

Министерство образования и науки  
федеральное государственное автономное образовательное учреждение  
высшего образования

«Национальный исследовательский университет ИТМО»

Факультет инфокоммуникационных технологий

Отчет по дисциплине: «Web-программирование»

## Практическая работа 2

Выполнила: Полтавец  
Елена Андреевна

Группа: К3322

Проверила: Марченко  
Елена Вадимовна

Санкт-Петербург

2024

**Цель:** научиться основам работы с Git, изучить практические аспекты использования gulp и написать программу клиент согласно заданию.

### Задачи:

- Ознакомиться с особенностями работы с Git и GitHub, научиться работать с репозиторием и файлами;
- Установить gulp и создать task;
- Написать программу клиент, которая показывает web-страницы одна за другой из списка с возможностью задания адресов страниц и интервалом показа.

## Ход работы

### Задание 1

Ссылка: [https://github.com/Malenago/WebDevelopment\\_2024-2025](https://github.com/Malenago/WebDevelopment_2024-2025)

В начале была осуществлена проверка на установку Git, далее настроены имя и адрес электронной почты для привязки и подписки действий, после был клонирован репозиторий с Практической работы 1 (Рисунок 1).

```
[drachioaelena@MacBook-Air-Elena-2 ~ % git --version
git version 2.39.3 (Apple Git-146)
[drachioaelena@MacBook-Air-Elena-2 ~ % git config --global user.name "Malenago"
[drachioaelena@MacBook-Air-Elena-2 ~ % git config --global user.email Malenago.kay@yandex.ru
[drachioaelena@MacBook-Air-Elena-2 ~ % cd Desktop
[drachioaelena@MacBook-Air-Elena-2 Desktop % md Git
[zsh: command not found: md
[drachioaelena@MacBook-Air-Elena-2 Desktop % mkdir Git
[drachioaelena@MacBook-Air-Elena-2 Desktop % cd Git
[drachioaelena@MacBook-Air-Elena-2 Git % git clone https://github.com/Malenago/WebDevelopment_2024-2025
Cloning into 'WebDevelopment_2024-2025'...
remote: Enumerating objects: 111, done.
remote: Counting objects: 100% (109/109), done.
remote: Compressing objects: 100% (81/81), done.
remote: Total 111 (delta 29), reused 87 (delta 23), pack-reused 2 (from 1)
Receiving objects: 100% (111/111), 24.75 MiB | 4.62 MiB/s, done.
Resolving deltas: 100% (29/29), done.
[drachioaelena@MacBook-Air-Elena-2 Git % █
```

Рисунок 1 – Вход и клонирование репозитория

Была переключена ветка на lab\_1, проверены все возможные ветки (Рисунок 2).

```
drachioaelena@MacBook-Air-Elena-2 WebDevelopment_2024-2025 % git checkout lab_1
branch 'lab_1' set up to track 'origin/lab_1'.
Switched to a new branch 'lab_1'
drachioaelena@MacBook-Air-Elena-2 WebDevelopment_2024-2025 % git branch
* lab_1
  main
```

## Рисунок 2 – Работа с ветками

После этого были изменены файлы и созданы коммиты (Рисунок 3).

```
drachioaelena@MacBook-Air-Elena-2 WebDevelopment_2024-2025 % git add works/K3322/Полтавец_Елена/lab_1/ex1.html
drachioaelena@MacBook-Air-Elena-2 WebDevelopment_2024-2025 % git commit -m "Commit 2"
[lab_1 c87967a] Commit 2
1 file changed, 5 insertions(+), 2 deletions(-)
drachioaelena@MacBook-Air-Elena-2 WebDevelopment_2024-2025 % git add works/K3322/Полтавец_Елена/lab_1/ex5.html
drachioaelena@MacBook-Air-Elena-2 WebDevelopment_2024-2025 % git commit -m "Commit 3"
[lab_1 82fe496] Commit 3
1 file changed, 1 insertion(+), 1 deletion(-)

drachioaelena@MacBook-Air-Elena-2 WebDevelopment_2024-2025 % git log
commit 53680a402579b463ea38b14788a595fb7cc11bed (HEAD -> lab_1)
Author: Malenago <Malenago.kay@yandex.ru>
Date: Wed Sep 18 10:47:37 2024 +0300

    Commit 1

commit 82fe496818045afde5d92442ea2465b01b2f403f
Author: Malenago <Malenago.kay@yandex.ru>
Date: Wed Sep 18 10:44:09 2024 +0300

    Commit 3

commit c87967a9530e3d624767b4da07715989b8b844b2
Author: Malenago <Malenago.kay@yandex.ru>
Date: Wed Sep 18 10:43:21 2024 +0300

    Commit 2
```

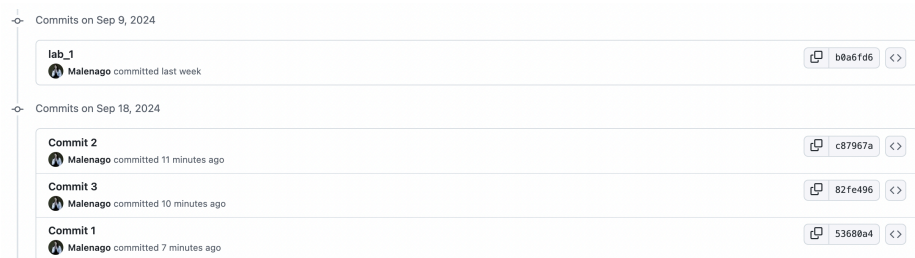
## Рисунок 3 – Работа с коммитами

Далее изменения были запущены на GitHub (Рисунок 4).

```
drachioaelena@MacBook-Air-Elena-2 WebDevelopment_2024-2025 % git push origin lab_1
Username for 'https://github.com': Malenago
Password for 'https://Malenago@github.com':
Enumerating objects: 29, done.
Counting objects: 100% (29/29), done.
Delta compression using up to 8 threads
Compressing objects: 100% (21/21), done.
Writing objects: 100% (21/21), 1.70 KiB | 1.70 MiB/s, done.
Total 21 (delta 12), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (12/12), completed with 6 local objects.
To https://github.com:Malenago/WebDevelopment_2024-2025
   b0a6fd6..53680a4  lab_1 -> lab_1
```

## Рисунок 4 – Пуш изменений

Проверив на GitHub, нашлись все выполненные коммиты (Рисунок 5).



## Рисунок 5 – Выполненные коммиты

## Задание 2

Документация: <https://webdesign-master.ru/blog/tools/gulp-beginners.html>

Установка включала в себя следующие этапы:

1. Для работы с с Gulp был установлен Node.js: `brew install node`;
2. После был установлен Gulp (Рисунок 6): `npm i gulp -g` (запускается менеджер пакетов npm (Node Package Manager), который командой `install` устанавливает Gulp в систему. Ключ `-g` говорит о том, что пакет установится в систему глобально, то-есть в систему, а не в папку проекта. Без ключа `-g` пакет устанавливаются в ту папку, в которой выполняются текущие команды).

```
[drachioaelena@MacBook-Air-Elena-2 ~ % npm i gulp -g
added 143 packages in 17s

14 packages are looking for funding
  run `npm fund` for details
npm notice
npm notice New patch version of npm available! 10.8.2 -> 10.8.3
npm notice Changelog: https://github.com/npm/cli/releases/tag/v10.8.3
npm notice To update run: npm install -g npm@10.8.3
npm notice
```

Рисунок 6 – Установка Gulp

Далее был создан базовый task:

1. Была выполнена инициализация проекта, заполнена метаданная о проекте (Рисунок 7);

```
drachioaelena@MacBook-Air-Elena-2 ~ % cd Desktop/gulp
drachioaelena@MacBook-Air-Elena-2 gulp % npm init
This utility will walk you through creating a package.json file.
It only covers the most common items, and tries to guess sensible defaults.

See `npm help init` for definitive documentation on these fields
and exactly what they do.

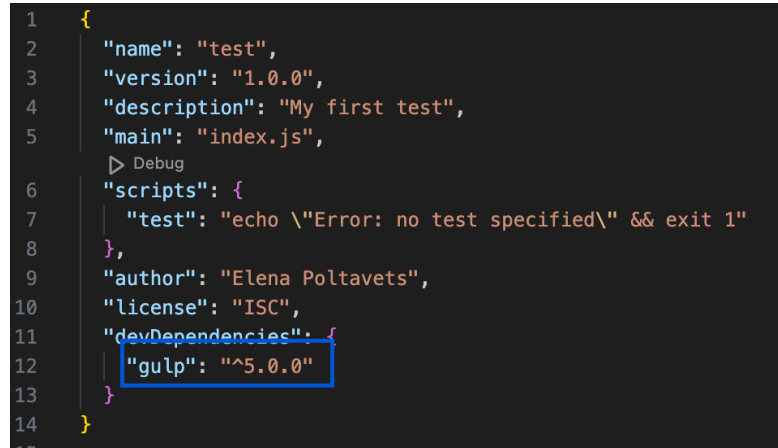
Use `npm install <pkg>` afterwards to install a package and
save it as a dependency in the package.json file.

Press ^C at any time to quit.
package name: (gulp) MyProject
Sorry, name can no longer contain capital letters.
package name: (gulp) test
version: (1.0.0)
description: My first test
entry point: (index.js)
test command:
git repository:
keywords:
author: Elena Poltavets
license: (ISC)
About to write to /Users/drachioaelena/Desktop/gulp/package.json:
{
  "name": "test",
  "version": "1.0.0",
  "description": "My first test",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "Elena Poltavets",
  "license": "ISC"
}

Is this OK? (yes) yes
```

Рисунок 7 – Инициализация проекта

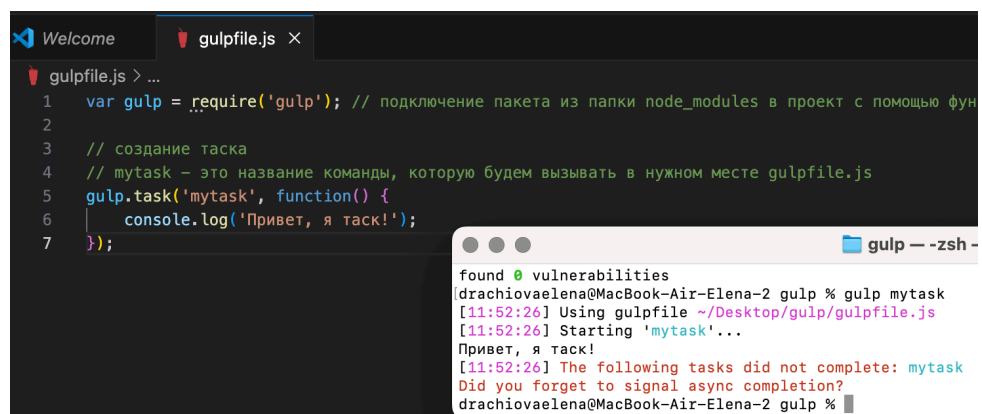
2. Далее в проект был установлен Gulp (Рисунок 8): `npm i gulp --save-dev`. npm устанавливает пакет gulp в текущую папку (потому, что нет ключа -g, устанавливающий пакет глобально в систему) и сохраняет название пакета с версией в файл `package.json`;



```
1 {
2   "name": "test",
3   "version": "1.0.0",
4   "description": "My first test",
5   "main": "index.js",
6   "scripts": {
7     "test": "echo \\\"Error: no test specified\\\" && exit 1"
8   },
9   "author": "Elena Poltavets",
10  "license": "ISC",
11  "devDependencies": {
12    "gulp": "^5.0.0"
13  }
14 }
```

Рисунок 8 – Установка gulp в проект

3. Далее был создан простой task (Рисунок 9);



```
1 var gulp = require('gulp'); // подключение пакета из папки node_modules в проект с помощью фун
2
3 // создание таска
4 // mytask - это название команды, которую будем вызывать в нужном месте gulpfile.js
5 gulp.task('mytask', function() {
6   console.log('Привет, я таск!');
7 });
```

Terminal output:

```
found 0 vulnerabilities
drachiovaelena@MacBook-Air-Elena-2 gulp % gulp mytask
[11:52:26] Using gulpfile ~/Desktop/gulp/gulpfile.js
[11:52:26] Starting 'mytask'...
Привет, я таск!
[11:52:26] The following tasks did not complete: mytask
Did you forget to signal async completion?
drachiovaelena@MacBook-Air-Elena-2 gulp %
```

Рисунок 9 – Простой таск и вывод в консоль

Далее был написан task для автоматического обновления страниц с использованием Bbrowser Sync.

Browser Sync – это отличное решение для LiveReload страниц при сохранении файлов. При чем релоад происходит не только в одном браузере, но и во всех браузерах сети, будь это мобильные устройства или другие компьютеры в одной Wi-Fi сети.

При выполнении в терминале команды `gulp` в консоли появляется подробная информация, а также добавляется файл `css`, стили которого применяются на страницу `html` (Рисунок 10).

```
[drachioaelena@MacBook-Air-Elena-2 gulp % gulp
[12:35:08] Using gulpfile ~/Desktop/gulp/gulpfile.js
[12:35:08] Starting 'default'...
[12:35:08] Starting 'sass'...
[12:35:08] Starting 'browser-sync'...
[12:35:08] Starting 'watch'...
[12:35:08] Finished 'sass' after 29 ms
[12:35:08] Finished 'browser-sync' after 29 ms
[12:35:08] Finished 'watch' after 29 ms
[12:35:08] Finished 'default' after 30 ms
Deprecation Warning: The legacy JS API is deprecated and will
Sass 2.0.0.

[Browsersync] Access URLs:
-----
    Local: http://localhost:3000
  External: http://192.168.0.101:3000
-----
    UI: http://localhost:3001
  UI External: http://192.168.0.101:3001
-----
[Browsersync] Serving files from: app
```

Рисунок 10 – Запуск командой

Для того, чтобы страница обновлялась при сохранении `html`, был добавлен дополнительный код, здесь была использована функция `browserSync.reload`, которую предоставил пакет `Browser Sync` (Рисунок 11).

```
[drachioaelena@MacBook-Air-Elena-2 gulp % gulp
[12:41:11] Using gulpfile ~/Desktop/gulp/gulpfile.js
[12:41:11] Starting 'default'...
[12:41:11] Starting 'sass'...
[12:41:11] Starting 'browser-sync'...
[12:41:11] Starting 'watch'...
[12:41:12] Finished 'sass' after 29 ms
[12:41:12] Finished 'browser-sync' after 29 ms
[12:41:12] Finished 'watch' after 29 ms
[12:41:12] Finished 'default' after 31 ms
Deprecation Warning: The legacy JS API is deprecated and will
ss 2.0.0.

[Browsersync] Access URLs:
-----
    Local: http://localhost:3002
  External: http://192.168.0.101:3002
-----
    UI: http://localhost:3003
  UI External: http://192.168.0.101:3003
-----
[Browsersync] Serving files from: app
[12:41:54] Starting 'code'...
[12:41:54] Finished 'code' after 22 ms
[Browsersync] 1 file changed (index.html)
[Browsersync] Reloading Browsers...
```

Рисунок 11 – Обновления в терминале после изменения `html`

### Задание 3

В рамках задания был написан сайт на `html`, `css`, `js`. В `js` реализована логика и передача информации о ссылках и интервалах, которая хранится в массиве, а затем используется для передачи на `iframe` и вывода. Показ

осуществляется в `iframe`, поэтому не все сайты можно через него открыть. Одной из распространенных причин является правило безопасности браузера X-Frame-Options, которое запрещает загрузку содержимого из определенных источников.

Пример работы с сайтом приведен на Рисунке 12.

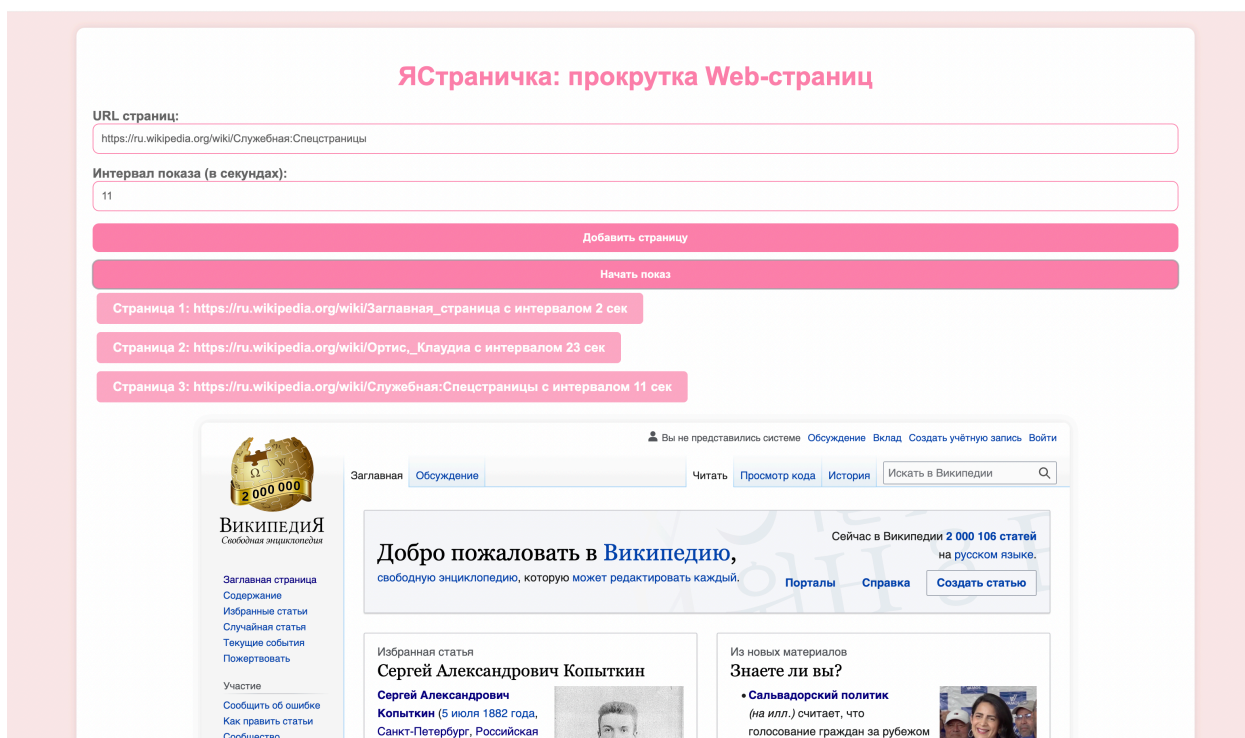


Рисунок 12 – Сайт для показа страниц по ссылкам и интервалам

**Вывод:** были изучены основы работы с Git и Gulp, была написана программа клиент согласно заданию.