Git

Muammolarni hal qiladi

* ﻿﻿Kod versiyasini nazorat qilish
* ﻿﻿Qanday o'zgarishlar qilindi
* ﻿﻿Qachon va kim tomonidan kodga o'zgarishlar kiritildi
* ﻿﻿Loyihalar bilan jamoaviy ishlash
* ﻿﻿Open Source - loyihalarga yordam berish
* ﻿﻿Portfolio va ish

**Git Config**

git config - gitni konfiguratsiya qilish uchun ishlatiladi

Sozlanmani tekshirish

* ﻿﻿git config --list - barcha sozlanmalarni ko'rsatadi
* ﻿﻿git config user.name - git foydalanuvchining ismi
* ﻿﻿git config user.email - git foydalanuvchining emaili

**Git Sozlanmasi o'rnatish**

Shaxsni tasdiqlash

* ﻿﻿git config --globakuser.name "Sizning ismingiz"
* ﻿﻿git config --global user.email ism@email.com

**Git repository**

* ﻿﻿Repository - git yangi papka/loyiha
* ﻿﻿**git init** - yangi git loyihasini boshlash uchun ishlatiladi faqat 1 marta ishlating loyiha uchun
* ﻿﻿**git status** - o'zgarishga uchragan holatni bilish uchun

**git add**

* ﻿﻿git add - loyihadagi yangi fayllar, o'zgarishlar va o'chirilishlarni xotiraga olish uchun navbatga qo'yadi
* ﻿﻿--command lar - flag deyiladi
* **git add . ->**barcha o'zgarish, o'chirilish va qo'shilish
* ﻿-A--all - barcha o'zgarish, o'chirilish va qo'shilish

**git commit**

* **﻿﻿git commit** - xotiraga olingan o'zgarishlarni saqlab qo'yadi
* **﻿﻿-m** - xabar qoldirish uchun flag
* **﻿﻿git log / git log --pretty** - commit'lar ro'yxatini ko'rish

**git commit -m"some changes"**

**Fayl - o'zgarishlarni bekor qilish**

* ﻿﻿git checkout -- faylname- fayda bolgan o'zgarishlarni bekor qiladi
* ﻿﻿**git checkout – .** ->barcha fayllardagi o'zgarishlarni bekor qiladi

**Git add - bekor qilish**

* ﻿﻿git reset - staged - fayllarni unstage qiladi lekin fayl qiymatlari o'zgarmaydi
* ﻿﻿**git reset myFile.txt** - faqat berilgan fayini unstage qiladi
* ﻿﻿git reset HEAD

**git commit** - o'zgarishlarni bekor qilish

**git commit --amend** - so'nggi commit'ni qayta nomlash

**BRANCH** NIMA?

Yangi xususiyatlar bilan mustaqil ravishda ishlash

**branch yaratish**

* **﻿﻿git branch** - majud hozirgi branch nomi
* **﻿﻿git branch myBranch** - yangi branch yaratish
* **﻿﻿git checkout branchNomi** - branchga o'tish
* **﻿﻿git merge branchNom**i - berilgan branchni mavjud branchga yuklab olish

**git clone -** github dan loyihani ko'chirib olish uchun

**git remote add origin https://…..**

**git push -u origin main**

**Git push**

* **﻿﻿git remote add origin URL** - yangi github repository manzil
* **﻿﻿git push -u origin master** - git local loyihani githubga joylash
* **﻿﻿git remote -v** - github project manzilini bilish uchun

**Git pull**

**• git pull origin master** - so'ngi o'zgarishlarni githubdan oladi