Ответы на контрольные вопросы

1. Git – это одна из самых распространённый систем контроля версий, т.е. ПО, позволяющее отслеживать изменения в документах, при необходимости производить их откат, определять, кто и когда внес исправления и т.п.
2. VCS позволяет:

1. Отслеживать изменения в проекте

2. Фиксировать изменения в проекте

3. Откатываться до более ранней версии проекта

4. Определять источник изменений в проекте

5. Создавать новые и объединять существующие ветви проекта

6. Производить контроль доступа пользователей к проекту

1. Commit - это команда отправки индексированных изменений в репозиторий. Commit выполняется после инициализации рабочих файлов.
2. Посмотреть историю коммитов можно с помощью команды git log, которая имеет большое количество опций для поиска коммитов по разным критериям.
3. Чистый репозиторий создаётся командой git init
4. Система контроля версий *git*имеет архитектуру трех деревьев. Суть ее заключается в том, что дополнительно к репозиторию и рабочей копии добавляется ещё одно место, которое можно назвать кэшем или *stage*в английской терминологии. Рабочая копия и репозиторий идейно не отличается от их аналогов в архитектуре двух деревьев.

Наличие кэша (stage) делает процесс разработки более гибким и позволяет вносить изменения в довольно большое количество файлов, но при этом непосредственно отправлять их в репозиторий в разных коммитах со своими специфическими комментариями.

1. Создать новую ветвь можно с помощью команды git branch <Название ветви> или командой git checkout –b <Название ветви> (при выполнении второй команды происходит переход на созданную ветвь, а при выполнении первой – нет). Объединение ветвей осуществляется при помощи команды git merge
2. **Staging area** —информация о том, что должно войти в следующий коммит. До того, как изменения будут отправлены в репозиторий, разработчик может добавлять и удалять файлы из области подготовленных файлов.
3. Сравнивать между собой файлы можно c помощью этой команды в git diff