**Лабораторная работа 1.**

**Установка и настройка Git.**

**Цель:**

1. Настроить git клиент на своей рабочей машине и изучить возможности git.

I. Установка и настройка Linux на VirtualBox (**этот пункт выполнять необязательно**)

Ссылка на скринкаст: <https://youtu.be/qEL2iHIDbhM>

Ссылка на презентацию из скринкаста: <https://docs.google.com/presentation/d/1AV6h4Ed4qef3bc9hKNRxJctbuHYLeiT6IfhClPM-5lU/edit?usp=sharing>

**II. Установка и настройка Git.**

[Git](https://git-scm.com/about) — распределённая система контроля версий, с помощью которой разработчики могут отслеживать изменения в файлах проекта и обмениваться ими внутри команды.

Git был создан в 2005 году Линусом Торвальдсом для работы над ядром Linux.

Сейчас трудно представить проект, который бы не использовал систему контроля версий, поэтому навык работы с системой контроля является одним из обязательных базовых навыков любого программиста.

**Шаги по установке и настройке клиента Git**

1. Чтобы поставить Git-клиент на свою машину выполните шаги описанные в этой [инструкции](https://git-scm.com/book/ru/v2/%D0%92%D0%B2%D0%B5%D0%B4%D0%B5%D0%BD%D0%B8%D0%B5-%D0%A3%D1%81%D1%82%D0%B0%D0%BD%D0%BE%D0%B2%D0%BA%D0%B0-Git).
2. Для того, чтобы можно было добавлять свои изменения в репозиторий необходимо произвести первичную настройку git, выполнив команды:

git config --global user.name "Your Name"

git config --global user.email "[your\_email@example.com](mailto:your_email@example.com)"

С помощью этих команд Вы укажите Ваши имя и e-mail. Когда вы будете вносить изменения в репозиторий git, он будет использовать эту информацию для обозначения автора правок.

**Изучение Git**

1. Для знакомство с git рекомендуется прочитать электронную книгу [Pro Git](https://git-scm.com/book/ru/v2) главы 2, 3 и 7. Это **необходимый минимум**.
2. Первая книга содержит достаточно много материала. Упорядочить его поможет [Git Handbook](https://guides.github.com/introduction/git-handbook/)
3. Для приобретения первичных навыков работы с Git необходимо полностью пройти [Learn Git Branching](https://learngitbranching.js.org/)
4. Изучить [Git Flow](https://habr.com/ru/post/106912/)

**Рекомендация!** В качестве шпаргалок по часто используемым командам используйте [Git Cheat Sheet](https://training.github.com/downloads/github-git-cheat-sheet.pdf)

**Рекомендация!** Для визуального представления истории коммитов можно воспользоваться <http://git-school.github.io/visualizing-git/>

**Критерии приемки задания**

Скрин с выполненными шагами.