1. Система контроля версий (СКВ) - это инструмент, который используется для управления изменениями в коде программы или любых других документах. Она позволяет отслеживать изменения, возвращаться к предыдущим версиям, сравнивать различные версии и объединять изменения от разных членов команды.

2. Git использует принцип хранения файлов в виде снимков (snapshots). Это означает, что при каждом коммите Git сохраняет полное состояние всех файлов в репозитории, а не только изменения относительно предыдущего состояния.

3. Отличие Git от других систем контроля версий заключается в его распределенной структуре. Это означает, что каждый пользователь имеет полную копию репозитория, что обеспечивает независимость и устойчивость к сбоям в сети.

4. В Git файлы могут находиться в трех основных состояниях: измененные (modified), подготовленные (staged) и зафиксированные (committed).

5. Индексация файла - это процесс подготовки файла к коммиту. Она выполняется командой "git add".

6. Фиксация файла (commit) - это процесс сохранения изменений в репозитории. Она выполняется командой "git commit".

7. Команда "git config –list" позволяет проверить выбранные настройки Git.

8. Для инициализации репозитория в существующей папке используется команда "git init".

9. Файлы, за которыми должна следить система, указываются в файле .gitignore.

10. Фиксацию изменений можно выполнить с помощью команды "git commit".

11. Состояния файлов определяются командой "git status".

12. Файл .gitignore создается для указания файлов и папок, которые Git должен игнорировать при отслеживании изменений. Его структура состоит из шаблонов для игнорирования файлов и папок.

13. Команда "git log" используется для просмотра истории коммитов в репозитории.

14. Ветвление (branching) - это возможность создания отдельной линии разработки, которая может быть объединена с основной линией в будущем.

15. Команда "git checkout" используется для переключения между ветками или восстановления файлов из предыдущих коммитов.

16. Указатель HEAD - это ссылка на текущий коммит или текущую ветку в репозитории.

17. Создание новых веток в Git осуществляется с помощью команды "git branch <branch\_name>".

18. Переключение между ветками осуществляется с помощью команды "git checkout <branch\_name>".

19. Команда merge используется для объединения изменений из одной ветки в другую.

20. Команда "git log" позволяет отобразить историю репозитория.

21. Слияние изменений из одной ветки в другую выполняется с помощью команды "git merge <branch\_name>".

22. При слиянии могут возникнуть проблемы с конфликтами изменений, которые разрешаются вручную путем выбора нужных изменений из разных версий файлов.