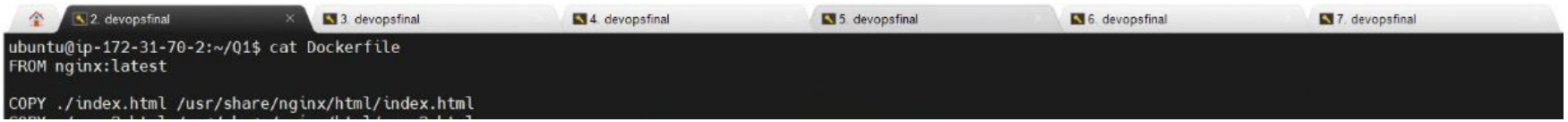
Final Exam (DevOps for Cloud Computing)- Part 2

NOTE: Please provide the files and scripts for each question in separate zip archives with names such as Q1 part2.zip, Q2 part2.zip, etc. You will present your work in class by sharing your screen, however I require that all of your files be uploaded to Blackboard prior to your presentation time.

You have 15 minutes to present all the answers. To save time, I recommend keeping the commands you wish to execute in separate tabs of your terminal, such as one tab for each question that requires executing commands.



1- Create three web pages (this is our application in this exam), each page has two links to other pages. Create a local repo and include the pages you have created in that repo. Through **Jenkins** create a **CI** pipeline in a way that when you push the pages to the remote repo (your Github repo), Jenkins creates an image. Use your choice of a web server (Nginx, Apache, etc) as its base image. **(2 marks)**
2- Through Jenkins push the image you have created in the previous step to your dockerhub account **(1 mark).**
3- Create a Helm chart for a deployment and a service and use the image you pushed in step 2. Deploy the application two times in the Kubernetes through Helm (i.e. we want to create two releases of the same application). For that, you need to parameterize parts of the chart (i.e. do not create two charts) and use **one** helm chart to deploy two releases of the application in Kubernetes **(2 marks)**

4- Design scenarios should correspond with manifest files that replicate the issues stated below.On exam day, you must explain why the problem exists and what the solution is. Create a table like the one below (**each question has 2 mark, total 20 marks**).

#	Issue/scenario	Reason	Files you have uploaded	Note
1	Pod status is in Pending mode and does not change to Running mode			
2	Pod status is in Waiting in container creation mode			
3	Pod status is in ImagePullBackOff mode			
4	Pod is in CrashLoopBackOff mode			
5	Pod is in Error mode			
6	The service is missing endpoints			
7	Run a temporary pod in cluster and try to access a non existing ClusterIP service from that pod			
8	Run a temporary pod in cluster to look up			

	inside a cluster for a ClusterIP service through ping, or nsloopup from inside that pod (assuming the service exists)			
9	Create a pod, then a service and check its `cat /etc/resolv.conf.` Do this experiment in reverse order: create a service and create a pod and then `cat /etc/resolv.conf`			
10	Select one of the scenarios between 1-6 and create a Kubernetes Cron job[1] that runs every minute and if it finds that issue it writes that issue to a file in a folder of host node			

NOTE: If I see similarity in responses that appears to violate academic integrity, I shall notify the office. Please refrain from consulting with one another, since it will result in identical replies, which is problematic.

[1] https://kubernetes.io/docs/concepts/workloads/controllers/cron-jobs/