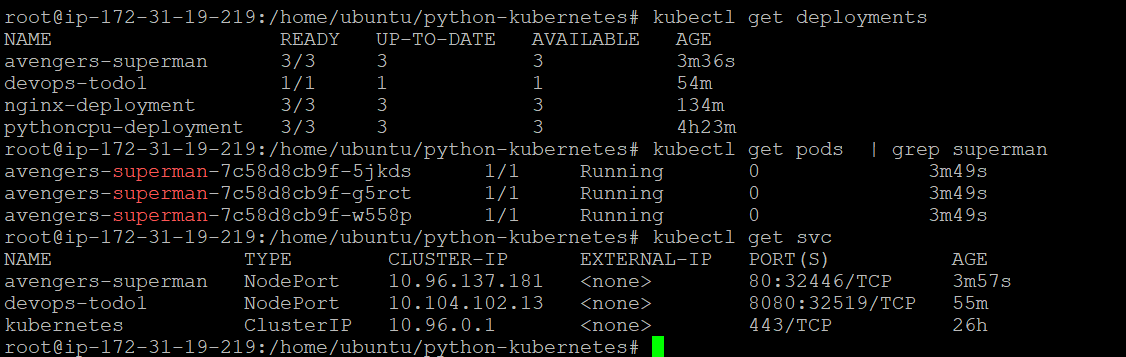
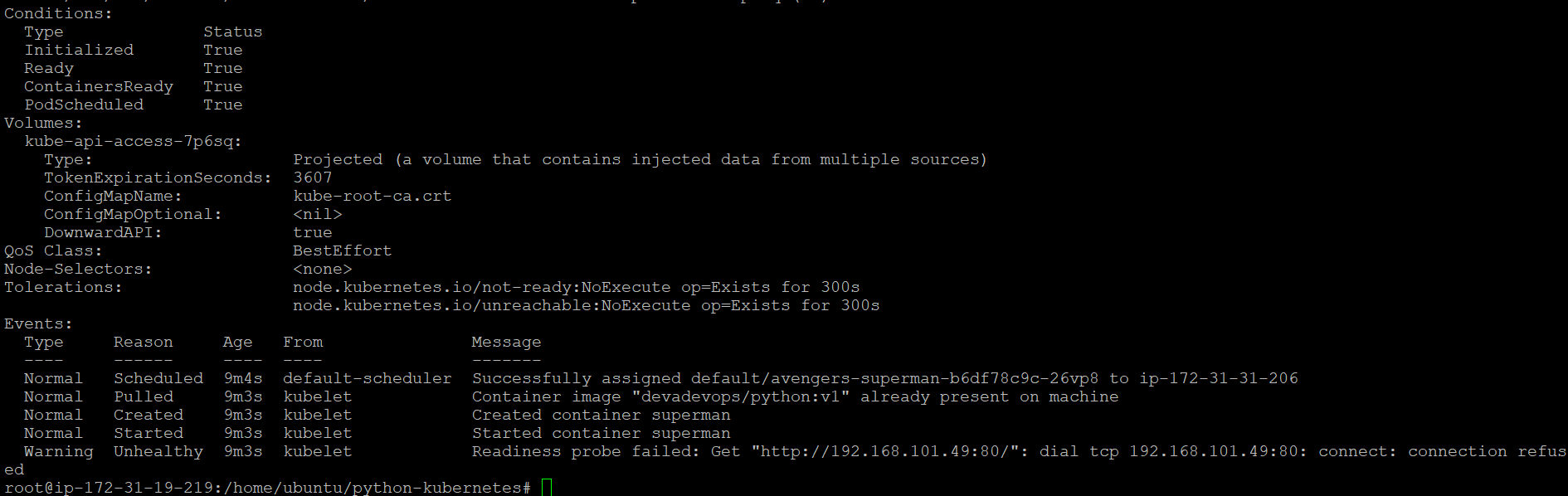
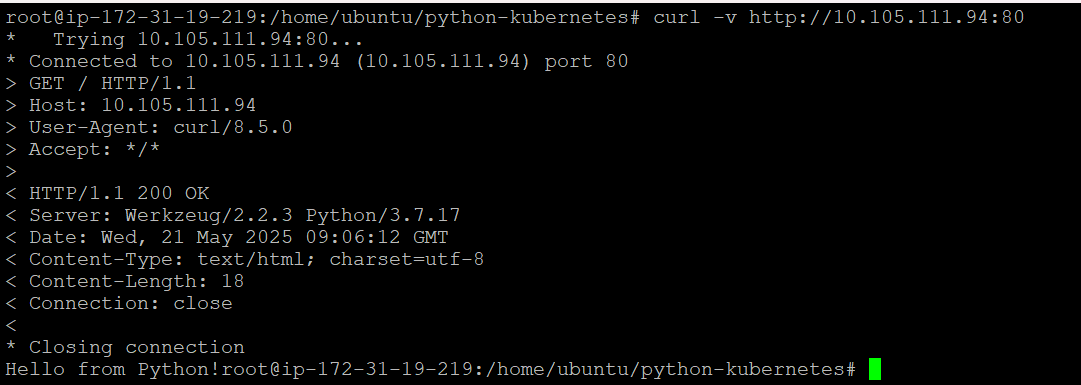
**Values.yml for python deployment using the Helm cart name superman.**

|  |
| --- |
| root@ip-172-31-19-219:/home/ubuntu/python-kubernetes# helm show values superman  # Default values for superman.  # This is a YAML-formatted file.  # Declare variables to be passed into your templates.  # This will set the replicaset count more information can be found here: https://kubernetes.io/docs/concepts/workloads/controllers/replicaset/  replicaCount: 3  # This sets the container image more information can be found here: https://kubernetes.io/docs/concepts/containers/images/  image:  repository: devadevops/python  # This sets the pull policy for images.  pullPolicy: IfNotPresent  # Overrides the image tag whose default is the chart appVersion.  tag: "v1"  # This is for the secrets for pulling an image from a private repository more information can be found here: https://kubernetes.io/docs/tasks/configure-pod-container/pull-image-private-registry/  imagePullSecrets: []  # This is to override the chart name.  nameOverride: ""  fullnameOverride: ""  # This section builds out the service account more information can be found here: https://kubernetes.io/docs/concepts/security/service-accounts/  serviceAccount:  # Specifies whether a service account should be created  create: true  # Automatically mount a ServiceAccount's API credentials?  automount: 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: ""  # This is for setting Kubernetes Annotations to a Pod.  # For more information checkout: https://kubernetes.io/docs/concepts/overview/working-with-objects/annotations/  podAnnotations: {}  # This is for setting Kubernetes Labels to a Pod.  # For more information checkout: https://kubernetes.io/docs/concepts/overview/working-with-objects/labels/  podLabels: {}  podSecurityContext: {}  # fsGroup: 2000  securityContext: {}  # capabilities:  # drop:  # - ALL  # readOnlyRootFilesystem: true  # runAsNonRoot: true  # runAsUser: 1000  # This is for setting up a service more information can be found here: https://kubernetes.io/docs/concepts/services-networking/service/  service:  # This sets the service type more information can be found here: https://kubernetes.io/docs/concepts/services-networking/service/#publishing-services-service-types  type: NodePort  # This sets the ports more information can be found here: https://kubernetes.io/docs/concepts/services-networking/service/#field-spec-ports  port: 80  # This block is for setting up the ingress for more information can be found here: https://kubernetes.io/docs/concepts/services-networking/ingress/  ingress:  enabled: false  className: ""  annotations: {}  # kubernetes.io/ingress.class: nginx  # kubernetes.io/tls-acme: "true"  hosts:  - host: chart-example.local  paths:  - path: /  pathType: ImplementationSpecific  tls: []  # - secretName: chart-example-tls  # hosts:  # - chart-example.local  resources: {}  # We usually recommend not to specify default resources and to leave this as a conscious  # choice for the user. This also increases chances charts run on environments with little  # resources, such as Minikube. If you do want to specify resources, uncomment the following  # lines, adjust them as necessary, and remove the curly braces after 'resources:'.  # limits:  # cpu: 100m  # memory: 128Mi  # requests:  # cpu: 100m  # memory: 128Mi  # This is to setup the liveness and readiness probes more information can be found here: https://kubernetes.io/docs/tasks/configure-pod-container/configure-liveness-readiness-startup-probes/  livenessProbe:  httpGet:  path: /  port: 80  readinessProbe:  httpGet:  path: /  port: 80  # This section is for setting up autoscaling more information can be found here: https://kubernetes.io/docs/concepts/workloads/autoscaling/  autoscaling:  enabled: false  minReplicas: 1  maxReplicas: 100  targetCPUUtilizationPercentage: 80  # targetMemoryUtilizationPercentage: 80  # Additional volumes on the output Deployment definition.  volumes: []  # - name: foo  # secret:  # secretName: mysecret  # optional: false  # Additional volumeMounts on the output Deployment definition.  volumeMounts: []  # - name: foo  # mountPath: "/etc/foo"  # readOnly: true  nodeSelector: {}  tolerations: []  affinity: {}  root@ip-172-31-19-219:/home/ubuntu/python-kubernetes# |



Error :





Assessment 2: -

Deployment Name : antman

Image : devadevops/python:v1

Container Port : 80

Replica : 3

Create Service Type : NodePort

Create HPA : min 5 Max 20 CPU percentage 60

|  |
| --- |
| root@ip-172-31-19-219:/home/ubuntu/python-kubernetes# cat antman.yaml  apiVersion: apps/v1  kind: Deployment  metadata:  name: antman-deployment  spec:  replicas: 3  selector:  matchLabels:  app: antman  template:  metadata:  labels:  app: antman  spec:  containers:  - name: antman  image: devadevops/python:v1  ports:  - containerPort: 80  root@ip-172-31-19-219:/home/ubuntu/python-kubernetes# |

