|  |  |
| --- | --- |
| Clear  git add [Markdown\_Instruction.md](http://markdown_instruction.md/)  git commit -m "initial commit  ctrl+s  git branch  git branch имя\_новой\_ветки  git checkout имяНовойВетки  git log  git checkout имяВетки  git checkout draft(имя ветки)  git merge имяВетки  git branch -d имяВетки git branch -d draft  файл .gitignore (без расширения)  git log -- graph | **Лекция 2 Контроль версий**  clear - очищает терминал Папка git уже отслеживается  1. Создали новый файл [Markdown\_Instruction.md](http://markdown_instruction.md/) 2. git add [Markdown\_Instruction.md](http://markdown_instruction.md/) 3. git commit -m "initial commit" 4. Открыли файл [Markdown\_Instruction.md](http://markdown_instruction.md/) и ввели основные заголовки после чего сохранили файл ctrl+s 5. Повторяем пункт 2 и 3 с новым комментарием "Добавили структуру данных" 6. git branch - показывает ветки, отмечает звёздочкой ту, на которой мы находимся 7. Сделаем черновик. Для этого создадим новую ветку: git branch имя\_новой\_ветки git branch draft(черновик) 8. Теперь если 2ветки git branch - Покажет какие ветки есть и отметит звёздочкой ту на которой мы находимся 9. Перейдем на новую ветку git checkout имяНовойВетки git checkout draft Теперь можно работать с файлом предыдущая версия сохранена на старой ветке 10. Внесли изменения в файл теперь git status покажет что изменения не сохранены для этого добавим файл к отслеживанию и новый commit git add [Markdown\_Instruction.md](http://markdown_instruction.md/) git commit -m "Добавили типы выделений текста" 11. git log покажет все добавленные изменения на всех существующих ветках 12. Для перехода на необходимую ветку введем git checkout имяВетки git checkout master Таким образом вернёмся к "основному" файлу до изменений (в общем я этого не сделал и создал новую ветку из ветки draft) 13. Добавим ещё ветку lists git branch lists и перейдем на нее, и отредактируем наш файл и сохраним ctrl+s, добавим к отслеживанию и добавим commit git add [Markdown\_Instruction.md](http://markdown_instruction.md/) git commit -m "Добавили списки" 14. Перейдем к ветке draft git checkout draft добавим ещё про выделения текста информацию, сохраняем файл git status проверим  добавим к отслеживанию 15. Произведем "слияние" веток master и draft, для этого перейдем в ветку master git checkout master Проверим перешли? git branch теперь проведем слияние git merge имяВетки git merge draft После слияния информация из ветки draft добавилась в ветку master 16. git log показывает что у нас master и draft объединены 17. Посмотрим какие ветки у нас есть git branch Все ветки на месте, но ветка draft нам больше не нужна 18. Удалим не нужную ветку git branch -d имяВетки git branch -d draft Удалять можно и с заглавной -D но лучше со строчной так удаление происходи когда все штатно Проверим что у нас с ветками git branch Покажет оставшиеся 2 ветки 19. Поработаем с изображениями для этого создадим новую ветку images git branch images Проверим git branch Теперь у нас 3 ветки, перейдем к той с которой нужно работать 20. git checkout images проверим где мы git branch Мы перешли в ветку images Добавим информацию про изображения и сохраним git status У нас в папке появилось изображение которое не нужно отслеживать, его нужно добавить в исключения но для начала добавим основной файл для отслеживания git add [Markdown\_Instruction.md](http://markdown_instruction.md/) git commit -m "working with imaged" git log покажет что мы добавили новый commit 21. Вернёмся к изображению которое не нужно отслеживать для этого создадим новый файл .gitignore (без расширения) и в нем перечисляем файлы которые не нужно отслеживать, теперь добавим этот файл в commit git add .gitignore git commet -m "added .gitignore" git status Теперь файл с изображением git игнорирует 22. Теперь перельем ветку images в ветку master, для этого перейдем в ветку мастер git checkout master Проверим где мы git branch мы в master git merge images Слили git log Показывает что слиты 2 ветки 23. Добавим информацию в файл про не нумерованные списки, сделаем с файлом add и commit git add [Markdown\_Instruction.md](http://markdown_instruction.md/) git commit -m "добавили информацию про списки" Перельем из ветки lists в master git merge list Так как у нас в обоих ветках была информация про списки, произошел конфликт. VScode нам предоставляет варианты: принять текущую версию Принять пришедшую Оставить оба варианта Сравнить Мы оставим оба варианта и отредактируем в ручную как нам надо git status git add [Markdown\_Instruction.md](http://markdown_instruction.md/) git commit -m "Слили 2 ветки, разрешили конфликт" git log -- graph  Покажет более наглядно какие сохранения в каких ветках и что как сливалось 24. Теперь все ветки слиты в master значит все прочии можно удалить  git branch -d lists git branch -d images Все ветки кроме master удалины git breanch - так и есть |
|  |  |