

Лабораторная работа №16 (2 ч.)

Публикация проекта на GitHub Pages

Цель работы: научиться публиковать веб-проект с помощью сервиса GitHub Pages, освоить базовые операции с репозиторием на GitHub (создание, загрузка файлов, настройка источника публикации), а также применить практики работы с ветками и историей изменений. Способствовать формированию соответствующих общих компетенций (ОК 01, ОК 02, ОК 03, ОК 04) и профессиональных компетенций (ПК 9.7, ПК 9.9, ПК 9.10).

Средства выполнения работы:

- аппаратные: ПК;
- СЭДО СВФУ (Moodle) - Портал электронного обучения СВФУ 09.02.07 Информационные системы и программирование, Квалификация: Разработчик веб и мультимедийных приложений МДК.09.02 Оптимизация веб-приложений;
- Редакторы кода/среды для веб-разработки (прописать свой, например VSCode);
- Браузер (прописать свой);

I. Теоретическая часть

1. [Документация GitHub](#)
2. Материалы из лекции по теме лабораторной работы.

Что такое GitHub Pages

- Бесплатный хостинг статических сайтов от GitHub.
- Поддерживает HTML, CSS, JavaScript.
- Автоматически публикует содержимое указанной ветки (обычно main или gh-pages).

Основные понятия Git и GitHub

- Репозиторий (repository) — хранилище проекта со всей историей изменений.
- Коммит (commit) — фиксация изменений с описанием.
- Ветка (branch) — параллельная версия кода для разработки без влияния на основную (main) ветку.
- Pull request — запрос на слияние изменений из одной ветки в другую.
- GitHub Pages settings — настройка источника (branch + folder) для публикации.

Процесс публикации сайта на GitHub Pages

- Создание репозитория.
- Загрузка файлов проекта.
- Настройка публикации через Settings → Pages.
- Ожидание сборки и проверка работоспособности по автоматически сгенерированной ссылке.

Преимущества использования GitHub Pages в образовательных и демонстрационных целях

- Бесплатно, просто, интегрировано с Git.
- Позволяет быстро делиться работами, портфолио, учебными проектами.

II. Правила выполнения заданий:

1. Ознакомиться с теоретическим материалом по теме лабораторной работы (*источники с блока теор. части, пояснения преподавателя, материал с лекций*).
2. Выполнить задания из раздела IV. Задания.
3. Оформить отчет. Отчет должен содержать:
 - 1) Титульный лист;
 - 2) Цель лабораторной работы;
 - 3) Задачи;
 - 4) Средства для выполнения работы (аппаратные/программные);
 - 5) Теоретическую сводку (ответы на контр. вопросы (если есть), основные понятия, применяемые в работе/при наличии теор. блок);
 - 6) Описание хода работы со скриншотами, **ссылка на опубликованный сайт и репозиторий**;

- 7) Результаты проделанной работы;
 - 8) Выводы по проделанной работе.
4. В СЭДО загрузить текстовый файл отчета.

III. Задание

- 1) Опубликовать созданный в лабораторной работе №15 сайт, используя GitHub Pages.
- 2) Редактировать сайт-пример из прикрепления в СДО в побочной ветке, посмотреть историю версий, провести слияние кода из побочной ветки в основную.

Ход работы:

Задание 1. Публикация сайта на GitHub Pages

1. Пройдите регистрацию на GitHub (если еще нет аккаунта).

Следуйте инструкции из офиц. документации GitHub:

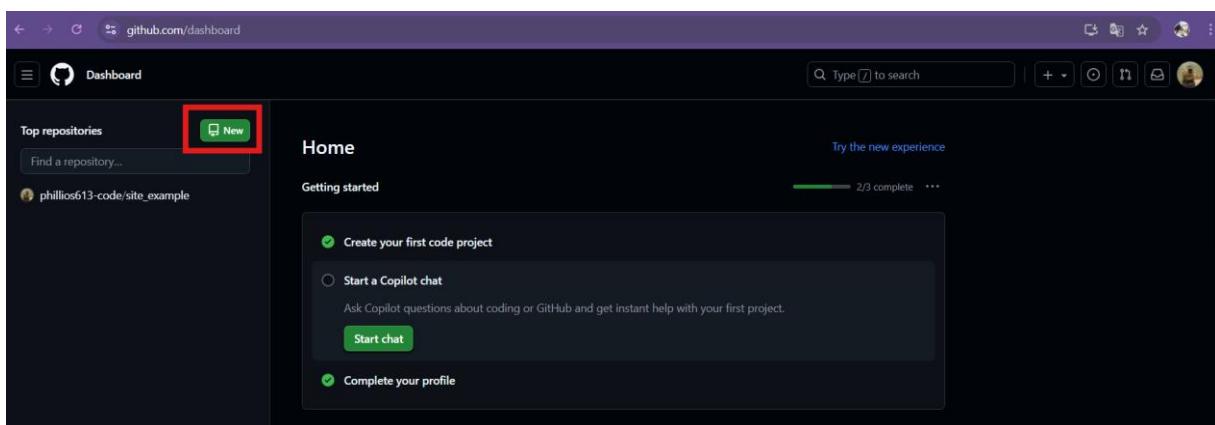
<https://docs.github.com/ru/get-started/start-your-journey/creating-an-account-on-github#signing-up-for-a-new-personal-account>

*Официальная документация на английском языке, пользуйтесь встроенным переводчиком в браузере.
Дополнительно: для начального ознакомления можете пройти урок/гайд Hello world!*

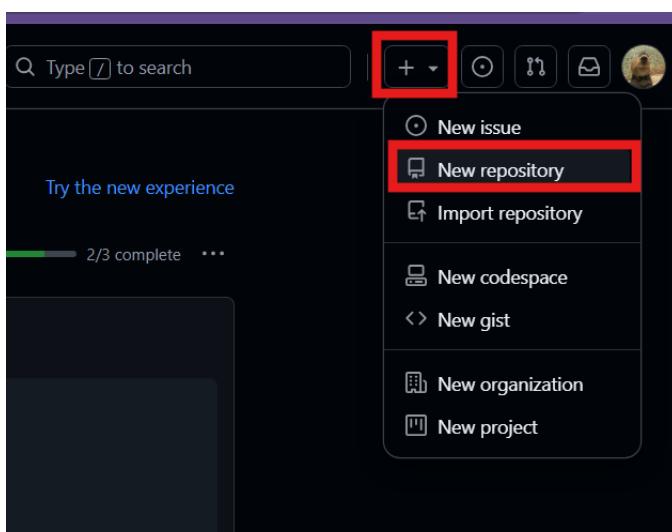
<https://docs.github.com/ru/get-started/start-your-journey/hello-world> - ссылка на урок

2. Создайте новый репозиторий с именем (сайта), например **my-shopy**.

Создать репозиторий можно, кликнув на кнопку “New”, в боковой панели в окне главной страницы,



или кликнув на плюсик вверху, выбрать пункт в меню “new repository”.



Заполните данные (название репозитория, краткое описание, настройки приватности, наличие специальных файлов в создаваемом репозитории). После внесения данных нажмите «Create repository».

Create a new repository

Repositories contain a project's files and version history. Have a project elsewhere? [Import a repository](#).

Required fields are marked with an asterisk (*).

1 General

Owner * Repository name *

phillios613-code / git-site-example
git-site-example is available.

Great repository names are short and memorable. How about [glowing-waddle](#)?

Description

Репозиторий для учебных целей

29 / 350 characters

2 Configuration

Choose visibility * Public

Choose who can see and commit to this repository

Add README Off

READMEs can be used as longer descriptions. [About READMEs](#)

Add .gitignore No .gitignore

.gitignore tells git which files not to track. [About ignoring files](#)

Add license No license

Licenses explain how others can use your code. [About licenses](#)

Create repository

3. Реализуйте добавление файлов проекта на созданный репозиторий в GitHub. Для этого нажмите выделенную ссылку.

phillios613-code / git-site-example

Type (I) to search

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

git-site-example Public

Pin Watch 0 Fork 0 Star 0

Set up GitHub Copilot Use GitHub's AI pair programmer to autocomplete suggestions as you code. Get started with GitHub Copilot

Add collaborators to this repository Search for people using their GitHub username or email address. Invite collaborators

Quick setup — if you've done this kind of thing before Set up in Desktop or HTTPS SSH https://github.com/phillios613-code/git-site-example.git Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line

```
echo "# git-site-example" >> README.md
git init
git add README.md
git commit -m "First commit"
git branch -M main
git remote add origin https://github.com/phillios613-code/git-site-example.git
git push -u origin main
```

Выберите файлы проекта сайта на ПК (медиа сайта, файлы html, css и js). Путем перетаскивания файлов в окошко или выбора “choose your files”.

git-site-example /

Drag files here to add them to your repository Or choose your files

Commit changes

Add files via upload

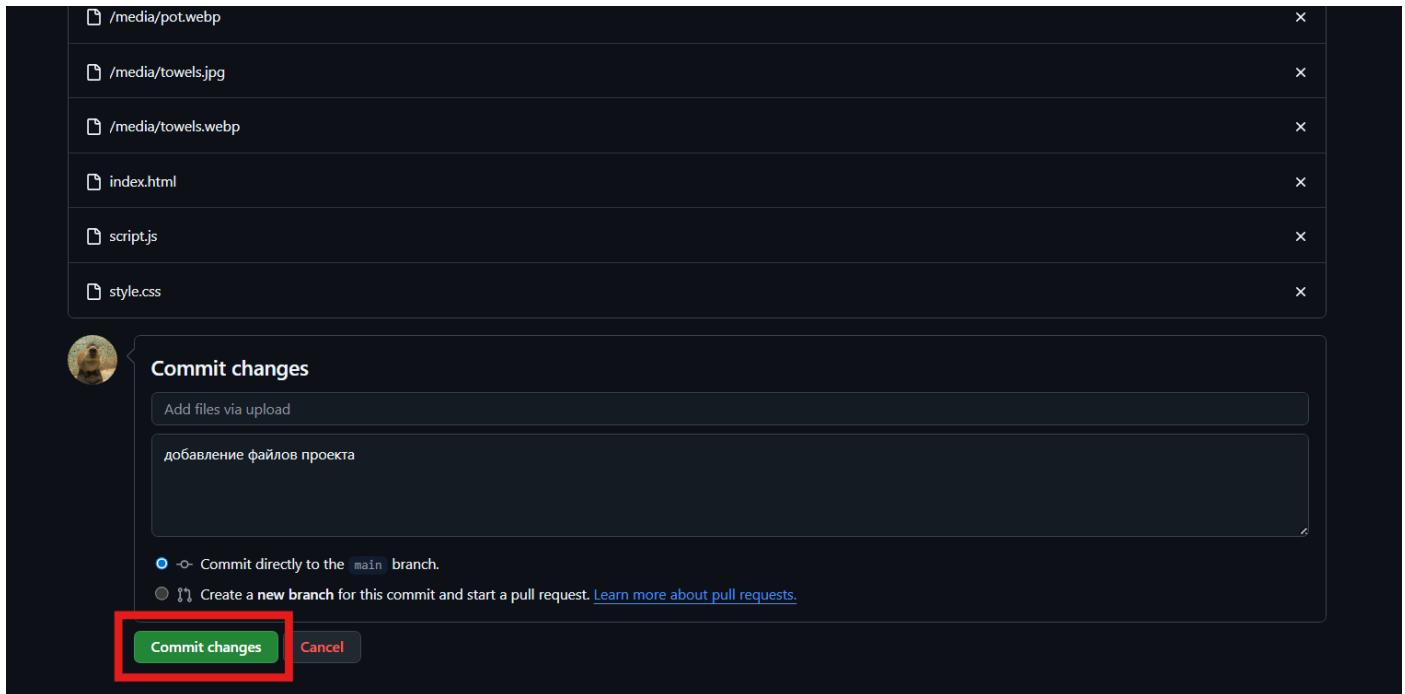
Add an optional extended description...

Commit directly to the main branch.

Create a new branch for this commit and start a pull request. [Learn more about pull requests](#)

Commit changes **Cancel**

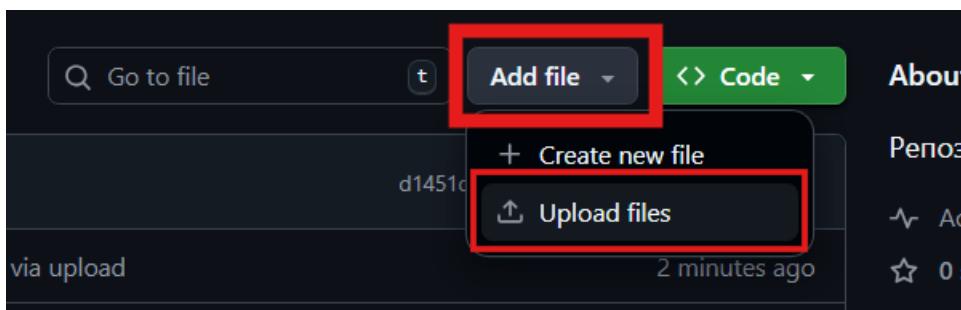
Целые папки можно добавить путем перетаскивания (автоматически будет создана отдельная директория). Когда файлы прогрузятся, можете внести описание в вводимые изменения в репозитории. Далее, делаете подтверждение изменений «Commit changes»



Как будет выглядеть репозиторий после добавления файлов:

A screenshot of a GitHub repository page for 'git-site-example'. The repository is public and has 9 commits. The commits list shows five files added via upload: 'media', 'index.html', 'script.js', and 'style.css', all added 9 minutes ago. Below the commits, there's a 'README' section with a 'Add a README' button. On the right side of the page, there are sections for 'About', 'Releases', 'Packages', and 'Languages', along with a 'Suggested workflows' sidebar.

Если нужно добавить что-то дополнительно нажимаете Add File → Upload files



4. Для публикации сайта в настройках репозитория проделайте следующее.

Перейдите в настройки репозитория “Settings”.

The screenshot shows a GitHub repository named "git-site-example" by user "phillios613-code". The repository is public and has 1 branch and 0 tags. The main branch is selected. Recent commits include "Add files via upload" by "phillios613-code" (d1451de) 5 minutes ago, and three other commits from the same user at different times: "media" (5 minutes ago), "index.html" (13 minutes ago), and "script.js" (13 minutes ago). The "Settings" tab is highlighted with a red box.

Найдите в списке раздел Pages.

The screenshot shows the "GitHub Pages" settings section within the repository's "Settings" tab. On the left, there is a sidebar with various repository settings like General, Access, Collaborators, etc. The "Pages" section is highlighted with a red box. The main area shows the "GitHub Pages" introduction, the "Build and deployment" section with a dropdown for "Source" set to "Deploy from a branch", and the "Visibility" section for GitHub Enterprise users. A "Start free for 30 days" button is visible.

Выберите разветвление - источник откуда GitHub Pages будет брать данные о публикуемом сайте.

The screenshot shows the "Select branch" modal window. It contains a search bar with the text "main" and a list of branches. The "main" branch is highlighted with a red box. Below the list is a "Save" button.

Если у вас html-файл сайта находится не в корневой папке /root, а внутри какой-то директории, выберите свою директорию, если же в корневой – то оставьте как есть.

Branch

GitHub Pages is currently disabled. Select a source below to enable GitHub Pages for this repository. [Learn more about configuring the publishing source for your site.](#)

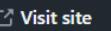
 main ▾  / (root) ▾ 

Сохраните “Save”.

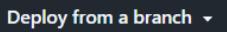
Через некоторое время сайт будет доступен по адресу (нужно подождать 2-3 минуты, обновить страницу).

GitHub Pages

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.

Your site is live at <https://phillios613-code.github.io/git-site-example/>  Visit site  Unpublish site
Last deployed by  phillios613-code 2 minutes ago

Build and deployment

Source  Deploy from a branch ▾

Branch
Your GitHub Pages site is currently being built from the `main` branch. [Learn more about configuring the publishing source for your site.](#)

 main ▾  / (root) ▾ 

Learn how to [add a Jekyll theme](#) to your site.

Your site was last deployed to the `github-pages` environment by the [pages build and deployment](#) workflow.
[Learn more about deploying to GitHub Pages using custom workflows](#)

Перейдите по ссылке, проверьте сайт.

Задание 2. Редактирование сайта и работа в альтернативной ветке

Создайте новый репозиторий. Туда поместите файлы из прикреплённой папки в СДО.

Создайте побочную ветку на основе главной (main). Во вкладке “< >Code”

The screenshot shows a GitHub repository page for 'git-site-example'. A red box highlights the dropdown menu at the top left, which is set to 'main'. An arrow points from the text '1. Переключение веток' (1. Switch branches) to this menu. Another red box highlights the search bar in the 'Switch branches/tags' modal, containing the text 'vetka-example'. An arrow points from the text '2. Имя новой ветки' (2. Name of the new branch) to this search bar. A third red box highlights the button 'Create branch vetka-example from main' in the modal. An arrow points from the text '3. Создание новой ветки на основе основной ветки main' (3. Creation of a new branch based on the main branch) to this button.

После совершенных действий появится новая ветка – копия от основной, где можно спокойно редактировать файлы, создавать, тестировать новую функциональность, не боясь что это затронет основной код.

Убедитесь, что у вас работа переключена на побочную ветку.

The screenshot shows the same GitHub repository page after the new branch was created. The dropdown menu now shows 'vetka-example'. A red box highlights this dropdown, with an arrow pointing from the text 'This branch is up to date with main.' The sidebar on the right shows the repository details: 'About' (Repository for learning purposes), 'Activity' (0 stars, 0 forks, 0 watching), 'Releases' (No releases published), 'Packages' (No packages published), and 'Deployments' (4 deployments, last updated 2 hours ago). The main content area shows the files 'media', 'index.html', 'script.js', and 'style.css'.

В самом GitHub встроен свой IDE (среда разработки по типу VScode но чуть попримитивнее), если нажать на файл с кодом, например index.html, отобразится свой редактор кода GitHub:

The screenshot shows a GitHub repository interface. On the left, there's a sidebar titled "Files" with a tree view of the project structure: "media", "index.html", "script.js", and "style.css". The "index.html" file is currently selected and shown in the main content area. The code editor shows the following HTML:

```
<!DOCTYPE html>
<html lang="ru">
<head>
<meta charset="UTF-8" />
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
<title>Домашний Магазин</title>
<meta name="description" content="Интернет-магазин полезных товаров для дома. Быстро, удобно, недорого." />
<link rel="stylesheet" href="style.css" />
</head>
<body>
<header>
<h1>Домашний Магазин</h1>
<nav>
<ul>
<li><a href="#catalog">Каталог</a></li>
<li><a href="#about">О нас</a></li>
<li><a href="#contacts">Контакты</a></li>
<li>
<button id="cart-btn" class="cart-icon" aria-label="Корзина">Корзина</button>
0
</li>
</ul>
</nav>
</header>
<main>
<section id="catalog">
```

Просмотреть элементы проекта, переключаться между файлами можно через боковую прилегающую панель.

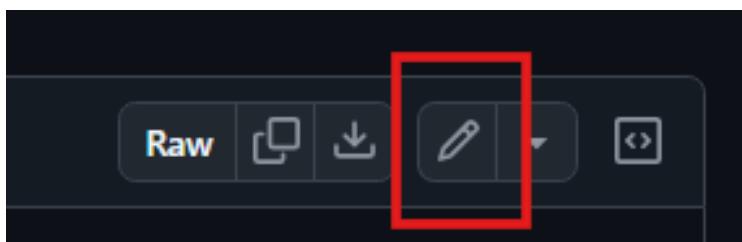
The screenshot shows a GitHub repository interface. On the left, there's a sidebar titled "Files" with a tree view of the project structure: "media", "index.html", "script.js", and "style.css". The "style.css" file is currently selected and shown in the main content area. The code editor shows the following CSS:

```
* {
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}

body {
```

Переключитесь на файл style.css. **Отредактируйте цвета кнопок и фона карточек товара на оттенки сиреневого.**

Для редактирования кода, нужно включить режим редактирования путем нажатия на иконку с карандашом.



После внесения изменений нужно их обязательно подтвердить «Commit changes». **Сделайте коммит.**

git-site-example / style.css in vetka-example

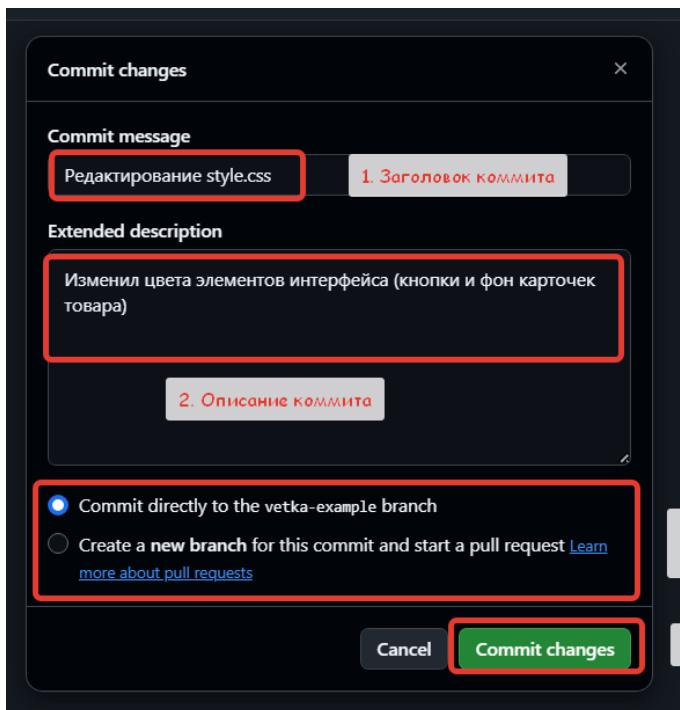
Cancel changes Commit changes...

Spaces 2 No wrap

Edit Preview

```
128 .object-title {  
129   border-radius: 6px;  
130   margin-bottom: 1rem;  
131 }  
132  
133 .price {  
134   font-size: 1.2rem;  
135   color: #e74c3c;  
136   font-weight: bold;  
137   margin: 0.6rem 0;  
138 }  
139  
140 button {  
141   background-color: #d747ff;  
142   color: white;  
143   border: none;  
144   padding: 0.5rem 1rem;  
145   border-radius: 4px;  
146   cursor: pointer;  
147   font-size: 1rem;  
148 }  
149  
150 button:hover {  
151   background-color: #3a5f47;  
152 }  
153  
154 /* Модальное окно корзины */  
155 .modal {  
156   position: fixed;  
157   top: 0;  
158   left: 0;  
159 }
```

В окне описания совершаемого коммита можете внести данные (о чём коммит, как называется и куда коммитить). Будет полезно потом просматривать/прослеживать в истории версий/изменений какое было изменение применено и зачем – как для вас, так и для других разработчиков.



Просмотрите историю версий/коммитов.

Историю версий можно посмотреть тут:

A screenshot of the GitHub code editor interface. At the top, it shows the repository 'phillios613-code' and the file 'style.css'. The status bar indicates '07a0fa9 · now'. A red box highlights the 'History' button in the top right corner. Below the header, there are tabs for 'Code' and 'Blame', with 'Code' selected. The code editor displays the following CSS:

```
1 * {  
2     margin: 0;  
3     padding: 0;  
4     box-sizing: border-box;  
5 }  
6  
7 body {  
8     font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;  
9     line-height: 1.6;  
10    color: #333;  
11    background-color: #fafafa;
```

Там будет отображаться история всех совершенных действий:

A screenshot of the GitHub commit history for the 'style.css' file. The title 'Commits' is at the top. Below it, a search bar shows 'History for git-site-example / style.css on vетка-example'. There are two dropdown menus: 'All users' and 'All time'. The commit list starts with a collapsed section 'Commits on Nov 13, 2025'. The first item is a collapsed commit titled 'Редактирование style.css' with a green 'Verified' badge, commit hash '07a0fa9', and a copy icon. The second item is a collapsed commit titled 'Add files via upload' with a green 'Verified' badge, commit hash '777561a', and a copy icon. The third item is a collapsed commit titled 'Delete style.css' with a green 'Verified' badge, commit hash '5ff41a6', and a copy icon. The fourth item is a collapsed commit titled 'Add files via upload' with a green 'Verified' badge, commit hash '736bc64', and a copy icon. The fifth item is a collapsed commit titled 'Delete style.css' with a green 'Verified' badge, commit hash '3ce8d8f', and a copy icon. The sixth item is a collapsed commit titled 'Add files via upload' with a green 'Verified' badge, commit hash '700401f', and a copy icon. At the bottom of the list, there is a note '-o End of commit history for this file'.

Например, зайдем в запись последнего совершенного нами коммита:

A screenshot of the GitHub commit details for commit '07a0fa9'. The commit was authored by 'phillios613-code' 2 minutes ago and is verified. The commit message is 'Редактирование style.css' with the note 'Изменения цвета элементов интерфейса (кнопки и фон карточки товара)'. The commit has 1 parent, '777561a'. The commit summary shows '+1 file changed +4 -3 lines changed'. The file changes are shown in a diff view for 'style.css':

```
diff --git a/style.css b/style.css  
--- a/style.css  
+++ b/style.css  
@@ -114,7 +114,7 @@ footer {  
114 114 }  
115 115  
116 116 .product-card {  
117 - background: white;  
117 + background: #F5d1ff;  
118 118 border-radius: 8px;  
119 119 box-shadow: 0 2px 8px rgba(0,0,0,0.1);  
120 120 padding: 1.2rem;  
121 121 }  
@@ -138,7 +138,7 @@ footer {  
138 138 }  
139 139  
140 140 button {  
141 - background-color: #94c669;  
141 + background-color: #d747ff;  
142 142 color: white;  
143 143 border: none;  
144 144 padding: 0.5rem 1rem;
```

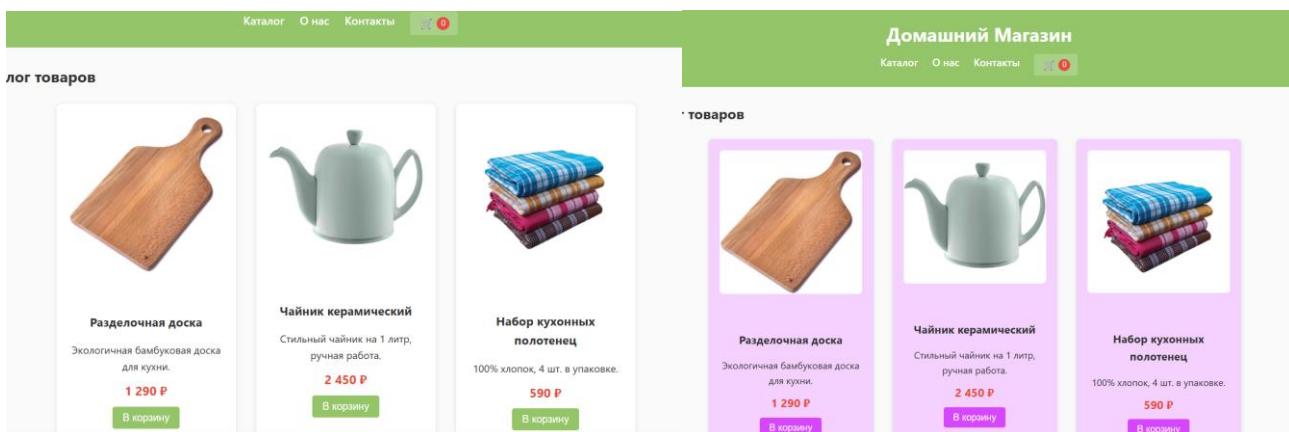
Просмотрите изменения на сайте.

Поскольку мы работаем на побочной ветке, нужно в настройках GitHub Pages, поменять ветку с которой берутся данные, вместо main укажите побочную, подтвердите Save

The screenshot shows the GitHub Pages settings page. On the left, there's a sidebar with various options like General, Access, Collaborators, Moderation options, Code and automation, and Pages (which is selected). The main area is titled "GitHub Pages" and contains a message about hosting personal or organization pages. Below that, there's a "Build and deployment" section with a "Source" dropdown set to "Deploy from a branch". A modal window is open, showing a "Select branch" dropdown with "vetka-example" selected (marked with a red box labeled 1). To the right of the dropdown is a "Save" button (marked with a red box labeled 2). At the bottom of the modal are "Save" and "Remove" buttons. A tooltip explains that the site is currently built from the main branch and provides links for configuring publishing source and custom workflows.

Далее, подождите некоторое время (2-3 минуты). Проверьте сайт по ссылке.

До/после:



Как видите изменения вступили в силу.

Проведите слияние кода из побочной ветки в основную, нажав «Compare&pull request», чтобы код из побочной ветки применился к основной.

The screenshot shows the GitHub repository page for 'git-site-example'. At the top, there's a header with links for Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. The repository name 'git-site-example' is shown with a public status. Below the header, there's a message about recent pushes and a 'Compare & pull request' button (highlighted with a red box). The repository stats show 2 branches and 0 tags. A message indicates the 'vetka-example' branch is 1 commit ahead of 'main'. The repository content list shows files like 'media', 'index.html', 'script.js', and 'style.css' with their last commit details. On the right side, there are sections for About (repository description), Releases (no releases published), Packages (no packages published), and Deployments (6 deployments, with one for 'github-pages' 4 minutes ago).

Перед слиянием можно проверить изменения на побочных ветках. Так же сам GitHub проверяет совместимость и возможность применения изменений от побочной ветки к основной. Если все устраивает и стоит плашка “Able to merge”, то вы можете подтвердить слияние.

The screenshot shows the GitHub interface for opening a pull request. At the top, it says "base: main" and "compare: vetka-example". A red box highlights the green checkmark icon followed by the text "Able to merge. These branches can be automatically merged.". Below this, there's a title field with a placeholder "Add a title" and a description field with a placeholder "Add a description". The description contains the text "Изменил цвета элементов интерфейса (кнопки и фон карточек товара)". On the right side, there are sections for "Reviewers" (No reviews), "Assignees" (No one—assign yourself), "Labels" (None yet), "Projects" (None yet), "Milestone" (No milestone), and "Development" (Use Closing keywords in the description to automatically close issues). At the bottom, there's a "Create pull request" button with a red border, and a note: "Remember, contributions to this repository should follow our GitHub Community Guidelines".

Все подтвердить (Merge pull request):

The screenshot shows the GitHub pull request details page for a PR titled "Редактирование style.css #1". It indicates that "phillios613-code wants to merge 1 commit into main from vetka-example". The PR has 0 comments, 1 commit, 3 checks, and 1 file changed. The commit message is "Изменил цвета элементов интерфейса (кнопки и фон карточек товара)". The PR is marked as "Verified" and "07a0fa9". Below the commit, it says "This branch was previously deployed" with 1 inactive deployment. The "Checks" section shows "All checks have passed" (3 successful checks) and "No conflicts with base branch" (Merging can be performed automatically). At the bottom, there's a "Merge pull request" button with a red border, and a note: "You can also merge this with the command line. View command line instructions." To the right, there are sections for "Reviewers" (No reviews), "Assignees" (No one—assign yourself), "Labels" (None yet), "Projects" (None yet), "Milestone" (No milestone), and "Development" (Successfully merging this pull request may close these issues). There are also "Notifications" and "Customize" buttons at the bottom right.

Подтвержденные изменения из побочной ветки применяются к основной:

Pull requests 1 Actions Projects Wiki Security Insights Settings

Редактирование style.css #1

Merged phillios613-code merged 1 commit into main from vetka-example now

Conversation 0 Commits 1 Checks 3 Files changed 1 +4 -3

phillios613-code commented now
Изменил цвета элементов интерфейса (кнопки и фон карточек товара)

phillios613-code Редактирование style.css ... Verified ✓ 07a0fa9
phillios613-code merged commit bf0448b into main now View details Revert
3 checks passed

This branch was previously deployed Show environments
1 inactive deployment

Pull request successfully merged and closed Delete branch
You're all set — the vetka-example branch can be safely deleted.

Add a comment Write Preview

Reviewers No reviews Still in progress? Convert to draft

Assignees No one — assign yourself

Labels None yet

Projects None yet

Milestone None

Development Successfully merging this pull request may close these issues.
None yet