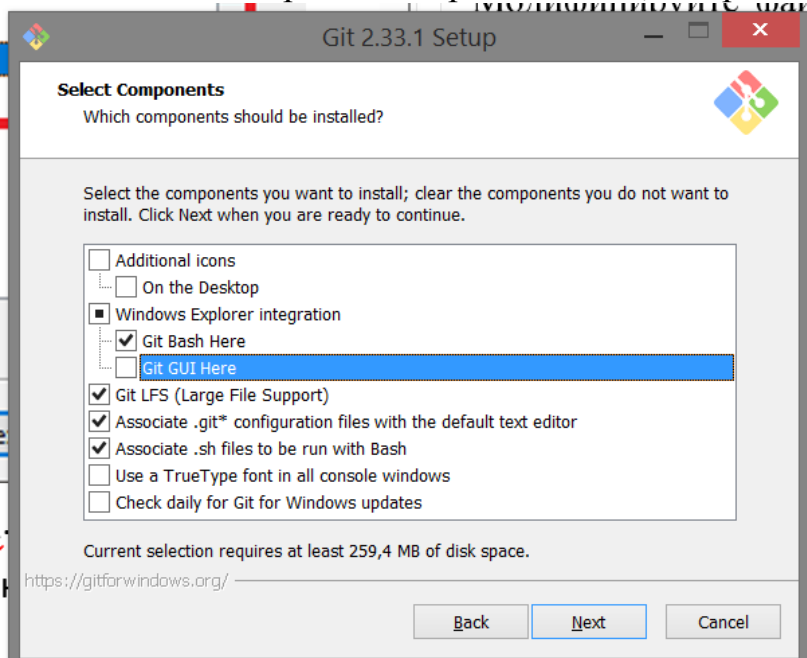


### Системы контроля версий. Основы работы в Git

**Цель работы:** ознакомление с основными возможностями системы контроля версий Git.

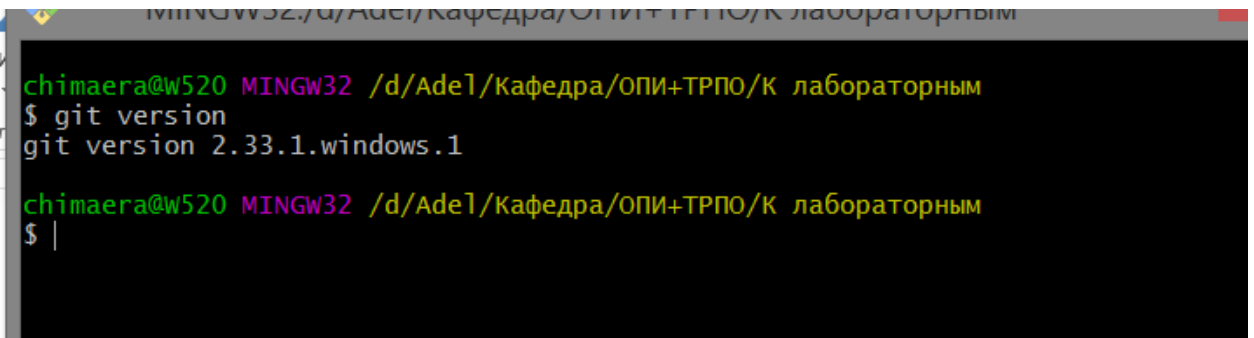
1. Используйте при выполнении лабораторной работы материал лекции 10 (часть1).
2. Отчет по лабораторной работе оформить в виде документа в MS Word.
3. Скачайте с сайта <http://git-scm.com/downloads/> и установите консольного клиента (*gitbush*):

Установить интеграцию с Проводником, выбрав чекбокс: Git Bash Here



**НЕ РЕКОМЕНДУЕТСЯ** изменять настройки для среды.

4. Проверьте установленную версию и зафиксируйте ее в отчете:



5. Получите справку по **Git**. Ознакомьтесь со списком команд. Какой командой для получения справки вы воспользовались? Описание поместите в отчет.
6. Создайте локальный репозиторий. Опишите последовательность действий в отчете.
7. Выполните конфигурацию гит. Для этого настройте конфигурационный файл, используя:

```
$ git config --global user.name "XXX"
$ git config --global user.email XXX@XXX.com
```

8. Проверить настройки конфигурационного файла командой `git config --list`
9. Инициализируйте локальный репозиторий в рабочей папке (см. лекцию), где будет он будет размещен. Исследуйте изменения, которые произошли в папке. Зафиксируйте их в отчете.
10. С помощью блокнота создайте в папке текстовый файл `Hello.txt`.
11. Научитесь получать информацию о статусе (`status`) репозитория. Какая информация отображается в терминальном окне `GitBash`? Прокомментируйте вывод команды в отчете.
12. Проиндексируйте файл `Hello.txt` Какие изменения произошли в репозитории?
13. Зафиксируйте изменения индексированных файлов в репозитории. Какие изменения произошли в репозитории? Как `Git` информирует об успешном создании нового коммита? В какую ветку добавлен файл?
14. Просмотрите журнал с помощью команды `git log`
15. Научитесь получать информацию о проиндексированных и неиндексированных файлах. Для этого измените существующий файл `Hello.txt`, добавьте в репозиторий новый файл и исследуйте состояние репозитория. Зафиксируйте изменения в репозитории.
16. Научитесь пользоваться командами управления и фиксации изменений (`add`, `commit`, `status`, `log`).
17. Ознакомьтесь с шаблонами `.gitignore` и настройте этот файл для вашего репозитория.

**Вопросы (сохранить в отчете).**

1. Что такое система контроля версий, для чего ее используют?
2. Какой принцип хранения файлов использует Git?
3. В чем отличие Git от других систем контроля версий?
4. В каких трех основных состояниях файлы могут находиться в Git ?
5. Что такое индексация файла? Какой командой она выполняется?
6. Что такое фиксация файла? Какой командой она выполняется?
7. Продемонстрируйте команду проверки выбранных настроек.
8. Как инициализировать репозиторий в существующей папке?
9. Как указать файлы, за которыми должна следить система?
10. Как выполнить фиксацию изменений?
11. Какой командой определить состояния файлов?
12. Для чего создается файл `.gitignore`? Поясните его структуру.