.yaml templates values.yaml
```



# Helm Chart 내 Chart.yaml 및 values.yaml에 설정된 정보 확인

## 05. Helm Values 및 Template 소개

### Helm Chart 내 Chart.yaml 및 values.yaml에 설정된 정보 확인 명령어

\$ **helm show all [Helm Chart 디렉토리]**

```
test@najeonghoui-MacBookPro ~/temp$ ls test/
Chart.yaml  templates  values.yaml
test@najeonghoui-MacBookPro ~/temp$ helm show all test/
apiVersion: v2
appVersion: 1.16.0
description: A Helm chart for Kubernetes
name: test
type: application
version: 0.1.0

---
# Default values for test.
# This is a YAML-formatted file.
# Declare variables to be passed into your templates.

replicaCount: 1

image:
  repository: nginx
  pullPolicy: IfNotPresent
  # Overrides the image tag whose default is the chart appVersion.
  tag: "1.16.0"

imagePullSecrets: []
nameOverride: ""
fullnameOverride: ""

serviceAccount:
  # Specifies whether a service account should be created
  create: true
  # Annotations to add to the service account
  annotations: {}
  # The name of the service account to use.
  # If not set and create is true, a name is generated using the fullname template
  name: ""

podAnnotations: {}
```