# **--- Git and GitHub ---**

**Git -** система розпределеного контроля различных версий кода. Контроль и отслеживание изменений в коде за период жизни проекта. Не хранит в себе файлы, только отслеживает изменения

**GitHub** - облачное хранилище предоставляющее доступ для совместной работы програмистов над проектов. Хранит в себе git - репозитории. Для глобальной разработки

**Рабочая Директория** - название папки с проектом, которая хранит состояние файлов проекта и находиться под контролем версий git

**Git Репозиторий** **.git** - область, где происходит само управление версиями проекта. Храниться внутри скрытой папки **.git** Делиться на две зоны **↓**

**Staging Area Репозиторий (черновая зона)** - область репозитория, где храниться индексный файл. Перед добавлением в commit файлы сохраняються здесь. При разовом добавлении файла и зделаному коммиту будет храниться всегда в етой области, если только не удалим сам файл из проекта.

**Commits Area Репозитория (зона коммитов) -** область репозитория, где содержаться комиты проекта

**Commit** - сохранение состояния проекта, кода. Каждый коммит имеет свой идентификатор

**Хеш Коммита** - уникальное название для каждого коммита -> 06beccaf9af64ea48b1ffd6

**Файл -> Staging Area Репозитория -> Commits Area Репозитория -> GitHub (сохранение в облаке)**

**Ветка -** паралельное состояние проекта с другими изменениями в коде. Нужен для предотвращениея ризка сломать текущую версию проекта (+ добавить баги). Можно работать

**Ветка master ( main )** - название первой ветки в репозиторее, по-умолчанию все комиты сохраняються в ету ветку

**Merge ( слияние )** - обьединение воедину розделеную историю двух веток, они становятса одинаковыми

**Указатель ( Head -> )** - указывате на определенный коммит в текущей ветке. Каждая ветка имеет свой указатель. По-умолчанию указывает на последний коммит в текущей ветке. Можна переключать указатель на любой выбранный коммит

**Смещенный Указатель** - показывает не на последний коммит в текущей ветке, а на указаный в етой ще ветке

**.gitignore** - папка содержит внутри файлы которые на которых не будет контроля версий git и которые не будут записываться в коммиты ( например файлы логинов - файлы настройки ide). Git не может взаемодествовать с етими файлами. Работает один раз только после самого создания файла и еще не добавлен в stage (потом не работает)

**Git Stash** - хранит в себе не подготовленные и не закоммиченные изменения в проекте. Можно создавать несколько стешей. Можно использовать одновременно только один стеш

**Git Tags** - теги позволяют зафиксировать в истории ветки оперделенные моменты разработотки (сообщения, подсказки)

**Легковесные Теги** - указатель с сообщением

**Анотированные Теги -** полноценные обьекты git, имеют свой хеш, автора и дату когда был зделан

**Git Hub Actions** - позволяет автоматизаировать прочти все процесы разработки приложения. Например тестирование, развертывание(deploy) приложения на сервера, сборку всего проекта и уведомление о событиях в коде. Позволяет запускать Linux и Windows команды внутри нашего репозиятория с проектом без поднятия сервера