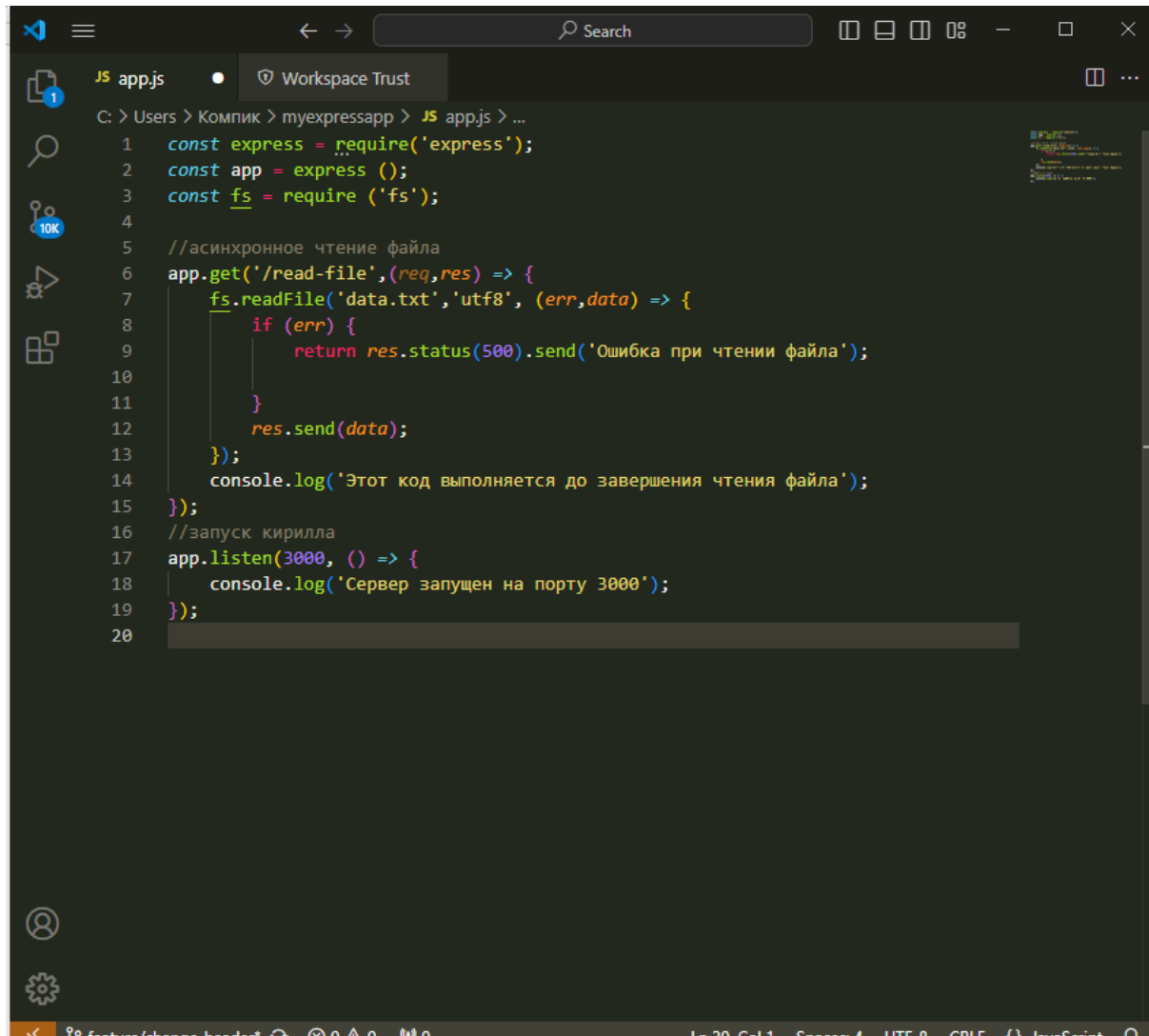


## лаба 6



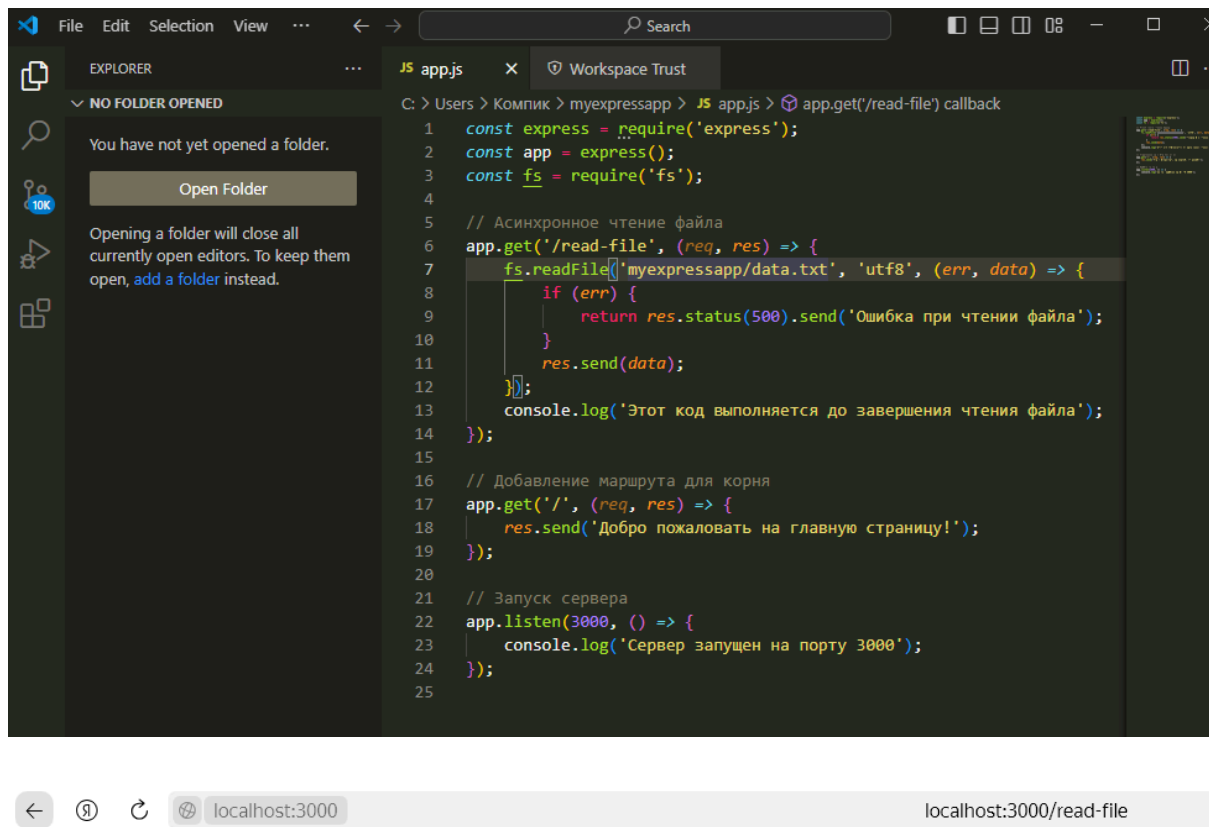
```
1  const express = require('express');
2  const app = express ();
3  const fs = require ('fs');
4
5  //асинхронное чтение файла
6  app.get('/read-file',(req,res) => {
7      fs.readFile('data.txt','utf8', (err,data) => {
8          if (err) {
9              return res.status(500).send('Ошибка при чтении файла');
10         }
11         res.send(data);
12     });
13     console.log('Этот код выполняется до завершения чтения файла');
14 });
15
16 //запуск кирилла
17 app.listen(3000, () => {
18     console.log('Сервер запущен на порту 3000');
19 });
20
```

node_modules	17 окт. 2024
app.js	17 окт. 2024
Data.txt	17 окт. 2024
package.json	17 окт. 2024
package-lock.json	17 окт. 2024

\*Data.txt – Блокнот

Файл Правка Формат Вид Справка

Тестовые каракули |



## Тестовые каракули

это была часть из “6 лекция”

а это идёт часть из практики

```
C:\Users\Компик\Desktop\laba6>npm init -y
Wrote to C:\Users\Компик\Desktop\laba6\package.json:
```

```
{
  "name": "laba6",
  "version": "1.0.0",
  "main": "server.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1",
    "start": "node server.js"
  },
  "keywords": [],
  "author": "",
  "license": "ISC",
  "description": ""
}
```

```
C:\Users\Компик\Desktop\laba6>npm install express mongoose body-parser
```

```
added 85 packages, and audited 86 packages in 19s
```

```
14 packages are looking for funding
  run `npm fund` for details
```

```
found 0 vulnerabilities
```

```
C:\Users\Компик\Desktop\laba6>_
```

```
C:\Users\Компик\Desktop\laba6>node server.js
(node:16224) [MONGODB DRIVER] Warning: useNewUrlParser is a deprecated option: useNewUrlParser has no effect since Node.js Driver version 4.0.0 and will be removed in the next major version
(Use 'node --trace-warnings ...' to show where the warning was created)
(node:16224) [MONGODB DRIVER] Warning: useUnifiedTopology is a deprecated option: useUnifiedTopology has no effect since Node.js Driver version 4.0.0 and will be removed in the next major version
Сервер запущен на порту 3000
Ошибка подключения: MongooseServerSelectionError: connect ECONNREFUSED ::1:27017, connect ECONNREFUSED 127.0.0.1:27017
    at _handleConnectionErrors (C:\Users\Компик\Desktop\laba6\node_modules\mongoose\lib\connection.js:909:11)
    at NativeConnection.openUri (C:\Users\Компик\Desktop\laba6\node_modules\mongoose\lib\connection.js:860:11) {
  reason: TopologyDescription {
    type: 'Unknown',
    servers: Map(1) { 'localhost:27017' => [ServerDescription] },
    status: false,
    compatible: true,
    heartbeatFrequencyMS: 10000,
    localThresholdMS: 15,
    setName: null,
    maxSetVersion: null,
    commonWireVersion: 0,
    logicalSessionTimeoutMinutes: null
  },
  code: undefined
}
```

не подключается :(

```
C:\Users\Компик\blog-app>npm install express mongoose bcrypt express-session body-parser connect-mongo
npm warn deprecated inflight@1.0.6: This module is not supported, and leaks memory. Do not use it. Check out lru-cache if you want a good and tested way to coalesce async requests by a key value, which is much more comprehensive and powerful.
npm warn deprecated glob@7.2.3: Glob versions prior to v9 are no longer supported
npm warn deprecated npmlog@5.0.1: This package is no longer supported.
npm warn deprecated rimraf@3.0.2: Rimraf versions prior to v4 are no longer supported
npm warn deprecated are-we-there-yet@2.0.0: This package is no longer supported.
npm warn deprecated gauge@3.0.2: This package is no longer supported.

added 157 packages, and audited 158 packages in 36s

17 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities

C:\Users\Компик\blog-app>npm install express mongoose bcrypt express-session body-parser connect-mongo
up to date, audited 158 packages in 3s

17 packages are looking for funding
  run `npm fund` for details

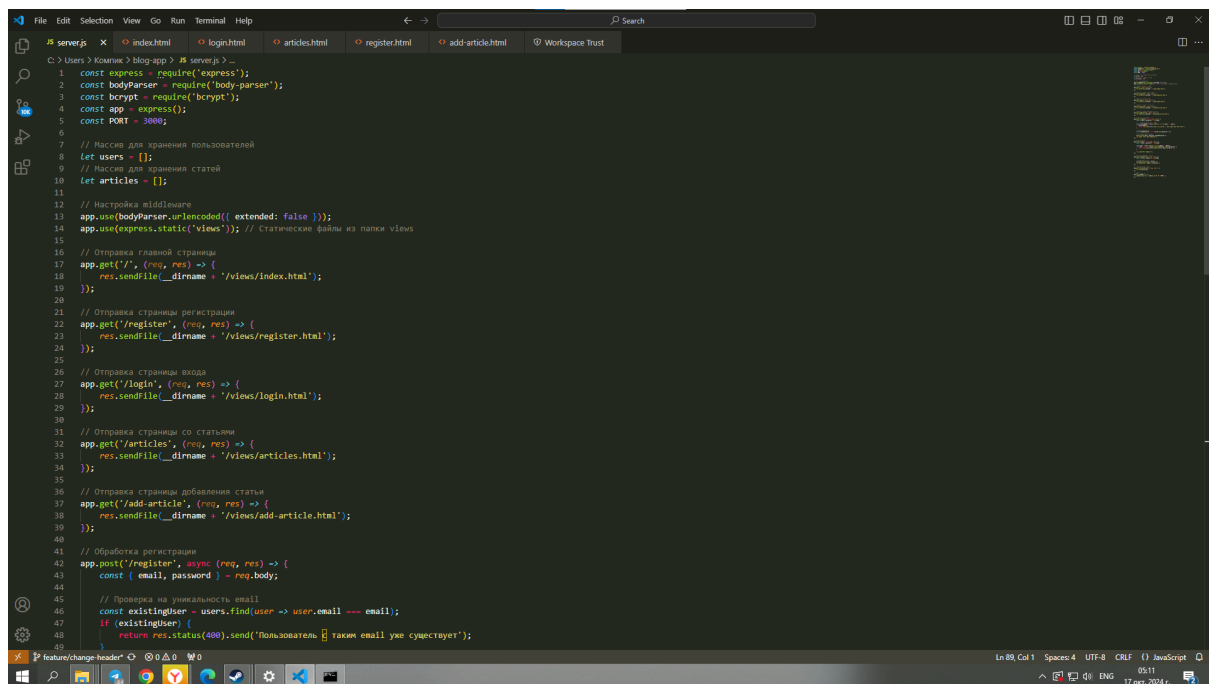
found 0 vulnerabilities

C:\Users\Компик\blog-app>_
```

```
localhost:3000 Error
MongoServerSelectionError: connect ECONNREFUSED ::1:27017, connect ECONNREFUSED 127.0.0.1:27017
    at Topology.selectServer (C:\Users\Компик\blog-app\node_modules\mongodb\lib\sdam\topology.js:303:38)
    at runNextTicks (node:internal/process/task_queues:60:5)
    at listOnTimeout (node:internal/timers:545:9)
    at process.processTimers (node:internal/timers:519:7)
    at async Topology._connect (C:\Users\Компик\blog-app\node_modules\mongodb\lib\sdam\topology.js:196:28)
    at async Topology.connect (C:\Users\Компик\blog-app\node_modules\mongodb\lib\sdam\topology.js:158:13)
    at async topologyConnect (C:\Users\Компик\blog-app\node_modules\mongodb\lib\mongo_client.js:209:17)
    at async MongoClient._connect (C:\Users\Компик\blog-app\node_modules\mongodb\lib\mongo_client.js:222:13)
    at async MongoClient.connect (C:\Users\Компик\blog-app\node_modules\mongodb\lib\mongo_client.js:147:13)
    at async MongoClient.connect (C:\Users\Компик\blog-app\node_modules\mongodb\lib\mongo_client.js:308:16)
```

не работает собака, поэтому попробую сделать попроще без монгода:

## ИТОГ

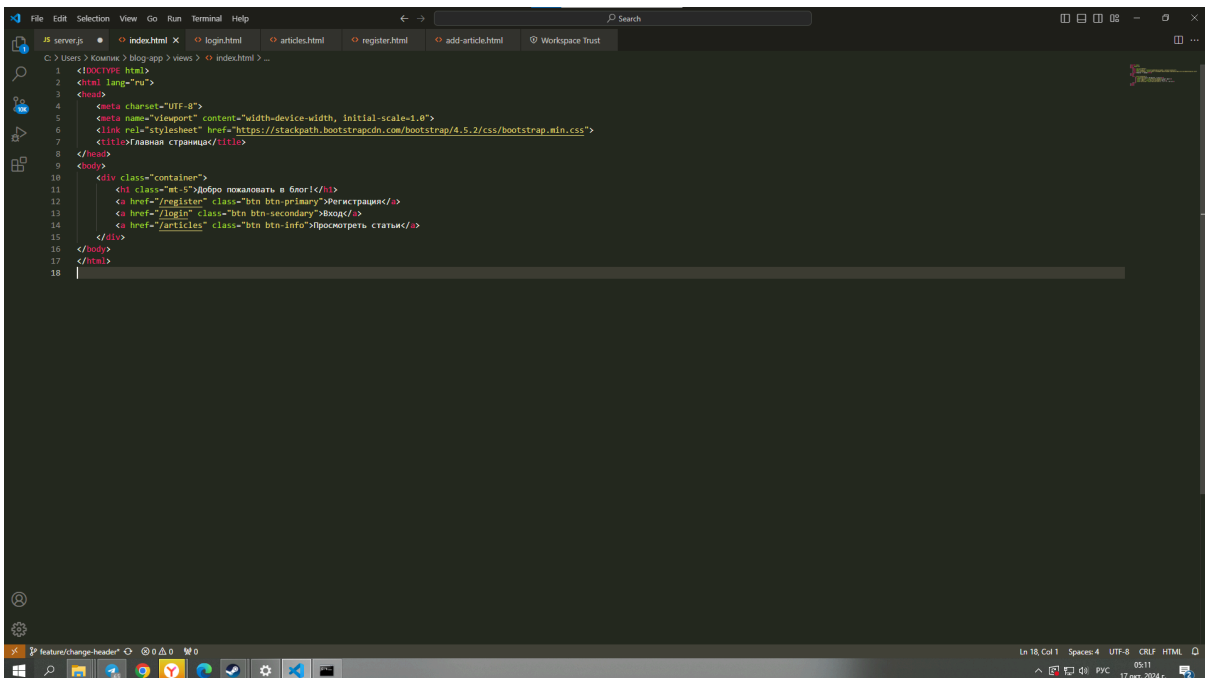


```
server.js
1 const express = require('express');
2 const bodyParser = require('body-parser');
3 const bcrypt = require('bcrypt');
4 const app = express();
5 const PORT = 3000;
6
7 // Массив для хранения пользователей
8 let users = [];
9 // Массив для хранения статей
10 let articles = [];
11
12 // Настройка middleware
13 app.use(bodyParser.urlencoded({ extended: false }));
14 app.use(express.static('views')); // Статические файлы из папки views
15
16 // Отправка главной страницы
17 app.get('/', (req, res) => {
18   res.sendFile(__dirname + '/views/index.html');
19 });
20
21 // Отправка страницы регистрации
22 app.get('/register', (req, res) => {
23   res.sendFile(__dirname + '/views/register.html');
24 });
25
26 // Отправка страницы входа
27 app.get('/login', (req, res) => {
28   res.sendFile(__dirname + '/views/login.html');
29 });
30
31 // Отправка страницы со статьями
32 app.get('/articles', (req, res) => {
33   res.sendFile(__dirname + '/views/articles.html');
34 });
35
36 // Отправка страницы добавления статьи
37 app.get('/add-article', (req, res) => {
38   res.sendFile(__dirname + '/views/add-article.html');
39 });
40
41 // Обработка регистрации
42 app.post('/register', async (req, res) => {
43   const { email, password } = req.body;
44
45   // Проверка на уникальность email
46   const existingUser = users.find(user => user.email === email);
47   if (existingUser) {
48     return res.status(400).send('Пользователь с таким email уже существует');
49   }
50 });
```

```

45 // Проверка на уникальность email
46 const existingUser = users.find(user => user.email === email);
47 if (existingUser) {
48   return res.status(400).send('Пользователь с таким email уже существует');
49 }
50
51 // Хеширование пароля
52 const hashedPassword = await bcrypt.hash(password, 10);
53
54 // Сохранение пользователя
55 users.push({ email, password: hashedPassword });
56 res.send('Регистрация успешна!');
57 });
58
59 // Обработка входа
60 app.post('/login', async (req, res) => {
61   const { email, password } = req.body;
62
63   const user = users.find(user => user.email === email);
64   if (!user || !(await bcrypt.compare(password, user.password))) {
65     return res.status(400).send('Неверный email или пароль');
66   }
67
68   res.send('Вход успешен!');
69 });
70
71 // Обработка добавления статьи
72 app.post('/add-article', (req, res) => {
73   const { title, content } = req.body;
74
75   // Добавление статьи в массив
76   articles.push({ title, content });
77   res.send('Статья успешно добавлена!');
78 });
79
80 // Отправка списка статей
81 app.get('/articles/list', (req, res) => {
82   res.json(articles);
83 });
84
85 // Запуск сервера
86 app.listen(PORT, () => {
87   console.log(`Сервер запущен на порту ${PORT}`);
88 });
89

```



```

1 <!DOCTYPE html>
2 <html lang="ru">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
7   <title>Главная страница</title>
8 </head>
9 <body>
10   <div class="container">
11     <div class="mt-5">Добро пожаловать в блог!</div>
12     <a href="/register" class="btn btn-primary">Регистрация</a>
13     <a href="/login" class="btn btn-secondary">Вход</a>
14     <a href="/articles" class="btn btn-info">Просмотреть статьи</a>
15   </div>
16 </body>
17 </html>
18

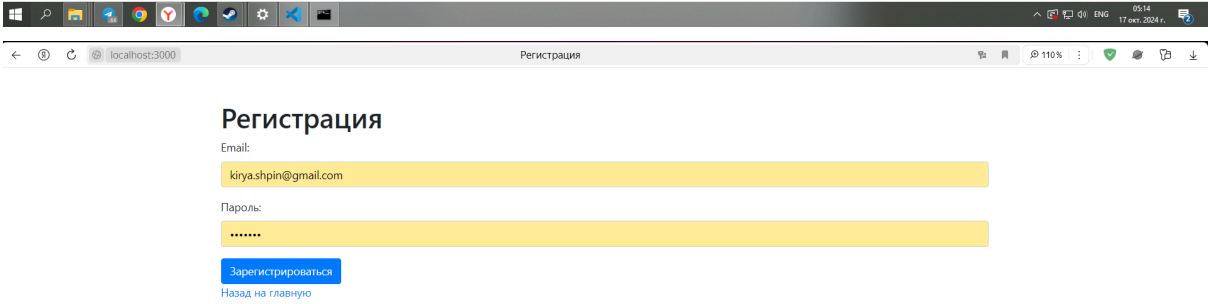
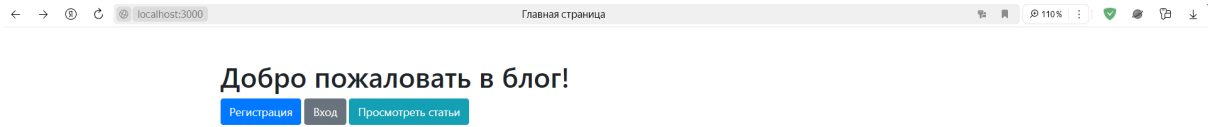
```

```
File Edit Selection View Go Run Terminal Help
server.js index.html login.html X articles.html register.html add-article.html Workspace Trust
C:\Users\Компик\blog-app\views> login.html > ...
1 <!DOCTYPE html>
2 <html lang="ru">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
7   <title>Вход</title>
8 </head>
9 <body>
10   <div class="container">
11     <h1 class="mt-5">Вход</h1>
12     <form action="/login" method="POST">
13       <div class="form-group">
14         <label for="email">Email:</label>
15         <input type="email" class="form-control" id="email" name="email" required>
16       </div>
17       <div class="form-group">
18         <label for="password">Пароль:</label>
19         <input type="password" class="form-control" id="password" name="password" required>
20       </div>
21       <button type="submit" class="btn btn-primary">Войти</button>
22     </form>
23     <a href="/">Назад на главную</a>
24   </div>
25 </body>
26 </html>
27
```

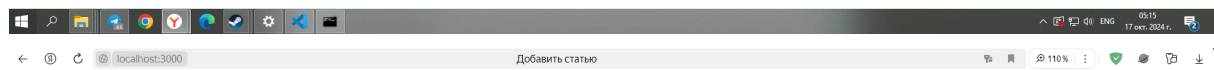
```
File Edit Selection View Go Run Terminal Help
server.js index.html login.html X articles.html register.html add-article.html Workspace Trust
C:\Users\Компик\blog-app\views> articles.html > ...
1 <!DOCTYPE html>
2 <html lang="ru">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
7   <title>Список статей</title>
8 </head>
9 <body>
10   <div class="container">
11     <h1 class="mt-5">Список статей</h1>
12     <a href="/add-article" class="btn btn-primary mb-3">Добавить статью</a>
13     <div id="articles-list"></div>
14     <a href="/">Назад на главную</a>
15   </div>
16
17   <script>
18     // получаем список статей с сервера
19     fetch('/articles/list')
20       .then(response => response.json())
21       .then(articles => {
22         const articlesList = document.getElementById('articles-list');
23         articles.forEach(article => {
24           const articleDiv = document.createElement('div');
25           articleDiv.classList.add('card', 'mb-3');
26           articleDiv.innerHTML = `
27             <div class="card-body">
28               <h5 class="card-title">${article.title}</h5>
29               <p class="card-text">${article.content}</p>
30             </div>
31           `;
32           articlesList.appendChild(articleDiv);
33         });
34       });
35   </script>
36 </body>
37 </html>
38
```

```
File Edit Selection View Go Run Terminal Help
JS server.js index.html login.html articles.html register.html X add-article.html Workspace Trust
C:\Users\Компик > blog-app > views > register.html > ...
1 <!DOCTYPE html>
2 <html lang="ru">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
7   <title>Регистрация</title>
8 </head>
9 <body>
10   <div class="container">
11     <h1 class="mt-5">Регистрация</h1>
12     <form action="/register" method="POST">
13       <div class="form-group">
14         <label for="email">Email:</label>
15         <input type="email" class="form-control" id="email" name="email" required>
16       </div>
17       <div class="form-group">
18         <label for="password">Пароль:</label>
19         <input type="password" class="form-control" id="password" name="password" required>
20       </div>
21       <button type="submit" class="btn btn-primary">Зарегистрироваться</button>
22     </form>
23     <a href="/">Назад на главную</a>
24   </div>
25 </body>
26 </html>
27
28
```

```
File Edit Selection View Go Run Terminal Help
JS server.js index.html login.html articles.html register.html add-article.html X Workspace Trust
C:\Users\Компик > blog-app > views > add-article.html > html > body > div.container > form > button.btn.btn-primary
1 <!DOCTYPE html>
2 <html lang="ru">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
7   <title>Добавить статью</title>
8 </head>
9 <body>
10   <div class="container">
11     <h1 class="mt-5">Добавить статью</h1>
12     <form action="/add-article" method="POST">
13       <div class="form-group">
14         <label for="title">Заголовок:</label>
15         <input type="text" class="form-control" id="title" name="title" required>
16       </div>
17       <div class="form-group">
18         <label for="content">Содержание:</label>
19         <textarea class="form-control" id="content" name="content" required></textarea>
20       </div>
21       <button type="submit" class="btn btn-primary">Добавить статью</button>
22     </form>
23   </div>
24 </body>
25 </html>
26
```







[Добавить статью](#)

Заголовок:

Содержание:

[Добавить статью](#)



localhost:3000

# Список статей

[Добавить статью](#)

**Вауууу новый пост от кiryshi**

dsadasda

[Назад на главную](#)