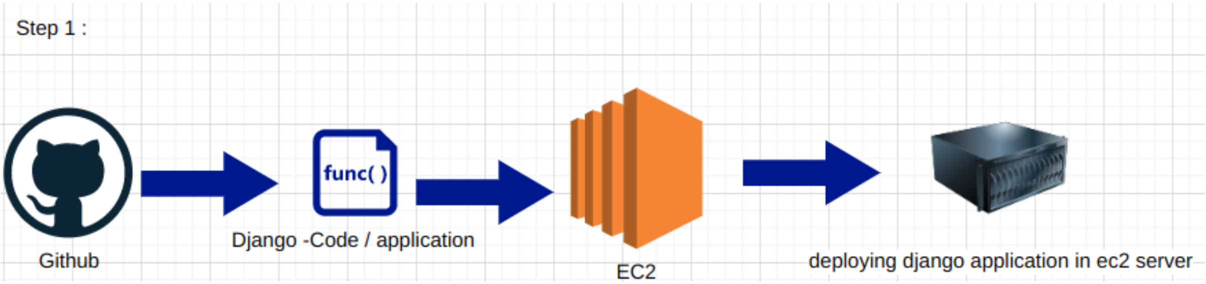
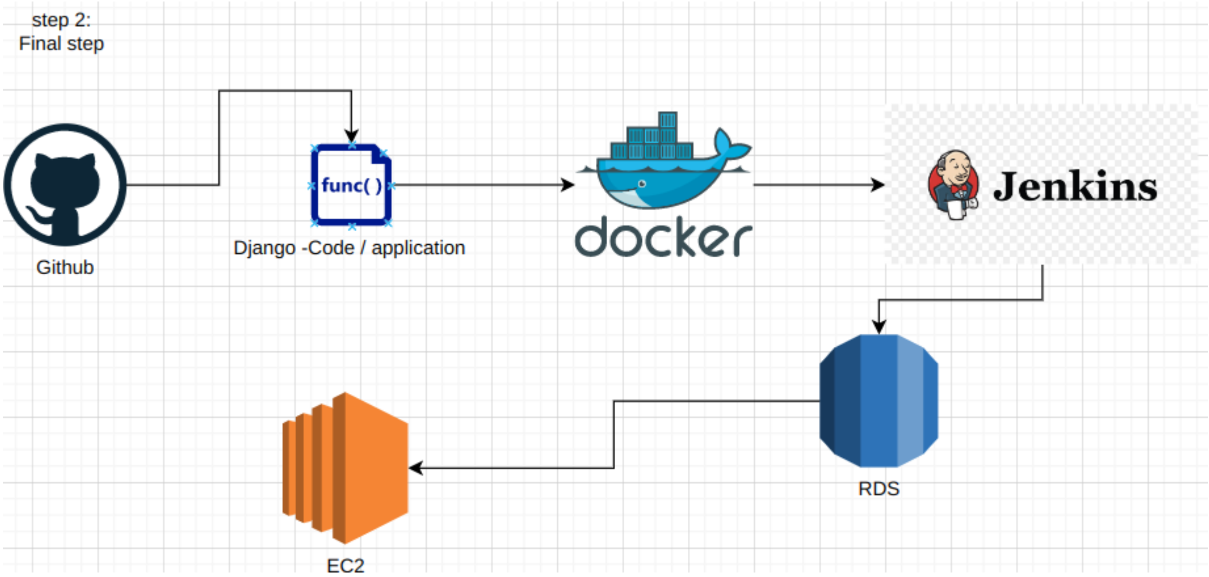
Devops Project 1 : Deploying django application into aws using docker , jenkins

Basic structure of what i did in this project : (as of now step 1 is done)


Step 1 :
func( )
Github
Django -Code / application
EC2
deploying django application in ec2 server

And final step of this project will be like this :



Full diagram can be found here : link

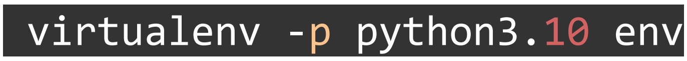
Basic step : I took one sample django application from github and i tested that application into my local machine with all environment dependencies and i uploaded that application into server (aws).

Basic django app link : app link

Step 1 : I've cloned the django app into my local machine using git clone and tested that application as well .

Step 2: I have created a virtual environment into my local machine where I have tested and automated this django application just to prevent my local machine from having a django application.

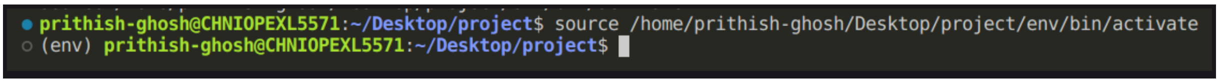
Code :



```
virtualenv -p python3.10 env
```

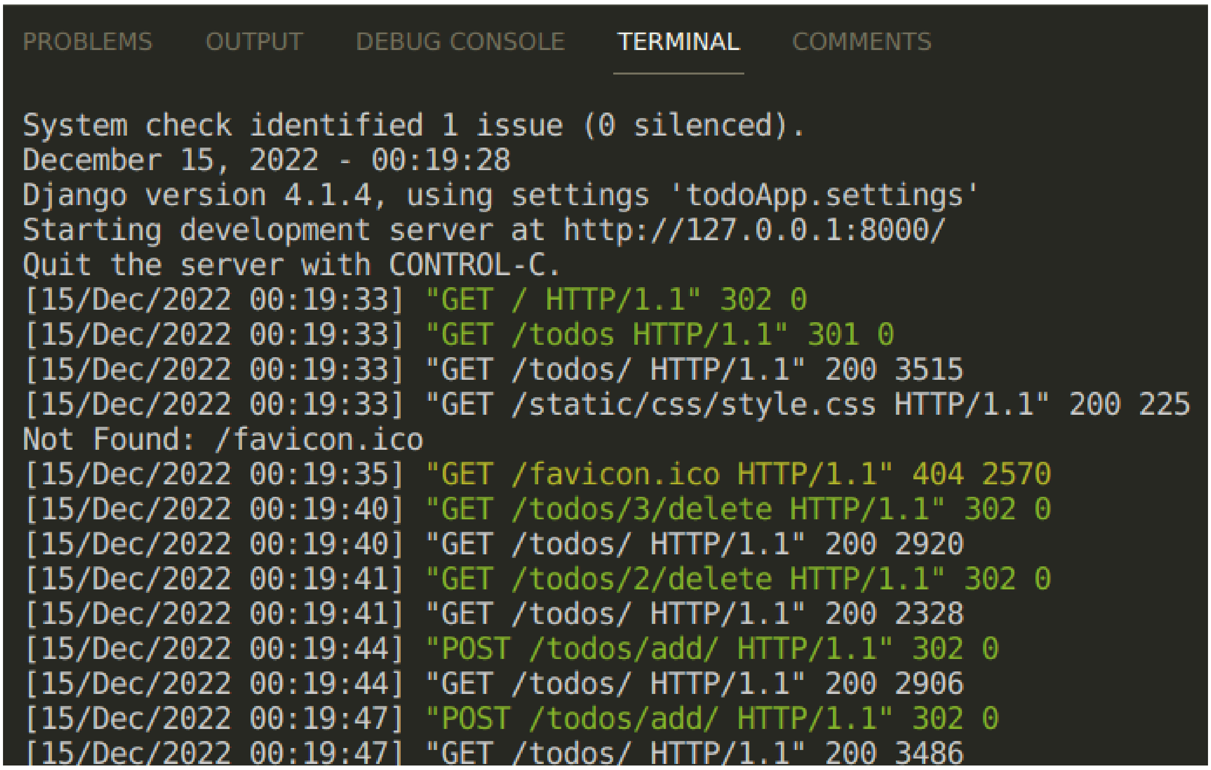
Then activated environment using this :

```
source/project/env/bin/activate
```



```
● prithish-ghosh@CHNIOPEXL5571:~/Desktop/project$ source /home/prithish-ghosh/Desktop/project/env/bin/activate
○ (env) prithish-ghosh@CHNIOPEXL5571:~/Desktop/project$ 
```

Step 3: While testing whether the app is working or not I found out a lot of missed requirements in the README.MD file and for reference purposes I have installed all dependencies and tested that application into my local machine .



```
System check identified 1 issue (0 silenced).
December 15, 2022 - 00:19:28
Django version 4.1.4, using settings 'todoApp.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CONTROL-C.
[15/Dec/2022 00:19:33] "GET / HTTP/1.1" 302 0
[15/Dec/2022 00:19:33] "GET /todos HTTP/1.1" 301 0
[15/Dec/2022 00:19:33] "GET /todos/ HTTP/1.1" 200 3515
[15/Dec/2022 00:19:33] "GET /static/css/style.css HTTP/1.1" 200 225
Not Found: /favicon.ico
[15/Dec/2022 00:19:35] "GET /favicon.ico HTTP/1.1" 404 2570
[15/Dec/2022 00:19:40] "GET /todos/3/delete HTTP/1.1" 302 0
[15/Dec/2022 00:19:40] "GET /todos/ HTTP/1.1" 200 2920
[15/Dec/2022 00:19:41] "GET /todos/2/delete HTTP/1.1" 302 0
[15/Dec/2022 00:19:41] "GET /todos/ HTTP/1.1" 200 2328
[15/Dec/2022 00:19:44] "POST /todos/add/ HTTP/1.1" 302 0
[15/Dec/2022 00:19:44] "GET /todos/ HTTP/1.1" 200 2906
[15/Dec/2022 00:19:47] "POST /todos/add/ HTTP/1.1" 302 0
[15/Dec/2022 00:19:47] "GET /todos/ HTTP/1.1" 200 3486
```

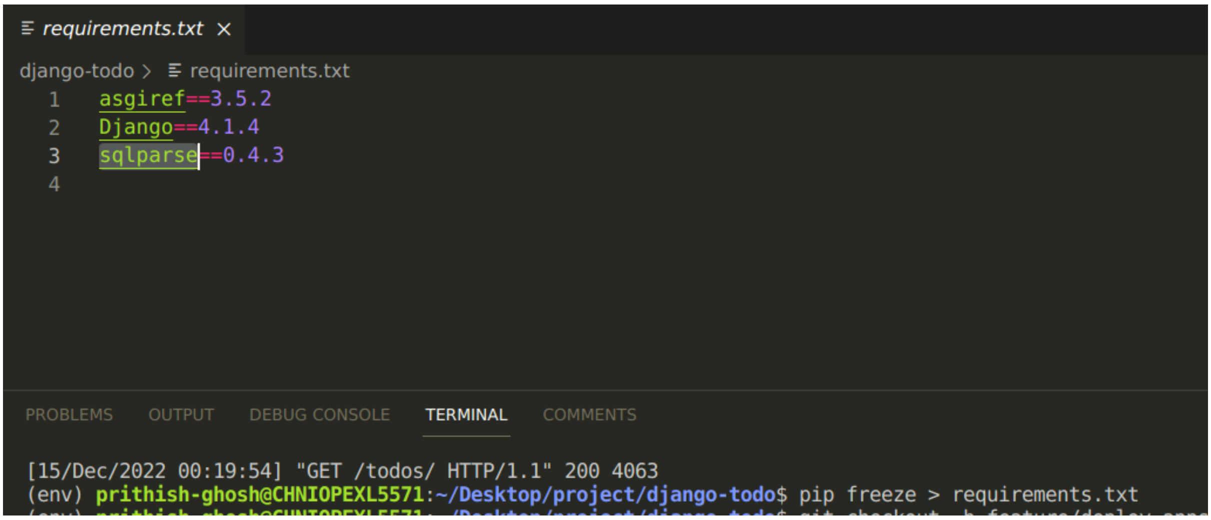
And the django app was working fine but on my local machine .

Dependencies i have used while running this django app :

```
Pip install django (cause this was a django application)
Pip install sqlparse
Python manage.py makemigrations
Python manage.py migrate
Created a superuser
Python manage.py runserver (locally working)
```

Step 3 : This app worked fine in the local machine and the app was functional as well .
I have created a requirement.txt where I have freezed all packages and dependencies to run this django app .

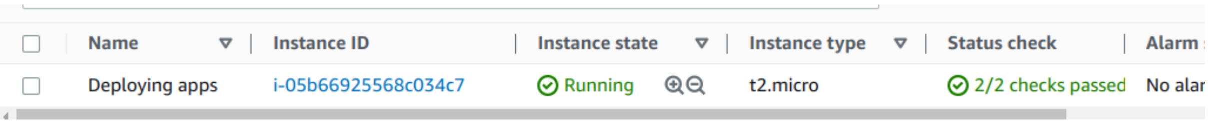
Commands i have used : `pip freeze > requirements.txt`



```
asgiref==3.5.2
Django==4.1.4
sqlparse==0.4.3

```

```
[15/Dec/2022 00:19:54] "GET /todos/ HTTP/1.1" 200 4063
(env) prithish-ghosh@CHNIOPEXL5571:~/Desktop/project/django-todo$ pip freeze > requirements.txt
(env) prithish-ghosh@CHNIOPEXL5571:~/Desktop/project/django-todo$ git checkout -b feature/deploy-apps
```

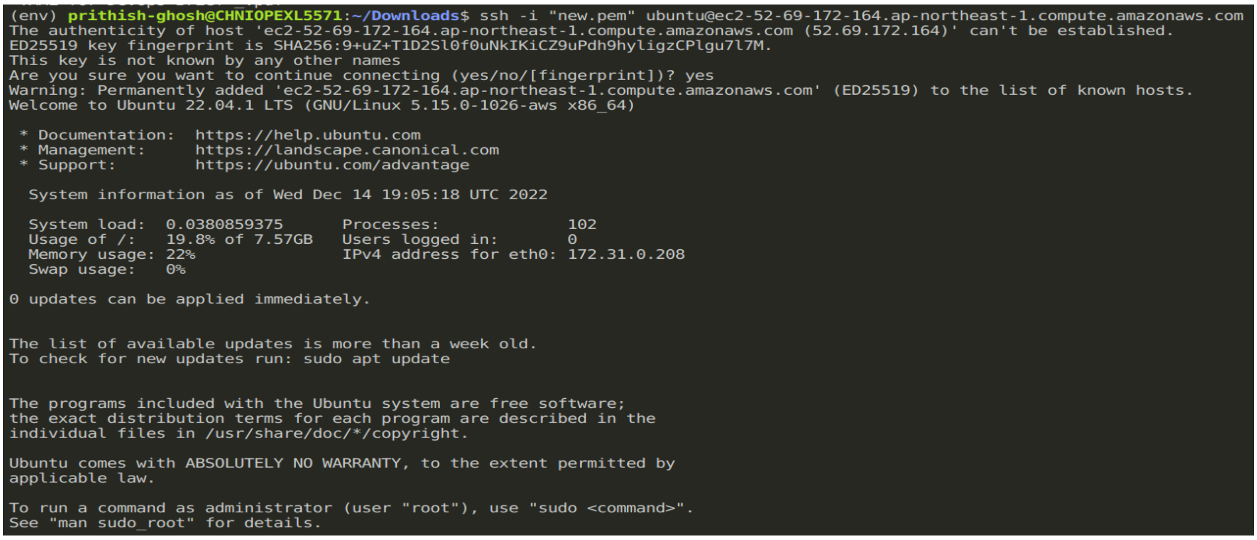
Step 4: Now I have created an EC2 instance in aws with a free tier system which is t2.micro and image set as : ubuntu , while installing i have created a key-pair .pem file for connecting ec2 instance in my local machine using ssh .



□	Name	Instance ID	Instance state	Instance type	Status check	Alarm
□	Deploying apps	i-05b66925568c034c7	✓ Running	t2.micro	✓ 2/2 checks passed	No alar

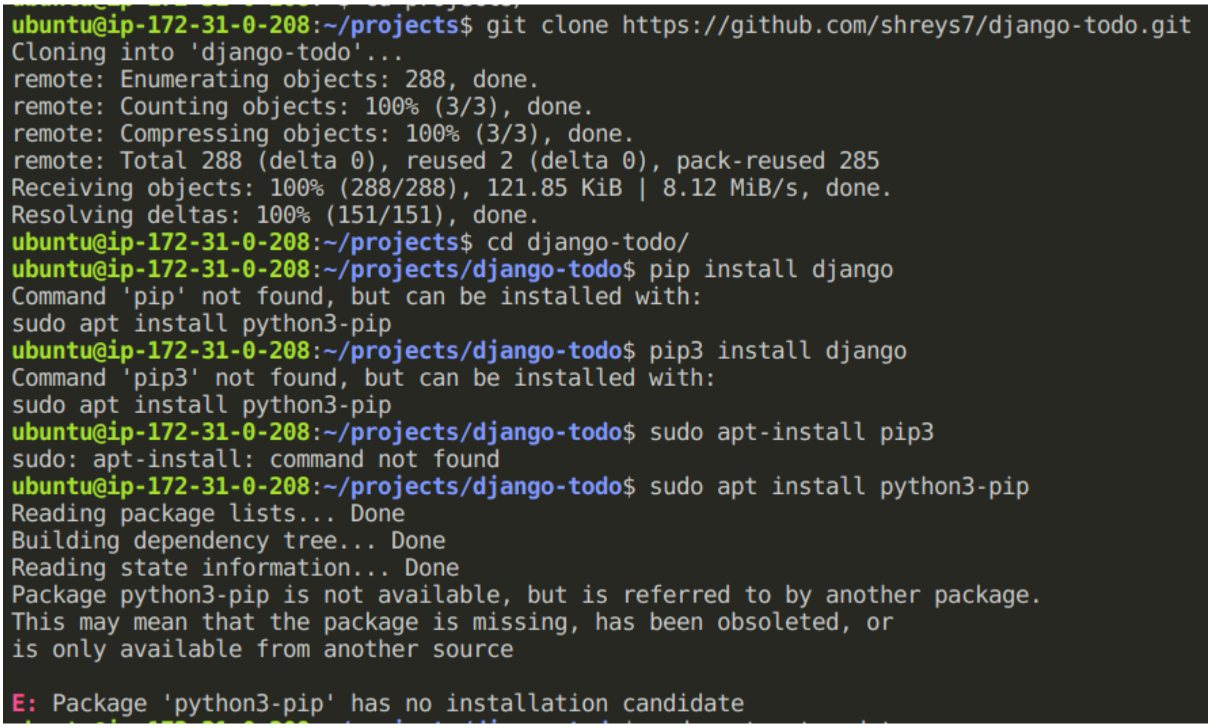
Step 5 : now with .pem file and ssh i have connected ec2 machine using instance id .

```
ssh -i "new.pem"
ubuntu@ec2-52-69-172-164.ap-northeast-1.compute.amazonaws.com
```

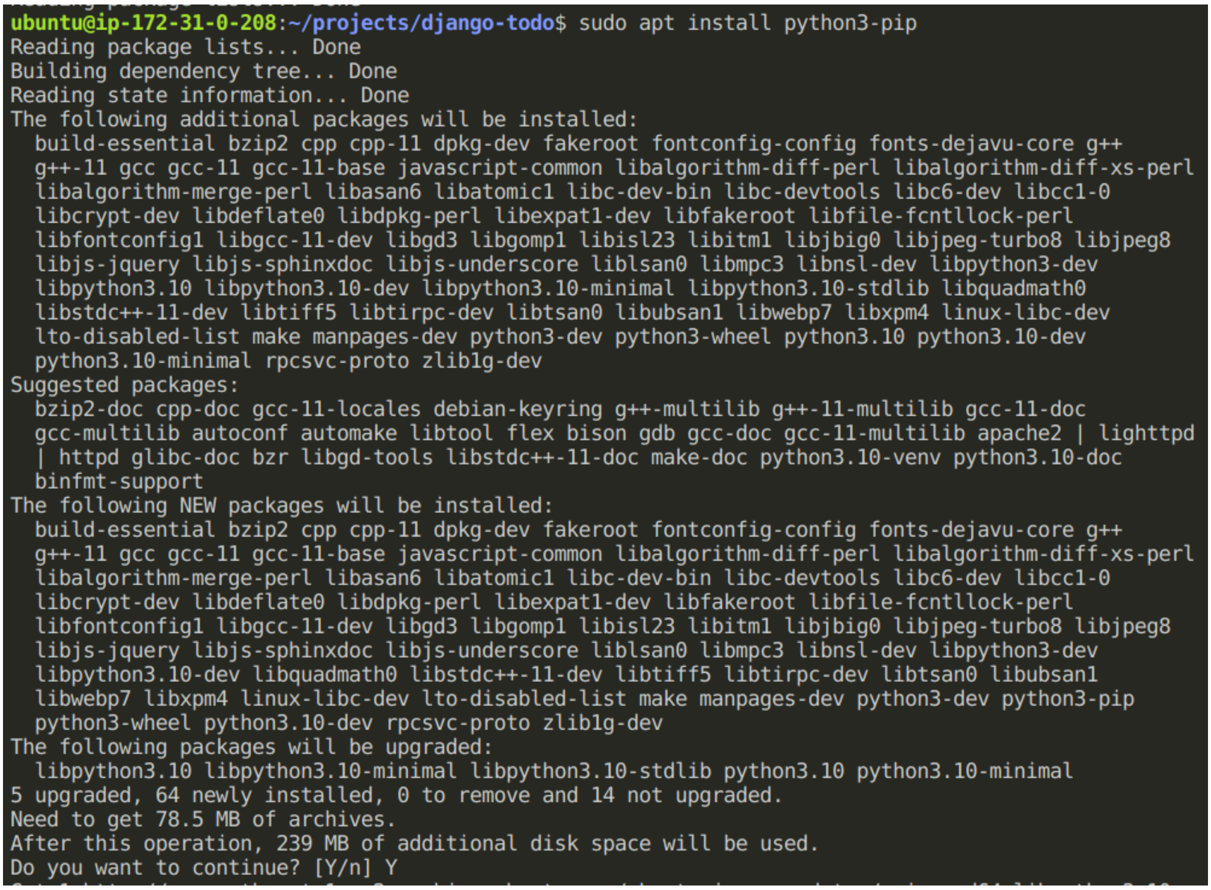


```
(env) prithish-ghosh@CHNIOPEXL5571:~/Downloads$ ssh -i "new.pem" ubuntu@ec2-52-69-172-164.ap-northeast-1.compute.amazonaws.com
The authenticity of host 'ec2-52-69-172-164.ap-northeast-1.compute.amazonaws.com (52.69.172.164)' can't be established.
ED25519 key fingerprint is SHA256:9+uZ+T1D2Sl0f0uNkIKiCZ9uPdh9hyligzCPlgu7l7M.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-52-69-172-164.ap-northeast-1.compute.amazonaws.com' (ED25519) to the list of known hosts.
Welcome to Ubuntu 22.04.1 LTS (GNU/Linux 5.15.0-1026-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Wed Dec 14 19:05:18 UTC 2022

  System load:  0.0380859375      Processes:             102
  Usage of /:   19.8% of 7.57GB   Users logged in:       0
  Memory usage: 22%               IPv4 address for eth0: 172.31.0.208
  Swap usage:   0%

0 updates can be applied immediately.


The list of available updates is more than a week old.
To check for new updates run: sudo apt update


The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.
```

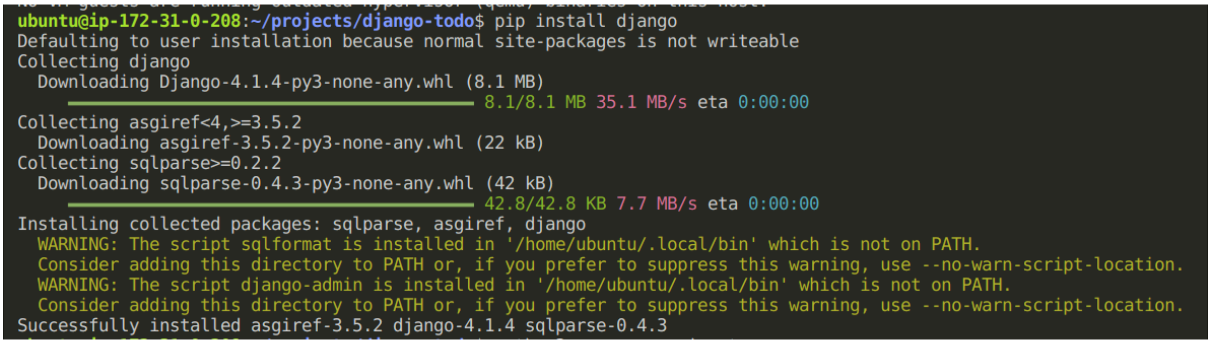
Step 6: Now into the EC2 machine I have created a folder where I have cloned the django app & installed all dependencies that are needed.



```
ubuntu@ip-172-31-0-208:~/projects$ git clone https://github.com/shreys7/django-todo.git
Cloning into 'django-todo'...
remote: Enumerating objects: 288, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 288 (delta 0), reused 2 (delta 0), pack-reused 285
Receiving objects: 100% (288/288), 121.85 KiB | 8.12 MiB/s, done.
Resolving deltas: 100% (151/151), done.
ubuntu@ip-172-31-0-208:~/projects$ cd django-todo/
ubuntu@ip-172-31-0-208:~/projects/django-todo$ pip install django
Command 'pip' not found, but can be installed with:
sudo apt install python3-pip
ubuntu@ip-172-31-0-208:~/projects/django-todo$ pip3 install django
Command 'pip3' not found, but can be installed with:
sudo apt install python3-pip
ubuntu@ip-172-31-0-208:~/projects/django-todo$ sudo apt-install pip3
sudo: apt-install: command not found
ubuntu@ip-172-31-0-208:~/projects/django-todo$ sudo apt install python3-pip
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Package python3-pip is not available, but is referred to by another package.
This may mean that the package is missing, has been obsoleted, or
is only available from another source

E: Package 'python3-pip' has no installation candidate
```



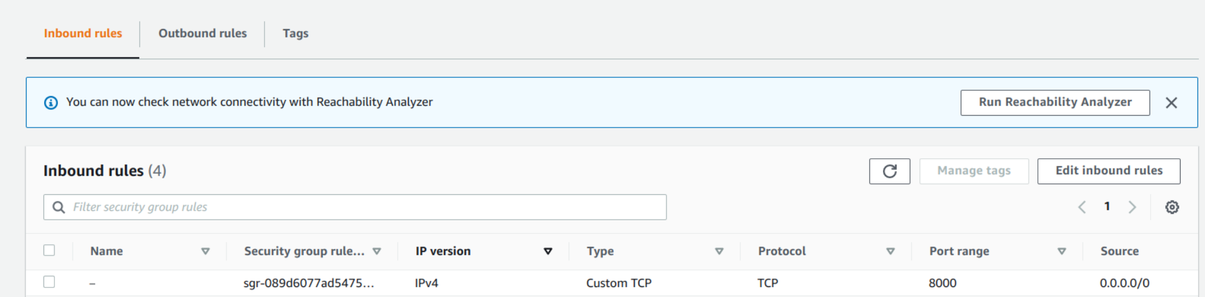
```
ubuntu@ip-172-31-0-208:~/projects/django-todo$ sudo apt install python3-pip
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  build-essential bzip2 cpp cpp-11 dpkg-dev fakeroot fontconfig-config fonts-dejavu-core g++
  g++-11 gcc gcc-11 gcc-11-base javascript-common libalgorithm-diff-perl libalgorithm-diff-xs-perl
  libalgorithm-merge-perl libasan6 libatomic1 libc-dev-bin libc-devtools libc6-dev libcc1-0
  libcrypt-dev libdeflate0 libdpkg-perl libexpat1-dev libfakeroot libfile-fcntllock-perl
  libfontconfig1 libgcc-11-dev libgd3 libgomp1 libisl23 libitm1 libjbig0 libjpeg-turbo8 libjpeg8
  libjs-jquery libjs-sphinxdoc libjs-underscore liblsan0 libmpc3 libnsl-dev libpython3-dev
  libpython3.10 libpython3.10-dev libpython3.10-minimal libpython3.10-stdlib libquadmath0
  libstdc++-11-dev libtiff5 libtirpc-dev libtsan0 libubsan1 libwebp7 libxpm4 linux-libc-dev
  lto-disabled-list make manpages-dev python3-dev python3-wheel python3.10 python3.10-dev
  python3.10-minimal rpcsvc-proto zlib1g-dev
Suggested packages:
  bzip2-doc cpp-doc gcc-11-locales debian-keyring g++-multilib g++-11-multilib gcc-11-doc
  gcc-multilib autoconf automake libtool flex bison gdb gcc-doc gcc-11-multilib apache2 | lighttpd
  | httpd glibc-doc bzr libgd-tools libstdc++-11-doc make-doc python3.10-venv python3.10-doc
  binfmt-support
The following NEW packages will be installed:
  build-essential bzip2 cpp cpp-11 dpkg-dev fakeroot fontconfig-config fonts-dejavu-core g++
  g++-11 gcc gcc-11 gcc-11-base javascript-common libalgorithm-diff-perl libalgorithm-diff-xs-perl
  libalgorithm-merge-perl libasan6 libatomic1 libc-dev-bin libc-devtools libc6-dev libcc1-0
  libcrypt-dev libdeflate0 libdpkg-perl libexpat1-dev libfakeroot libfile-fcntllock-perl
  libfontconfig1 libgcc-11-dev libgd3 libgomp1 libisl23 libitm1 libjbig0 libjpeg-turbo8 libjpeg8
  libjs-jquery libjs-sphinxdoc libjs-underscore liblsan0 libmpc3 libnsl-dev libpython3-dev
  libpython3.10-dev libquadmath0 libstdc++-11-dev libtiff5 libtirpc-dev libtsan0 libubsan1
  libwebp7 libxpm4 linux-libc-dev lto-disabled-list make manpages-dev python3-dev python3-pip
  python3-wheel python3.10-dev rpcsvc-proto zlib1g-dev
The following packages will be upgraded:
  libpython3.10 libpython3.10-minimal libpython3.10-stdlib python3.10 python3.10-minimal
5 upgraded, 64 newly installed, 0 to remove and 14 not upgraded.
Need to get 78.5 MB of archives.
After this operation, 239 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
```



```
ubuntu@ip-172-31-0-208:~/projects/django-todo$ pip install django
Defaulting to user installation because normal site-packages is not writeable
Collecting django
  Downloading Django-4.1.4-py3-none-any.whl (8.1 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 8.1/8.1 MB 35.1 MB/s eta 0:00:00
Collecting asgiref<4,>=3.5.2
  Downloading asgiref-3.5.2-py3-none-any.whl (22 kB)
Collecting sqlparse>=0.2.2
  Downloading sqlparse-0.4.3-py3-none-any.whl (42 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 42.8/42.8 KB 7.7 MB/s eta 0:00:00
Installing collected packages: sqlparse, asgiref, django
  WARNING: The script sqlformat is installed in '/home/ubuntu/.local/bin' which is not on PATH.
  Consider adding this directory to PATH or, if you prefer to suppress this warning, use --no-warn-script-location.
  WARNING: The script django-admin is installed in '/home/ubuntu/.local/bin' which is not on PATH.
  Consider adding this directory to PATH or, if you prefer to suppress this warning, use --no-warn-script-location.
Successfully installed asgiref-3.5.2 django-4.1.4 sqlparse-0.4.3
```

Step 7: Finally I deployed the django app into ec2 but this app is still running at `127.0.0.1:8000`/ which is localhost . Now for exposing this server it should be `0.0.0.0:8000` so that anyone can access this django app.

But the app is still not loading so for that i set inbound rules to allow inbound traffic in this app :

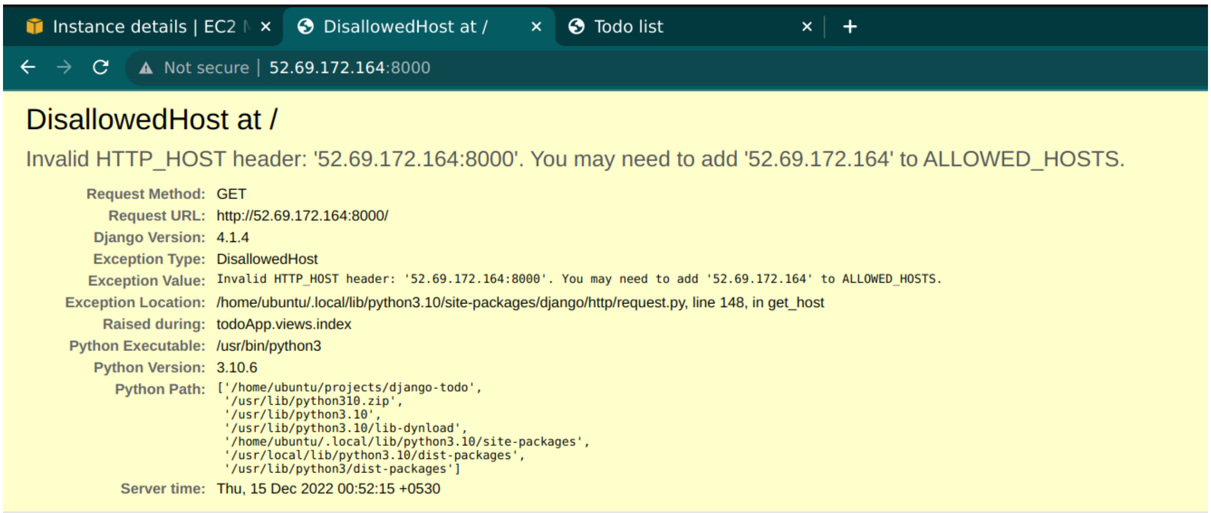


You can now check network connectivity with Reachability Analyzer

Inbound rules (4)

□	Name	Security group rule...	IP version	Type	Protocol	Port range	Source
□	–	sgr-089d6077ad5475...	IPv4	Custom TCP	TCP	8000	0.0.0.0/0

For this i have set command : `python3 manage.py runserver 0.0.0.0:8000`
Now the app is running and showing this :

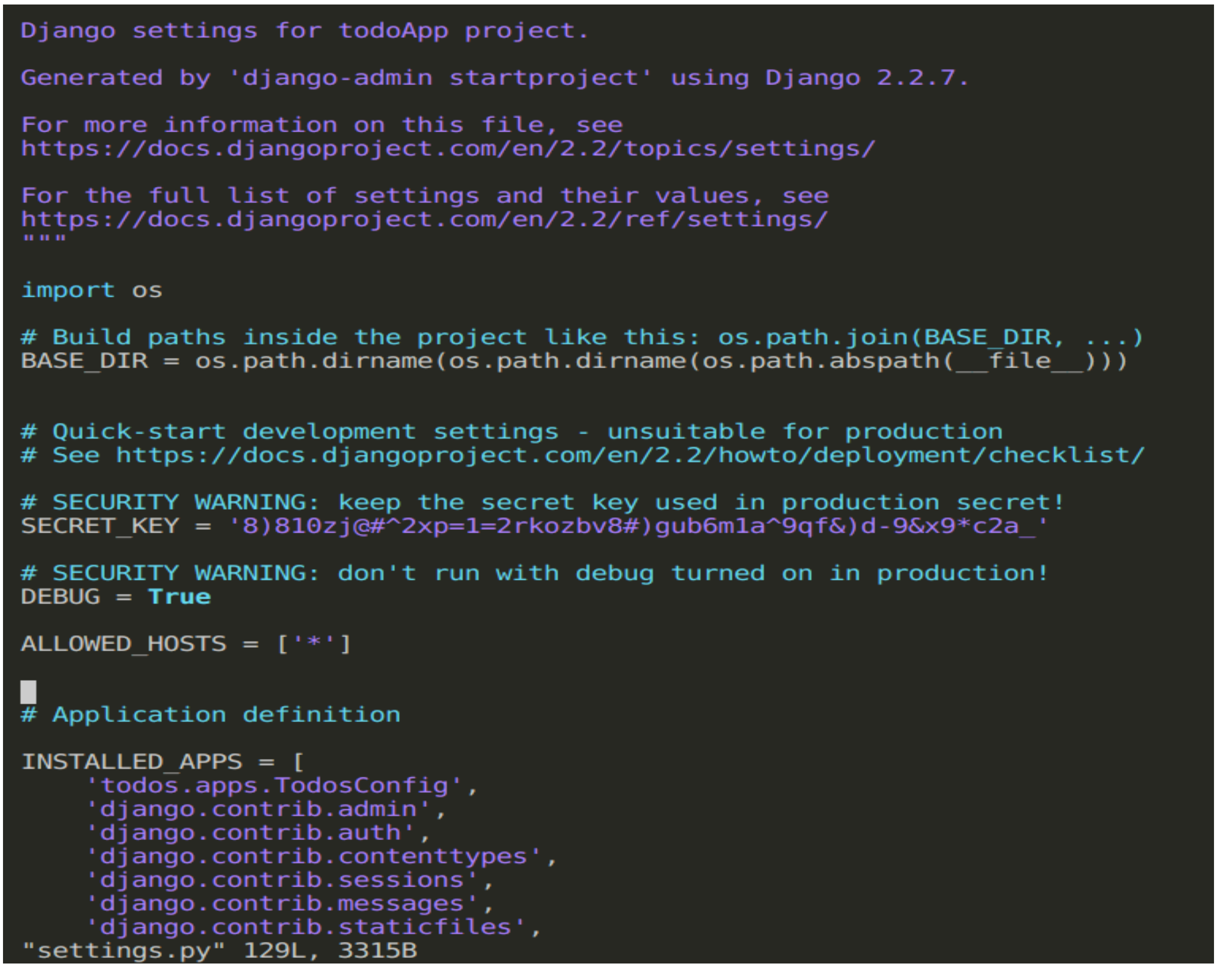


DisallowedHost at /

Invalid HTTP_HOST header: '52.69.172.164:8000'. You may need to add '52.69.172.164' to ALLOWED_HOSTS.

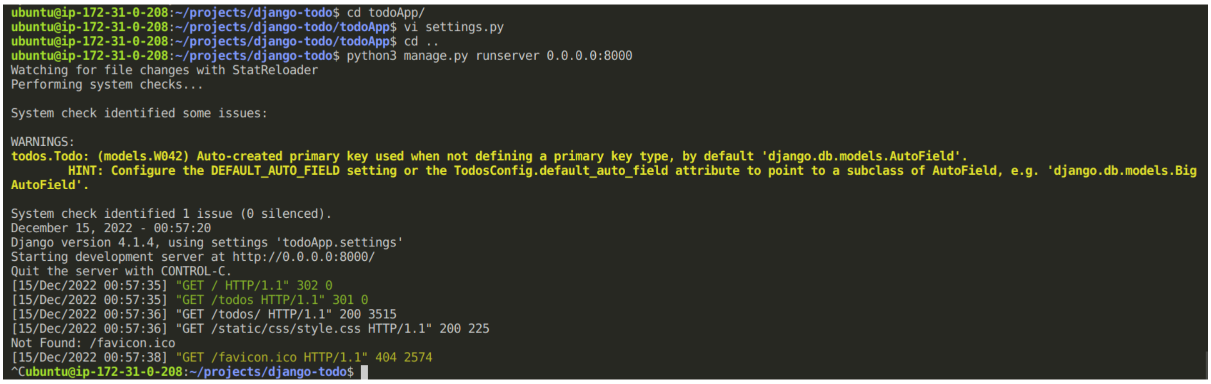
Request Method:	GET
Request URL:	http://52.69.172.164:8000/
Django Version:	4.1.4
Exception Type:	DisallowedHost
Exception Value:	`Invalid HTTP_HOST header: '52.69.172.164:8000'. You may need to add '52.69.172.164' to ALLOWED_HOSTS.`
Exception Location:	/home/ubuntu/.local/lib/python3.10/site-packages/django/http/request.py, line 148, in get_host
Raised during:	todoApp.views.index
Python Executable:	/usr/bin/python3
Python Version:	3.10.6
Python Path:	['/home/ubuntu/projects/django-todo', '/usr/lib/python310.zip', '/usr/lib/python3.10', '/usr/lib/python3.10/lib-dynload', '/home/ubuntu/.local/lib/python3.10/site-packages', '/usr/local/lib/python3.10/dist-packages', '/usr/lib/python3/dist-packages']
Server time:	Thu, 15 Dec 2022 00:52:15 +0530

So i have to set allowed host in settings.py page as ‘*’ as needed set to allow server host which presents in settings.py .(* = if ip changed then it will take auto)



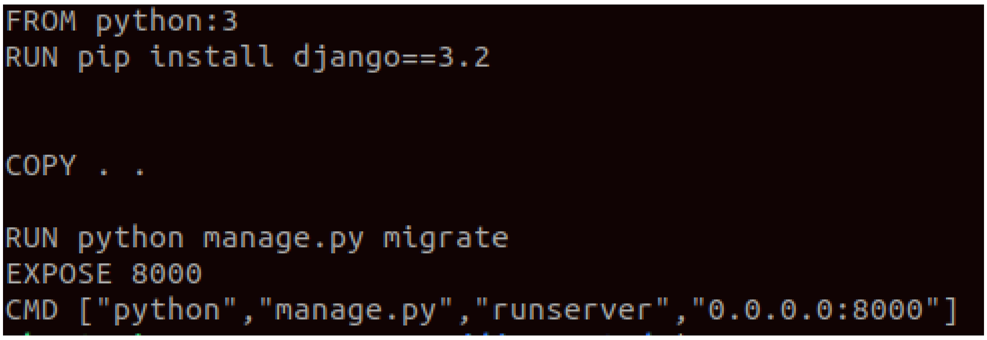
```
Django settings for todoApp project.

Generated by 'django-admin startproject' using Django 2.2.7.

For more information on this file, see
https://docs.djangoproject.com/en/2.2/topics/settings/

For the full list of settings and their values, see
https://docs.djangoproject.com/en/2.2/ref/settings/
"""

import os

# Build paths inside the project like this: os.path.join(BASE_DIR, ...)
BASE_DIR = os.path.dirname(os.path.dirname(os.path.abspath(__file__)))


# Quick-start development settings - unsuitable for production
# See https://docs.djangoproject.com/en/2.2/howto/deployment/checklist/

# SECURITY WARNING: keep the secret key used in production secret!
SECRET_KEY = '8)810zj@#^2xp=1=2rkozbv8#)gub6m1a^9qf&)d-9&x9*c2a_'

# SECURITY WARNING: don't run with debug turned on in production!
DEBUG = True

ALLOWED_HOSTS = ['*']


# Application definition

INSTALLED_APPS = [
    'todos.apps.TodosConfig',
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
"settings.py" 129L, 3315B
```

Now if I put : `python3 manage.py runserver 0.0.0.0:8000` so my django app is running successfully into aws ec2 server.



```
ubuntu@ip-172-31-0-208:~/projects/django-todo$ cd todoApp/
ubuntu@ip-172-31-0-208:~/projects/django-todo/todoApp$ vi settings.py
ubuntu@ip-172-31-0-208:~/projects/django-todo/todoApp$ cd ..
ubuntu@ip-172-31-0-208:~/projects/django-todo$ python3 manage.py runserver 0.0.0.0:8000
Watching for file changes with StatReloader
Performing system checks...

System check identified some issues:

WARNINGS:
todos.Todo: (models.W042) Auto-created primary key used when not defining a primary key type, by default 'django.db.models.AutoField'.
        HINT: Configure the DEFAULT_AUTO_FIELD setting or the TodosConfig.default_auto_field attribute to point to a subclass of AutoField, e.g. 'django.db.models.Big
AutoField'.

System check identified 1 issue (0 silenced).
December 15, 2022 - 00:57:20
Django version 4.1.4, using settings 'todoApp.settings'
Starting development server at http://0.0.0.0:8000/
Quit the server with CONTROL-C.
[15/Dec/2022 00:57:35] "GET / HTTP/1.1" 302 0
[15/Dec/2022 00:57:35] "GET /todos HTTP/1.1" 301 0
[15/Dec/2022 00:57:36] "GET /todos/ HTTP/1.1" 200 3515
[15/Dec/2022 00:57:36] "GET /static/css/style.css HTTP/1.1" 200 225
Not Found: /favicon.ico
[15/Dec/2022 00:57:38] "GET /favicon.ico HTTP/1.1" 404 2574
^Cubuntu@ip-172-31-0-208:~/projects/django-todo$
```

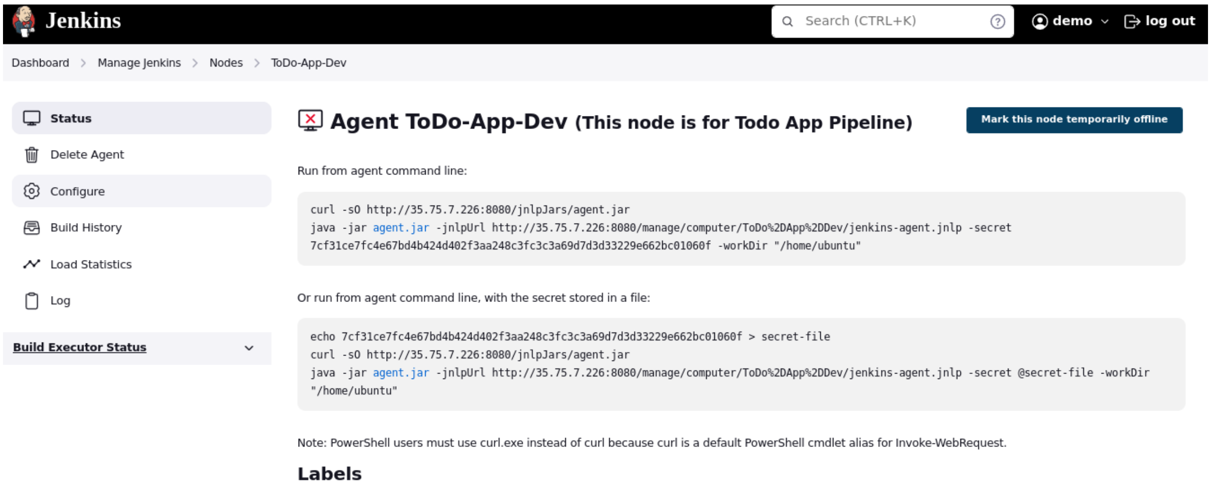
Next step is to build docker file where i have implemented these codes for CI :



```
FROM python:3
RUN pip install django==3.2


COPY . .

RUN python manage.py migrate
EXPOSE 8000
CMD ["python","manage.py","runserver","0.0.0.0:8000"]
```

Next step is to create a pipeline in Jenkins and push our todo app into the pipeline.

For that i have installed jenkins in AWS EC2 server and configured jenkins nodes along with job details .



Agent ToDo-App-Dev (This node is for Todo App Pipeline)

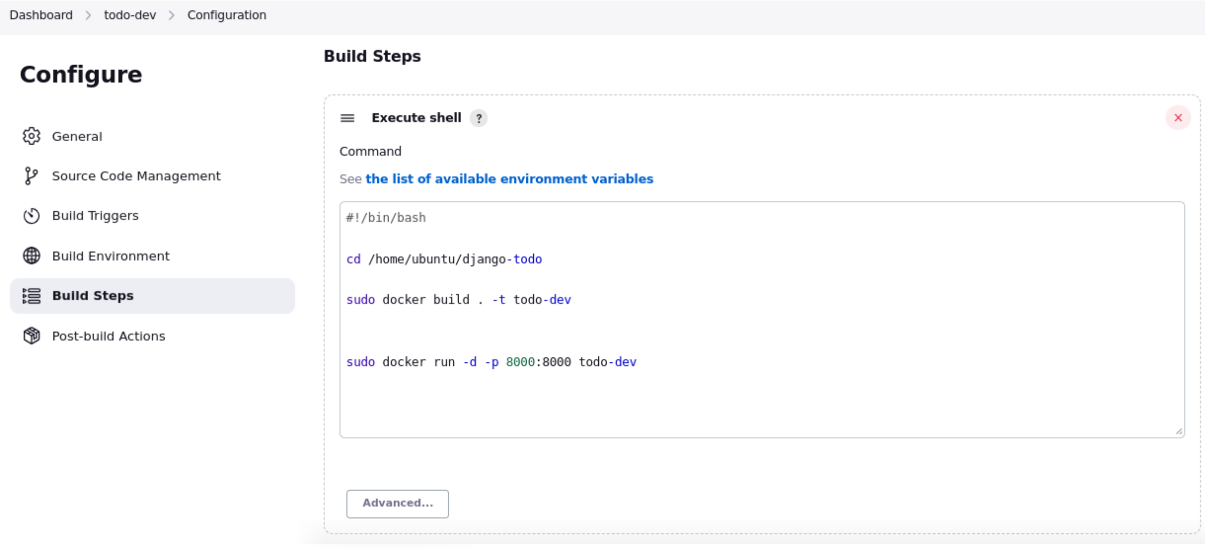
Run from agent command line:

```
curl -sO http://35.75.7.226:8080/jnlpJars/agent.jar
java -jar agent.jar -jnlpUrl http://35.75.7.226:8080/manage/computer/ToDo%2DApp%2DDev/jenkins-agent.jnlp -secret
7cf31ce7fc4e67bd4b424d402f3aa248c3fc3c3a69d7d3d33229e662bc01060f -workDir "/home/ubuntu"
```

Or run from agent command line, with the secret stored in a file:

```
echo 7cf31ce7fc4e67bd4b424d402f3aa248c3fc3c3a69d7d3d33229e662bc01060f > secret-file
curl -sO http://35.75.7.226:8080/jnlpJars/agent.jar
java -jar agent.jar -jnlpUrl http://35.75.7.226:8080/manage/computer/ToDo%2DApp%2DDev/jenkins-agent.jnlp -secret @secret-file -workDir
"/home/ubuntu"
```

Note: PowerShell users must use curl.exe instead of curl because curl is a default PowerShell cmdlet alias for Invoke-WebRequest.

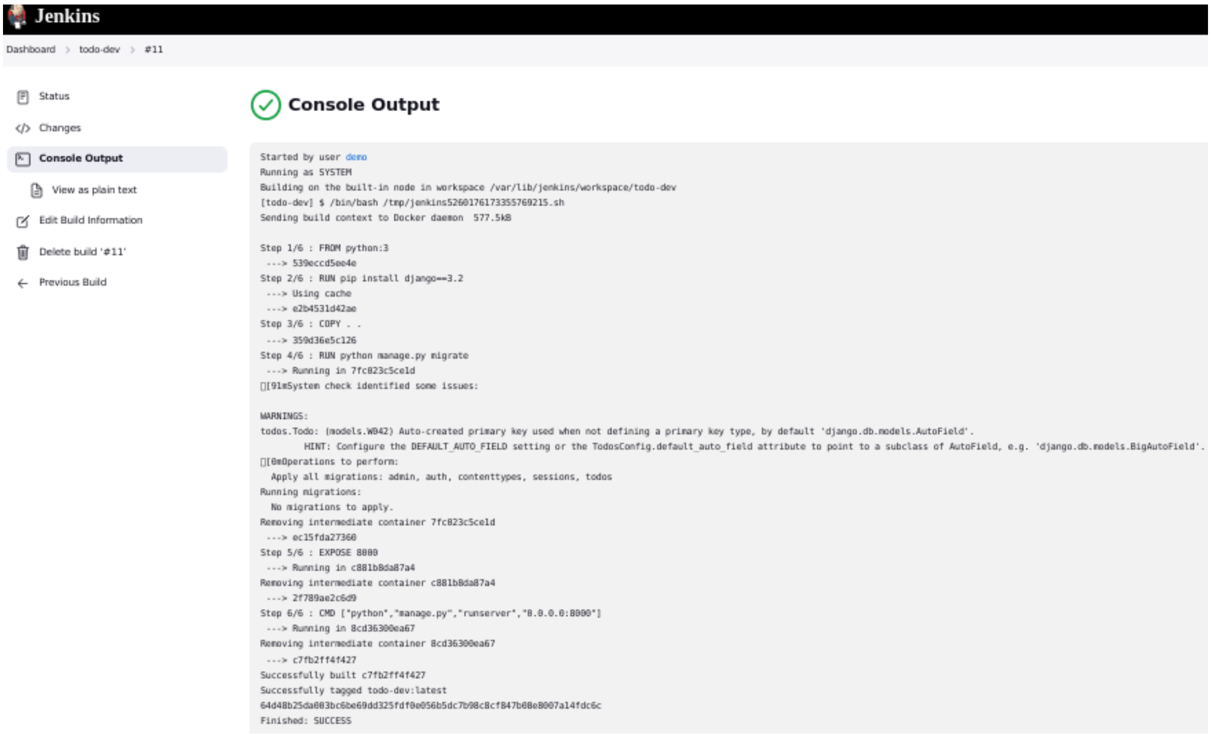


Command

See the list of available environment variables

```
#!/bin/bash

cd /home/ubuntu/django-todo

sudo docker build . -t todo-dev


sudo docker run -d -p 8000:8000 todo-dev
```

And after this my built was successful :



```
Started by user demo
Running as SYSTEM
Building on the built-in node in workspace /var/lib/jenkins/workspace/todo-dev
[todo-dev] $ /bin/bash /tmp/jenkins5260176173355769215.sh
Sending build context to Docker daemon  577.5kB

Step 1/6 : FROM python:3
 ---> 539eccd5ee4e
Step 2/6 : RUN pip install django==3.2
 ---> Using cache
 ---> e2b4531d42ae
Step 3/6 : COPY . .
 ---> 359d36e5c126
Step 4/6 : RUN python manage.py migrate
 ---> Running in 7fc823c5ce1d
[91mSystem check identified some issues:

WARNINGS:
todos.Todo: (models.W042) Auto-created primary key used when not defining a primary key type, by default 'django.db.models.AutoField'.
        HINT: Configure the DEFAULT_AUTO_FIELD setting or the TodosConfig.default_auto_field attribute to point to a subclass of AutoField, e.g. 'django.db.models.BigAutoField'.
[0mOperations to perform:
  Apply all migrations: admin, auth, contenttypes, sessions, todos
Running migrations:
  No migrations to apply.
Removing intermediate container 7fc823c5ce1d
 ---> ec15fda27360
Step 5/6 : EXPOSE 8000
 ---> Running in c881b8da87a4
Removing intermediate container c881b8da87a4
 ---> 2f789ae2c6d9
Step 6/6 : CMD ["python","manage.py","runserver","0.0.0.0:8000"]
 ---> Running in 8cd36300ea67
Removing intermediate container 8cd36300ea67
 ---> c7fb2ff4f427
Successfully built c7fb2ff4f427
Successfully tagged todo-dev:latest
64d48b25da003bc6be69dd325fdf0e056b5dc7b98c8cf847b08e8007a14fdc6c
Finished: SUCCESS
```

Now for checking purposes I have checked the url of ec2 with http method and 8000 port as i have defined 8000 port for my app in ec2 .

Small demo video : [deploying app into server]