

<https://kubernetes.io/docs/concepts/workloads/pods/init-containers/>

InitContainers

In a multi-container pod, each container is expected to run a process that stays alive as long as the POD's lifecycle. For example in the multi-container pod that we talked about earlier that has a web application and logging agent, both the containers are expected to stay alive at all times. The process running in the log agent container is expected to stay alive as long as the web application is running. If any of them fails, the POD restarts.

But at times you may want to run a process that runs to completion in a container. For example a process that pulls a code or binary from a repository that will be used by the main web application. That is a task that will be run only one time when the pod is first created. Or a process that waits for an external service or database to be up before the actual application starts. That's where **initContainers** comes in.

An **initContainer** is configured in a pod like all other containers, except that it is specified inside a **initContainers** section, like this:

```

1 | apiVersion: v1
2 | kind: Pod
3 | metadata:
4 |   name: myapp-pod
5 |   labels:
6 |     app: myapp
7 | spec:
8 |   containers:
9 |     - name: myapp-container
10 |       image: busybox:1.28
11 |       command: ['sh', '-c', 'echo The app is running! && sleep 3600']
12 |   initContainers:
13 |     - name: init-myservice
14 |       image: busybox
15 |       command: ['sh', '-c', 'git clone <some-repository-that-will-be-used-
    by-application> ; done;']

```

When a POD is first created the initContainer is run, and the process in the initContainer must run to a completion before the real container hosting the application starts.

You can configure multiple such initContainers as well, like how we did for multi-pod containers. In that case each init container is run **one at a time in sequential order**. If any of the initContainers fail to complete, Kubernetes restarts the Pod repeatedly until the Init Container succeeds.

```
1 | apiVersion: v1
2 | kind: Pod
3 | metadata:
4 |   name: myapp-pod
5 |   labels:
6 |     app: myapp
7 | spec:
8 |   containers:
9 |   - name: myapp-container
10 |     image: busybox:1.28
11 |     command: ['sh', '-c', 'echo The app is running! && sleep 3600']
12 |   initContainers:
13 |   - name: init-myservice
14 |     image: busybox:1.28
15 |     command: ['sh', '-c', 'until nslookup myservice; do echo waiting for
myservice; sleep 2; done;']
16 |   - name: init-mydb
17 |     image: busybox:1.28
18 |     command: ['sh', '-c', 'until nslookup mydb; do echo waiting for
mydb; sleep 2; done;']
```

Read more about initContainers here. And try out the upcoming practice test.

From <<https://www.udemy.com/course/certified-kubernetes-administrator-with-practice-tests/learn/lecture/15030046#overview>>