

Министерство цифрового развития, связи и массовых коммуникаций РФ  
Уральский технический институт связи и информатики (филиал) ФГБОУ ВО  
"Сибирский государственный университет телекоммуникаций и  
информатики" в г. Екатеринбурге (УрТИСИ СибГУТИ)



Уральский технический  
институт связи  
и информатики

ОТЧЕТ  
По дисциплине «Сетевое программирование»  
Практическое занятие №1  
«Работа с системой управления версиями»

Выполнила: студентка группы ПЕ-216

Морос Е.Е.

Проверил: преподаватель

Бурумбаев Д.И.

## 1 Цель работы:

- 1.1 Научиться работать с системой управления версиями;
- 1.2 Закрепить знания по теме «Системы управления версиями».

## 2 Перечень оборудования:

- 2.1 Персональный компьютер;
- 2.2 Система управления версиями Git;
- 2.3 Visual Studio Code.

## 3.Ход работы:

3.1 Перед началом работы с Git необходимо создать папку с фамилией и номером группы, в которой будут храниться наши элементы. Послеэтого, необходимо создать следующую структуру.

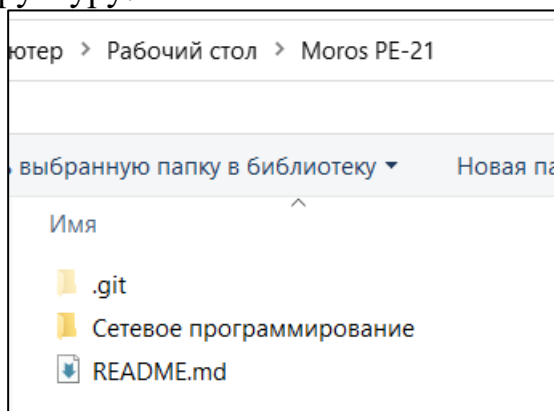


Рисунок 1 – Структура папки

Как видно из структуры, в корневой директории лежат:

1. Основная страница блога index.html
2. Папка img, в которой хранятся изображения (файлы\*.jpg) сайта. Тематика блога выбирается самостоятельно, картинки – тоже, но в рамках общественной нормы. Тогда структура файла index.html будет выглядеть как показано в листинге1.



Рисунок 2 – Внешний вид index.html

```
<!DOCTYPE html>
<html lang="ru">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Klik</title>
  <link href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" rel="stylesheet">
  <style>
```

```

body {
  background-color: #f7eacd;
  position: relative;
  min-height: 100vh;
  display: flex;
  flex-direction: column;
  justify-content: space-between;
  align-items: center;
}

.game-container {
  margin-top: 50px;
  text-align: center;
}

.score {
  font-size: 36px;
  margin-bottom: 20px;
}

.click-button {
  font-size: 24px;
  padding: 20px 40px;
}

.cat-image {
  width: 100px;
  height: 100px;
  transition: width 0.1s, height 0.1s;
}

.cat-container {
  position: fixed;
  bottom: 20px;
  width: 100%;
  text-align: center;
}
</style>
</head>
<body>
  <div class="container game-container">
    <h1>klik</h1>
    <div class="score" id="score">0</div>
    <button class="btn btn-primary click-button" id="clickButton">click on me!</button>
  </div>

  <div class="cat-container">
    
  </div>

  <script>
    let score = 0;
    const scoreDisplay = document.getElementById('score');
    const clickButton = document.getElementById('clickButton');
    const catImage = document.getElementById('catImage');

    clickButton.addEventListener('click', () => {
      score++;

```

```

        scoreDisplay.textContent = ' ' + score;
        let newWidth = catImage.width + 1;
        let newHeight = catImage.height + 1;

        catImage.style.width = newWidth + 'px';
        catImage.style.height = newHeight + 'px';
    });
</script>

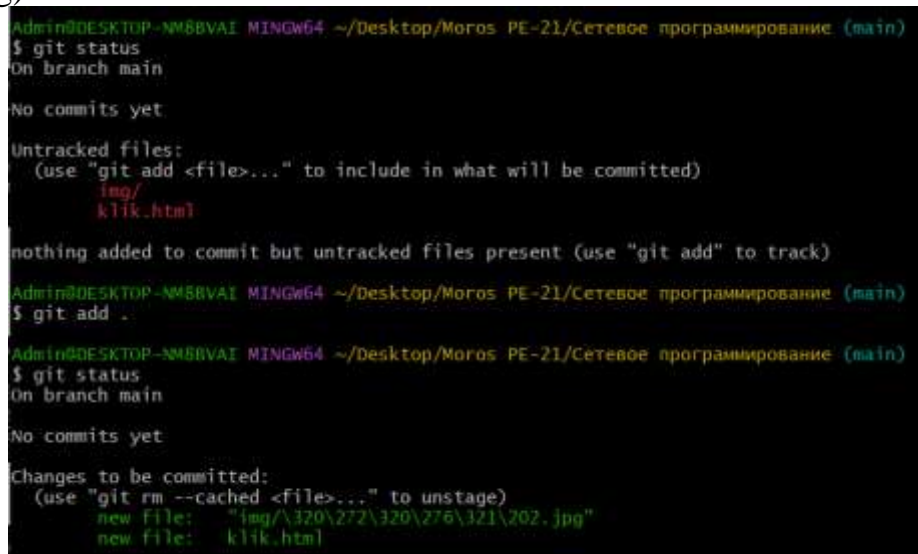
<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>
<script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.9.3/dist/umd/popper.min.js"></script>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
</body>
</html>

```

Листинг 1

Далее необходимо прописать команду git. Как видно из листинга, есть не отслеживаемые файлы. Как сделать отслеживаемыми – будет чуть-чуть попозже. Далее необходимо выполнить следующее задание – настроить пользователя Git на уровне локального репозитория:

- 1) Изучите содержимое файла конфигурации Git для текущего репозитория (.git/config)



```

Admin@DESKTOP-NM8BVAI MINGW64 ~/Desktop/Moros PE-21/Сетевое программирование (main)
$ git status
On branch main

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        img/
        klik.html

nothing added to commit but untracked files present (use "git add" to track)
Admin@DESKTOP-NM8BVAI MINGW64 ~/Desktop/Moros PE-21/Сетевое программирование (main)
$ git add .
Admin@DESKTOP-NM8BVAI MINGW64 ~/Desktop/Moros PE-21/Сетевое программирование (main)
$ git status
On branch main

No commits yet

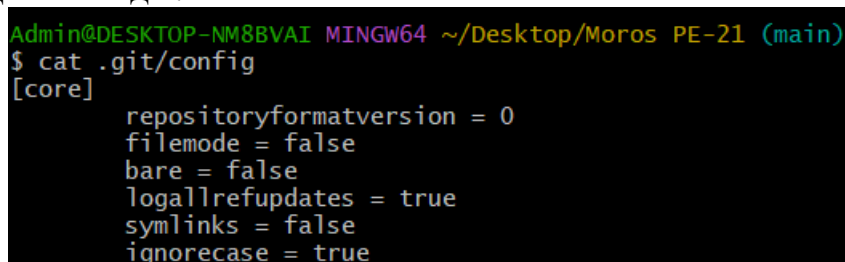
Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   img/\320\272\320\276\321\202.jpg"
        new file:   klik.html

```

Рисунок 3 - Проверка текущего состояния директории

- 2) Настройте имя и email пользователя для текущего репозитория
- 3) Убедитесь, что файл .git/config изменился соответствующим образом. Для вывода содержимого файла конфигурации Git для текущего репозитория необходимо воспользоваться командой

Далее необходимо задать имя и email пользователя для текущего репозитория при помощи команды.



```

Admin@DESKTOP-NM8BVAI MINGW64 ~/Desktop/Moros PE-21 (main)
$ cat .git/config
[core]
    repositoryformatversion = 0
    filemode = false
    bare = false
    logallrefupdates = true
    symlinks = false
    ignorecase = true

```

Рисунок 4 - Вывод содержимого файла конфигурации

```
Admin@DESKTOP-NM8BVAI MINGW64 ~/Desktop/Moros PE-21 (main)
$ git config --global user.email
moooros@yandex.ru

Admin@DESKTOP-NM8BVAI MINGW64 ~/Desktop/Moros PE-21 (main)
$ git config --global user.name
lisa
```

Рисунок 5 - Имя и email

```
Admin@DESKTOP-NM8BVAI MINGW64 ~/Desktop/Moros PE-21 (main)
$ git log
commit 138f4d279086a1c5cc63ae4369981edb5d32172a (HEAD -> main)
Author: lisa <moooros@yandex.ru>
Date: Fri Feb 21 19:08:09 2025 +0500
```

Рисунок 6 – проверка

Следующим заданием будет необходимо создать первый коммит. Для этого необходимо посмотреть состояние рабочей директории, как. После того, как было проверено текущее состояние директории, необходимо добавить все файлы в индекс при помощи команды `git add*`. После этого, снова необходимо проверить статус репозитория. Если все сделано, верно, то файлы попали в индекс и теперь готовы к коммиту

```
Admin@DESKTOP-NM8BVAI MINGW64 ~/Desktop/Moros PE-21/Сетевое программирование (main)
$ git add .

Admin@DESKTOP-NM8BVAI MINGW64 ~/Desktop/Moros PE-21/Сетевое программирование (main)
$ git status
On branch main

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   "img/\320\272\320\276\321\202.jpg"
    new file:   klik.html
```

Рисунок 7 - Статус репозитория после добавления в индекс

Далее необходимо выполнить коммит при помощи команды `git commit-m` "Project initial". После этого должен быть ответ. Далее можно проверить последнее изменение при помощи команды `git log`. В конце необходимо проверить статус репозитория.

```
Admin@DESKTOP-NM8BVAI MINGW64 ~/Desktop/Moros PE-21/Сетевое программирование (main)
$ git commit -m "Project initial"
[main (root-commit) 4531020] Project initial
2 files changed, 75 insertions(+)
create mode 100644 "img/\320\272\320\276\321\202.jpg"
create mode 100644 klik.html

Admin@DESKTOP-NM8BVAI MINGW64 ~/Desktop/Moros PE-21/Сетевое программирование (main)
$ git status
On branch main
nothing to commit, working tree clean
```

Рисунок 8 -Результат выполнения команды `git status`

3.1. В существующий файл `index.html` необходимо добавить еще одну картинку и дополнительный текст и сделать второй коммит самостоятельно. Результат выполнения может быть представлен в форме листинга/скриншота.

```

Admin@DESKTOP-NM8BVAI MINGW64 ~/Desktop/Moros PE-21/Сетевое программирование (main)
$ git commit -m "Project initial"
[main (root-commit) 4531020] Project initial
2 files changed, 75 insertions(+)
create mode 100644 "img\320\272\320\276\321\202.jpg"
create mode 100644 klik.html

Admin@DESKTOP-NM8BVAI MINGW64 ~/Desktop/Moros PE-21/Сетевое программирование (main)
$ git status
On branch main
nothing to commit, working tree clean

Admin@DESKTOP-NM8BVAI MINGW64 ~/Desktop/Moros PE-21/Сетевое программирование (main)
$ git remote add origin https://github.com/llastochka/setevoye

Admin@DESKTOP-NM8BVAI MINGW64 ~/Desktop/Moros PE-21/Сетевое программирование (main)
$ git remote -v
origin https://github.com/llastochka/setevoye (fetch)
origin https://github.com/llastochka/setevoye (push)

```

Рисунок 9 – Результат выполнения

### 3.2 Индивидуальное задание студента.

- 1) Зарегистрируйтесь на GitHub и подтвердите свою почту.
  - 2) Настройте SSH или HTTPS подключение (на ваш выбор). Для того, чтобы настроить SSH, можно прочитать здесь: <https://timeweb.cloud/tutorials/windows/kak-sgenerirovat-ssh-klyuch-dlya-Windows>
  - 3) Отправьте ваш локальный репозиторий (созданный в задании 3.1) На аккаунт GitHub (при помощи командной строки или вспомогательного ПО, не имеет значения).
  - 4) Внесите изменения в ваш репозиторий на локальном компьютере, сделайте коммит и отправьте его на GitHub.
  - 5) Создайте ветку с указанием Вашей фамилии и инициалов, например, BurumbaevDI
  - 6) Далее необходимо переключиться на ветку с вашей фамилией, изменить ваш файл index.html и отправить его на GitHub (именно с второй ветки).
  - 7) Необходимо выполнить Pull Request, указав комментарий.
  - 8) После этого необходимо произвести слияние версий от главной ветки(master/main) и второстепенной (в случае примера, это BurumbaevDI).
- Данный раздел необходимо изучить самостоятельно.

```

Admin@DESKTOP-NM8BVAI MINGW64 ~/Desktop/Moros PE-21 (main)
$ git commit -m "klik"
[main 51fe7cd] klik
1 file changed, 1 insertion(+)
create mode 100644 u.txt

```

Рисунок 10 – Результат выполнения

```

Admin@DESKTOP-NM8BVAI MINGW64 ~/Desktop/Moros PE-21 (main)
$ git pull origin main --allow-unrelated-histories
From https://github.com/llastochka/Moros-PE-21
 * branch      main      -> FETCH_HEAD
Merge made by the 'ort' strategy.
 README.md | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 README.md

Admin@DESKTOP-NM8BVAI MINGW64 ~/Desktop/Moros PE-21 (main)
$ git push -u origin main
Enumerating objects: 12, done.
Counting objects: 100% (12/12), done.
Delta compression using up to 8 threads
Compressing objects: 100% (9/9), done.
Writing objects: 100% (11/11), 44.76 KiB | 3.20 MiB/s, done.
Total 11 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/llastochka/Moros-PE-21
 5bc01d0..2de67b8 main -> main
branch 'main' set up to track 'origin/main'.

```

Рисунок 11 – Результат выполнения