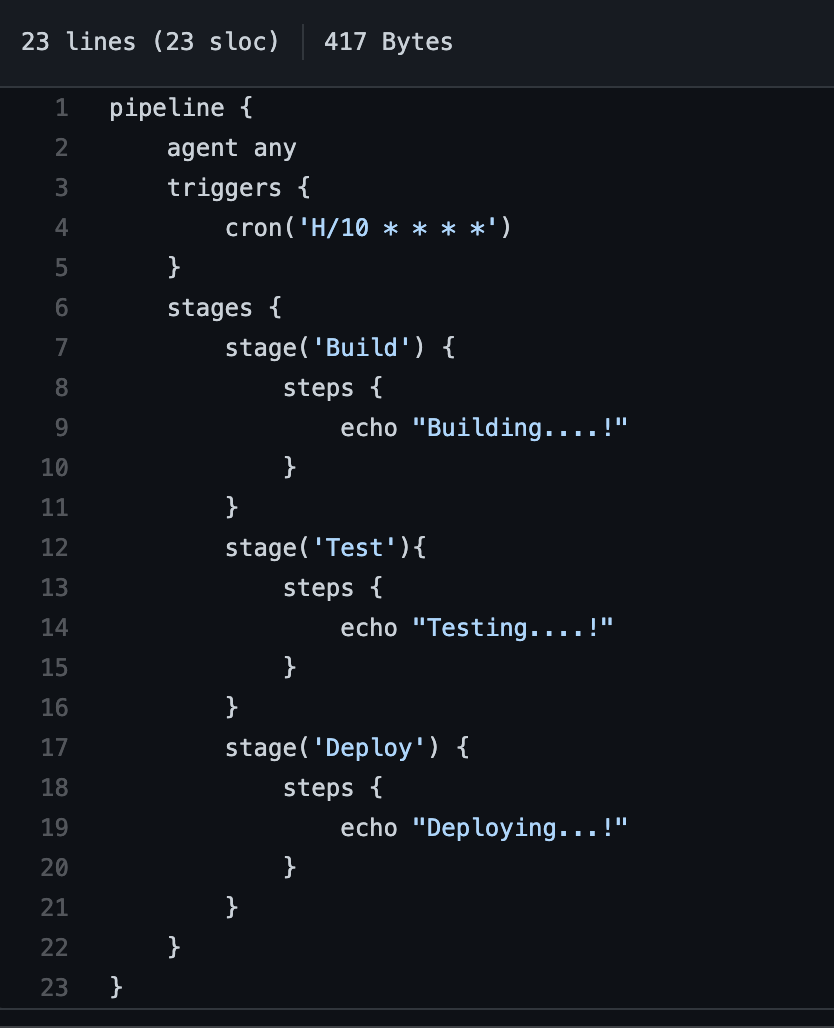
**Documentation for Deployment (Cron Job)**

Objectives

Create a Pipeline with a Build, Test, and Development stage.

Trigger the build for the Pipeline every 10 minutes.

Schedule your EC2 instance to shutdown by the end of class.



Creating the Pipeline & Planting the Trigger

Step 1: Write the script for the Pipeline in the Jenkinsfile in

Github.

Step 2: Include the trigger: cron(‘H/10 \* \* \* \*’)

The syntax means every 10 minutes, and the H is a

hash that is meant to prevent multiple timed jobs from

running at the same time.

Step 3: Fire up your EC2 instance using the AWS dashboard.

Connect to it using “it’s public ip”:8080 as the address.

Step 4: Log into Jenkins, and select New Item > Multibranch

Pipeline. Under “New Branch” select Add Source >

Github > Add > Jenkins. In the “Add Credentials”

window, enter your github username in Username, your github personal access token in Password, and an identifier in ID. Next to the “Add” button, click the drop down menu > your username. Select Repository Scan and in Owner type your github username. Select the repository holding your Jenkinsfile from step 1. Hit Save! The Pipeline has been built and will be built every 10 minutes.

Scheduling your EC2’s shutdown

Step 1: SSH into your EC2 instance.

Step 2: Create a script, shutdown.sh that will stop your instance. The -h will ensure not only the OS, but also the VM will shut down. You will need to mod it to make it executable, chmod +x shutdown.sh.

#!/bin/bash

sudo shutdown now -h

Step 3: Create a cron job so that your script can execute at a set time. Use, env EDITOR=nano crontab -e to open up a cron file. Using the syntax, 0 21 \* \* \* /home/ec2-user/shutdown.sh. This will execute the shutdown script at 9 PM.  
  
Probem: In order to test the cron job, I would set the time such that the EC2 would stop by the next minute. This repeatedly failed until I checked the date. I noticed that the time zone of the EC2 was 4 hours ahead. Adjusting my value in the hour slot for the cron expression, allowed the cron job to behave as expected. Kawang has since given me the command to change an instance’s time zone.

List a time zone: timedatectl list-timezones

Change the time zone: sudo timedatectl set-timezone <your\_time\_zome>