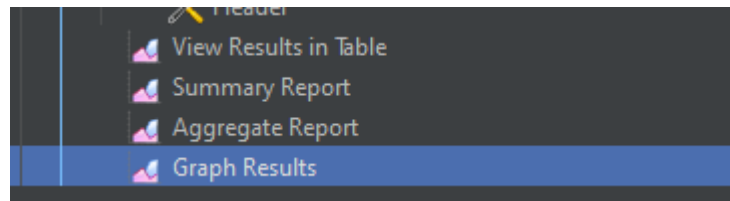
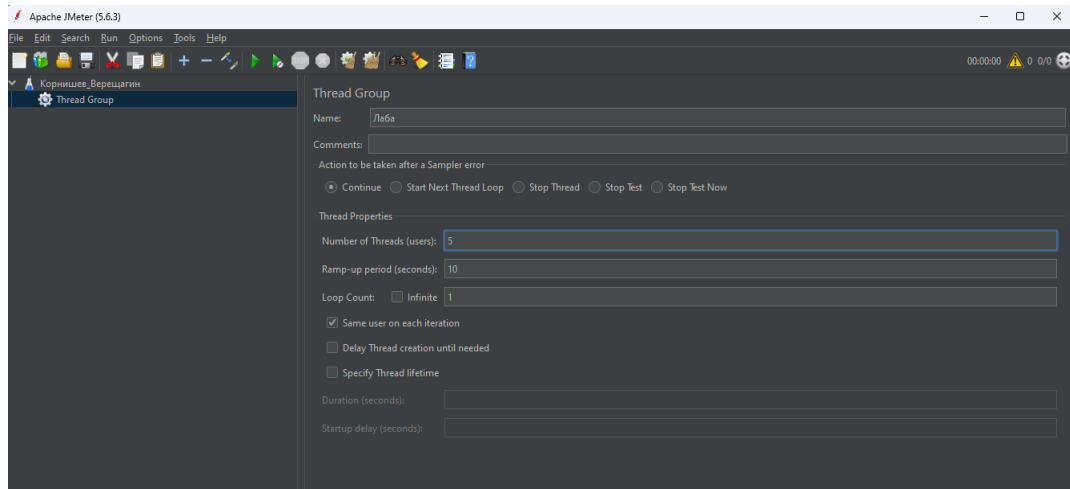


Лабораторная работа №5

Apache JMeter.

Первым действием скачиваем и распаковываем Apache JMeter с официального сайта. Затем его открываем, создаем тестовый план. Добавляем Thread Group, затем Sampler, под конец Listener и запускаем сам тест.



Рисунки 1 – 3 -Первоначальная настройка Jmeter

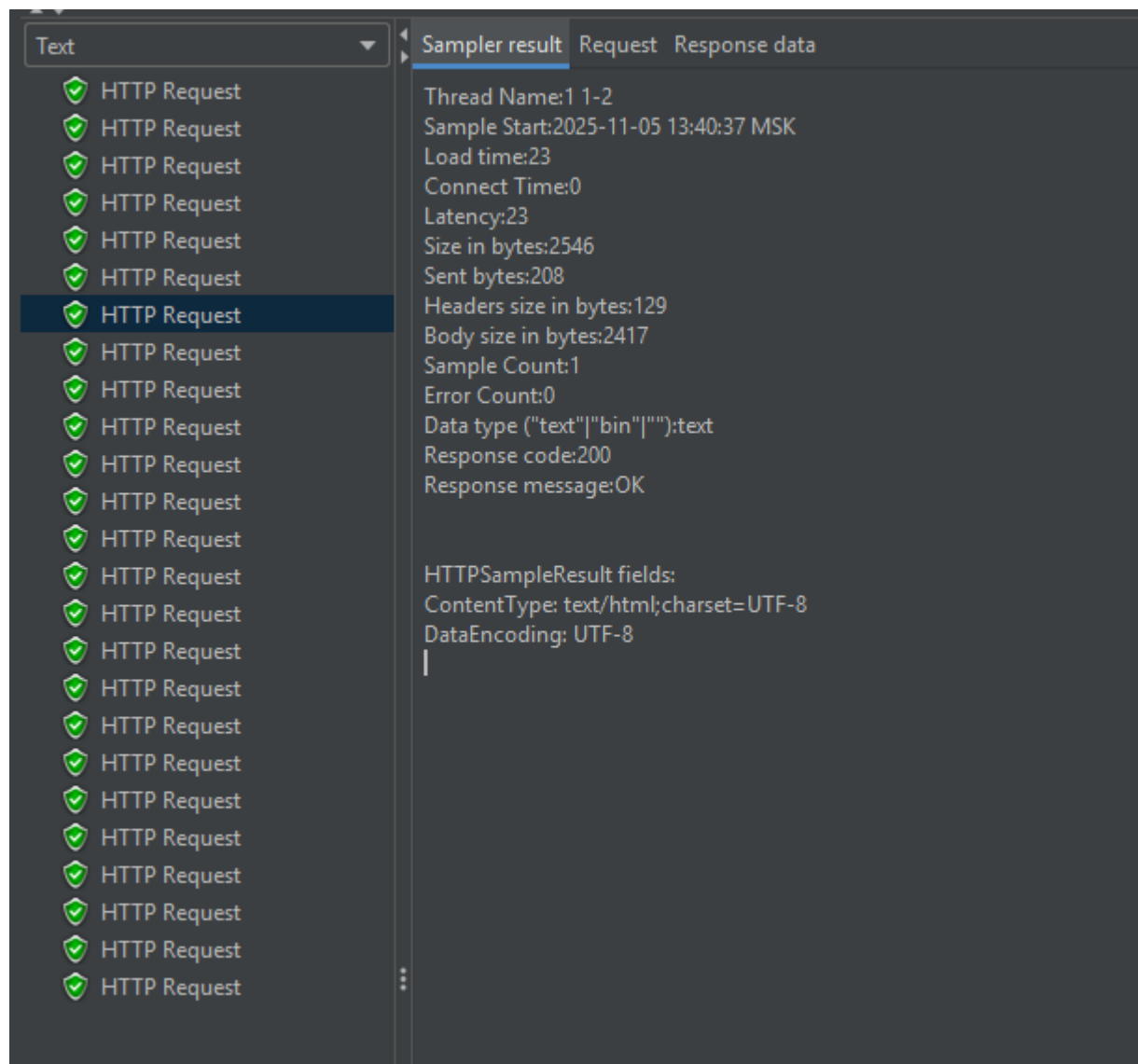


Рисунок 4 – Готовый тест

Затем, создаем и запускаем сервер на Python. Сначала устанавливаем зависимости через командную строку, при помощи следующей команды: `pip install fastapi[all] uvicorn`.

```

Collecting watchfiles>=0.13 (from uvicorn[standard]>=0.12.0; extra == "all"-->fastapi[all])
  Downloading watchfiles-1.1.1-cp313-cp313-win_amd64.whl.metadata (5.0 kB)
Collecting websockets>=10.4 (from uvicorn[standard]>=0.12.0; extra == "all"-->fastapi[all])
  Downloading websockets-15.0.1-cp313-cp313-win_amd64.whl.metadata (7.0 kB)
Downloading fastapi-0.121.0-py3-none-any.whl (109 kB)
Downloading httpx-0.28.1-py3-none-any.whl (73 kB)
Downloading httpcore-1.0.9-py3-none-any.whl (78 kB)
Downloading starlette-0.49.3-py3-none-any.whl (74 kB)
Downloading anyio-4.11.0-py3-none-any.whl (109 kB)
Downloading uvicorn-0.38.0-py3-none-any.whl (68 kB)
Downloading annotated_doc-0.0.3-py3-none-any.whl (5.5 kB)
Downloading click-8.3.0-py3-none-any.whl (107 kB)
Downloading email_validator-2.3.0-py3-none-any.whl (35 kB)
Downloading dnspython-2.8.0-py3-none-any.whl (331 kB)
Downloading fastapi_cli-0.0.14-py3-none-any.whl (11 kB)
Downloading fastapi_cloud_cli-0.3.1-py3-none-any.whl (19 kB)
Using cached h11-0.16.0-py3-none-any.whl (37 kB)
Downloading itsdangerous-2.2.0-py3-none-any.whl (16 kB)
Downloading jinja2-3.1.6-py3-none-any.whl (134 kB)
Downloading markupsafe-3.0.3-cp313-cp313-win_amd64.whl (15 kB)
Downloading orjson-3.11.4-cp313-cp313-win_amd64.whl (131 kB)
Downloading pydantic_extra_types-2.10.6-py3-none-any.whl (40 kB)
Downloading pydantic_settings-2.11.0-py3-none-any.whl (48 kB)
Downloading python_multipart-0.0.20-py3-none-any.whl (24 kB)
Downloading pyyaml-6.0.3-cp313-cp313-win_amd64.whl (154 kB)
Downloading rich_toolkit-0.15.1-py3-none-any.whl (29 kB)
Downloading rich-14.2.0-py3-none-any.whl (243 kB)
Using cached pygments-2.19.2-py3-none-any.whl (1.2 MB)
Downloading markdown_it_py-4.0.0-py3-none-any.whl (87 kB)
Downloading mdurl-0.1.2-py3-none-any.whl (10.0 kB)
Downloading rignore-0.7.5-cp313-cp313-win_amd64.whl (726 kB)
 726.2/726.2 kB 3.2 MB/s eta 0:00:00
Downloading sentry_sdk-2.43.0-py2.py3-none-any.whl (400 kB)
Using cached sniffio-1.3.1-py3-none-any.whl (10 kB)
Downloading typer-0.20.0-py3-none-any.whl (47 kB)
Downloading shellingham-1.5.4-py2.py3-none-any.whl (9.8 kB)
Downloading ujson-5.11.0-cp313-cp313-win_amd64.whl (43 kB)
Using cached urllib3-2.5.0-py3-none-any.whl (129 kB)
Using cached colorama-0.4.6-py2.py3-none-any.whl (25 kB)
Downloading httptools-0.7.1-cp313-cp313-win_amd64.whl (85 kB)
Downloading watchfiles-1.1.1-cp313-cp313-win_amd64.whl (288 kB)
Downloading websockets-15.0.1-cp313-cp313-win_amd64.whl (176 kB)
Installing collected packages: websockets, urllib3, ujson, sniffio, shellingham, rignore, pyyaml, python-multipart, pygm
ents, orjson, mdurl, MarkupSafe, itsdangerous, httptools, h11, dnspython, colorama, annotated-doc, sentry-sdk, markdown-
it-py, jinja2, httpcore, email-validator, click, anyio, watchfiles, uvicorn, starlette, rich, pydantic-settings, pydanti
c-extra-types, httpx, typer, rich-toolkit, fastapi, fastapi-cloud-cli, fastapi-cli
Successfully installed MarkupSafe-3.0.3 annotated-doc-0.0.3 anyio-4.11.0 click-8.3.0 colorama-0.4.6 dnspython-2.8.0 emai
l-validator-2.3.0 fastapi-0.121.0 fastapi-cli-0.0.14 fastapi-cloud-cli-0.3.1 h11-0.16.0 httpcore-1.0.9 httptools-0.7.1 h
ttpx-0.28.1 itsdangerous-2.2.0 jinja2-3.1.6 markdown-it-py-4.0.0 mdurl-0.1.2 orjson-3.11.4 pydantic-extra-types-2.10.6 p
ydantic-settings-2.11.0 pygments-2.19.2 python-multipart-0.0.20 pyyaml-6.0.3 rich-14.2.0 rich-toolkit-0.15.1 rignore-0.7
.5 sentry-sdk-2.43.0 shellingham-1.5.4 sniffio-1.3.1 starlette-0.49.3 typer-0.20.0 ujson-5.11.0 urllib3-2.5.0 uvicorn-0.
38.0 watchfiles-1.1.1 websockets-15.0.1

[notice] A new release of pip is available: 25.1.1 -> 25.3
[notice] To update, run: python.exe -m pip install --upgrade pip

```

Рисунок 5 – Установка зависимостей

Затем, в самом Python, вписываем код и запускаем сервер с помощью команды - `uvicorn main:app --reload`. У нас откроется пустая страница в браузере. После чего, снова входим в Apache JMeter и создаем там 2 запроса – на создание и получение предмета.

Thread Group

Name:

Comments:

Action to be taken after a Sampler error

☒ Continue
 ☐ Start Next Thread Loop
 ☐ Stop Thread
 ☐ Stop Test
 ☐ Stop Test Now

Thread Properties

Number of Threads (users):

Ramp-up period (seconds):

Loop Count: ☐ Infinite

☒ Same user on each iteration
☐ Delay Thread creation until needed
☐ Specify Thread lifetime

Duration (seconds):

Startup delay (seconds):

HTTP Request

Name:

Comments:

Basic Advanced

Web Server

Protocol [http]: Server Name or IP:

HTTP Request

POST Path:

☐ Redirect Automatically
 ☒ Follow Redirects
 ☒ Use KeepAlive
 ☐ Use multipart/form-data
 ☐ Browser-compatible headers

Parameters Body Data Files Upload

```

1 {
2   "name": "Item ${__threadNum}",
3   "description": "From Meter",
4   "price": "${__Random(50,500)}"
5 }
  
```

HTTP Request

Name:

Comments:

Basic Advanced

Web Server

Protocol [http]: Server Name or IP:

HTTP Request

GET Path:

☐ Redirect Automatically
 ☒ Follow Redirects
 ☒ Use KeepAlive
 ☐ Use multipart/form-data
 ☐ Browser-compatible headers

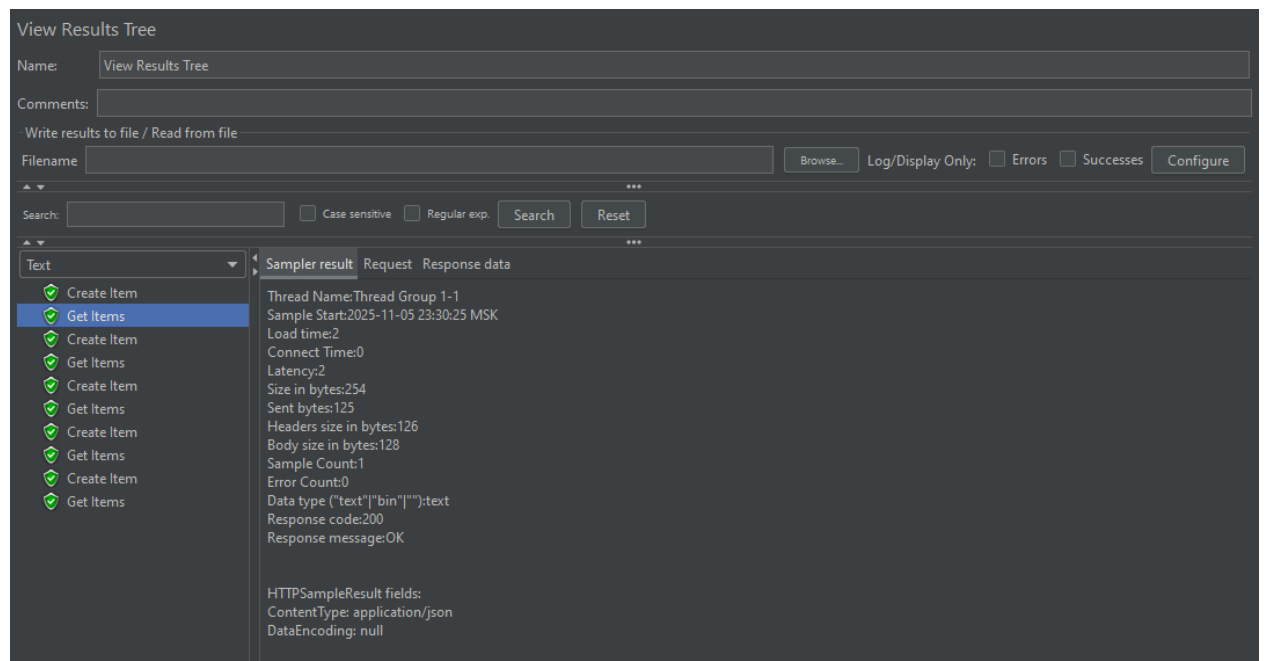
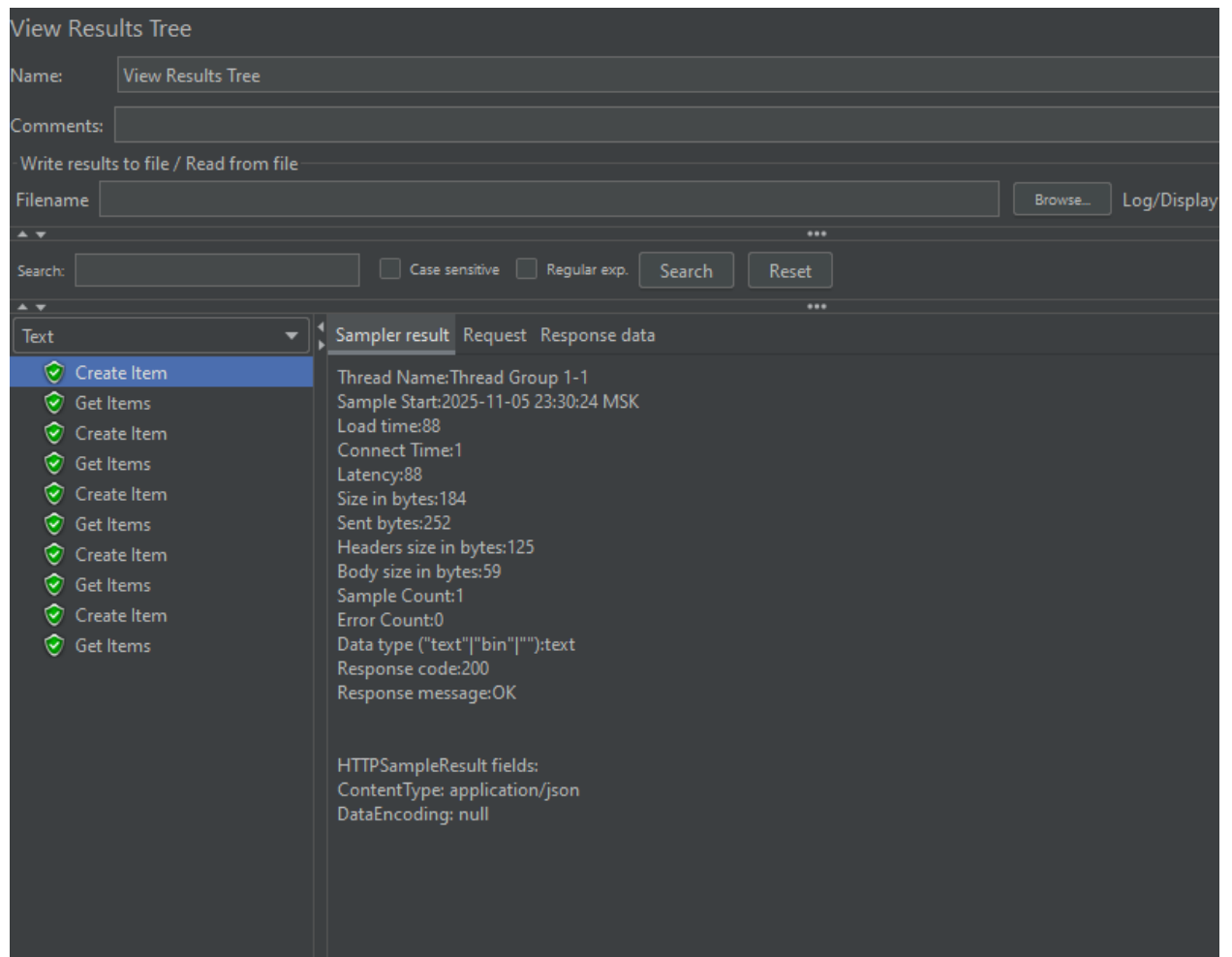
Parameters Body Data Files Upload

Send Parameters With

Name:	Value

Рисунки 6 – 9 – создание запросов

И запускаем тест.



Рисунки 10 – 11 – Успешные тест