

Rest API

Install and activate environment

```
garfield@LAPTOP-R002ACDB x + v
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 5.15.167.4-microsoft-standard-WSL2 x86_64)

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

System information as of Thu Sep 25 22:40:51 AEST 2025

This message is shown once a day. To disable it please create the
/home/garfield/.hushlogin file.
garfield@LAPTOP-R002ACDB:~$ python3 -m venv
usage: venv [-h] [--system-site-packages] [--symlinks | --copies] [--clear] [--upgrade] [--without-pip]
            [--prompt PROMPT] [--upgrade-deps]
            ENV_DIR [ENV_DIR ...]
venv: error: the following arguments are required: ENV_DIR
garfield@LAPTOP-R002ACDB:~$ python3 -m venv venv
garfield@LAPTOP-R002ACDB:~$ source venv/bin/activate
(venv) garfield@LAPTOP-R002ACDB:~$ pip install fastapi uvicorn asyncpg
Collecting fastapi
  Downloading fastapi-0.117.1-py3-none-any.whl.metadata (28 kB)
Collecting uvicorn
  Downloading uvicorn-0.37.0-py3-none-any.whl.metadata (6.6 kB)
Collecting asyncpg
  Downloading asyncpg-0.30.0-cp312-cp312-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (5.0 kB)
Collecting starlette<0.49.0,>=0.40.0 (from fastapi)
  Downloading starlette-0.48.0-py3-none-any.whl.metadata (6.3 kB)
Collecting pydantic!=1.8,!>1.8.1,!>2.0.0,!>2.0.1,!>2.1.0,<3.0.0,>=1.7.4 (from fastapi)
  Downloading pydantic-2.11.9-py3-none-any.whl.metadata (68 kB)
68.4/68.4 kB 765.7 kB/s eta 0:00:00
Collecting typing-extensions>=4.8.0 (from fastapi)
  Downloading typing_extensions-4.15.0-py3-none-any.whl.metadata (3.3 kB)
Collecting click>=7.0 (from uvicorn)
  Downloading click-8.3.0-py3-none-any.whl.metadata (2.6 kB)
Collecting h11>=0.8 (from uvicorn)
  Downloading h11-0.16.0-py3-none-any.whl.metadata (8.3 kB)
```

Create Python file, including cache

```
garfield@LAPTOP-R002ACDB x + v
** (pluma:2908): WARNING **: 23:54:06.630: Could not load theme icon system-file-manager: Icon 'system-file-manager' not present in theme Yaru
(venv) garfield@LAPTOP-R002ACDB:~$ cat main.py
from fastapi_cache import FastAPICache
from fastapi_cache.backends.inmemory import InMemoryBackend
from fastapi_cache.decorator import cache

@app.on_event("startup")
async def startup():
    global pool
    pool = await asyncpg.create_pool(
        host=os.getenv("DATABASE_HOST", "127.0.0.1"),
        port=int(os.getenv("DATABASE_PORT", "8812")),
        user=os.getenv("DATABASE_USER", "admin"),
        password=os.getenv("DATABASE_PASS", "quest"),
        database=os.getenv("DATABASE_NAME", "qdb")
    )
    FastAPICache.init(InMemoryBackend(), prefix="fastapi-cache")

@app.get("/cached_trades")
@cache(expire=60) # cache i 60 sekunder
async def cached_query():
    async with pool.acquire() as conn:
        row = await conn.fetchrow("SELECT * FROM cpp_trades LIMIT -1")
        return {"Row name": row["symbol"], "Price": row["price"]}

(venv) garfield@LAPTOP-R002ACDB:~$ pwd
/home/garfield
(venv) garfield@LAPTOP-R002ACDB:~$
```

Install cache library

```
garfield@LAPTOP-R002ACDB: X + v
config.load()
File "/home/garfield/venv/lib/python3.12/site-packages/uvicorn/config.py", line 438, in load
self.loaded_app = import_from_string(self.app)
File "/home/garfield/venv/lib/python3.12/site-packages/uvicorn/importer.py", line 22, in import_from_string
raise exc from None
File "/home/garfield/venv/lib/python3.12/site-packages/uvicorn/importer.py", line 19, in import_from_string
module = importlib.import_module(module_str)
File "/usr/lib/python3.12/importlib/_init_.py", line 90, in import_module
return _bootstrap.gcd_import(name[level:], package, level)
File "<frozen importlib._bootstrap>", line 1387, in _gcd_import
File "<frozen importlib._bootstrap>", line 1360, in _find_and_load
File "<frozen importlib._bootstrap>", line 1331, in _find_and_load_unlocked
File "<frozen importlib._bootstrap>", line 935, in _load_unlocked
File "<frozen importlib._bootstrap_external>", line 995, in exec_module
File "<frozen importlib._bootstrap>", line 488, in _call_with_frames_removed
File "/home/garfield/main.py", line 1, in <module>
from fastapi_cache import FastAPICache
ModuleNotFoundError: No module named 'fastapi_cache'
^C[INFO] Stopping reloader process [2963]
(venv) garfield@LAPTOP-R002ACDB:~$ pip install fastapi-cache2
Collecting fastapi-cache2
  Downloading fastapi-cache2-0.2.2-py3-none-any.whl.metadata (8.8 kB)
Requirement already satisfied: fastapi in ./venv/lib/python3.12/site-packages (from fastapi-cache2) (0.117.1)
Collecting pendulum<4.0.0,>=3.0.0 (from fastapi-cache2)
  Downloading pendulum-3.1.0-cp312-cp312-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (6.8 kB)
Requirement already satisfied: typing-extensions>=4.1.0 in ./venv/lib/python3.12/site-packages (from fastapi-cache2) (4.15.0)
Requirement already satisfied: uvicorn in ./venv/lib/python3.12/site-packages (from fastapi-cache2) (0.37.0)
Collecting python-dateutil>=2.6 (from pendulum<4.0.0,>=3.0.0->fastapi-cache2)
  Downloading python-dateutil-2.9.0.post0-py2.py3-none-any.whl.metadata (8.4 kB)
Collecting tzdata>=2020.1 (from pendulum<4.0.0,>=3.0.0->fastapi-cache2)
  Downloading tzdata-2025.2-py2.py3-none-any.whl.metadata (1.4 kB)
Requirement already satisfied: starlette<0.49.0,>=0.40.0 in ./venv/lib/python3.12/site-packages (from fastapi->fastapi-cache2) (0.48.0)
Requirement already satisfied: pydantic!=1.8,!1.8.1,!2.0.0,!2.0.1,!2.1.0,<3.0.0,>=1.7.4 in ./venv/lib/python3.12/site-packages (from fastapi->fastapi-cache2) (2.11.9)
Requirement already satisfied: click>=7.0 in ./venv/lib/python3.12/site-packages (from uvicorn->fastapi-cache2) (8.3.0)
Requirement already satisfied: h11>=0.8 in ./venv/lib/python3.12/site-packages (from uvicorn->fastapi-cache2) (0.16.0)
Requirement already satisfied: annotated-types>=0.6.0 in ./venv/lib/python3.12/site-packages (from pydantic!=1.8,!1.8.1,!2.0.0,!2.0.1,!2.1.0,<3.0.0,>=1.7.4->fastapi->fastapi-cache2) (0.7.0)
```

Set up requirements.txt & dockerfile, & .yaml file

```
garfield@LAPTOP-R002ACDB: X + v
fastapi
uvicorn
asynpg
fastapi-cache2

(venv) garfield@LAPTOP-R002ACDB:~$ cat docker-compose.yml
version: "3.9"
services:
  questdb:
    image: questdb/questdb:latest
    ports:
      - "8812:8812"
      - "9000:9000"
      - "9009:9009"

  fastapi:
    build: .
    ports:
      - "8000:8000"
    depends_on:
      - questdb
    environment:
      - DATABASE_HOST=questdb
      - DATABASE_PORT=8812
      - DATABASE_USER=admin
      - DATABASE_PASS=quest
      - DATABASE_NAME=qdb

(venv) garfield@LAPTOP-R002ACDB:~$ cat Dockerfile
FROM python:3.12-slim

WORKDIR /app
COPY requirements.txt .
RUN pip install -r requirements.txt

COPY . .

CMD ["uvicorn", "main:app", "--host", "0.0.0.0", "--port", "8000"]

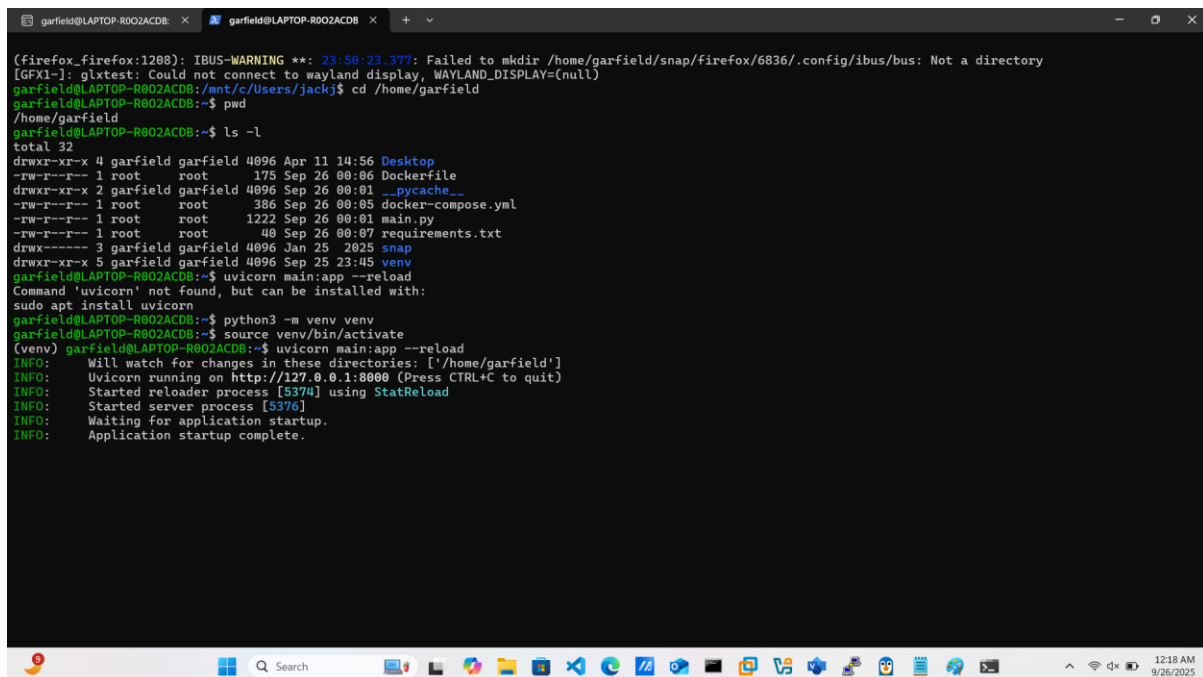
(venv) garfield@LAPTOP-R002ACDB:~$
```


Commands til activate envoiement:

python3 -m venv venv source venv/bin/activate

pip install fastapi uvicorn asynccpg

uvicorn main:app --reload



```
(firefox_firefox:1208): IBUS-WARNING **: 23:50:23.377: Failed to mkdir /home/garfield/snap/firefox/6836/.config/ibus/bus: Not a directory
[GFX1-]: glxtest: Could not connect to wayland display, WAYLAND_DISPLAY=(null)
garfield@LAPTOP-R802ACDB: /mnt/c/Users/jackj$ cd /home/garfield
garfield@LAPTOP-R802ACDB:~$ pwd
/home/garfield
garfield@LAPTOP-R802ACDB:~$ ls -l
total 32
drwxr-xr-x 4 garfield garfield 4096 Apr 11 14:56 Desktop
-rw-r--r-- 1 root root 175 Sep 26 00:06 Dockerfile
drwxr-xr-x 2 garfield garfield 4096 Sep 26 00:01 __pycache__
-rw-r--r-- 1 root root 386 Sep 26 00:05 docker-compose.yml
-rw-r--r-- 1 root root 1222 Sep 26 00:01 main.py
-rw-r--r-- 1 root root 40 Sep 26 00:07 requirements.txt
drwx----- 3 garfield garfield 4096 Jan 25 2025 snap
drwxr-xr-x 5 garfield garfield 4096 Sep 25 23:45 venv
garfield@LAPTOP-R802ACDB:~$ uvicorn main:app --reload
Command 'uvicorn' not found, but can be installed with:
sudo apt install uvicorn
garfield@LAPTOP-R802ACDB:~$ python3 -m venv venv
garfield@LAPTOP-R802ACDB:~$ source venv/bin/activate
(venv) garfield@LAPTOP-R802ACDB:~$ uvicorn main:app --reload
INFO: Will watch for changes in these directories: ['/home/garfield']
INFO: Uvicorn running on http://127.0.0.1:8000 (Press CTRL+C to quit)
INFO: Started reloader process [5374] using StatReload
INFO: Started server process [5376]
INFO: Waiting for application startup.
INFO: Application startup complete.
```

Dockerfile:

FROM python:3.12-slim

WORKDIR /app

COPY requirements.txt .

RUN pip install -r requirements.txt

COPY . .

CMD ["uvicorn", "main:app", "--host", "0.0.0.0", "--port", "8000"]

Requirements.txt

fastapi

uvicorn

asyncpg

fastapi-cache2

docker-compose.yml

version: "3.9"

services:

questdb:

image: questdb/questdb:latest

ports:

- "8812:8812"

- "9000:9000"

- "9009:9009"

fastapi:

build: .

ports:

- "8000:8000"

depends_on:

- questdb

environment:

- DATABASE_HOST=questdb

- DATABASE_PORT=8812

- DATABASE_USER=admin

- DATABASE_PASS=quest

- DATABASE_NAME=qdb

Main.py

```
from fastapi import FastAPI
```

```
import asyncpg
```

```
import os
```

```
from fastapi_cache import FastAPICache
```

```
from fastapi_cache.backends.inmemory import InMemoryBackend
```

```
from fastapi_cache.decorator import cache
```

```
# Create FastAPI app
```

```
app = FastAPI()
```

```
# Global connection pool
```

```
pool = None
```

```
# Startup event: initialize DB pool + cache
```

```
@app.on_event("startup")
```

```
async def startup():
```

```
    global pool
```

```
    pool = await asyncpg.create_pool(
```

```
        host=os.getenv("DATABASE_HOST", "127.0.0.1"),
```

```
        port=int(os.getenv("DATABASE_PORT", "8812")),
```

```
        user=os.getenv("DATABASE_USER", "admin"),
```

```

    password=os.getenv("DATABASE_PASS", "quest"),

    database=os.getenv("DATABASE_NAME", "qdb")

)

FastAPICache.init(InMemoryBackend(), prefix="fastapi-cache")


# Shutdown event: close pool

@app.on_event("shutdown")

async def shutdown():

    await pool.close()


# Root endpoint

@app.get("/")

def read_root():

    return {"Hello": "World"}


# Cached query endpoint

@app.get("/cached_trades")

@cache(expire=60) # cache for 60 seconds

async def cached_query():

    async with pool.acquire() as conn:

        row = await conn.fetchrow("SELECT * FROM cpp_trades LIMIT -1")

    return {"Row name": row["symbol"], "Price": row["price"]}

```