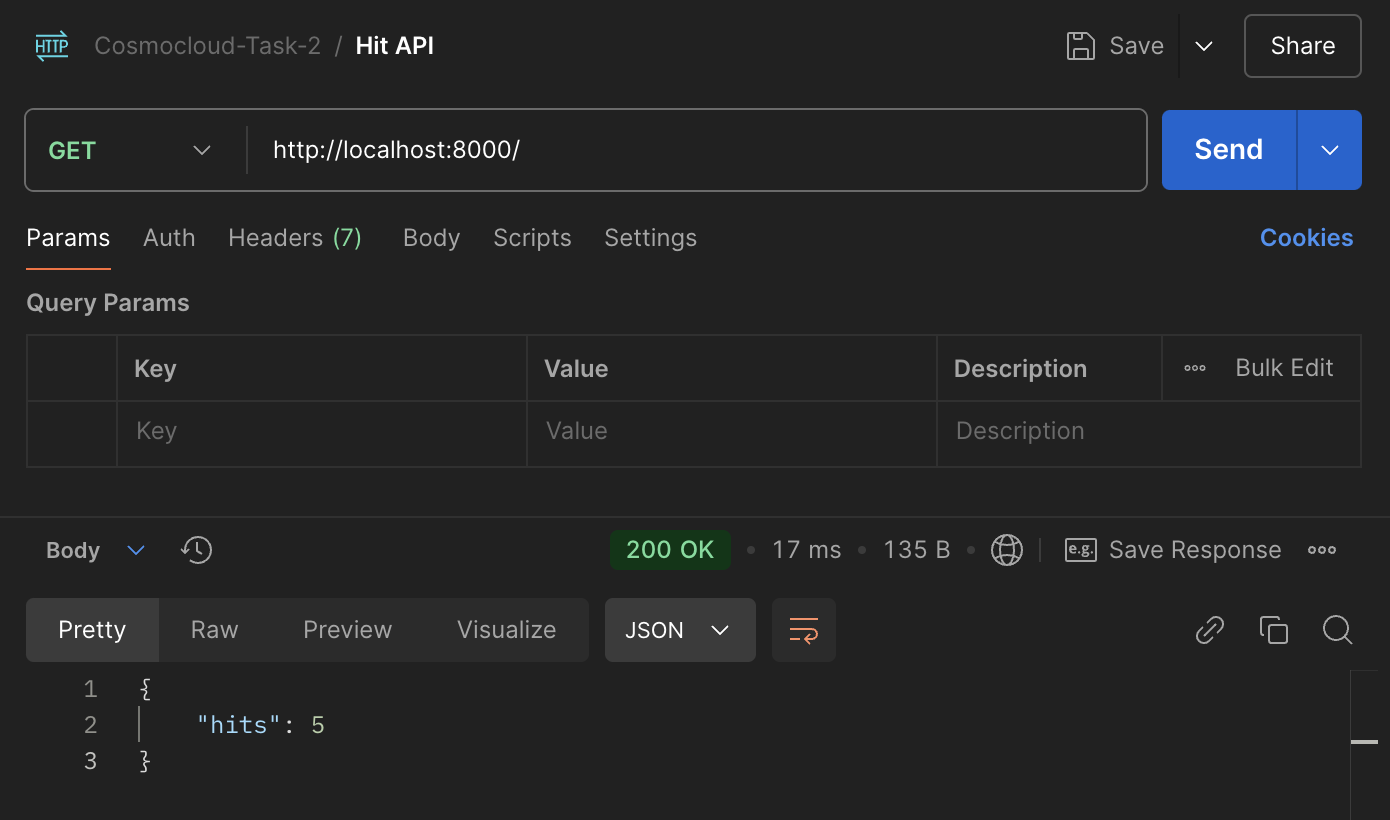
Cosmocloud Task 2

Debug & Fix Application/Helm Chart

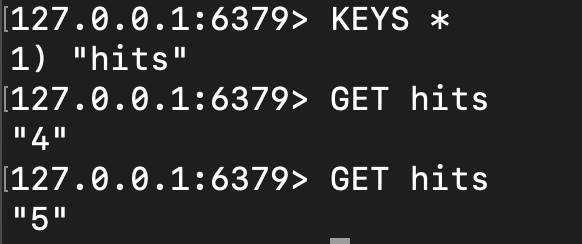
By- Jay Oswal

## Working API Call Screenshots

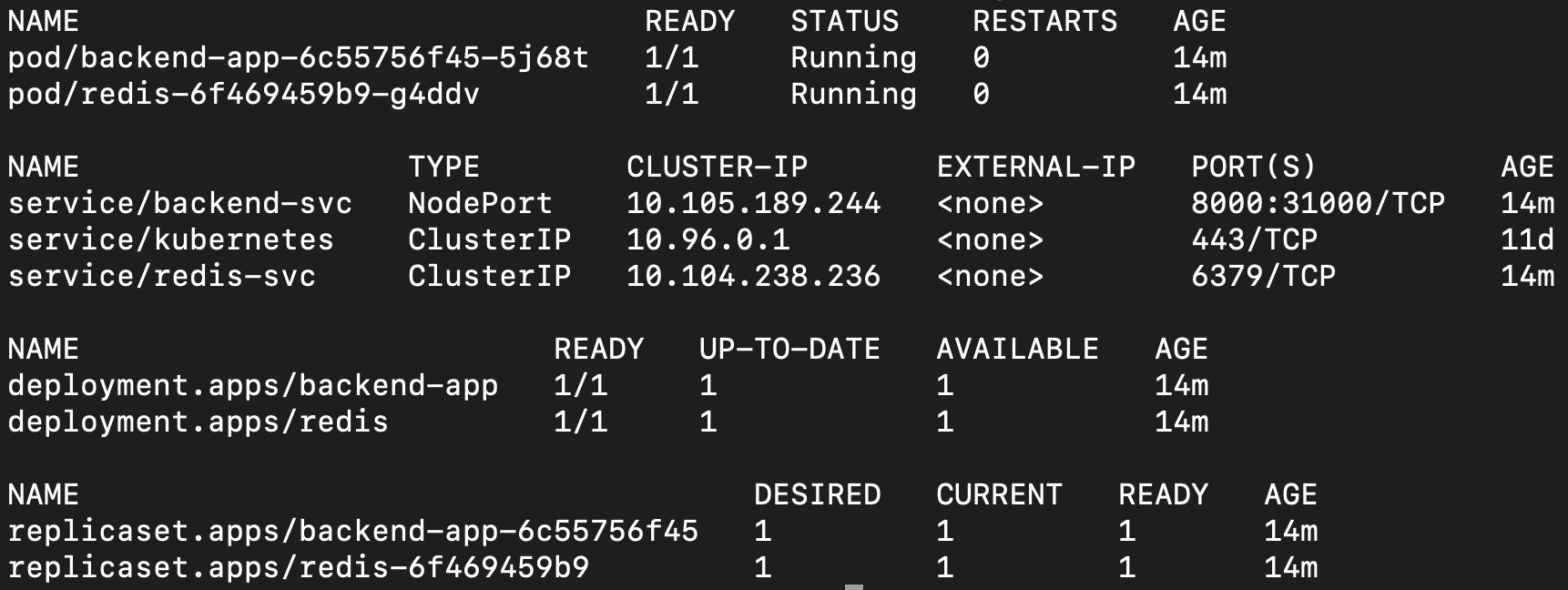
1. Postman



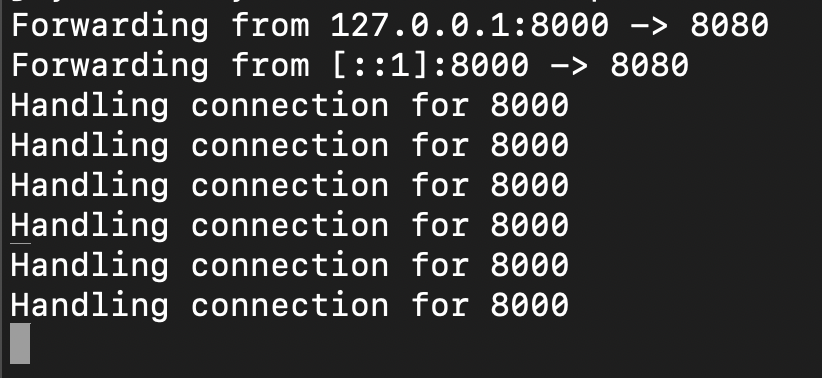
1. Redis through CLI
   1. *‘kubectl exec -it <redis-pod-name> -- redis-cli’*



1. ‘*kubectl get all*’



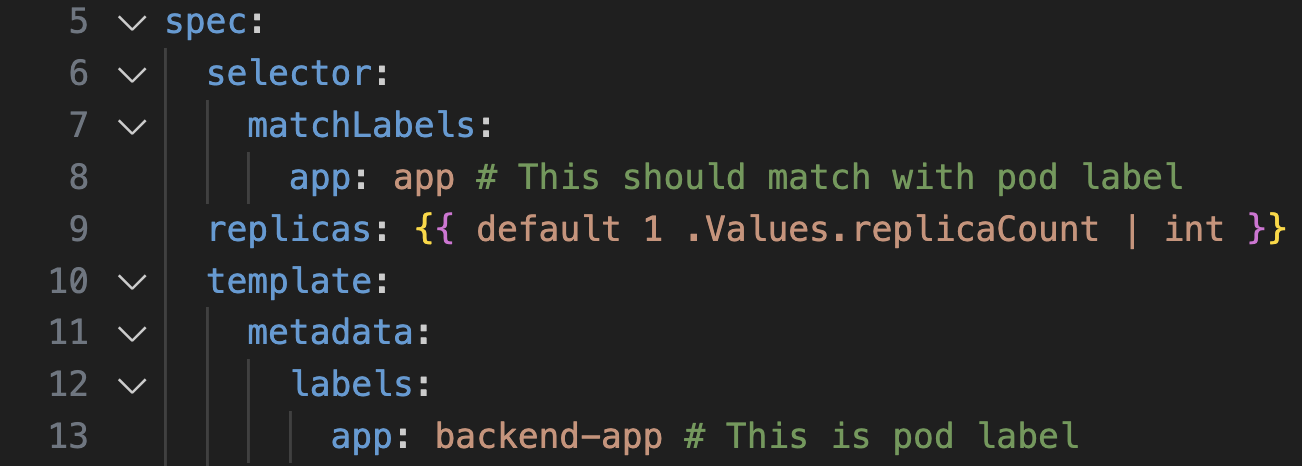
1. Kubernetes port forwarding → ‘*kubectl port-forward svc/backend-svc 8000:8000*’



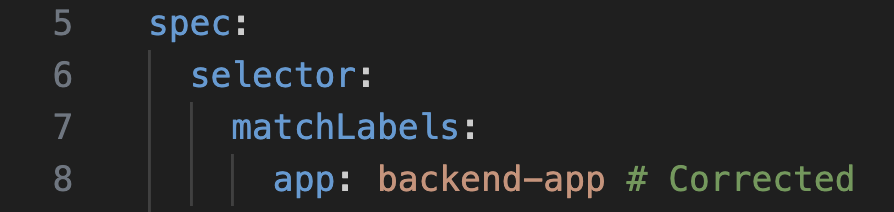
## Total Bugs Found: 10

### Bug 1

* 1. File: deployment.yaml
  2. How did I find this?: By Executing command ‘helm install sample-helm-chart .’
  3. Error: *Deployment.apps "backend-app" is invalid: spec.template.metadata.labels: Invalid value: map[string]string{"app":"backend-app"}: `selector` does not match template `labels`*

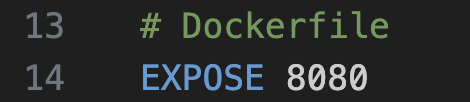


* 1. Explanation: The deployment uses matchLabels to manage the correct set of pods. If it doesn’t matches the deployment will be unable to manage pods and service will not route traffic correctly.
  2. Solution : Use the same pod name in the selector

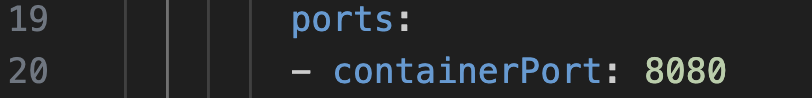


### Bug 2

* 1. File: deployment.yaml
  2. How did I find this?: By Observation
  3. Error: As per Docker file of the backend application port ‘8080’ is supposed to be exposed. So in deployment.yaml file the container port is mentioned incorrectly as 8000

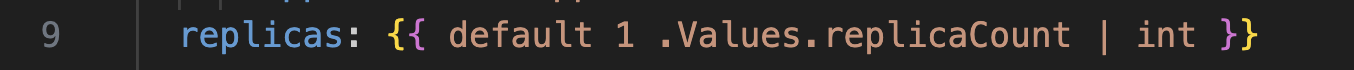


* 1. Explanation: There will be port conflic leading to networking & request routing issues as the container’s port do not match(8000) to what it is exposed(8080).
  2. Solution : Keep port consistent to avoid routing traffic & networking issues. So using Dockerfile as source of truth, all related port is used as 8080



### Bug 3

* 1. File: deployment.yaml
  2. How did I find this?: By Observation
  3. Error: Incorrect syntax for default when values not specified in values.yaml

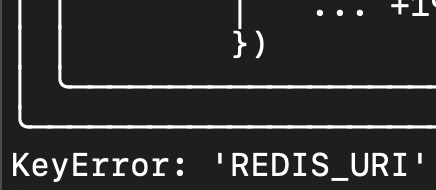


* 1. Explanation: Possible incorrect syntax
  2. Solution: Changed to recommended syntax to handle default

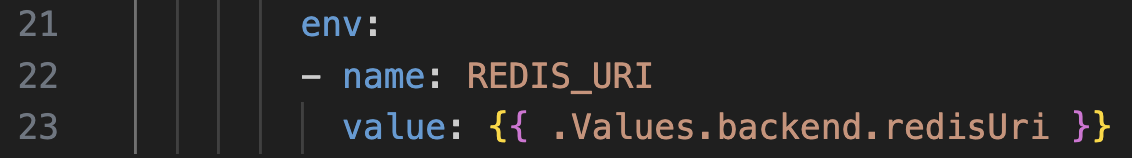


### Bug 4

* 1. File: deployment.yaml
  2. How did I find this?: By checking log of backend pod, when pod failed to start and went into CrashLoopBackOff. By using command ‘*kubectl logs <backend pod name*>’
  3. Error: REDIS\_URI not specified in env



* 1. Explanation: The backend image needs REDIS\_URI as it looks into env, failing to specified will not start the backend pod
  2. Solution: Mention it in the deployment file through the values.yaml file
     1. deployment.yaml

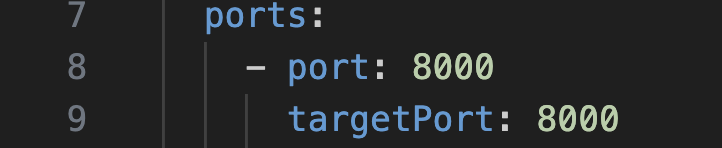


* + 1. values.yaml

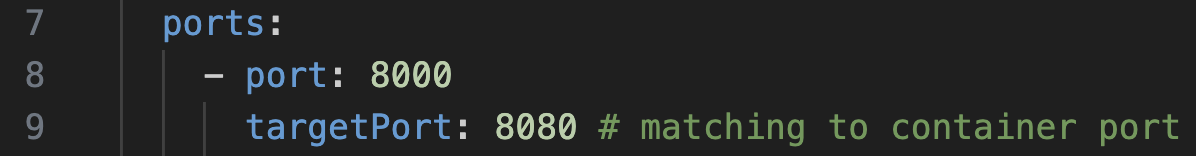


### Bug 5

* 1. File: backend-svc.yaml
  2. How did I find this?: By Observation
  3. Error: Incorrect port for service. Target port not matching to exposed port of backend image (Dockerfile)



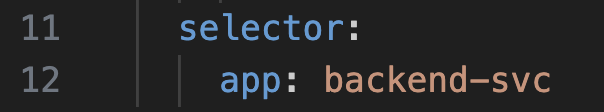
* 1. Explanation: This will lead to incorrect routing of traffic to non-desired port.
  2. Solution: Match the port to that of Image’s port



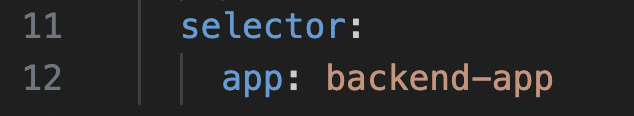
Note:- As on which port should the backend app be accessible from outside has not been mentioned, I am using port 8000 as service port

### Bug 6

* 1. File: backend-svc.yaml
  2. How did I find this?: By Observation
  3. Error: Incorrect selector for backend app. We’ve named the pod as ‘backend-app’ and not ‘backend-svc’

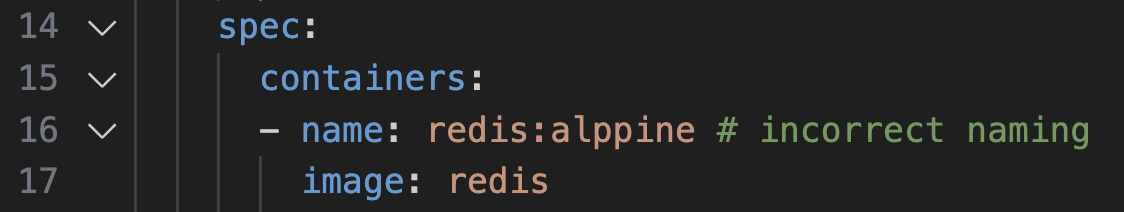


* 1. Explanation: Selector should match with our correct desired backend app name, else will result in incorrect network routing.
  2. Solution: Match the app selector to correct pod name

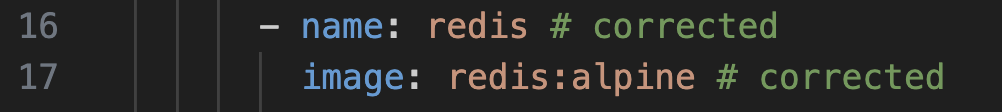


### Bug 7

* 1. File: redis.yaml
  2. How did I find this?: By Executing command ‘*helm install sample-helm-chart .*’
  3. Error: *Deployment.apps "redis" is invalid: spec.template.spec.containers[0].name: Invalid value: "redis:alppine": a lowercase RFC 1123 label must consist of lower case alphanumeric characters or '-', and must start and end with an alphanumeric character (e.g. 'my-name', or '123-abc', regex used for validation is '[a-z0-9]([-a-z0-9]\*[a-z0-9])?')*

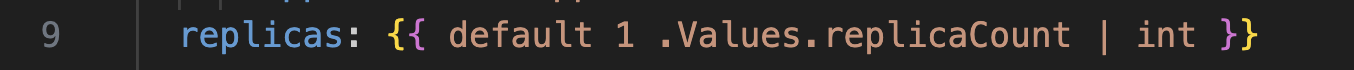
**

* 1. Explanation: Incorrect naming convention for container OR container’s name & image name are swapped.
  2. Solution : Just kept name as redis and image name as redis:alpine



### Bug 8

* 1. File: redis.yaml
  2. How did I find this?: By Observation
  3. Error: Incorrect syntax for default when values not specified in values.yaml

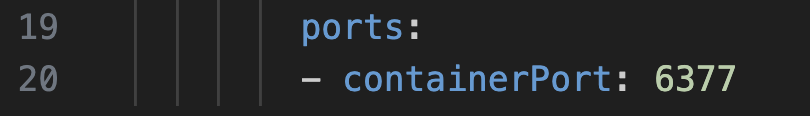


* 1. Explanation: Possible incorrect syntax
  2. Solution: Changed to recommended syntax to handle default



### Bug 9

* 1. File: redis.yaml
  2. How did I find this?: By Observation
  3. Error: Incorrect container port

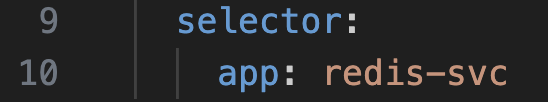


* 1. Explanation: We are using default port of redis which is 6379 in the redis docker image. So 6377 will not point to redis image resulting in networking issues
  2. Solution: Use correct redis port in its deployment file

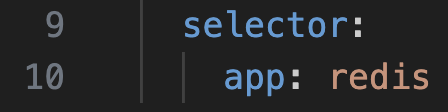


### Bug 10

* 1. File: redis-svc.yaml
  2. How did I find this?: By Observation
  3. Error: Incorrect app selector name

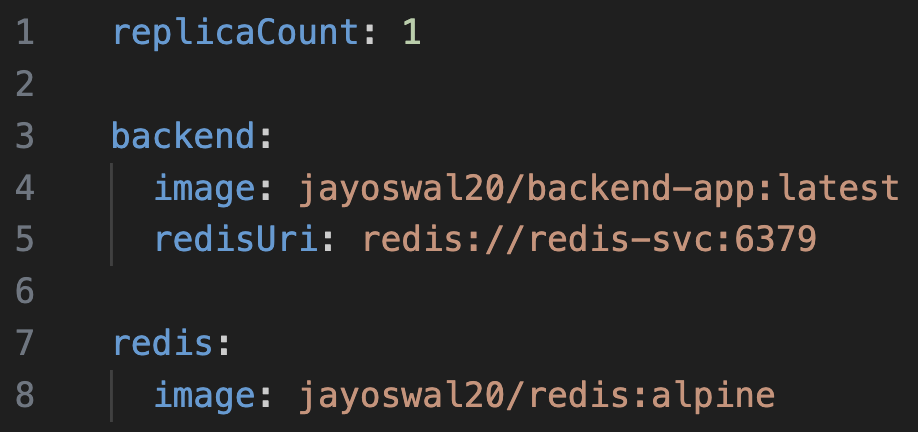


* 1. Explanation: We are using redis as our pod name for redis container hence, using redis-svc will result in incorrect network routing.
  2. Solution: Use correct redis pod name

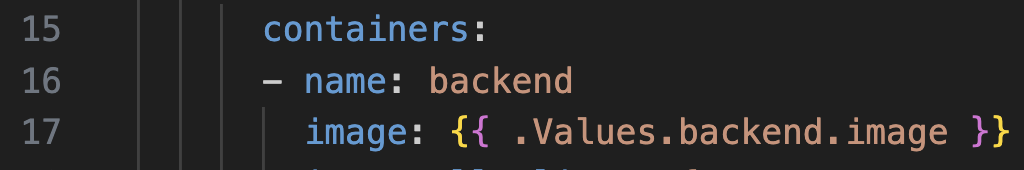


### **Values.yaml File**

I utilized values.yaml file for image name and redis url as it will easy to maintain for CI/CD purpose. I also pushed the images into my docker repository ‘jayoswal20’ as it was already configured on my machine, else I had to point minikube to local docker repository.



Pointed to values.yaml wherever applicable



### Github Repo with bug fixed Helm Charts → <https://github.com/jayoswal/DevOps-Task-2>