# Annotations

1. Create a pod from a yaml file.
2. Check its annotation kubectl.kubernetes.io/last-applied-configuration
3. Add own annotation and check that its has been applied.
4. Change your annotation and check that new value has been applied.

## Solution

1. Create a pod from a yaml file.

It can be done using either kubectl create --save-config or kubectl apply:

$ kubectl create --save-config -f labels/labels-demo.yaml  
$ kubectl apply -f labels/labels-demo.yaml  
pod/demo1 created

1. Check its annotation kubectl.kubernetes.io/last-applied-configuration

$ kubectl get pod demo1 -o yaml | yq -C e .metadata.annotations -  
kubectl.kubernetes.io/last-applied-configuration: |  
 {"apiVersion":"v1","kind":"Pod","metadata":{"annotations":{},"labels":{"key1":"value1","key2":"value2"},"name":"demo1","namespace":"msuslov"},"spec":{"containers":[{"command":["sleep","1d"],"image":"busybox","imagePullPolicy":"IfNotPresent","name":"bc"}],"restartPolicy":"Never"}}

It is inline version. It is possible to unford it:

$ kubectl get pod demo1 -o yaml | yq e '.metadata.annotations."kubectl.kubernetes.io/last-applied-configuration"' - | jq .  
{  
 "apiVersion": "v1",  
 "kind": "Pod",  
 "metadata": {  
 "annotations": {},  
 "labels": {  
 "key1": "value1",  
 "key2": "value2"  
 },  
 "name": "demo1",  
 "namespace": "msuslov"  
 },  
 "spec": {  
 "containers": [  
 {  
 "command": [  
 "sleep",  
 "1d"  
 ],  
 "image": "busybox",  
 "imagePullPolicy": "IfNotPresent",  
 "name": "bc"  
 }  
 ],  
 "restartPolicy": "Never"  
 }  
}

1. Add own annotation and check that its has been applied.

$ $ kubectl annotate pod demo1 create=MSuslov git=a1b2c3d4  
pod/demo1 annotated  
  
$ kubectl get pod demo1 -o yaml | yq -C e .metadata.annotations -  
create: MSuslov  
git: a1b2c3d4  
kubectl.kubernetes.io/last-applied-configuration: |  
 {"apiVersion":"v1","kind":"Pod","metadata":{"annotations":{},"labels":{"key1":"value1","key2":"value2"},"name":"demo1","namespace":"msuslov"},"spec":{"containers":[{"command":["sleep","1d"],"image":"busybox","imagePullPolicy":"IfNotPresent","name":"bc"}],"restartPolicy":"Never"}}

1. Change your annotation and check that new value has been applied.

Rename isn’t supported:

$ $ kubectl annotate pod demo1 create-  
pod/demo1 annotated  
  
$ kubectl annotate pod demo1 creator=MSuslov  
pod/demo1 annotated

Update git commit hash value:

$ kubectl annotate pod demo1 git=A1B2C3D4 --overwrite  
pod/demo1 annotated  
  
$ kubectl get pod demo1 -o yaml | yq -C e .metadata.annotations -  
creator: MSuslov  
git: A1B2C3D4  
kubectl.kubernetes.io/last-applied-configuration: |  
 {"apiVersion":"v1","kind":"Pod","metadata":{"annotations":{},"labels":{"key1":"value1","key2":"value2"},"name":"demo1","namespace":"msuslov"},"spec":{"containers":[{"command":["sleep","1d"],"image":"busybox","imagePullPolicy":"IfNotPresent","name":"bc"}],"restartPolicy":"Never"}}

Clean up:

$ kubectl delete -f labels/labels-demo.yaml  
pod "demo1" deleted