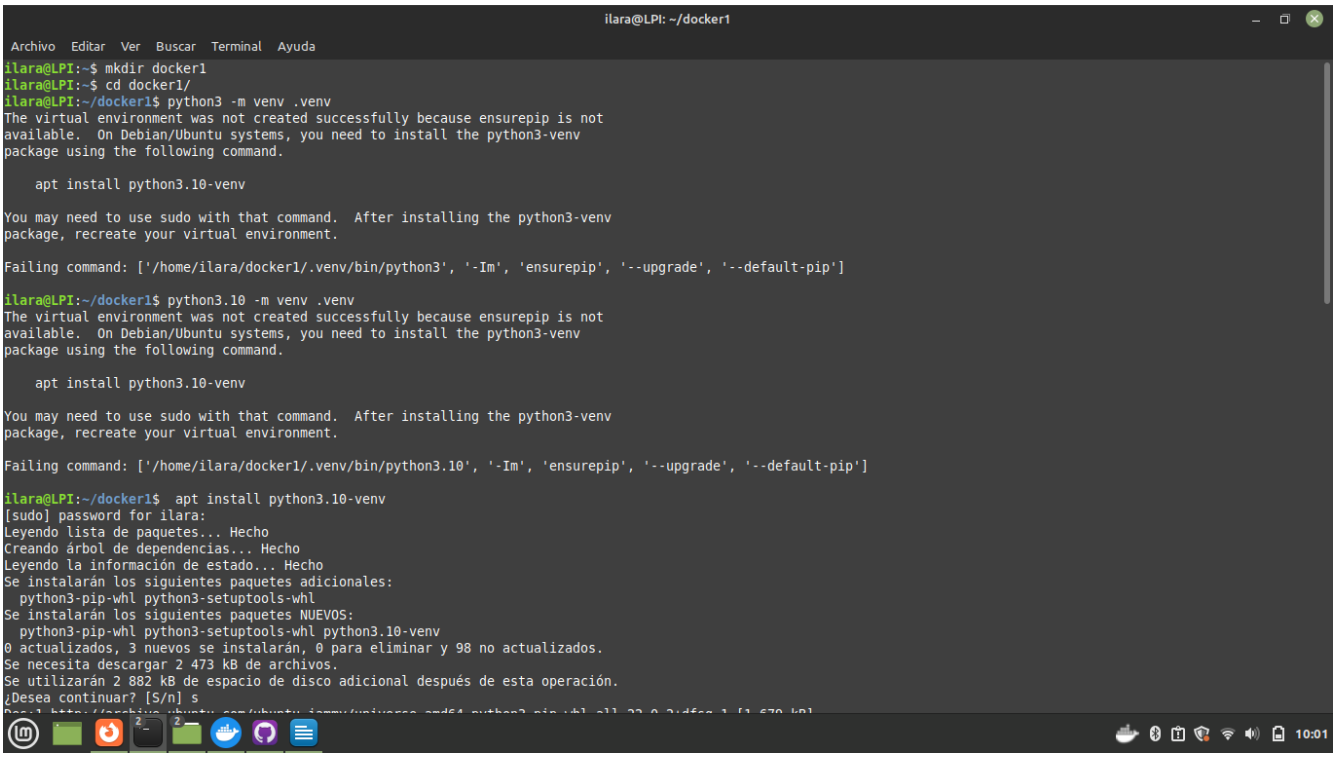
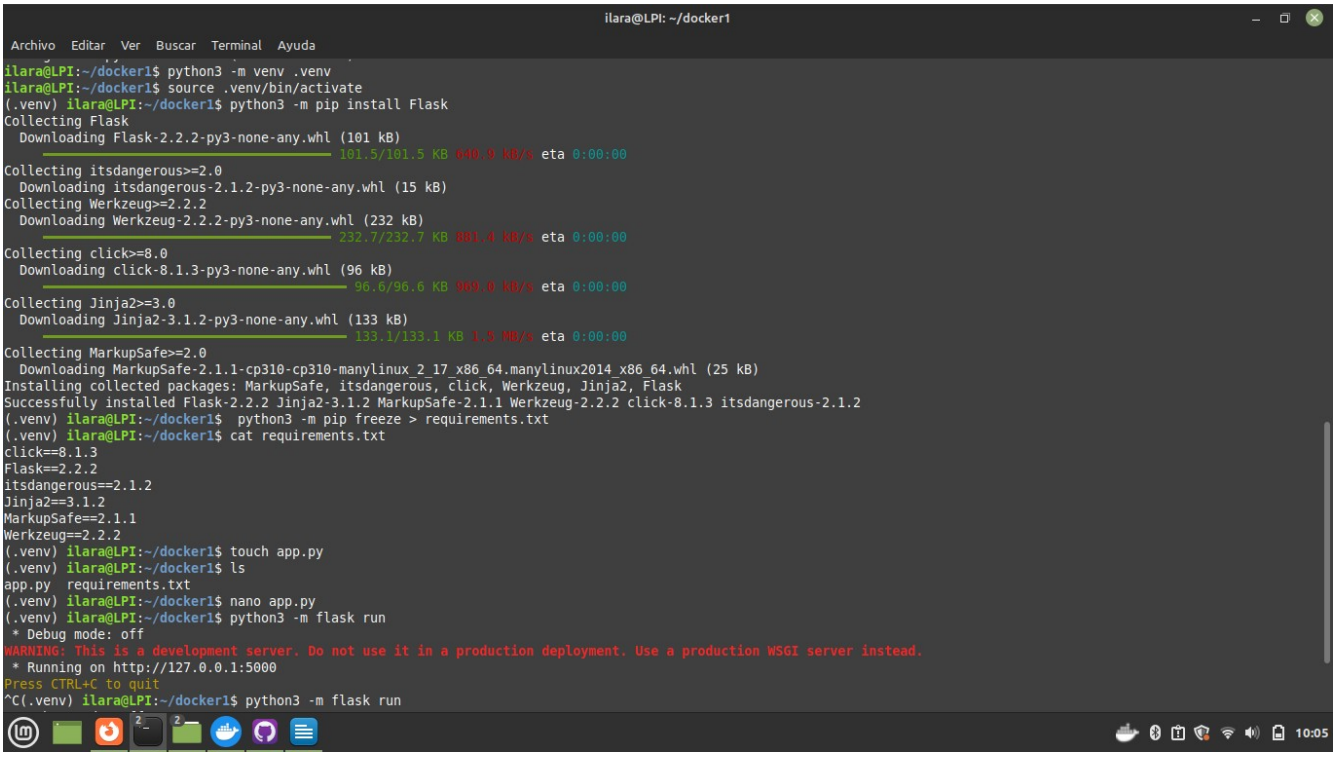
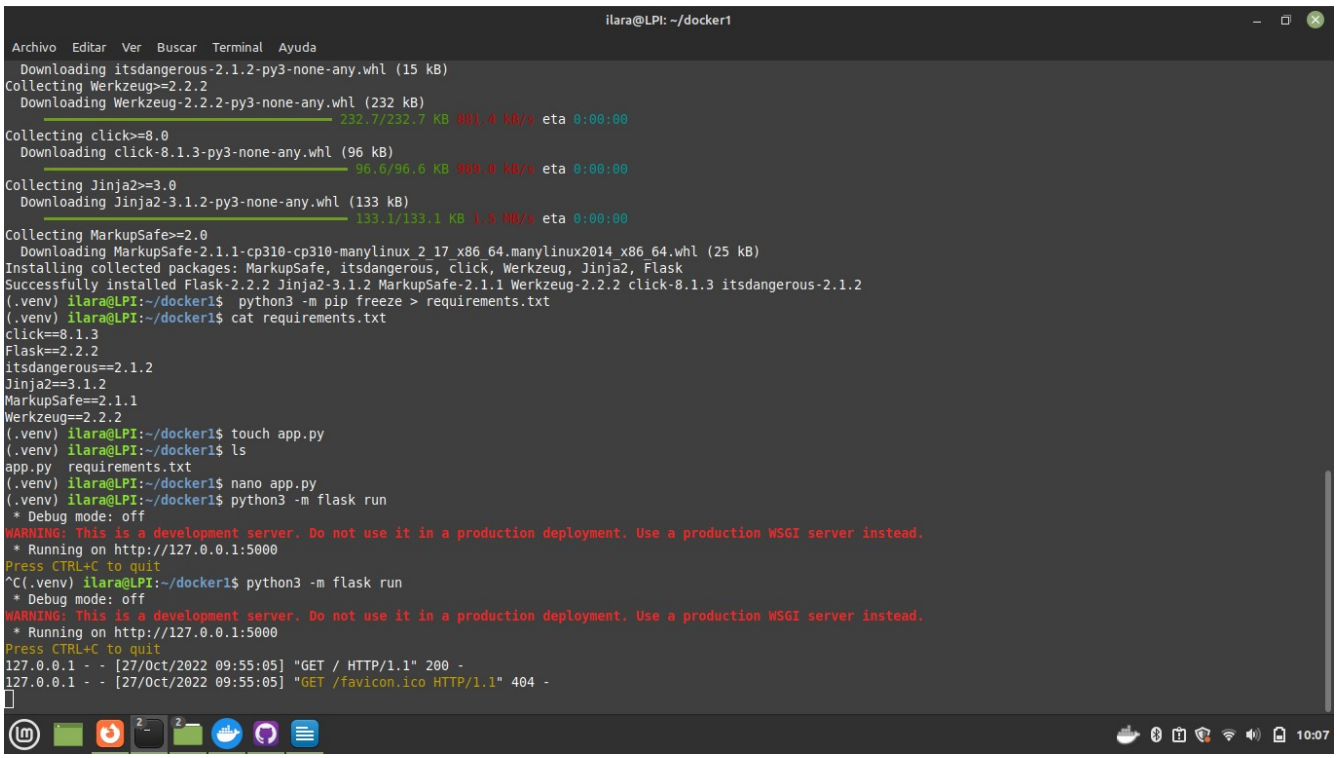
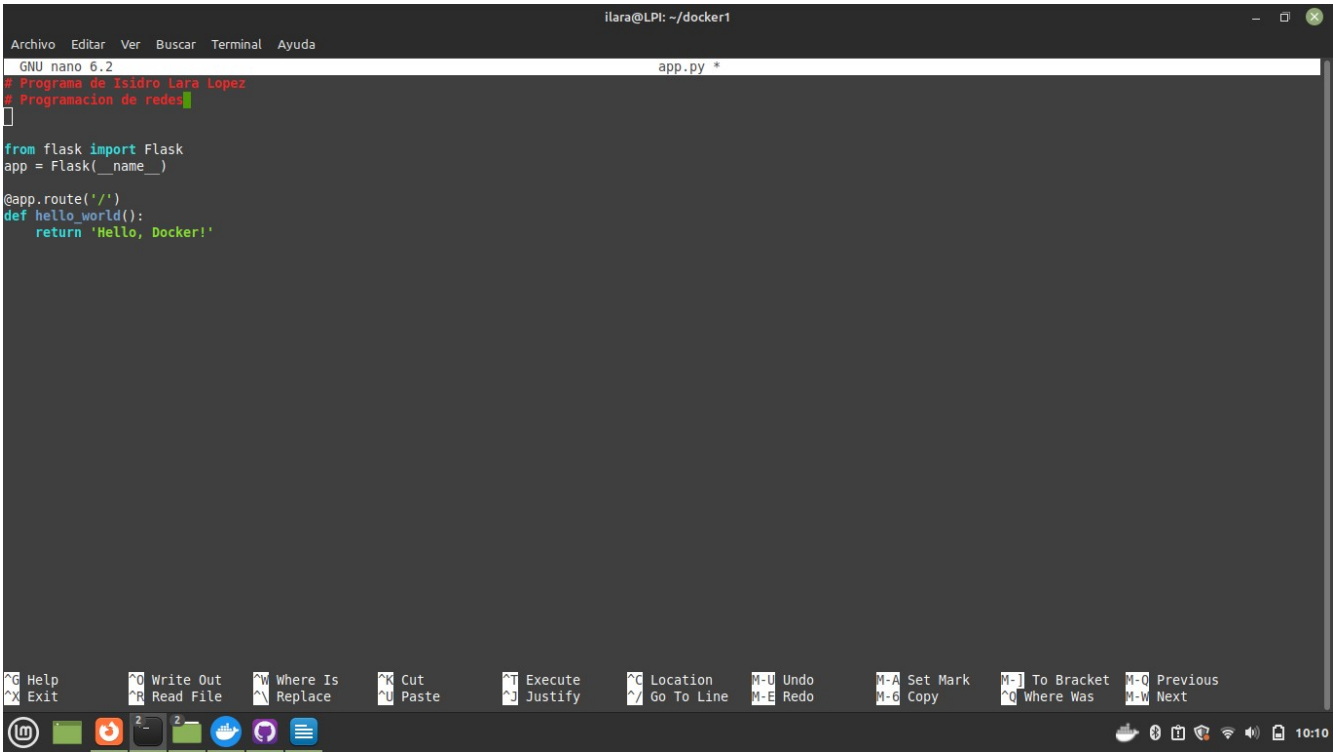
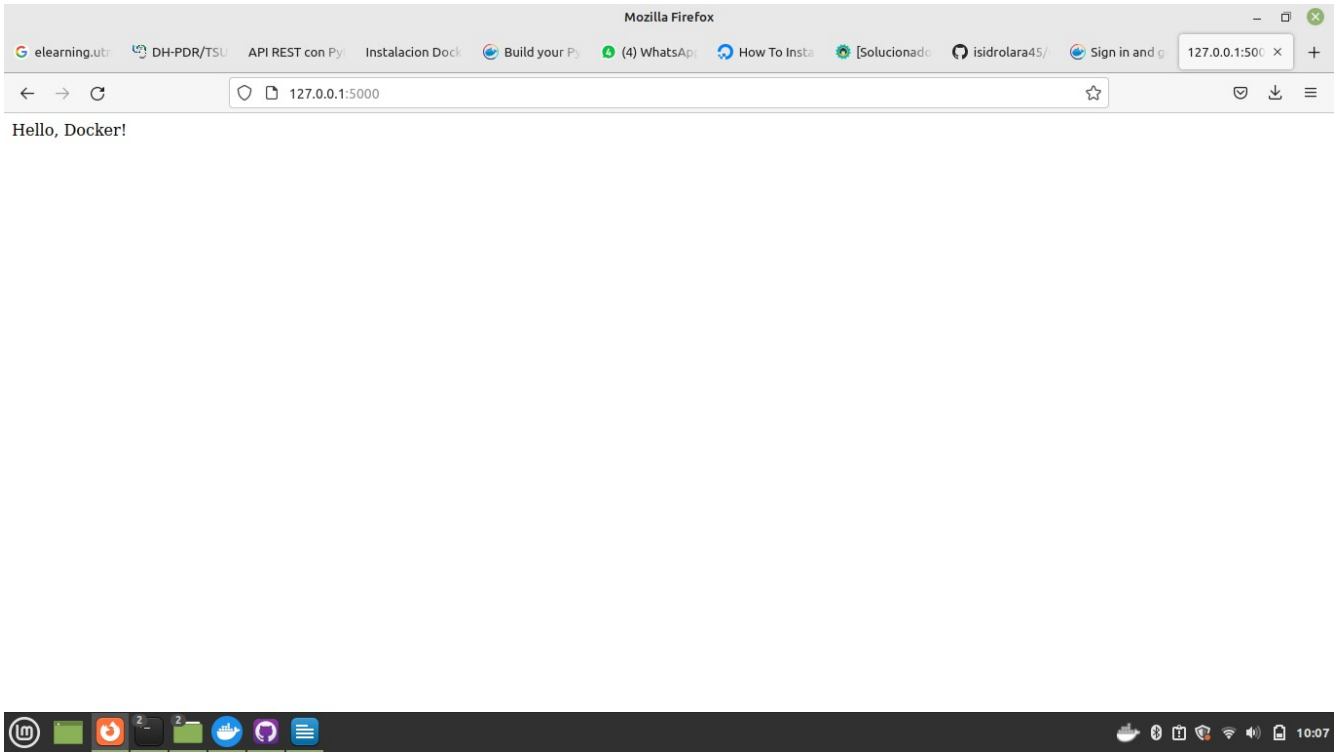
Docker and Python






ilara@LPI: ~/docker1
Archivo Editar Ver Buscar Terminal Ayuda
Downloading itsdangerous-2.1.2-py3-none-any.whl (15 kB)
Collecting Werkzeug>=2.2.2
Downloading Werkzeug-2.2.2-py3-none-any.whl (232 kB)
232.7/232.7 KB 881.4 kB/s eta 0:00:00
Collecting click>=8.0
Downloading click-8.1.3-py3-none-any.whl (96 kB)
96.6/96.6 KB 969.0 kB/s eta 0:00:00
Collecting Jinja2>=3.0
Downloading Jinja2-3.1.2-py3-none-any.whl (133 kB)
133.1/133.1 KB 1.5 MB/s eta 0:00:00
Collecting MarkupSafe>=2.0
Downloading MarkupSafe-2.1.1-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (25 kB)
Installing collected packages: MarkupSafe, itsdangerous, click, Werkzeug, Jinja2, Flask
Successfully installed Flask-2.2.2 Jinja2-3.1.2 MarkupSafe-2.1.1 Werkzeug-2.2.2 click-8.1.3 itsdangerous-2.1.2
(.venv) ilara@LPI:~/docker1$ python3 -m pip freeze > requirements.txt
(.venv) ilara@LPI:~/docker1$ cat requirements.txt
click==8.1.3
Flask==2.2.2
itsdangerous==2.1.2
Jinja2==3.1.2
MarkupSafe==2.1.1
Werkzeug==2.2.2
(.venv) ilara@LPI:~/docker1$ touch app.py
(.venv) ilara@LPI:~/docker1$ ls
app.py requirements.txt
(.venv) ilara@LPI:~/docker1$ nano app.py
(.venv) ilara@LPI:~/docker1$ python3 -m flask run
* Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
^C(.venv) ilara@LPI:~/docker1$ python3 -m flask run
* Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
127.0.0.1 - - [27/Oct/2022 09:55:05] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [27/Oct/2022 09:55:05] "GET /favicon.ico HTTP/1.1" 404 -
10:07


ilara@LPI: ~/docker1
Archivo Editar Ver Buscar Terminal Ayuda
GNU nano 6.2 app.py *
# Programa de Isidro Lara Lopez
# Programacion de redes
from flask import Flask
app = Flask(__name__)
@app.route('/')
def hello_world():
return 'Hello, Docker!'
^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location M-U Undo M-A Set Mark M-] To Bracket M-Q Previous
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^/ Go To Line M-E Redo M-6 Copy ^Q Where Was M-W Next
10:10



Run your image as a container: