docker version

```
igor@ubuntu-server:~$ docker version
Client: Docker Engine - Community
 Version:           20.10.13
 API version:       1.41
 Go version:        go1.16.15
 Git commit:        a224086
 Built:             Thu Mar 10 14:07:51 2022
 OS/Arch:           linux/amd64
 Context:           default
 Experimental:      true

Server: Docker Engine - Community
 Engine:
  Version:          20.10.13
  API version:      1.41 (minimum version 1.12)
  Go version:       go1.16.15
  Git commit:       906f57f
  Built:            Thu Mar 10 14:05:44 2022
  OS/Arch:          linux/amd64
  Experimental:     false
 containerd:
  Version:          1.5.10
  GitCommit:        2a1d4dbdb2a1030dc5b01e96fb110a9d9f150ecc
 runc:
  Version:          1.0.3
  GitCommit:        v1.0.3-0-gf46b6ba
 docker-init:
  Version:          0.19.0
  GitCommit:        de40ad0
igor@ubuntu-server:~$ ^C
igor@ubuntu-server:~$ 
```

vi Dockerfile

```
FROM python:3.9

RUN pip install Django==3.2

ADD . /app

EXPOSE 8000

ENTRYPOINT python /app/manage.py runserver 0.0.0.0:8000
~
:wq
```

docker build -t my-first-image .

```
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ docker build -t my-first-image .
Sending build context to Docker daemon  11.26kB
Step 1/5 : FROM python:3.9
3.9: Pulling from library/python
5492f66d2700: Pull complete
540ff8c0841d: Pull complete
a0bf850a0df0: Pull complete
d751dc38ae51: Pull complete
9720a112e886: Pull complete
f97b81fbdbd9: Pull complete
8f5bb67e9e73: Pull complete
0ae6cb39a048: Pull complete
f8c61b949cf8: Pull complete
Digest: sha256:279e513ea3784cb0effa3c9ee0aa30d13bf49e8297038e1f718d5e97e31e26cd
Status: Downloaded newer image for python:3.9
 ---> 47abcf270bef
Step 2/5 : RUN pip install Django==3.2
 ---> Running in c2c52002dd9f
Collecting Django==3.2
  Downloading Django-3.2-py3-none-any.whl (7.9 MB)
                                         7.9/7.9 MB 19.9 MB/s eta 0:00:00
Collecting pytz
  Downloading pytz-2022.1-py2.py3-none-any.whl (503 kB)
                                         503.5/503.5 KB 25.7 MB/s eta 0:00:00
Collecting sqlparse>=0.2.2
  Downloading sqlparse-0.4.2-py3-none-any.whl (42 kB)
                                         42.3/42.3 KB 3.6 MB/s eta 0:00:00
Collecting asgiref<4,>=3.3.2
  Downloading asgiref-3.5.0-py3-none-any.whl (22 kB)
Installing collected packages: pytz, sqlparse, asgiref, Django
Successfully installed Django-3.2 asgiref-3.5.0 pytz-2022.1 sqlparse-0.4.2
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is recommended
to use a virtual environment instead: https://pip.pypa.io/warnings/venv
Removing intermediate container c2c52002dd9f
 ---> 5a5d0f1551a7
Step 3/5 : ADD . /app
 ---> 2b21c3a7b948
Step 4/5 : EXPOSE 8000
 ---> Running in f7b7e21de61f
Removing intermediate container f7b7e21de61f
 ---> cf0e6248cdc0
Step 5/5 : ENTRYPOINT python /app/manage.py runserver 0.0.0.0:8000
 ---> Running in 9b54423cbb98
Removing intermediate container 9b54423cbb98
 ---> 7195fe0eefe0
Successfully built 7195fe0eefe0
Successfully tagged my-first-image:latest
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$
```

```
docker image ls
```

```
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ docker image ls
REPOSITORY                        TAG       IMAGE ID       CREATED         SIZE
my-first-image                    latest    7195fe0eefe0   2 minutes ago   956MB
python                            3.9       47abcf270bef   5 days ago      914MB
hello-world                       latest    feb5d9fea6a5   6 months ago    13.3kB
gcr.io/k8s-minikube/kicbase       v0.0.27   9fa1cc16ad6d   6 months ago    1.08GB
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$
```

```
docker run -d -p 80:8000 my-first-image
```

```
docker ps
```

```
docker logs 6bd5872d22a1
```

```
hello-world                        latest      feb5d9fea6a5    6 months ago     13.3kB
gcr.io/k8s-minikube/kicbase        v0.0.27     9fa1cc16ad6d    6 months ago     1.08GB
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ docker run -d -p 80:8000 my-first-image
6bd5872d22a1537a8775c2ef90488ae2e1f29311d0f647a651056ae3c1a25d5d
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ docker ps
CONTAINER ID   IMAGE            COMMAND               CREATED              STATUS             PORTS                                         NAMES
6bd5872d22a1   my-first-image   "/bin/sh -c 'python …"   About a minute ago   Up About a minute   0.0.0.0:80->8000/tcp, :::80->8000/tcp    intelligent_elion
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ docker logs 6bd5872d22a1
Watching for file changes with StatReloader
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ 
```

curl 0.0.0.0:80

```
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ curl 0.0.0.0:80
<!DOCTYPE html>
<html lang="en">
<head>
  <meta http-equiv="content-type" content="text/html; charset=utf-8">
  <meta name="robots" content="NONE,NOARCHIVE">
  <title>DisallowedHost
          at /</title>
  <style type="text/css">
    html * { padding:0; margin:0; }
    body * { padding:10px 20px; }
    body * * { padding:0; }
    body { font:small sans-serif; background-color:#fff; color:#000; }
    body>div { border-bottom:1px solid #ddd; }
    h1 { font-weight:normal; }
    h2 { margin-bottom:.8em; }
    h3 { margin:1em 0 .5em 0; }
    h4 { margin:0 0 .5em 0; font-weight: normal; }
    code, pre { font-size: 100%; white-space: pre-wrap; }
    table { border:1px solid #ccc; border-collapse: collapse; width:100%; background:white; }
    tbody td, tbody th { vertical-align:top; padding:2px 3px; }
    thead th {
      padding:1px 6px 1px 3px; background:#fefefe; text-align:left;
      font-weight:normal; font-size:11px; border:1px solid #ddd;
    }
    tbody th { width:12em; text-align:right; color:#666; padding-right:.5em; }
    table.vars { margin:5px 0 2px 40px; }
    table.vars td, table.req td { font-family:monospace; }
    table td.code { width:100%; }
    table td.code pre { overflow:hidden; }
    table.source th { color:#666; }
    table.source td { font-family:monospace; white-space:pre; border-bottom:1px solid #eee; }
    ul.traceback { list-style-type:none; color: #222; }
    ul.traceback li.frame { padding-bottom:1em; color:#4f4f4f; }
    ul.traceback li.user { background-color:#e0e0e0; color:#000 }
    div.context { padding:10px 0; overflow:hidden; }
    div.context ol { padding-left:30px; margin:0 10px; list-style-position: inside; }
    div.context ol li { font-family:monospace; white-space:pre; color:#777; cursor:pointer; padding-left: 2px; }
    div.context ol li pre { display:inline; }
    div.context ol.context-line li { color:#464646; background-color:#dfdfdf; padding: 3px 2px; }
    div.context ol.context-line li span { position:absolute; right:32px; }
    .user div.context ol.context-line li { background-color:#bbb; color:#000; }
    .user div.context ol li { color:#666; }
    div.commands { margin-left: 40px; }
    div.commands a { color:#555; text-decoration:none; }
    .user div.commands a { color: black; }
    #summary { background: #ffc; }
    #summary h2 { font-weight: normal; color: #666; }
    #explanation { background:#eee; }
    #template, #template-not-exist { background:#f6f6f6; }
    #template-not-exist ul { margin: 0 0 10px 20px; }
    #template-not-exist .postmortem-section { margin-bottom: 3px; }
    #unicode-hint { background:#eee; }
    #traceback { background:#eee; }
    #requestinfo { background:#f6f6f6; padding-left:120px; }
    #summary table { border:none; background:transparent; }
    #requestinfo h2, #requestinfo h3 { position:relative; margin-left:-100px; }
    #requestinfo h3 { margin-bottom:-1em; }
    .error { background: #ffc; }
    .specific { color:#cc3300; font-weight:bold; }
    h2 span.commands { font-size:.7em; font-weight:normal; }
    span.commands a:link {color:#5E5694;}
    pre.exception_value { font-family: sans-serif; color: #575757; font-size: 1.5em; margin: 10px 0 10px 0; }
    .append-bottom { margin-bottom: 10px; }
    .fname { user-select: all; }
  </style>

  <script>
    function hideAll(elems) {
      for (var e = 0; e < elems.length; e++) {
        elems[e].style.display = 'none';
      }
    }
    window.onload = function() {
      hideAll(document.querySelectorAll('table.vars'));
      hideAll(document.querySelectorAll('ol.pre-context'));
      hideAll(document.querySelectorAll('ol.post-context'));
      hideAll(document.querySelectorAll('div.pastebin'));
    }
    function toggle() {
      for (var i = 0; i < arguments.length; i++) {
        var e = document.getElementById(arguments[i]);
        if (e) {
          e.style.display = e.style.display == 'none' ? 'block': 'none';
        }
      }
      return false;
    }
    function varToggle(link, id) {
      toggle('v' + id);
      var s = link.getElementsByTagName('span')[0];
      var uarr = String.fromCharCode(0x25b6);
      var darr = String.fromCharCode(0x25bc);
      s.textContent = s.textContent == uarr ? darr : uarr;
      return false;
    }
    function switchPastebinFriendly(link) {
      s1 = "Switch to copy-and-paste view";
      s2 = "Switch back to interactive view";
      link.textContent = link.textContent.trim() == s1 ? s2: s1;
      toggle('browserTraceback', 'pastebinTraceback');
      return false;
    }
  </script>

</head>
<body>
<div id="summary">
  <h1>DisallowedHost
          at /</h1>
  <pre class="exception_value">Invalid HTTP_HOST header: &#x27;0.0.0.0&#x27;. You may need to add &#x27;0.0.0.0&#x27; to ALLOWED_HOSTS.</pre>
  <table class="meta">

    <tr>
      <th>Request Method:</th>
      <td>GET</td>
    </tr>
    <tr>
      <th>Request URL:</th>
```

```
docker logs 6bd5872d22a1
```

Файл Правка Вид Терминал Вкладки Справка

```
</body>
</html>
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ docker logs 6bd5872d22a1
Watching for file changes with StatReloader
Invalid HTTP_HOST header: '0.0.0.0'. You may need to add '0.0.0.0' to ALLOWED_HOSTS.
Bad Request: /
[24/Mar/2022 15:32:35] "GET / HTTP/1.1" 400 58203
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$
```

```
docker exec -t -i 6bd5872d22a1 bash

ls

cd app

ls

cd myapp

ls
```

Файл Правка Вид Терминал Вкладки Справка

```
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ docker exec -t -i 6bd5872d22a1 bash
root@6bd5872d22a1:/# ls
app  bin  boot  dev  etc  home  lib  lib64  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var
root@6bd5872d22a1:/# cd app
root@6bd5872d22a1:/app# ls
Dockerfile  db.sqlite3  manage.py  myapp
root@6bd5872d22a1:/app# cd myapp
root@6bd5872d22a1:/app/myapp# ls
__init__.py  __pycache__  settings.py  urls.py  wsgi.py
root@6bd5872d22a1:/app/myapp#
```

```
vi settings.py

nano settings.py

exit
```

Файл Правка Вид Терминал Вкладки Справка

```
root@6bd5872d22a1:/app/myapp# vi settings.py
bash: vi: command not found
root@6bd5872d22a1:/app/myapp# nano settings.py
bash: nano: command not found
root@6bd5872d22a1:/app/myapp# exit
exit
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$
```

```
ls

cd myapp

ls
```

Файл   Правка   Вид   Терминал   Вкладки   Справка

```
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ ls
Dockerfile  manage.py  myapp
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ cd myapp
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django/myapp$ ls
__init__.py  settings.py  urls.py  wsgi.py
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django/myapp$ vi settings.py
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django/myapp$ █
```

```
vi settings.py

ALLOWED_HOSTS = [0.0.0.0]
```

Файл   Правка   Вид   Терминал   Вкладки   Справка

```
"""
Django settings for myapp project.

Generated by 'django-admin startproject' using Django 2.1.5.

For more information on this file, see
https://docs.djangoproject.com/en/2.1/topics/settings/

For the full list of settings and their values, see
https://docs.djangoproject.com/en/2.1/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/2.1/howto/deployment/checklist/

# SECURITY WARNING: keep the secret key used in production secret!
SECRET_KEY = 'h!i)j*@*1@b*k!mewnj$*oe5(-841e(7h_)34_$3jn5nr-!23q'

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

ALLOWED_HOSTS = [0.0.0.0]


# Application definition

INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
]

MIDDLEWARE = [
    'django.middleware.security.SecurityMiddleware',
    'django.contrib.sessions.middleware.SessionMiddleware',
    'django.middleware.common.CommonMiddleware',
    'django.middleware.csrf.CsrfViewMiddleware',
    'django.contrib.auth.middleware.AuthenticationMiddleware',
    'django.contrib.messages.middleware.MessageMiddleware',
    'django.middleware.clickjacking.XFrameOptionsMiddleware',
]

ROOT_URLCONF = 'myapp.urls'

TEMPLATES = [
    {
        'BACKEND': 'django.template.backends.django.DjangoTemplates',
        'DIRS': [],
        'APP_DIRS': True,
        'OPTIONS': {
            'context_processors': [
                'django.template.context_processors.debug',
                'django.template.context_processors.request',
                'django.contrib.auth.context_processors.auth',
                'django.contrib.messages.context_processors.messages',
```

`-- ВСТАВКА --`   28,25   Наверху

```
cd ..
```

```
vi Dockerfile

RUN apt update
```

```
FROM python:3.9

RUN pip install Django==3.2

ADD . /app

EXPOSE 8000

RUN apt update

ENTRYPOINT python /app/manage.py runserver 0.0.0.0:8000
~
:wq
```

Файл   Правка   Вид   Терминал   Вкладки   Справка

```
docker container stop 6bd5872d22a1

docker container rm 6bd5872d22a1

docker ps -a
```

Файл   Правка   Вид   Терминал   Вкладки   Справка

```
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ docker container stop 6bd5872d22a1
6bd5872d22a1
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ docker container rm 6bd5872d22a1
6bd5872d22a1
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ docker ps -a
CONTAINER ID   IMAGE                             COMMAND                 CREATED        STATUS                    PORTS       NAMES
3041bc99fb37   gcr.io/k8s-minikube/kicbase:v0.0.27   "/usr/local/bin/entr…"  4 months ago   Exited (130) 4 months ago             minikube
0a4bdf412265   hello-world                       "/hello"                4 months ago   Exited (0) 4 months ago               happy_brahmagupta
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$
```

```
docker build -t my-first-image .
```

```
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ docker build -t my-first-image .
Sending build context to Docker daemon  11.26kB
Step 1/6 : FROM python:3.9
 ---> 47abcf270bef
Step 2/6 : RUN pip install Django==3.2
 ---> Using cache
 ---> 5a5d0f1551a7
Step 3/6 : ADD . /app
 ---> 2a86e9d27540
Step 4/6 : EXPOSE 8000
 ---> Running in d7a3380f67e5
Removing intermediate container d7a3380f67e5
 ---> ced858f882cf
Step 5/6 : RUN apt update
 ---> Running in 75b3aa1c917d

WARNING: apt does not have a stable CLI interface. Use with caution in scripts.

Get:1 http://deb.debian.org/debian bullseye InRelease [116 kB]
Get:2 http://security.debian.org/debian-security bullseye-security InRelease [44.1 kB]
Get:3 http://deb.debian.org/debian bullseye-updates InRelease [39.4 kB]
Get:4 http://security.debian.org/debian-security bullseye-security/main amd64 Packages [122 kB]
Get:5 http://deb.debian.org/debian bullseye/main amd64 Packages [8183 kB]
Get:6 http://deb.debian.org/debian bullseye-updates/main amd64 Packages [2596 B]
Fetched 8507 kB in 2s (5159 kB/s)
Reading package lists...
Building dependency tree...
Reading state information...
All packages are up to date.
Removing intermediate container 75b3aa1c917d
 ---> cf4235f31b5b
Step 6/6 : ENTRYPOINT python /app/manage.py runserver 0.0.0.0:8000
 ---> Running in 1afbaabab1c1
Removing intermediate container 1afbaabab1c1
 ---> 26e85e62462f
Successfully built 26e85e62462f
Successfully tagged my-first-image:latest
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$
```

docker image ls

```
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ docker image ls
REPOSITORY                   TAG       IMAGE ID       CREATED          SIZE
my-first-image               latest    26e85e62462f   About a minute ago   974MB
<none>                       <none>    7195fe0eefe0   54 minutes ago   956MB
python                       3.9       47abcf270bef   5 days ago       914MB
hello-world                  latest    feb5d9fea6a5   6 months ago     13.3kB
gcr.io/k8s-minikube/kicbase  v0.0.27   9fa1cc16ad6d   6 months ago     1.08GB
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$
```

docker build -t my-first-image:latest .

```
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ docker build -t my-first-image:latest .
Sending build context to Docker daemon  11.26kB
Step 1/6 : FROM python:3.9
 ---> 47abcf270bef
Step 2/6 : RUN pip install Django==3.2
 ---> Using cache
 ---> 5a5d0f1551a7
Step 3/6 : ADD . /app
 ---> Using cache
 ---> 2a86e9d27540
Step 4/6 : EXPOSE 8000
 ---> Using cache
 ---> ced858f882cf
Step 5/6 : RUN apt update
 ---> Using cache
 ---> cf4235f31b5b
Step 6/6 : ENTRYPOINT python /app/manage.py runserver 0.0.0.0:8000
 ---> Using cache
 ---> 26e85e62462f
Successfully built 26e85e62462f
Successfully tagged my-first-image:latest
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$
```

docker image ls

```
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ docker image ls
REPOSITORY                  TAG       IMAGE ID       CREATED         SIZE
my-first-image              latest    26e85e62462f   5 minutes ago   974MB
<none>                      <none>    7195fe0eefe0   58 minutes ago  956MB
python                      3.9       47abcf270bef   6 days ago      914MB
hello-world                 latest    feb5d9fea6a5   6 months ago    13.3kB
gcr.io/k8s-minikube/kicbase v0.0.27   9fa1cc16ad6d   6 months ago    1.08GB
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$
```

docker tag 7195fe0eefe0 my-first-image:v1

docker image ls

docker run -d -p 80:8000 my-first-image

```
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ docker tag 7195fe0eefe0 my-first-image:v1
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ docker image ls
REPOSITORY                  TAG       IMAGE ID       CREATED            SIZE
my-first-image              latest    26e85e62462f   8 minutes ago      974MB
my-first-image              v1        7195fe0eefe0   About an hour ago  956MB
python                      3.9       47abcf270bef   6 days ago         914MB
hello-world                 latest    feb5d9fea6a5   6 months ago       13.3kB
gcr.io/k8s-minikube/kicbase v0.0.27   9fa1cc16ad6d   6 months ago       1.08GB
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ docker run -d -p 80:8000 my-first-image
56a6cf0b90ecced003e7233ed301e3bec03135ff6f3f449e43ca34f2341493fa
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$
```

docker run -d -p 80:8000 my-first-image

docker ps

docker run -d -p 80:8000 26e85e62462f

docker ps

docker run -d -p 80:8000 7195fe0eefe0

docker ps

```
hello-world                 latest    feb5d9fea6a5   6 months ago       13.3kB
gcr.io/k8s-minikube/kicbase v0.0.27   9fa1cc16ad6d   6 months ago       1.08GB
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ docker run -d -p 80:8000 my-first-image
56a6cf0b90ecced003e7233ed301e3bec03135ff6f3f449e43ca34f2341493fa
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ docker run -d -p 80:8000 26e85e62462f
e273cc18a91a59918a768263df9c504d99d9e6bc82583717941e5eb4c0fafd9b
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ docker run -d -p 80:8000 7195fe0eefe0
8ee1bb383397718e5affbe05f5cfb1d020064ef52da2cd290cba3349d62da32a2
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ docker ps
CONTAINER ID   IMAGE          COMMAND            CREATED         STATUS         PORTS                                       NAMES
8ee1bb383977   7195fe0eefe0   "/bin/sh -c 'python …"  15 seconds ago  Up 15 seconds  0.0.0.0:80->8000/tcp, :::80->8000/tcp       wonderful_chatelet
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$
```

curl 0.0.0.0:80

Файл   Правка   Вид   Терминал   Вкладки   Справка

```
        <tr>
          <td>USE_TZ</td>
          <td class="code"><pre>True</pre></td>
        </tr>

        <tr>
          <td>USE_X_FORWARDED_HOST</td>
          <td class="code"><pre>False</pre></td>
        </tr>

        <tr>
          <td>USE_X_FORWARDED_PORT</td>
          <td class="code"><pre>False</pre></td>
        </tr>

        <tr>
          <td>WSGI_APPLICATION</td>
          <td class="code"><pre>&#x27;myapp.wsgi.application&#x27;</pre></td>
        </tr>

        <tr>
          <td>X_FRAME_OPTIONS</td>
          <td class="code"><pre>&#x27;DENY&#x27;</pre></td>
        </tr>

        <tr>
          <td>YEAR_MONTH_FORMAT</td>
          <td class="code"><pre>&#x27;F Y&#x27;</pre></td>
        </tr>

      </tbody>
    </table>
  </div>

  <div id="explanation">
    <p>
      You're seeing this error because you have <code>DEBUG = True</code> in your
      Django settings file. Change that to <code>False</code>, and Django will
      display a standard page generated by the handler for this status code.
    </p>
  </div>

</body>
</html>
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$
```

```
docker ps

docker logs 8ee1bb383977
```

Файл   Правка   Вид   Терминал   Вкладки   Справка

```
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ docker ps
CONTAINER ID   IMAGE          COMMAND            CREATED        STATUS         PORTS                                        NAMES
8ee1bb383977   7195fe0eefe0   "/bin/sh -c 'python …"   5 minutes ago   Up 5 minutes   0.0.0.0:80->8000/tcp, :::80->8000/tcp   wonderful_chatelet
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ docker logs 8ee1bb383977
Watching for file changes with StatReloader
Invalid HTTP_HOST header: '0.0.0.0'. You may need to add '0.0.0.0' to ALLOWED_HOSTS.
Bad Request: /
[24/Mar/2022 16:29:42] "GET / HTTP/1.1" 400 58203
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$
```

```
docker image ls

docker image rm 7195fe0eefe0

docker image rm feb5d9fea6a5

docker image rm 26e85e62462f

docker image rm 9fa1cc16ad6d
```

```
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ docker image ls
REPOSITORY                  TAG       IMAGE ID       CREATED            SIZE
my-first-image              latest    26e85e62462f   22 minutes ago     974MB
my-first-image              v1        7195fe0eefe0   About an hour ago  956MB
python                      3.9       47abcf270bef   6 days ago         914MB
hello-world                 latest    feb5d9fea6a5   6 months ago       13.3kB
gcr.io/k8s-minikube/kicbase v0.0.27   9fa1cc16ad6d   6 months ago       1.08GB
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ docker image rm 7195fe0eefe0
Error response from daemon: conflict: unable to delete 7195fe0eefe0 (cannot be forced) - image is being used by running container 8ee1bb383977
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ docker image rm feb5d9fea6a5
Error response from daemon: conflict: unable to delete feb5d9fea6a5 (must be forced) - image is being used by stopped container 0a4bdf412265
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ docker image rm 26e85e62462f
Error response from daemon: conflict: unable to delete 26e85e62462f (must be forced) - image is being used by stopped container 56a6cf0b90ec
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ docker image rm 9fa1cc16ad6d
Error response from daemon: conflict: unable to delete 9fa1cc16ad6d (must be forced) - image is being used by stopped container 3041bc99fb37
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$
```

```
docker pull redhat/ubi8-minimal
```

```
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ docker pull redhat/ubi8-minimal
Using default tag: latest
latest: Pulling from redhat/ubi8-minimal
510abfcdf6bc: Pull complete
4a3604715398: Pull complete
Digest: sha256:574f201d7ed185a9932c91cef5d397f5298dff9df08bc2ebb266c6d1e6284cd1
Status: Downloaded newer image for redhat/ubi8-minimal:latest
docker.io/redhat/ubi8-minimal:latest
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$
```

```
docker image ls
```

```
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$ docker image ls
REPOSITORY                  TAG       IMAGE ID       CREATED            SIZE
my-first-image              latest    26e85e62462f   30 minutes ago     974MB
my-first-image              v1        7195fe0eefe0   About an hour ago  956MB
python                      3.9       47abcf270bef   6 days ago         914MB
redhat/ubi8-minimal         latest    281e62f51d47   2 weeks ago        103MB
hello-world                 latest    feb5d9fea6a5   6 months ago       13.3kB
gcr.io/k8s-minikube/kicbase v0.0.27   9fa1cc16ad6d   6 months ago       1.08GB
igor@ubuntu-server:~/geekbrains/practice/2.docker/python-django$
```