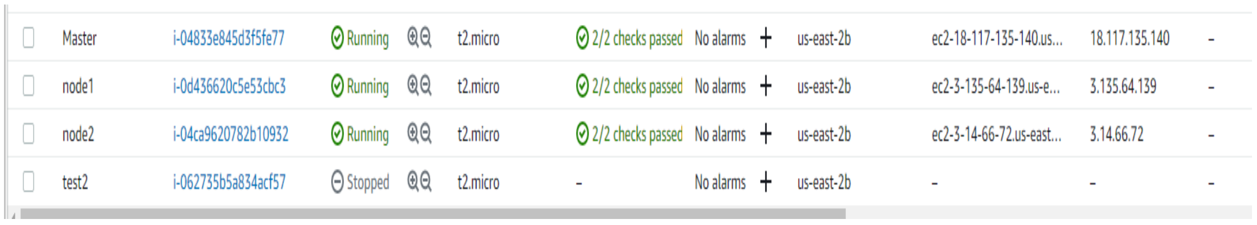
Assignment:22/09/2022

TO BUILD .WAR ON MASTER AND DEPLOY IT ON TWO NODES FOR TWO DIFFERENT PORTS (8080 AND 8090) USING DOCKER-COMPOSE FILE.

STEP1) launch Master and two slave

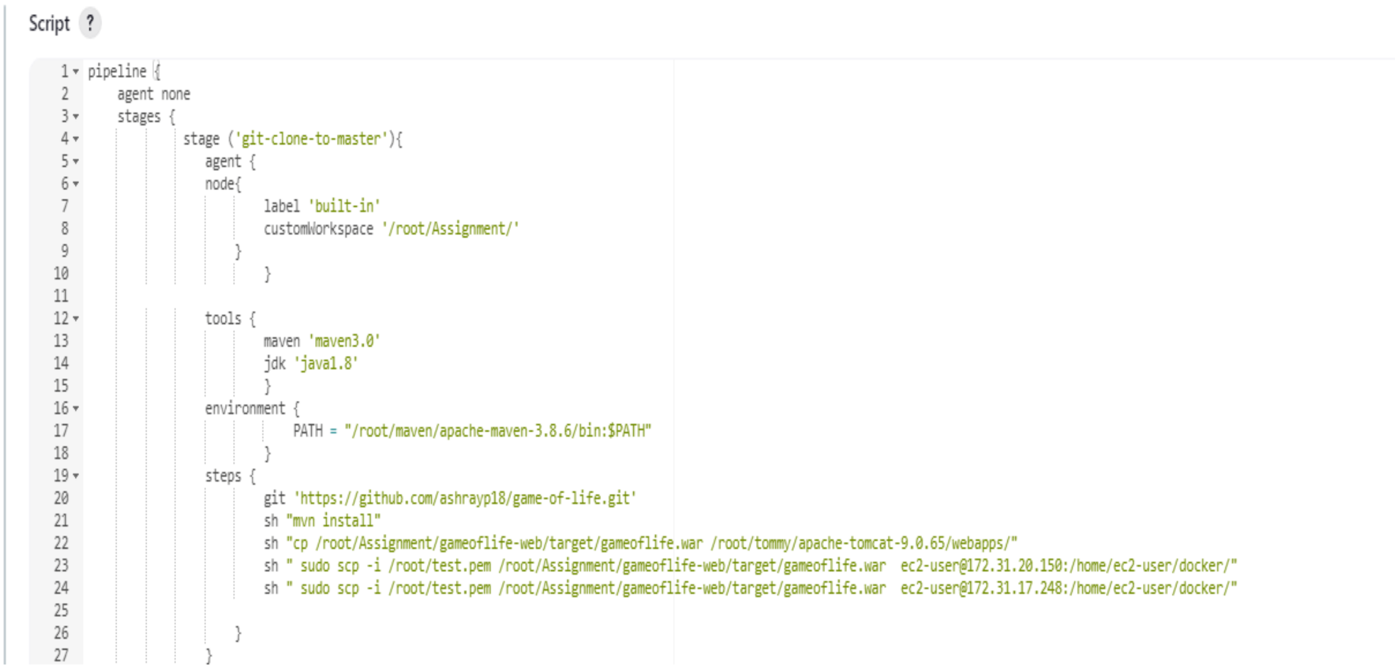


☐	Master	i-04833e845d3f5fe77	✓ Running	t2.micro	✓ 2/2 checks passed	No alarms +	us-east-2b	ec2-18-117-135-140.us...	18.117.135.140	-
☐	node1	i-0d436620c5e53cbc3	✓ Running	t2.micro	✓ 2/2 checks passed	No alarms +	us-east-2b	ec2-3-135-64-139.us-e...	3.135.64.139	-
☐	node2	i-04ca9620782b10932	✓ Running	t2.micro	✓ 2/2 checks passed	No alarms +	us-east-2b	ec2-3-14-66-72.us-east...	3.14.66.72	-
☐	test2	i-062735b5a834acf57	⊖ Stopped	t2.micro	-	No alarms +	us-east-2b	-	-	-

Step2) write code for stage one

Clone the code from Git repo

Build gameoflife.war on Master and also copy it on both the nodes simultaneously



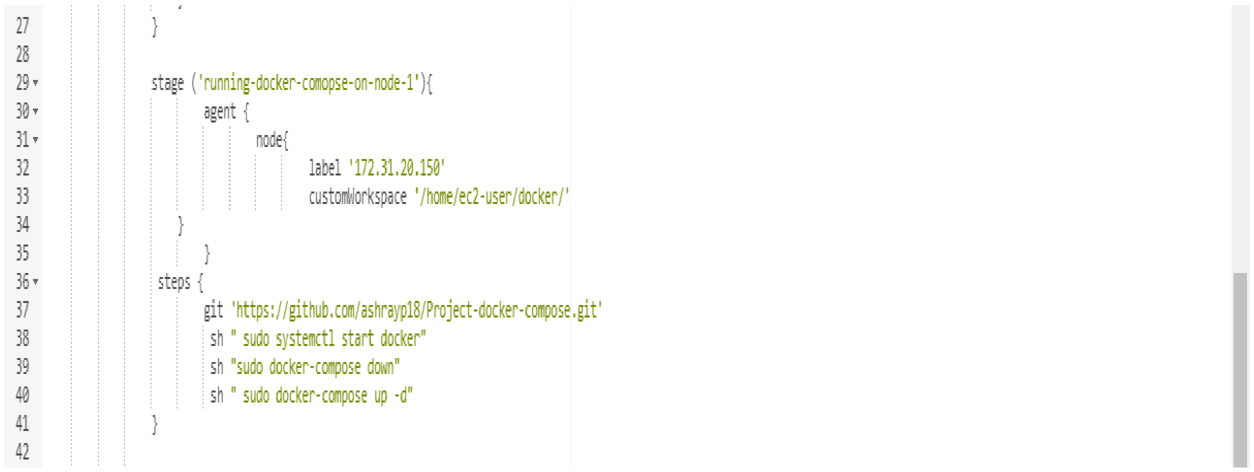
Script ?

```
pipeline {
    agent none
    stages {
            stage ('git-clone-to-master'){
                agent {
                node{
                        label 'built-in'
                        customWorkspace '/root/Assignment/'
                    }
                        }

                tools {
                        maven 'maven3.0'
                        jdk 'java1.8'
                        }
                environment {
                            PATH = "/root/maven/apache-maven-3.8.6/bin:$PATH"
                        }
                steps {
                        git 'https://github.com/ashrayp18/game-of-life.git'
                        sh "mvn install"
                        sh "cp /root/Assignment/gameoflife-web/target/gameoflife.war /root/tommy/apache-tomcat-9.0.65/webapps/"
                        sh " sudo scp -i /root/test.pem /root/Assignment/gameoflife-web/target/gameoflife.war  ec2-user@172.31.20.150:/home/ec2-user/docker/"
                        sh " sudo scp -i /root/test.pem /root/Assignment/gameoflife-web/target/gameoflife.war  ec2-user@172.31.17.248:/home/ec2-user/docker/"

                    }
                }
```

STEP3)

COPY THE GAMEOFLIFE.WAR to Node1 (DEV ENV)

CLONE THE GIT REPO (DOCKER-COMPOSE FILE) to NODE1

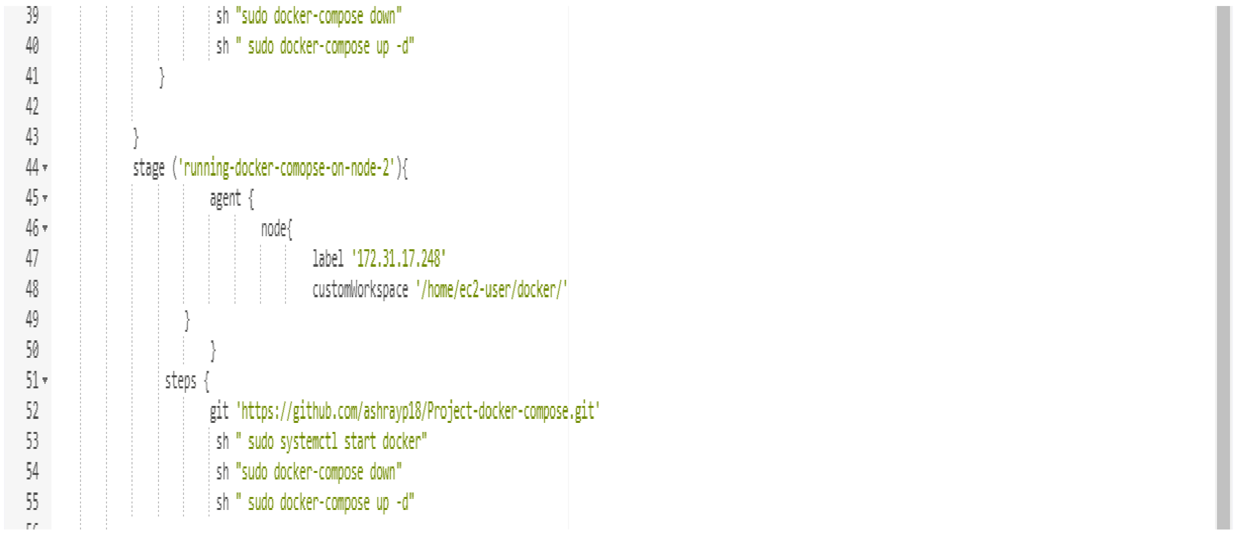


```
        }

        stage ('running-docker-comopse-on-node-1'){
            agent {
                node{
                    label '172.31.20.150'
                    customWorkspace '/home/ec2-user/docker/'
            }
              }
         steps {
              git 'https://github.com/ashrayp18/Project-docker-compose.git'
               sh " sudo systemctl start docker"
               sh "sudo docker-compose down"
               sh " sudo docker-compose up -d"
        }
```

STEP4)

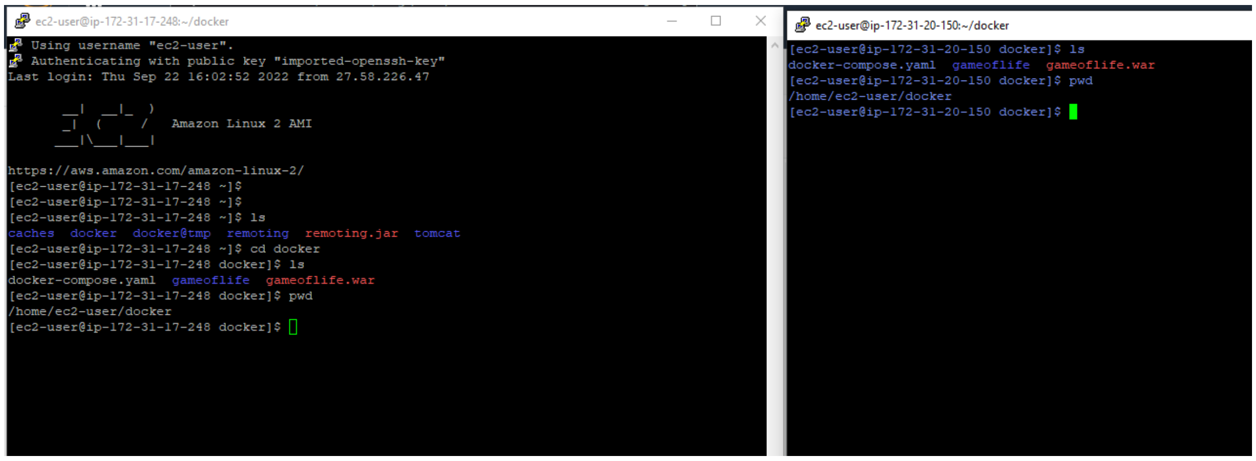
COPY THE GAMEOFLIFE.WAR to Node2 (QA ENV)

CLONE THE GIT REPO (DOCKER-COMPOSE FILE) to NODE2



```
               sh "sudo docker-compose down"
               sh " sudo docker-compose up -d"
          }

        }
        stage ('running-docker-comopse-on-node-2'){
              agent {
                  node{
                      label '172.31.17.248'
                      customWorkspace '/home/ec2-user/docker/'
              }
                }
           steps {
                git 'https://github.com/ashrayp18/Project-docker-compose.git'
                 sh " sudo systemctl start docker"
                 sh "sudo docker-compose down"
                 sh " sudo docker-compose up -d"
```

STEP5) LOGIN AND CHECK ON NODES IF FILES ARE PRESENT



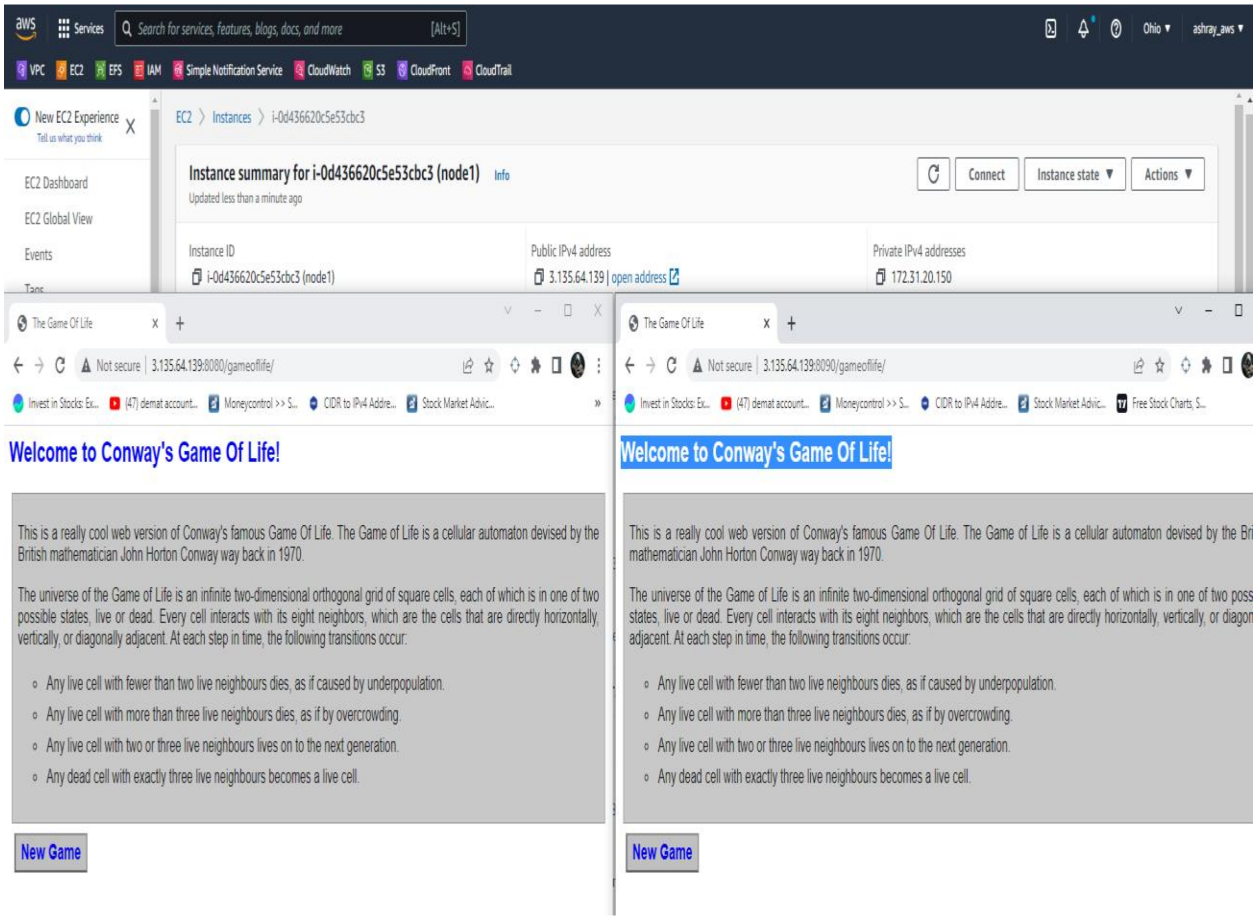
ec2-user@ip-172-31-17-248:~/docker

```
Using username "ec2-user".
Authenticating with public key "imported-openssh-key"
Last login: Thu Sep 22 16:02:52 2022 from 27.58.226.47

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-17-248 ~]$
[ec2-user@ip-172-31-17-248 ~]$
[ec2-user@ip-172-31-17-248 ~]$ ls
caches  docker  docker@tmp  remoting  remoting.jar  tomcat
[ec2-user@ip-172-31-17-248 ~]$ cd docker
[ec2-user@ip-172-31-17-248 docker]$ ls
docker-compose.yaml  gameoflife  gameoflife.war
[ec2-user@ip-172-31-17-248 docker]$ pwd
/home/ec2-user/docker
[ec2-user@ip-172-31-17-248 docker]$
```

ec2-user@ip-172-31-20-150:~/docker

```
[ec2-user@ip-172-31-20-150 docker]$ ls
docker-compose.yaml  gameoflife  gameoflife.war
[ec2-user@ip-172-31-20-150 docker]$ pwd
/home/ec2-user/docker
[ec2-user@ip-172-31-20-150 docker]$
```

STEP6) CHECK THE PUBLIC IP FOR NODE ONE ON PORT 8080 and 8090



EC2 > Instances > i-0d436620c5e53cbc3

Instance summary for i-0d436620c5e53cbc3 (node1) Info

Updated less than a minute ago

Instance ID	Public IPv4 address	Private IPv4 addresses
i-0d436620c5e53cbc3 (node1)	3.135.64.139 | open address	172.31.20.150

The Game Of Life — Not secure | 3.135.64.139:8080/gameoflife/

Welcome to Conway's Game Of Life!

This is a really cool web version of Conway's famous Game Of Life. The Game of Life is a cellular automaton devised by the British mathematician John Horton Conway way back in 1970.

The universe of the Game of Life is an infinite two-dimensional orthogonal grid of square cells, each of which is in one of two possible states, live or dead. Every cell interacts with its eight neighbors, which are the cells that are directly horizontally, vertically, or diagonally adjacent. At each step in time, the following transitions occur:

- Any live cell with fewer than two live neighbours dies, as if caused by underpopulation.
- Any live cell with more than three live neighbours dies, as if by overcrowding.
- Any live cell with two or three live neighbours lives on to the next generation.
- Any dead cell with exactly three live neighbours becomes a live cell.

The Game Of Life — Not secure | 3.135.64.139:8090/gameoflife/

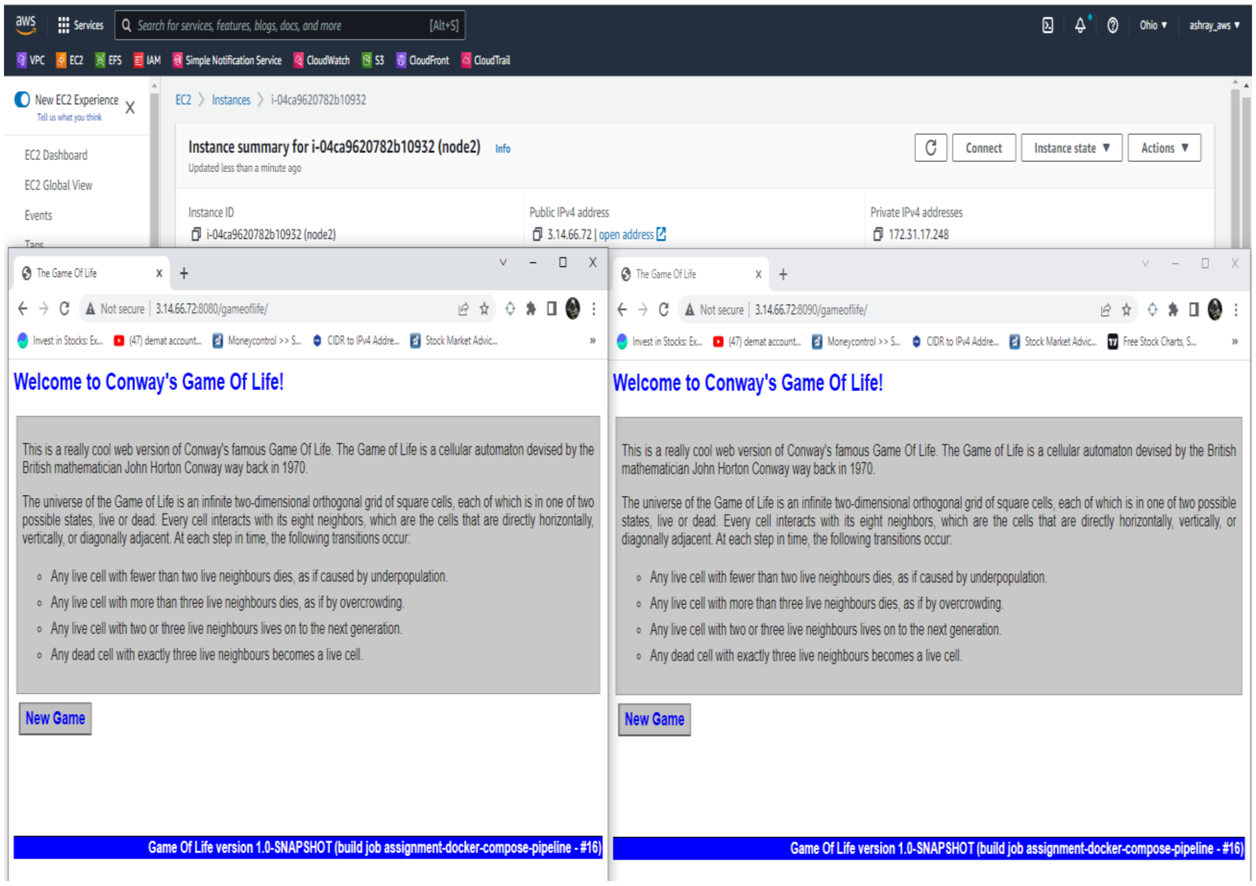
Welcome to Conway's Game Of Life!

This is a really cool web version of Conway's famous Game Of Life. The Game of Life is a cellular automaton devised by the Bri mathematician John Horton Conway way back in 1970.

The universe of the Game of Life is an infinite two-dimensional orthogonal grid of square cells, each of which is in one of two poss states, live or dead. Every cell interacts with its eight neighbors, which are the cells that are directly horizontally, vertically, or diagon adjacent. At each step in time, the following transitions occur:

- Any live cell with fewer than two live neighbours dies, as if caused by underpopulation.
- Any live cell with more than three live neighbours dies, as if by overcrowding.
- Any live cell with two or three live neighbours lives on to the next generation.
- Any dead cell with exactly three live neighbours becomes a live cell.

STEP7) CHECK THE PUBLIC IP FOR NODE ONE ON PORT 8080 and 8090


Instance summary for i-04ca9620782b10932 (node2)

Instance ID: i-04ca9620782b10932 (node2)

Public IPv4 address: 3.14.66.72 | open address

Private IPv4 addresses: 172.31.17.248

3.14.66.72:8080/gameoflife/

3.14.66.72:8090/gameoflife/

Welcome to Conway's Game Of Life!

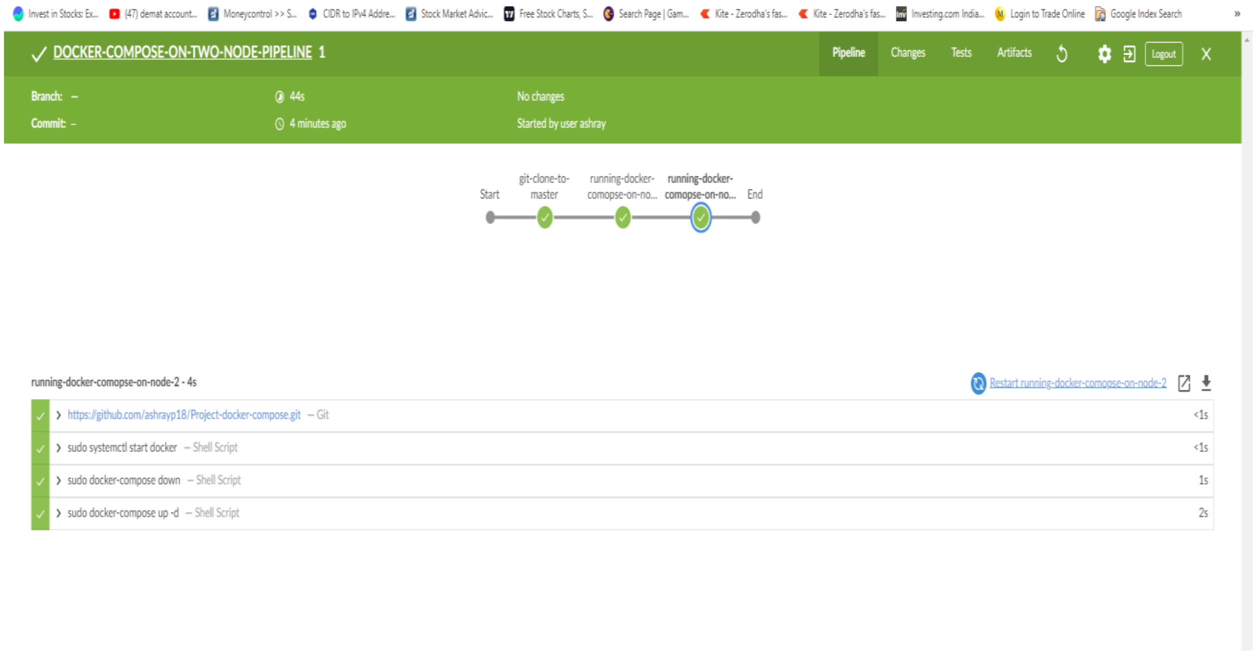
This is a really cool web version of Conway's famous Game Of Life. The Game of Life is a cellular automaton devised by the British mathematician John Horton Conway way back in 1970.

The universe of the Game of Life is an infinite two-dimensional orthogonal grid of square cells, each of which is in one of two possible states, live or dead. Every cell interacts with its eight neighbors, which are the cells that are directly horizontally, vertically, or diagonally adjacent. At each step in time, the following transitions occur:

- Any live cell with fewer than two live neighbours dies, as if caused by underpopulation.
- Any live cell with more than three live neighbours dies, as if by overcrowding.
- Any live cell with two or three live neighbours lives on to the next generation.
- Any dead cell with exactly three live neighbours becomes a live cell.

New Game

Game Of Life version 1.0-SNAPSHOT (build job assignment-docker-compose-pipeline - #16)


DOCKER-COMPOSE-ON-TWO-NODE-PIPELINE 1

Branch: – 44s No changes
Commit: – 4 minutes ago Started by user ashray

Start → git-clone-to-master → running-docker-compose-on-no... → running-docker-compose-on-no... → End

running-docker-compose-on-node-2 - 4s

```
https://github.com/ashrayp18/Project-docker-compose.git – Git   <1s
sudo systemctl start docker – Shell Script                       <1s
sudo docker-compose down – Shell Script                           1s
sudo docker-compose up -d – Shell Script                          2s
```