apiVersion: apps/v1

kind: Deployment

metadata:

labels:

run: nginx01

name: nginx01

spec:

replicas: 1

selector:

matchLabels:

run: nginx01

template:

metadata:

labels:

run: nginx01

spec:

containers:

- image: sreeharshav/rollingupdate:v5

name: nginx01

volumeMounts:

- mountPath: /scripts

name: scripts

initContainers:

- image: busybox:latest

name: init-busybox-1

command: ["/bin/sh"]

args: ["-c", "sleep 5"]

- image: busybox:latest

name: init-busybox-2

volumeMounts:

- mountPath: /scripts

name: scripts

command:

- wget

- "-O"

- "/scripts/script.sh"

- https://sree-kubernetes-data.s3.amazonaws.com/script.sh

#args:

# - "wget https://sree-kubernetes-data.s3.amazonaws.com/script.sh"

restartPolicy: Always

volumes:

- name: scripts

emptyDir: {}

**MySQL DB Seeding with Init Containers:**

<https://www.magalix.com/blog/kubernetes-patterns-the-init-container-pattern#:~:text=In%20Kubernetes%2C%20an%20init%20container,database%20schemas%20and%20so%20on>.

apiVersion: v1

kind: Pod

metadata:

name: mydb

labels:

app: db

spec:

initContainers:

- name: fetch

image: mwendler/wget

command: ["wget","--no-check-certificate","https://sample-videos.com/sql/Sample-SQL-File-1000rows.sql","-O","/docker-entrypoint-initdb.d/dump.sql"]

volumeMounts:

- mountPath: /docker-entrypoint-initdb.d

name: dump

containers:

- name: mysql

image: mysql

env:

- name: MYSQL\_ROOT\_PASSWORD

value: "example"

volumeMounts:

- mountPath: /docker-entrypoint-initdb.d

name: dump

volumes:

- emptyDir: {}

name: dump