**3.6 Практика по теме «Откладывание изменений»**

Описание

В этом курсе мы предусмотрели опциональные мини-задания для самостоятельной практики. Их не нужно сдавать на проверку. Нажав на кнопку внизу страницы, вы найдёте алгоритм и советы по выполнению этого задания с перечнем использованных команд. Постарайтесь решить задание самостоятельно, но если возникнет вопрос, то вы всегда сможете посмотреть в эту подсказку.

В предыдущем видео показана общая концепция работы с откладыванием изменений — использование стека. Здесь вы сможете повторить эти команды на практике в любом репозитории.

В этом задании вам необходимо, используя связанные локальный и удалённый репозитории:

1. создать новую ветку, внести в неё изменения по удалению файлов или строк кода, а затем отложить их;
2. внести новые строки кода или файлы в ветку и добавить их в индекс;
3. произвести изменения в этой же ветке, а затем отложить их;
4. посмотреть на весь список отложенных изменений и проверить, что у вас нет незакоммиченных изменений;
5. достать изменения из стэша, а затем вернуть этот стэш обратно;
6. удалить стэш с индексом 1 и проверить, что у вас остался только стеш с изменениями, а не с удалениями.

Для выполнения этого задания воспользуйтесь проектом из предыдущего задания для самостоятельного выполнения или создайте новый проект с использованием Git.

Эталонное решение

1. Создайте новую ветку от master и перейдите в неё.
2. Попробуйте удалить какой-нибудь файл или строки кода, а потом отложите эти изменения: git stash.
3. Создайте новый файл или напишите новые строки кода в существующих файлах, а потом добавьте их в индекс командой git add “имя файла”.
4. Измените существующие строки кода в файлах на что-нибудь другое, затем отложите такие изменения.
5. Посмотрите на весь список отложенных изменений: git stash list.
6. Проверьте, что у вас нет незакоммиченных изменений, с помощью команды git status.
7. Достаньте изменения из стэша: git stash pop.
8. Удостоверьтесь, что в выводе этой команды показаны все незакоммиченные изменения, в том числе те, которые раньше были проиндексированы.
9. Верните этот стэш обратно: git stash.  
   Обратите внимание, что индексы стэшей в списке формируются от раннего к позднему. Таким образом, под индексом 1 лежит стэш с удалениями.
10. Перейдите обратно в ветку master.
11. Удалите стэш с индексом 1: git stash drop 1.
12. Достаньте оставшийся стэш: git stash pop. Проверьте, что там именно изменения с добавлением и изменениями в файлах.
13. Проверьте, что у вас не осталось стэшей: git stash list.