Зеркала: /etc/apt/sources.list:

Pip3 – sudo apt install python3-pip

Environment building

* apt-get install python3-venv
* python3 –m venv <name>
* source <name>/bin/activate
* Upgrade python3
* Upgrade pip3
* pip3 install aiogram

Upgrade python3

|  |
| --- |
| * sudo apt update * sudo apt install python3.9 * sudo update-alternatives --install /usr/bin/python3 python3 /usr/bin/python3.[old-version] 1 * sudo update-alternatives --install /usr/bin/python3 python3 /usr/bin/python3.9 2 * sudo update-alternatives --config python3 * sudo rm /usr/bin/python3 * sudo ln -s python3.9 /usr/bin/python3 * python3 --version |

Upgrade pip3

|  |
| --- |
| * sudo apt-get install python3-apt –reinstall * cd /usr/lib/python3/dist-packages * ls -l | grep apt\_pkg * sudo ln -s apt\_pkg.cpython-{3.7}-x86\_64-linux-gnu.so apt\_pkg.so * sudo apt install --reinstall python3-apt * python3 -m pip --version * python3 -m pip install --upgrade pip |

If apt\_pkg no modile error appear

|  |
| --- |
| * sudo apt remove python3-apt * sudo apt autoremove * sudo apt autoclean * sudo apt install python3-apt |

GitHub

|  |  |
| --- | --- |
| Git management | Installation to EC2 |
| 1. git clone <link\_to\_the\_repository> - Clone project to the local  2. echo “Hello, World” > file.txt – Create new file in the local repo  3. git status – Check the commits  4. git add .  5. git commit –m “message”  6. git push origin main  7. git pull <link to the repository>  7. To remove the last commit from git, you can simply run git reset --hard HEAD^ | |  | | --- | | Installation to EC2  1. sudo yum update  2. sudo yum install git  3. git version (git version 2.39.2)  4. mkdir mygit |  |  | | --- | | SSH keys (they must be in ~/.ssh/ directory)  1. touch keys  2. ssh-keygen –o –t rsa –C “mail”  3. Add the public key to the ssh in the github  4. eval "$(ssh-agent -s)"  5. ssh-add ./privatekey |  |  | | --- | | "Authentication Failed" – You may also need to update the origin for your repository if it is set to HTTPS. Do this to switch to SSH:  git remote -v  git remote set-url origin git@github.com:annihilatoratm/test\_tg\_bot.git | |

EC2

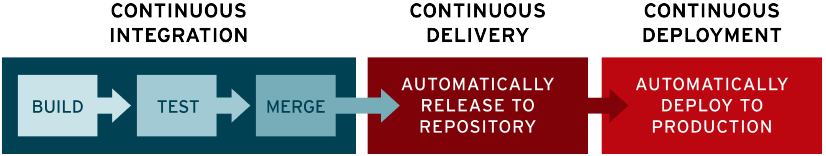
|  |
| --- |
| To copy the file from local machine to EC2 instance:  #scp -i identity\_file.pem source\_file.extention username@public\_ipv4\_dns:/remote\_path  To copy the file from EC2 instance to local machine:  #scp -i identity\_file.pem username@public\_ipv4\_dns:/remote\_path/source\_file.extension ~/destination\_local\_path  To copy the directory from local machine to EC2 instance:  #scp -i identity\_file.pem -r source\_dir username@public\_ipv4\_dns:/remote\_dir\_path  To copy the directory from EC2 instance to local machine:  #scp -i identity\_file.pem -r username@public\_ipv4\_dns:/remote\_dir\_path destination\_dir |

Docker

|  |
| --- |
| Installation  1. sudo apt update / sudo yum update  2. sudo apt install apt-transport-https ca-certificates curl software-properties-common / sudo yum search docker  3. curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add - / sudo yum info docker  4. sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu bionic stable" / sudo yum install docker  5. sudo apt update / sudo usermod -a -G docker ec2-user  6. apt-cache policy docker-ce  7. sudo apt install docker-ce  8. sudo systemctl status docker  1. docker build –t <name> <path to directory>  2. docker images – list of the images  3. docker run <image> - run the image  4. docker ps – a – list of containers  5. docker exec <container> - connect to the container  6. docker cp src\_file <container>:dest\_path – copy the file from local to container  7. apt install dos2unix – switch format from dos to unix system  1. docker login <> docker logout  2. docker tag <image name> <usernameofdocker>/<repository>  3. docker pull <usernameofdocker>/<repository>:latest |

CI/CD – GitHub Actions

**The main concepts attributed to CI/CD are continuous integration, continuous delivery and continuous deployment.**



|  |
| --- |
| 1. Build Code > Make a docker image > Push the image to Docker Hub (#docker push)  2. Pull the image in AWS EC2  3. A Github Actions workflow is trigged when an event occurs in your repository. All the configurations are set by YAML syntax and each workspace is stored in your repository, in a directory called .github/workflows.  4. Example of |

Wordpress

|  |
| --- |
| LAMP  sudo apt install lamp-server^  mysql –u root –p  create database wpss\_test\_db default character set utf8;  create user 'wpss\_user'@'localhost' identified by 'qwerty123456A%';  grant all on wpss\_test\_db.\* to 'wpss\_user'@'localhost';  flush privileges;  exit  apt install php-curl php-curl php-gd php-intl php-mbstring php-soap php-xml php-zip php-xmlrpc php-mysql php-cli  wget -c <http://wordpress.org/latest.tar.gz>  tar -xzvf latest.tar.gz  rsync -av wordpress/\* /var/www/html/  chown -R www-data:www-data /var/www/html  chmod -R 755 /var/www/html/  rm /var/www/html/index.html  username: studentraizer  passwd for wordpress: qwerty123456A%  #!/bin/bash  # This will remove Apache  sudo service apache2 stop  sudo apt-get purge apache2 apache2-utils apache2.2-bin  sudo apt remove apache2.\*  sudo apt-get autoremove  whereis apache2  sudo rm -rf /etc/apache2  # This will remove PHP version. Type your php version before run below command. I am using php 8.0.x change yours...  php --version  sudo apt-get purge `dpkg -l | grep php8.0| awk '{print $2}' |tr "\n" " "`  sudo apt-get purge php8.\*  sudo apt-get autoremove --purge  whereis php  sudo rm -rf /etc/php  sudo apt update -y  sudo apt upgrade -y  php --version  # This will remove MYSql  sudo service mysql stop  sudo apt-get remove --purge \*mysql\\*  sudo apt-get remove --purge mysql-server mysql-client mysql-common -y  sudo rm -rf /etc/mysql  sudo apt-get autoremove  sudo apt-get autoclean  sudo reboot |