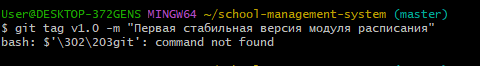
1. Создание тега для стабильной версии

После завершения модуля расписания была создана стабильная точка с помощью тега v1.0:

git tag v1.0 -m "Первая стабильная версия модуля расписания"

Это позволяет сохранить важную версию проекта для дальнейшего использования или проверки.



2. Просмотр списка тегов

Для просмотра всех созданных тегов использовалась команда:

git tag

Пример вывода:

v1.0

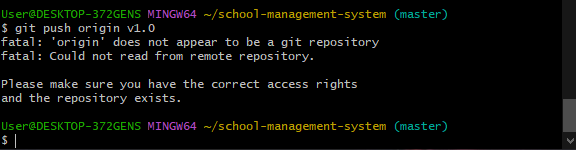


Так можно увидеть, какие версии были помечены.

3. Отправка тега на удалённый репозиторий (GitHub)

Чтобы тег был доступен не только локально, но и на сервере:

git push origin v1.0



Теперь тег доступен всем участникам команды.

4. Просмотр информации о теге

Чтобы посмотреть детали тега:

git show v1.0

Пример вывода:

commit bad54b23609dfdbe5e286b3187771a0f87a1682b (tag: v1.0)

Author: MrZhr4ik <starchik8116@gmail.com>

Date: Thu Jun 5 14:50:53 2025 +0300

Revert "Добавлено расписание на среду"

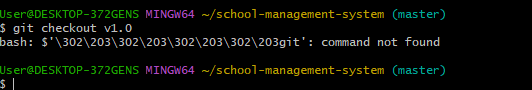
This reverts commit f639bbde866f63c9d9dbbcae6734b243206cf4b6.

Это помогает понять, к какой версии проекта относится тег.

5. Переход к версии проекта по тегу

Если нужно посмотреть, как выглядел проект на момент создания тега:

git checkout v1.0

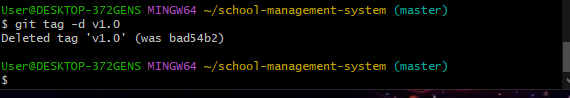


Таким образом, можно быстро переключиться на нужную версию системы.

6. Удаление локального тега

Если тег больше не нужен локально:

git tag -d v1.0



Это удаляет тег из локального репозитория.

7. Удаление удалённого тега (на сервере)

Если тег уже был отправлен на GitHub и его нужно удалить там:

git push origin :refs/tags/v1.0

или короткая запись:

git push origin --delete tag v1.0

Это важно, если ты хочешь удалить старые или ненужные метки на сервере.