1. Что такое система контроля версий, для чего ее используют?   
  
Система контроля версиями (от англ. Version Control System, VCS или Revision Control System) — программное обеспечение для облегчения работы с изменяющейся информацией. Используется для синхронизации кодовой базы, совместной работы,а также bug трекинговой системы. Удобно узнать, кто, когда и зачем сделал изменения, что вытекает из совместной работы.

2. Какой принцип хранения файлов использует Git?   
  
Хранит снимки состояний, а не изменений.

3. В чем отличие git от других систем контроля версий?

 Быстродействие и размер

 Безопасность и целостность (хэш SHA)

 Достоверность

 Гибкость (нелинейные рабочие процессы – слияние, ветвление) 

Производительность (простое ветвление) 

Функциональность

4. В каких трех основных состояниях файлы могут находиться в Git ?  
 В модифицированном(modified(отредактирован)), индексированное(staged), зафиксированное (committed)

5. Что такое индексация файла? Какой командой она выполняется?   
Выполняется командой git add. Индексация файла - это определение системой измененных файлов и добавление их в отслеживаемые файлы системы контроля версий.

6. Что такое фиксация файла? Какой командой она выполняется?   
Выполняется командой git commit. Фиксация файла — это добавление индексированных файлов в историю снимков состояния.

7. Продемонстрируйте команду проверки выбранных настроек.   
git config --list

8. Как инициализировать репозиторий в существующей папке?   
Находясь в этой папке, выполнить команду git init

9. Как указать файлы, за которыми должна следить система?   
Не добавлять их в .gitignore  
Проиндексировать их.

10. Как выполнить фиксацию изменений?   
С помощью команды git commit

11. Какой командой определить состояния файлов?   
git status

12. Для чего создается файл .gitignore? Поясните его структуру.  
Данный файл создается для того, чтобы не загружать ненужное в удаленный репозиторий и не засорять его. При индексации файлы, указанные в .gitignore не будут задействованы.   
Вот некоторые правила синтаксиса этого файла:

* Одна строчка - одно правило,
* Пустые строки игнорируются,
* Комментарии доступны через решётку(#) в начале строки,
* Символ "/" в начале строки указывает, что правило применяется только к файлам и папкам, которые располагаются в той же папке, что и сам файл .gitignore,
* Доступно использовать спецсимволы: звёздочка(\*) заменяет любое количество символов(ноль или больше), вопросик(?) заменяет от нуля до одного символа. Можно размещать в любом месте правила,
* Две звёздочки(\*\*) используются для указания любого количества поддиректорий, подробнее смотри ниже в примерах,
* Восклицательный знак(!) в начале строки означает инвертирование правила, необходим для указания исключений из правил игнорирования,
* Символ "\" используется для экранирования спецсимволов, например, чтобы игнорировать файл с именем "!readme!.txt", нужно написать такое правило: "\!readme!.txt",
* Для игнорирования всей директории, правило должно оканчиваться на слэш(/), в противном случае правило считается именем файла.

13. Как используется команда git diff?   
Для просмотра измененных фрагментов.

14. Как используется команда git commit?   
Для фиксации изменений. При git commit -m в кавычках после можно написать сообщение к зафиксированному изменению.

15. Как используется команда git log? Какие у нее есть параметры?   
 Используется для отслеживания в обратном хронологическом порядке списка сохраненных в данный репозиторий версий. Первым показываются самые первые фиксации. Показывается так же данные о том, кто сделал фиксацию и сообщение о ней.

16. Как используется команда git commit –amend?   
Команда позволяет объединить проиндексированные изменения с предыдущим коммитом без создания нового коммита. Ее также можно использовать для простого редактирования комментария к предыдущему коммиту без изменения состояния кода в нем.  

17. Как отобразить удаленные репозитории?   
 С помощью команды git remote

18. Как извлечь данные из удаленного репозитория?   
 Это можно сделать с помощью команды git pull

19. Как отправить данные в удаленный репозитроий?   
С помощью команды git push

20. Какая команда позволяет отобразить удаленные репозитории, связанные с текущим локальным?

Git remote

21. Каким образом можно получить изменения из удаленного репозитория в локальный?

С помощью команды git fetch

22. Для чего используется команда fetch? В чем отличие команды fetch от pull?   
Данная команда связывается с указанным удалённым проектом и забирает все те данные проекта, которых у вас ещё нет.   
Выполнение git pull извлекает (fetch) данные с сервера, с которого изначально склонировали, и автоматически пытается слить (merge) их с кодом, над которым в данный момент идет работа. git pull — это, по сути, команда git fetch, после которой сразу же следует git merge.

23. Для чего используется команда merge? В чем отличие merge от rebase?   
Команда merge используется для слияния текущей ветви с заданной ветвью.   
Отличие команд merge и rebase в том, что rebase нужен для порядка (линейность истории), чтоб потом смотреть и понимать. rebase переносит коммиты поверх того, что было в результирующей ветке. merge оставляет коммиты, как есть, просто добавляя связующий коммит между ветками.

24. Что такое pull request?

pull request — предложение изменения кода в чужом репозитории с целью исправления багов, ошибок, конфликтов с другими приложениями, добавления новых функций, возможностей, стилевых правок и тд.

25. Какая команда позволяет отобразить историю репозитория?   
Git log

26. Продемонстируйте создание новых веток в Git? Что такое ветвление? Что такое указатель HEAD?   
Создание новой ветки в Гит:   
git branch [имя\_ветки]  
Ветка в Git — это простой перемещаемый указатель на один из коммитов. По умолчанию, имя основной ветки в Git — master.    
Указатель HEAD – специальный указатель, указывающий на текущую локальную ветку.

27. Как используется команда git checkout?   
Используется для перехода в новую ветку или же отмены фиксации файла.(?)

28. Как выполнить включение изменений из одной ветки в другую?   
Git merge

29. Какие проблемы могут быть при слиянии и как они разрешаются?

При слиянии могут возникать конфликты, когда один и тот же файл отредактирован и закомичен, но еще не добавлен в удаленный репозиторий. Гит показывает в каком месте файла возникают сложности и нужно тогда из двух или более измененных файлов оставить один, или же отредактировать так, как это будет лучше для кода и проекта в целом.

30. Что такое GitLab?  
GitLab — веб-приложение и система управления репозиториями программного кода для Git. Появилась позже GirHub на 3 года(в 2011-ом).