**Лабораторная работа №1.**

Тема: **Настройка работы системы контроля версий (типов импортируемых файлов, путей, фильтров и др. параметров импорта в репозиторий)**

Цель работы: освоить навыки работы с системой контроля GIT

1. **Индивидуальные задания**

**ЧАСТЬ А**

1. Зарегистрироваться в github, если Вы ещё не зарегистрированы
2. Разбиться на группы по два человека для выполнения семестрового проекта
3. Один человек из каждой группы создаёт приватный репозиторий для проекта на java (mdk-02-02)
4. Владелец репозитория даёт доступ к нему второму студенту и преподавателю
5. Каждый студент создаёт свою ветку, в которой ведёт дальнейшую разработку своей части проекта до конца семестра, отправляя в master рабочие версии файлов.

**ЧАСТЬ Б**

1. Создать три файла: 1.txt, 2.txt, 3.txt.
2. Проиндексировать первый файл и сделать коммит с комментарием “add
3. 1.txt file”.
4. Проиндексировать второй и третий файлы.
5. Удалить из индекса второй файл.
6. Перезаписать уже сделанный коммит с новым комментарием “add 1.txt
7. and 3.txt”
8. Создать аннотированную метку с названием v0.01.
9. Создать легковесную ветку указывающую на первый коммит в репозитории.

**ЧАСТЬ С**

1. Создать папку temp в своем репозитории.
2. Создать папку log и добавить в нее 2 файла: main.html и some.tmp.
3. Создать файл .gitignore и добавить в игнорирование папку temp и файлы
4. с расширением .tmp из папки log.
5. Закоммитить добавление файла .gitignore.
6. Внести изменения в файл my\_first\_file.txt, добавив строчку “row to
7. index”, проиндексировать данные изменения. Еще раз внести изменения в файл, добавив строчку “row no index”.
8. Посмотреть индексированные и неиндексированные изменения используя команду git diff.
9. Удалить файл my\_first\_file.txt, зафиксировать данное удаление.
10. Переименовать файл my\_second\_file.txt в my\_first\_file.txt, зафиксировать изменение

31 git init

1932 git add 1.txt

1933 git add 2.txt

1934 git add 3.txt

1935 git add 3.txt

1936 git status

1937 git reset -- 2.txt

1938 git status

1939 git commit --amend -m "add 1.txt"

1940 git commit -m "add.1.txt"

1941 git reflog

1942 git tag -a v0.01 -m "Release version 0.01"

1943 git branch bagFig

1944 git reflog

1945 git reflog

1946 git branch -f bagFix e63de45

1947 git reflog

1948 history

1949 git commit

1950 git add commit

1951 git add commit good11

1952 git add commit good11

1953 git add commit good

1954 git commit good

1955 git initgit init

1956 git init

1957 git add commit good

1958 git add commit

1959 git branch -m good

1960 git branch -m 1. txt

1961 git add 1.txt

1962 git add 1.txt

1963 git add 1.txt

1964 git add 1.txt

1965 git add <1.txt>

1966 git add 1.txt.

1967 git add

1968 git add 1.txt

1969 git add 1.txt

1970 ls

1971 git add 1.txt

1972 git status

1973 git add 2.txt

1974 git add 3.txt

1975 git status

1976 git branch -m 1. txt

1977 git branch 1.txt

1978 git commit 1.txt file

1979 git commit 1.txt

1980 git reset 2

1981 git reset 2.txt

1982 git commit 1.txt gg

1983 git commit 1.txt file

1984 git reset 2

1985 git reset 2.txt

1986 git commit

1987 git commit

1988 txt file

1989 git commit txt file

1990 git commit

1991 git commit 1.txt

1992 git commit -m txt gile

1993 git commit -m 'txt file'

1994 git commit -m "txt file"

1995 git commit -m "txt file"

1996 git commit -m "1.txt"

1997 git reset 2.txt

1998 git status

1999 git reset

2000 git -rf ./. git

2001 git -rf ./. git

2002 git -rf ./. git

2003 rm -rf ./. git

2004 rm-rf ./. git

2005 rm -rf ./.git

2006 git init

2007 git add txt.1 txt.2 txt.3

2008 git status

2009 git add commit txt.1 txt.2 txt.3

2010 git status

2011 git add txt.1

2012 git status

2013 git add commit txt.1 txt.23 3.txt

2014 git add commit 1.txt 2.txt 3.txt

2015 git status

2016 git add commit 1.txt

2017 git status

2018 git add 1.txt 2.txt 3.txt

2019 git status

2020 git reset 2.txt

2021 git status

2022 git commit --amend -m "add 1.txt and 2.txt

git commit --amend -m "add 1.txt and 3.txt

2023 git commit --amend -m "add 1.txt and 2.txt

git commit --amend -m "add 1.txt and 3.txt"

git status

git status

git satatus

git commit --amend -m "add 1.txt and 2.txt

2024 git commit --amend -m "add 1.txt and 3.txt"

2025 git status

2026 git status

2027 git status

2028 git commit --amend -m "add 1.txt and 3.txt"

2029 git commit --ament -m "add 1.txt and 3.txt"

2030 git commit --amend -m 'add 1.txt and 3.txt'

2031 git commit -m "add 1.txt and 3.txt"

2032 git tag V0.01.

2033 git tEg V0.01.

2034 git tag v0.01.

2035\* git tag v0.01.

2036 git tag v0.01

2037 git branch bugFix

2038 git status

2039 git status

2040 git reflog

2041 git reflog

2042 git reflog

2043 git reflog

2044 git branch -f bugFix 380b9b2

2045 git reflog

2046 history

user@localhost good11 $