Lab5

Q1. Deploy pod.yml. Get the details of the pod using commands: *kubectl get pods hello-pod -o wide, kubectl describe pods hello-pod* (1 mark)

Q2. Apply a watch on the pods using command: *kubectl get pods –watch* (1 mark)

Q3. Expose the pod **imperatively** and check if the website is working correctly, then delete the service (don’t delete the pod) (1 mark)

Q4. Apply the service declaratively (NodePort) and check if the website is working correctly, then delete the service (don’t delete the pod) (1 mark)

Q5. Apply the service declaratively (LoadBalancer) and check if the website is working correctly. (Keep the service and the pod) (1 mark)

Q6. Use file rs1.yml to show how to create a replicaset (declaratively) and scale it up and down. You can open another terminal with a watch on the pods to see details about pods (pending, terminating, running). (1 mark)

Q7. Delete one of the pods and see how K8S self-heal the cluster, then delete the replicaset declaratively. (1 mark)

Q8. Modify the web app, build the docker image and push it to your docker hub account, write a yaml file to create a repliaset with 5 pods of the new app, and expose it using a Load balancer service. (1 mark)

Q9. Refer to this post <https://medium.com/google-cloud/kubernetes-nodeport-vs-loadbalancer-vs-ingress-when-should-i-use-what-922f010849e0> (and chap7 in the book) to demonstrate the differences between ClusterIP service, NodePort service, Load Balancer Service, and Ingress Service. Explain the sequence of steps that happen when a request arrives to each one of the mentioned services (2 marks)