**Git 1,9**

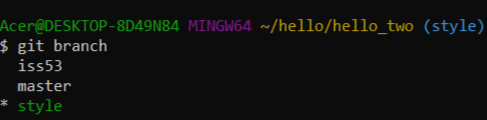
**1 Просмотр списка веток**

**Проверяю, какие ветки существуют в моём репозитории.**

**Acer@DESKTOP-8D49N84 MINGW64 ~/hello/hello\_two (style)**

**$ git branch**

**\* Показываю ветки, которые были слиты в master или ещё не слиты.**

****

**2 Фильтрация веток по истории коммитов**

* **Показать ветки, слитые в master:**

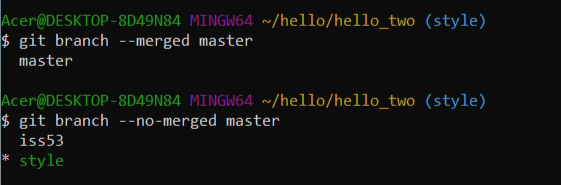
**Acer@DESKTOP-8D49N84 MINGW64 ~/hello/hello\_two (style)**

**$ git branch --merged master**

* **Показать ветки, не слитые в master:**

**Acer@DESKTOP-8D49N84 MINGW64 ~/hello/hello\_two (style)**

**$ git branch --no-merged mster**

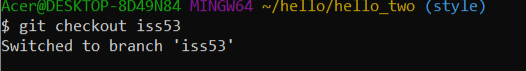
****

**3 Переключение между ветками**

**Перехожу на нужную ветку для работы.**

**Acer@DESKTOP-8D49N84 MINGW64 ~/hello/hello\_two (style)**

**$ git checkout iss53**

****

**4 Перебазирование (rebase)**

**Обновляю ветку, перенося её изменения поверх другой ветки (например, master).**

**Перебазировать style на master:**

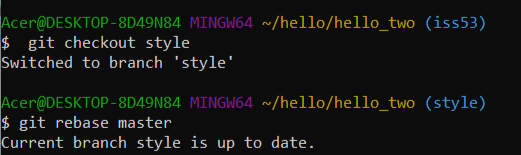
**Acer@DESKTOP-8D49N84 MINGW64 ~/hello/hello\_two (iss53)**

**$ git checkout style**

**Switched to branch 'style'**

**Acer@DESKTOP-8D49N84 MINGW64 ~/hello/hello\_two (style)**

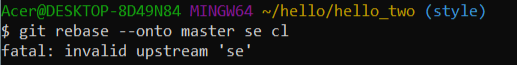
**$ git rebase master**

****

**Перебазирование с параметром --onto (для конкретных коммитов):**

**Acer@DESKTOP-8D49N84 MINGW64 ~/hello/hello\_two (style)**

**$ git rebase --onto master se cl**

****

**Описание:**

* **se — стартовая точка.**
* **cl — конечная точка.**
* **master — целевая ветка.**

**5 Слияние веток (merge)**

**Объединяю изменения из одной ветки в другую.**

**Слить cl в master:**

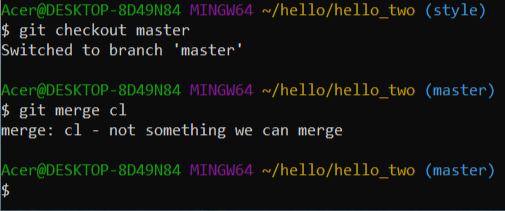
**Acer@DESKTOP-8D49N84 MINGW64 ~/hello/hello\_two (style)**

**$ git checkout master**

**Switched to branch 'master'**

**Acer@DESKTOP-8D49N84 MINGW64 ~/hello/hello\_two (master)**

**$ git merge cl**

****

**6 Удаление веток**

**Удаляю ветку после успешного слияния.**

**Acer@DESKTOP-8D49N84 MINGW64 ~/hello/hello\_two (master)**

**$ git branch -d iss53**

****