

# Лабораторная работа 5

## Ильин ВА 1гр 1пдгр

### Часть 1

Модифицированный код борда №1

```
from http.server import HTTPServer, BaseHTTPRequestHandler
from datetime import datetime

class SimpleHTTPRequestHandler(BaseHTTPRequestHandler):

    def do_GET(self):
        self.send_response(200)
        self.send_header('Content-Type', 'text/plain; charset=utf-8')
        self.end_headers()
        result = f"1148905, {datetime.now().strftime('%d.%m.%y %H:%M:%S')}}"
        self.wfile.write(bytes(result, "utf-8"))

httpd = HTTPServer(('0.0.0.0', 8080), SimpleHTTPRequestHandler)
print('server is running ')
httpd.serve_forever()
```

Модифицированный код борда №2

```
from flask import Flask
from datetime import datetime
app = Flask(__name__)

@app.route('/')
def do_get():
    return f"<h1>1148905, {datetime.now().strftime('%d.%m.%y %H:%M:%S')}}</h1>"

app.run('127.0.0.1', 8080) # aka serve_forever()
```

Код сервера на PHP

```
<?php
echo "<h2>1148905, ". date("d.m.Y h:i:s") . "</h2>" ;
?>
```

### Часть 2.1

Код веб-приложения

```
from flask import Flask, render_template, request
import csv

app = Flask(__name__)

try:
    with open('data.csv', 'a', newline='') as file:
        writer = csv.writer(file)
        if file.tell() == 0: # Если файл новый
            writer.writerow(['moodle_login', 'timestamp'])
except Exception as e:
    print(f"Ошибка при создании CSV файла: {e}")

@app.route('/', methods=['GET', 'POST'])
def index():
    if request.method == 'POST':
        try:
```

```

moodle_login = request.form.get('moodle_login', '')
timestamp = request.form.get('timestamp', '')

if not moodle_login or not timestamp:
    return render_template('index.html', error='Все поля обязательны для заполнения')

with open('data.csv', 'a', newline='') as file:
    writer = csv.writer(file)
    writer.writerow([moodle_login, timestamp])

    return render_template('index.html', success='Данные успешно сохранены')
except Exception as e:
    return render_template('index.html', error=f'Ошибка при сохранении данных: {e}')

return render_template('index.html')

app.run('127.0.0.1', 8080)

```

templates/index.html

```

<!DOCTYPE html>
<html lang="ru">
<head>
    <meta charset="UTF-8">
    <title>Форма отправки данных</title>
    <style>
        .error { color: red; }
        .success { color: green; }
        form { margin: 20px 0; }
        label { display: block; margin: 10px 0; }
        input { margin-left: 10px; }
    </style>
</head>
<body>
    <h2>Форма отправки данных</h2>

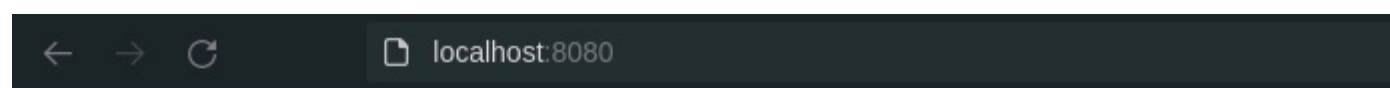
    {% if error %}
        <p class="error">{{ error }}</p>
    {% endif %}
    {% if success %}
        <p class="success">{{ success }}</p>
    {% endif %}

    <form method="POST">
        <label>Логин в системе Moodle:
            <input type="text" name="moodle_login" required>
        </label>
        <label>Текущее время:
            <input type="text" name="timestamp" id="timestamp" readonly required>
        </label>
        <input type="submit" value="Отправить">
    </form>

    <script>
        function updateTime() {
            const now = new Date();
            document.getElementById('timestamp').value =
                now.toLocaleString('ru-RU');
        }
        setInterval(updateTime, 1000);
        updateTime();
    </script>
</body>
</html>

```

Запуск в браузере



## Форма отправки данных

Данные успешно сохранены

Логин в системе Moodle:

Текущее время:

Отправить

The screenshot shows the Chrome DevTools Network tab. The top toolbar includes icons for Inspector, Console, Debugger, Network (active), Style Editor, Performance, and Memory. Below the toolbar is a filter bar with 'Filter URLs' and a search icon. The main area displays a list of network requests. The first request is a POST to 'loc... /' with status 200, type 'html', and size 1.33 kB. The second request is a GET for 'favicon.ico' with status 200, type 'html', and size 655 bytes. The right sidebar shows the 'Headers' tab for the selected POST request. It displays the status '200 OK', version 'HTTP/1.1', and various headers including 'Content-Type: text/html; charset=utf-8', 'Date: Wed, 19 Mar 2025 09:29:53 GMT', and 'Server: Werkzeug/3.1.3 Python/3.12.9'. The bottom status bar shows '2 requests', '1.81 kB / 1.33 kB transferred', and 'Finish: 468 ms'.

## Часть 2.2

Код веб-приложения

```
from http.server import HTTPServer, BaseHTTPRequestHandler
from urllib.parse import parse_qs
import csv

try:
    with open('data.csv', 'a', newline='') as file:
        writer = csv.writer(file)
        if file.tell() == 0: # Если файл новый
            writer.writerow(['moodle_login', 'timestamp'])
except Exception as e:
    print(f"Ошибка при создании CSV файла: {e}")
```

```

class SimpleHTTPRequestHandler(BaseHTTPRequestHandler):
    def do_GET(self):
        self.send_response(200)
        self.send_header('Content-Type', 'text/html; charset=UTF-8')
        self.end_headers()
        with open("index-default.html") as f:
            result = f.readlines()

        self.wfile.write(bytes("".join(result).replace("\n", ""), "UTF-8"))

    def do_POST(self):
        self.send_response(200)
        self.send_header('Content-type', 'text/html; charset=UTF-8')
        self.end_headers()

        content_length = int(self.headers['Content-Length'])
        post_data = parse_qs(self.rfile.read(content_length).decode('utf-8'))

        moodle_login = post_data.get('moodle_login', [''])[0]
        timestamp = post_data.get('timestamp', [''])[0]

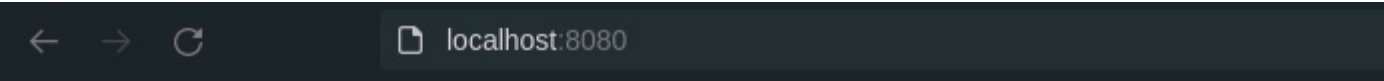
        if moodle_login and timestamp:
            with open('data.csv', 'a', newline='') as file:
                writer = csv.writer(file)
                writer.writerow([moodle_login, timestamp])
            with open("index-success.html") as f:
                result = f.readlines()
        else:
            with open("index-error.html") as f:
                result = f.readlines()

        self.wfile.write(bytes("".join(result).replace("\n", ""), "UTF-8"))

httpd = HTTPServer(('localhost', 8080), SimpleHTTPRequestHandler)
httpd.serve_forever()

```

Отличия index-default.html, index-success.html и index-error.html только в наличии/отсутствии текста об ошибке/успешной отправке запроса.



# Форма отправки данных

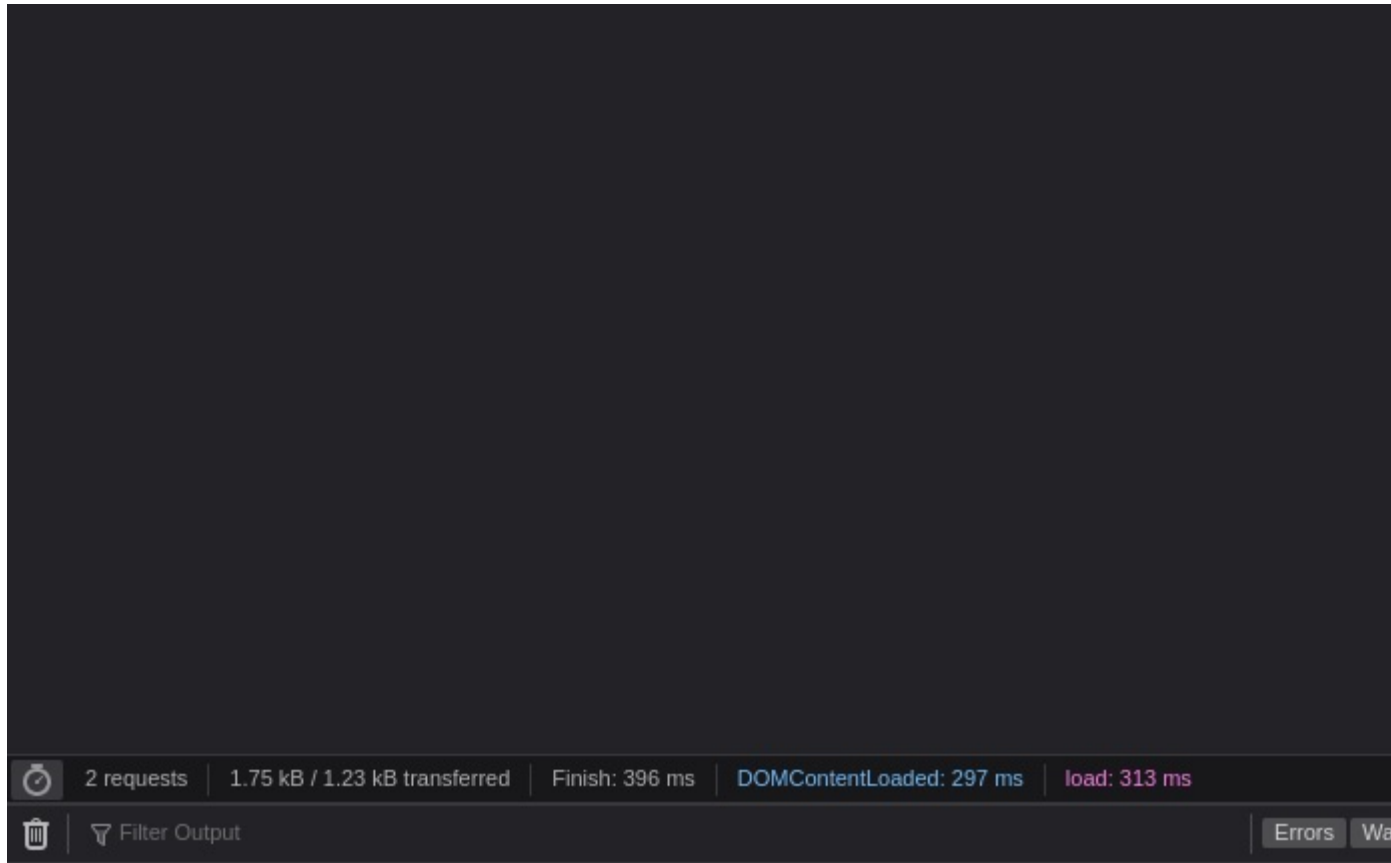
Данные успешно отправлены!

Логин в системе Moodle:

Текущее время:

Отправить

Status	Method	Domain	File	Initiator
200	POST	localhost:8080	/	document
200	GET	localhost:8080	favicon.ico	FaviconLoader.jsm:1..



POST без данных

```
femman@twinkpad ~/uni/comp-pract/lr5/code/2.2 $ python main.py
127.0.0.1 - - [19/Mar/2025 13:03:34] "POST / HTTP/1.1" 200 -
```

```
femman@twinkpad ~/uni/comp-pract/lr5/code/2.2 $ curl -d '' localhost:8080
<!DOCTYPE html><html lang="ru"><head>    <meta charset="UTF-8">    <title>Формы Moodle
    .error { color: red; }    .success { color: green; }    form {
y: block; margin: 10px 0; }    input { margin-left: 10px; }    </style></h2>
x</h2>    <p class="error">Оба поля должны быть заполнены!</p>    <form name Moodle:
:    <input type="text" name="moodle_login" required>
:    <input type="text" name="timestamp" id="timestamp" readonly required="submit" value="Отправить">    </form>    <script>        function updateTime
);    document.getElementById('timestamp').value =    document.getElementById('timestamp').value +
setInterval(updateTime, 1000);    updateTime();    </script></body></html>
```



