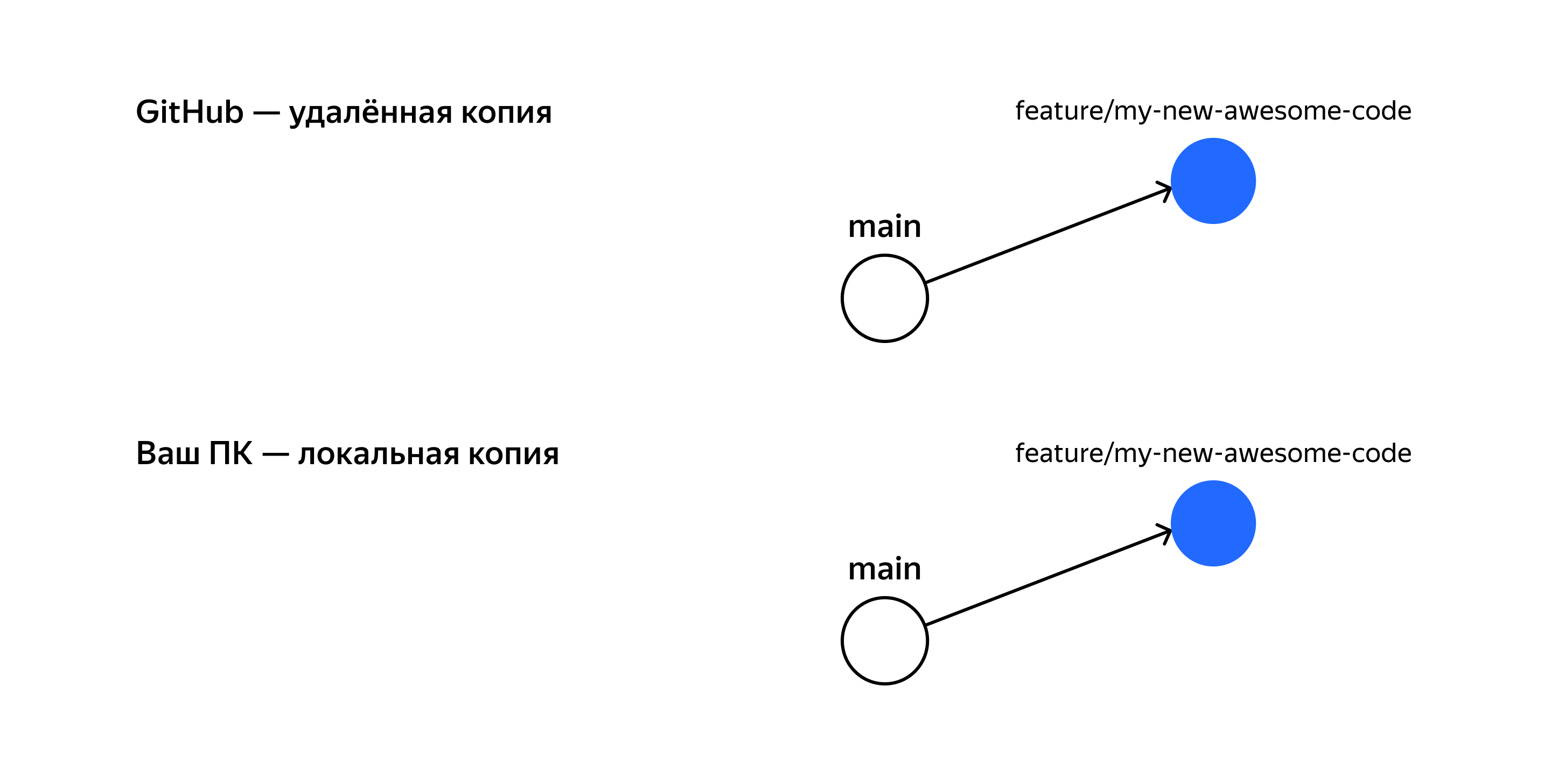
# **Что делать, если основная ветка «убежала» вперёд в процессе ревью**

Конфликт — это не ошибка, а особенность командной работы. Процесс не стоит на месте, и проекты постоянно развиваются: в них появляются новые функциональности или дорабатываются старые. Так что угнаться за изменениями бывает непросто.

В этом уроке рассмотрим одну из таких типичных рабочих ситуаций, которые могут привести к конфликту в Git.

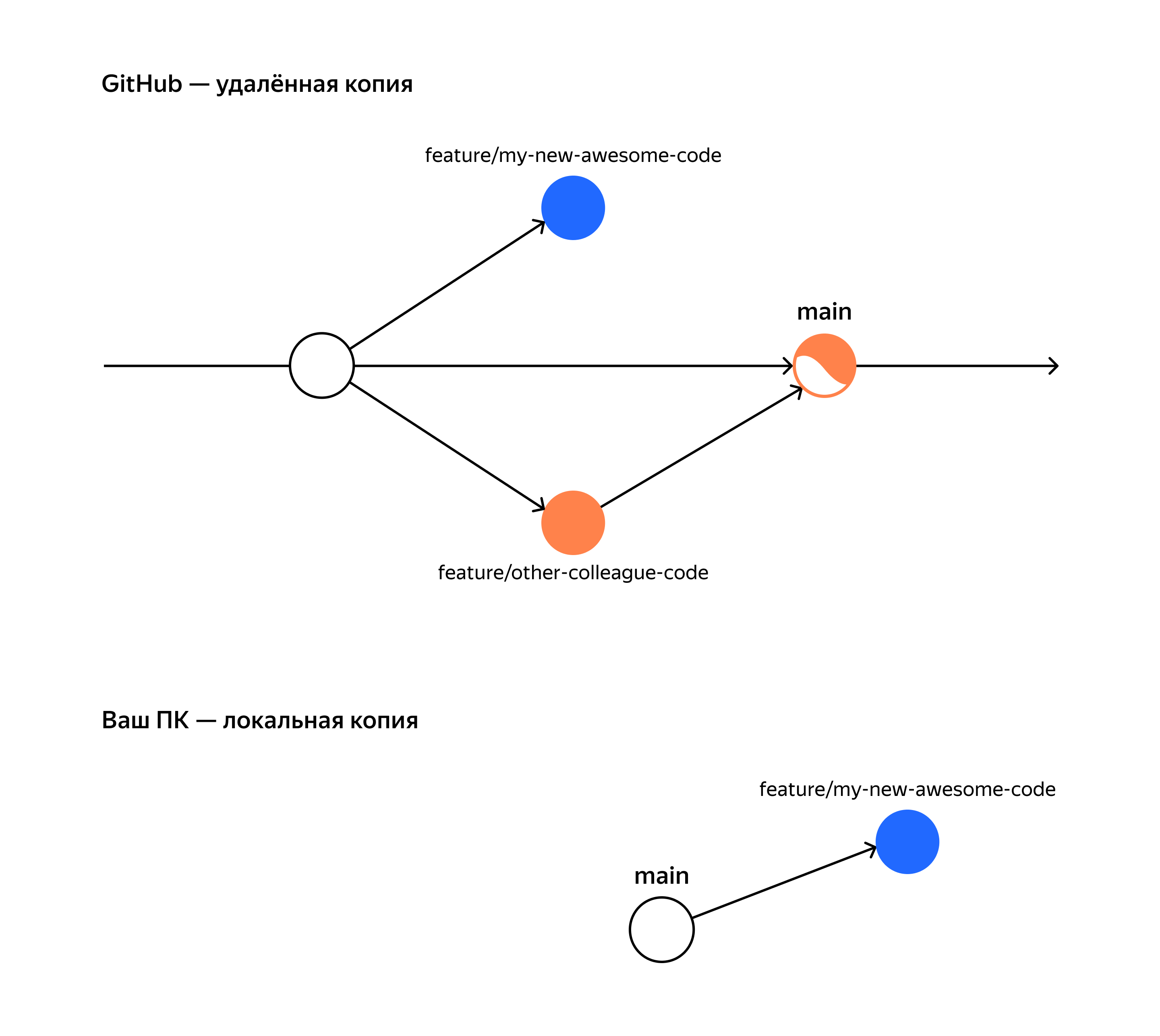
### **Основная ветка «убежала» во время ревью**

Представьте: вы создаёте пул-реквест, чтобы добавить свои изменения из ветки feature/my-new-awesome-code в основную ветку main. Схематически это выглядит так.



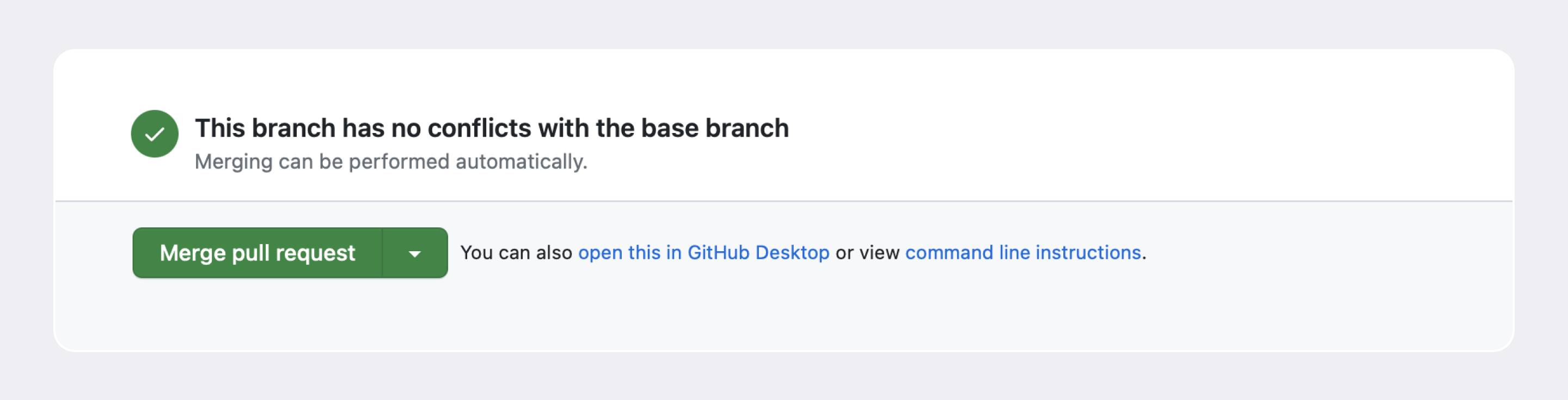
В вашей команде все изменения должны проходить ревью, поэтому вам нужно подождать, пока один из коллег одобрит ваш пул-реквест.

В это время в main вливается ветка другого коллеги (feature/other-colleague-code), которая попала на ревью раньше вашей, и передвигает main вперёд. Теперь на GitHub новая версия репозитория. А локально у вас всё ещё старая.

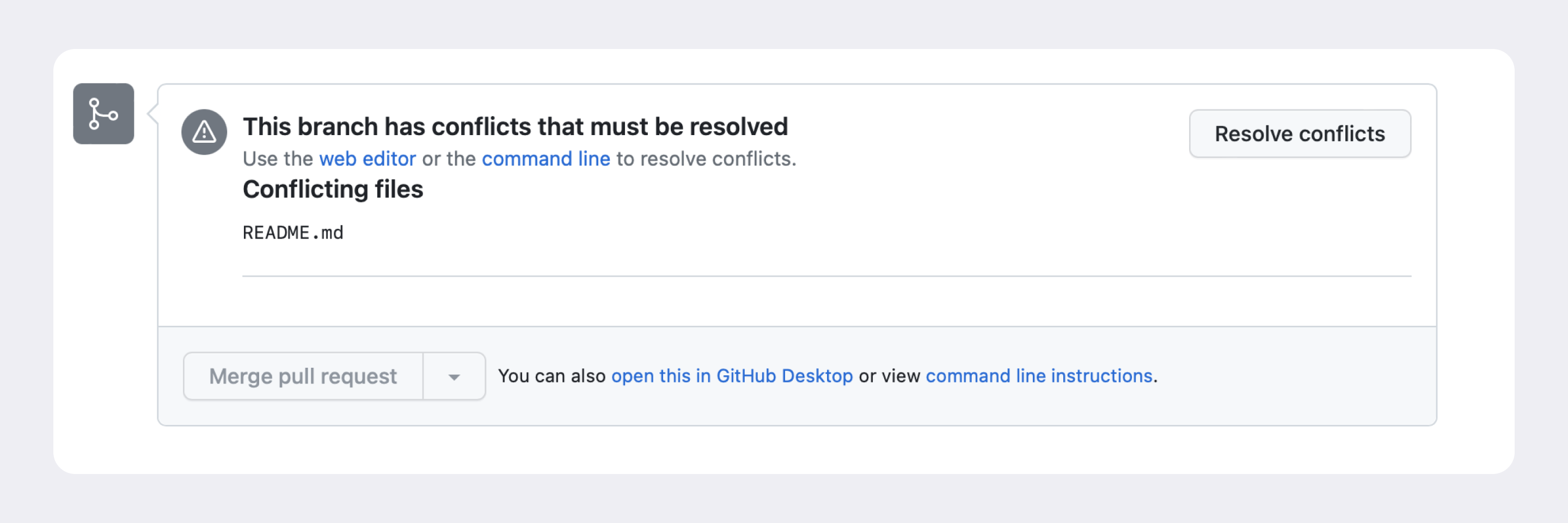


Теперь ваши изменения отстают от ветки main. Дальше может быть два варианта развития событий.

Первый: ваши изменения и текущие изменения в main получится «смёржить» автоматически, потому что GitHub сможет самостоятельно разрешить конфликт. В этом случае кнопка **Merge pull request** в интерфейсе пул-реквеста всё ещё будет зелёной.



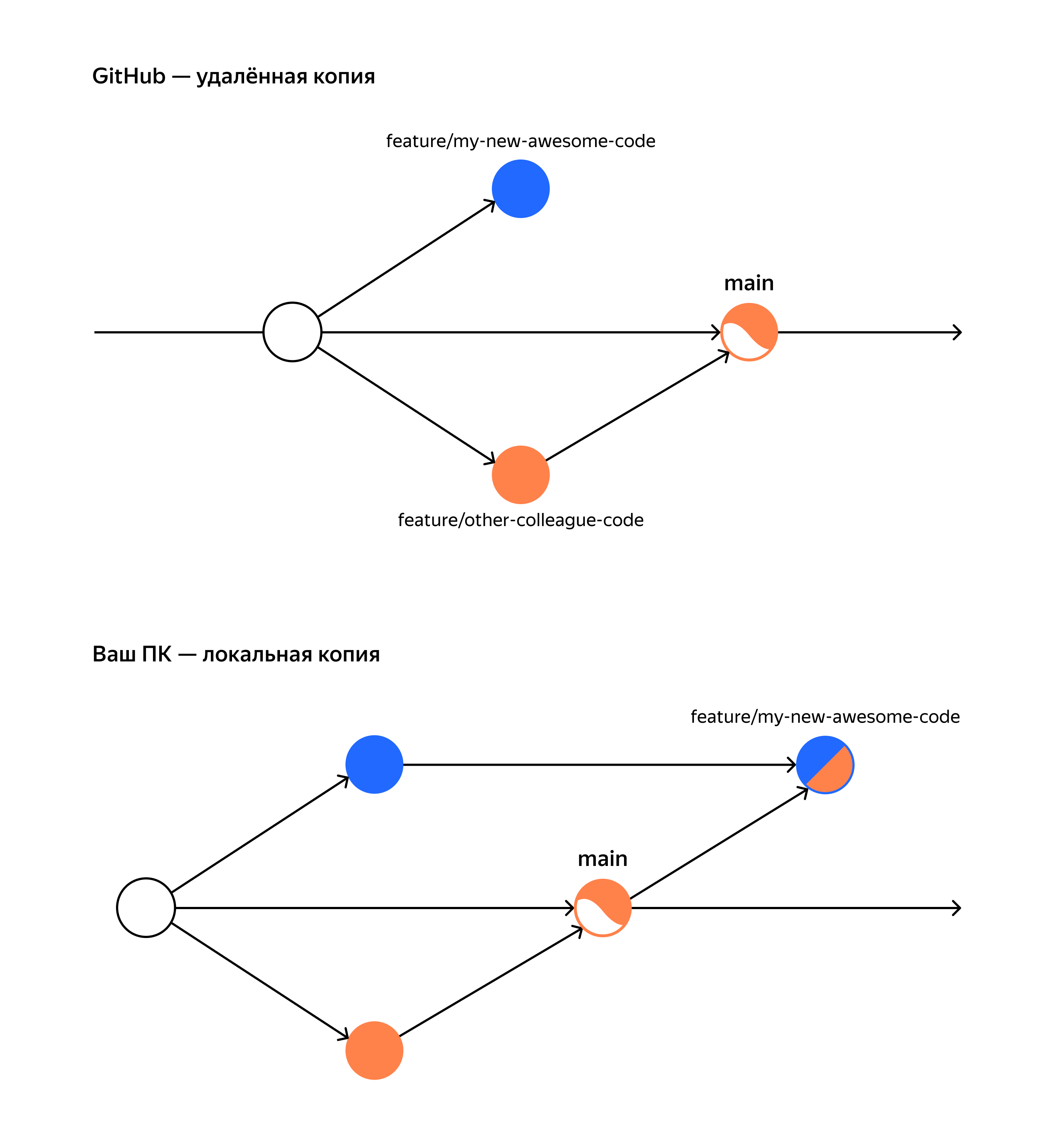
Второй: два изменения не могут существовать вместе, и вы увидите следующее предупреждение.



Хотя GitHub предоставляет интерфейс для разрешения конфликтов, обычно это делается локально. На своём локальном компьютере вы можете использовать более удобные инструменты для разрешения конфликтов, такие как Visual Studio Code.

Разрешить конфликт можно так:

* Перейти в ветку main.
* Загрузить новые изменения из неё с помощью git pull. При этом вы также загрузите ветку вашего коллеги.
* Снова перейти в вашу ветку feature/my-new-awesome-code.
* Выполнить git merge main и разрешить конфликт локально. В результате будет создан локальный коммит слияния.



* Отправить изменения из вашей ветки в репозиторий с помощью git push. Так коммит слияния попадёт в удалённый репозиторий и в пул-реквест.

Теперь, когда ваши изменения пройдут ревью и окажутся в main, удалённый и локальный репозитории будут выглядеть так.

