

## «Командный рецепт»

**Роли:**

- **Студент А (Владелец):** Создает репозиторий и управляет доступом.
- **Студент Б (Коллaborатор):** Присоединяется к разработке.

### Часть 1: Подготовка площадки (Только Студент А)

1. Зайти на [GitHub](#) и нажать на плюс (+) в верхнем правом углу -> **New repository**.
2. **Repository name:** написать prazdnik-recipe.
3. Выбрать **Public**.
4. Поставить галочку **Add a README.md**.
5. Нажать зеленую кнопку **Create repository**.
6. Теперь добавить напарника:
  - На странице репозитория нажать вкладку **Settings** (в верхнем меню под заголовком).
  - Слева в меню выбрать **Collaborators**.
  - Нажать кнопку **Add people**.
  - Ввести никнейм или почту **Студента Б** и нажать **Add [имя] to this repository**.

**Студент Б:** Зайти в свою почту или в уведомления на GitHub и нажать **Accept Invitation**, чтобы получить права на запись.

### Часть 2: Клонирование и создание веток (Оба студента)

1. **На GitHub:** Нажать зеленую кнопку **Code** и скопировать ссылку (HTTPS).
2. **На компьютере:** Откройте терминал введите:

```
git clone https://github.com/логин-студента-A/praznik-recipe.git  
cd praznik-recipe
```

3. Создайте свои ветки, чтобы не мешать друг другу в main:
  - **Студент А:** git checkout -b feature-ingredients
  - **Студент Б:** git checkout -b feature-steps

### Часть 3: Работа с файлами (Оба студента)

1. Откройте папку проекта в любом текстовом редакторе (VS Code).
2. **Оба студента:** Создайте файл с названием recipe.txt.
3. **Студент А (в своей ветке):** Напишите в файле:  
Ингредиенты:
  - Мука
  - Сахар

4. **Студент Б (в своей ветке):** Напишите в файле:

Инструкция:

1. Смешать все в миске.
2. Выпекать 30 минут.

5. **Оба студента:** Сохраните файл и отправьте код в облако:

```
git add recipe.txt
```

```
git commit -m "Добавил свою часть рецепта"  
git push origin [название-вашей-ветки]
```

### Часть 4: Слияние и Конфликт (Через браузер)

1. Зайдите на страницу репозитория на GitHub. Вы увидите желтую плашку с кнопкой **Compare & pull request**. Нажмите её.
2. **Студент А:** Создайте Pull Request (PR) и сами же его примите:
  - Нажмите **Create pull request**.
  - Нажмите зеленую кнопку **Merge pull request**, а затем **Confirm merge**.
  - Теперь файл recipe.txt в ветке main содержит ингредиенты.
3. **Студент Б:** Создайте свой Pull Request.
  - GitHub напишет красным: "**Can't automatically merge**". Это конфликт! Нажмите все равно **Create pull request**.

## Часть 5: Разрешение конфликта (Выполняет Студент Б)

Теперь Студенту Б нужно вручную соединить две версии файла.

1. В терминале переключитесь на главную ветку и скачайте то, что сделал Студент А:

```
git checkout main
```

```
git pull origin main
```

2. Вернитесь в свою рабочую ветку и попробуйте "влить" в неё обновленную main:

```
git checkout feature-steps
```

```
git merge main
```

3. Git скажет: CONFLICT (add/add): Merge conflict in recipe.txt. Откройте файл recipe.txt в редакторе. Вы увидите это:

```
<<<<< HEAD
```

Инструкция:

1. Смешать все в миске...

=====

Ингредиенты:

- Мука...

>>>>> main

4. **Вручную удалите** строки <<<<<, =====, >>>>>. Оставьте текст так, чтобы сначала шли ингредиенты, а потом инструкция. Сохраните файл.

5. Завершите слияние:

```
git add recipe.txt
```

```
git commit -m "Исправил конфликт: объединил ингредиенты и шаги"
```

```
git push origin feature-steps
```

6. **На GitHub:** Зайдите в свой Pull Request. Теперь кнопка стала зеленой! Нажмите **Merge pull request**.

Теперь в ветке main лежит полный рецепт