Git – control verification system.Projectlarni jamoaviy bo’lib ishlashda saqlaydigan local tizim

Afzalliklari quyidagilar:

* Codeni kim tomonidan qachon (vaqti bilan) o’zgartirilganligini o’zida saqlaydi
* Agar code xato kiritilgan bo’lsa ixtiyorish vaqtdagi codega qaytarish imkonini beradi
* Jamoa bilan ishlashda qulay

1.git config – gitning sozlamalari ya’ni git foydalanuvchisinig ismi,electron pochtasi,git uchun manzili,saqlanadigan ma’lumotlar to’plami

2.git config –list – gitning sozlamalari ro’yxatini ko’rish

3.git config – user.name – git foydalanuvchisining nomini ko’rish

4.git config – user.email – git foydalanuvchisining electron pochtasini ko’rish

5.git config – global user.name “Faxrullo” – git foydalanuvchisining nomini kiritish

6.git config – global user.email “faxrullofayzullayev@gamil.com” – git foydalanuvchisining electron pochtasini kiritish

7.git init – git loyihani initsialia]zatsiya qilish(ishga tushirish)

8.touch filename – git loyiha o’rnatilgan papkaga fay[ qo’shish

9.git add filename – git loyihaga ko’rsatilgan faylni navbatga qo’shish

10.git add. – git loyihaga barcha fayllarni qo’shish

11.git status – gitning holatini ko’rish

12.git commit -m “message” – git loyihaga kiritilgan o’zgarishlarni “message” izohi bilan saqlash

13.git log – git loyihaga kiritilgan o’zgarishlarni yozilgan izohlarni va ularning vqatni ko’rish

14.git checkout code – oxirgi bo’lgan o’zgarishlarni toki codegacha bo’lganlarini bekor qilish

15.git reset – oxirgi navbatga qo’yilgan(git addning buyrug’ini bekor qilib beradi) o’zgarishlarni bekor qilish

16.git branch – mavjud branch(filial)larni ko’rastish uchun ishlaydi

17.git branch new\_branch – new\_branch nomli yangi filialni qo’shish uchun ishlatiladi

18.git switch branch\_name – joriy branchdan ko’ratilgan branchga o’tish uchun ishlatiladi

19.merge branch\_name – joriy filial va branch\_nameni o’zaro birlashtirish