|  |  |
| --- | --- |
| **DOCUMENT RULES:** | |
| **Task Number / Name:** | **Omarm** |
| **Task name & column name should be written:** | **Bold (CTRL+B)** |
| **Commands should be written in the after # sign:** | *Italic (CTRL+I) #hostname* |
| **Output photo should be cropped or compressed:**  **Photo could be more than one:**  **If you need extra lines, add the line next after it:** | ***Description photo should be with title bar (CTRL + I + B)*** |
| **All other text should be written:** | Standard |
| **Font name and text size:** | Calibri and 9 |
| **Group name:** | Dev\_ops\_ |
| **Student name and surname:** | Murad Abbaszade |
| **E-mail:** | [muradabbaszade6@gmail.com](mailto:muradabbaszade6@gmail.com) |
| **WhatsApp number:** | **+994703664205** |

|  |  |
| --- | --- |
| 1.Install Jenkins and Docker |  |
| docker run -d -u root --name jenkins -p 8080:8080 -p 50000:50000 -v /root/jenkins:/var/jenkins\_home omarmohsen/jenkins-sa |  |
| docker ps |  |
| docker logs jenkins -f |  |
| 2.Access Jenkins and install plugins |  |
| Install Java on your machine which is a dependency for the jenkins agent to work:  sudo apt install openjdk-11-jre |  |
| 1- Click New Item .  2- Select Pipeline .  3- Give it a name, django-pipeline .  4- Click Ok , this will open a new page for you. |  |
| 5- Scroll to the bottom to the Pipeline section, this is where you will type your groovy code in.  6- Paste the following in the Pipeline section, this is the basic structure of the pipeline | pipeline {  agent { //Agent name specification  node {  label 'test' //Specify agent name  }  }  stages { //State stages of pipeline  stage('') { //Declare your first stage. (all stages must be inside the stages block)  steps { //Declare your first stage. (all stages must be inside the stages block)  }  }  stage('') {  steps {}  }  stage('') {  steps {}  }  stage('') {  steps {}  }  stage('') {  steps {}  }  }  } |
| as your first stage, you will need to paste the following: | stage('Cloning the repository') {  steps {  git url: 'https://github.com/omarmohsen/django-blog'  }  } |
| In this step, you will configure a stage to run unit test on the code you cloned from the repository. | stage('run tests') {  steps {  sh 'python3 manage.py test blog'  }  } |
| we will substitute the run python app stage with the following: | stage('Build docker image') {  steps {  sh 'docker build -t django-blog -f docker/Dockerfile . '  }  } |
| Now lets Run the pipeline and test the stages you made.  Now check your container if its running by executing  docker ps | stage('Run docker container') {  steps {  sh 'docker run --rm -d -p 80:8000 django-blog'  }  } |
| 2.Django App deployment |  |
| git clone https://github.com/omarmohsen/django-blog.git |  |
| ls -ltr django-blog |  |
| django-blog/requirements.txt{{open}} The file that contain all Python requirements for the application to run.  django-blog/manage.py{{open}} The main python file used for the application to run. |  |
| pip3 install -r requirements.txt |  |
| Python3 manage.py migrate  To run it, execute this command python3 manage.py runserver 0.0.0.0:8000 |  |