|  |  |
| --- | --- |
|  | **Tugas** |
|  | *Task* |

Mata Kuliah / *Course Name* : **Dasar Pemrograman**

Sesi Pertemuan/ *Session Number* : **II (Dua)**

Judul Materi / *Title* : **Pengenalan GIT dan IDE Python**

Nama Dosen / *Lecturer* : **SHINTA AYUNINGTIAS, S.Kom., M.Kom.**

Jenis Tugas / Type of task : **Case Based / Project Based**

NIM / *Student ID Number* : 20240040032

Nama Mahasiswa/*Student Name* : Ahmad Hasan Maki

Kelas/*Class*  : TI24B

**Study Case Based 2:**

1. Berdasarkan materi yang telah diberikan, apa yang anda pahami tentang Git dan IDE Python?
2. Untuk melatih pemahaman anda, buatlah sebuah panduan lengkap tentang langkah-langkah melakukan sinkronisasi file ke dalam github menggunakan git command.

Jawaban Soal 1:

1. Git adalah sistem control atau sebuah tracker file, atau repository sebuah project atau pengelolaan catatan dengan kemampuan mendeteksi setiap perubahan, contros versi dan lain nya.
2. IDE Python adalah Integrated Development Software untuk pengembangan python, dengan beberapa fitur yang sangat baik seperti , auto completion , syntax highlight, linter dan lain lain.
3. Jika bekerja dengan data biasa nya menggunakan
4. Juptyer notebook
5. Spyder
6. Google Collabs

Jika bekerja dengan software , API dan lain lain lain

1. VSCODE + Extension +package manager (UV, Poetry dan lain2)
2. PYCHARM
3. IDLE : Command Line bawaan Python
4. VIM

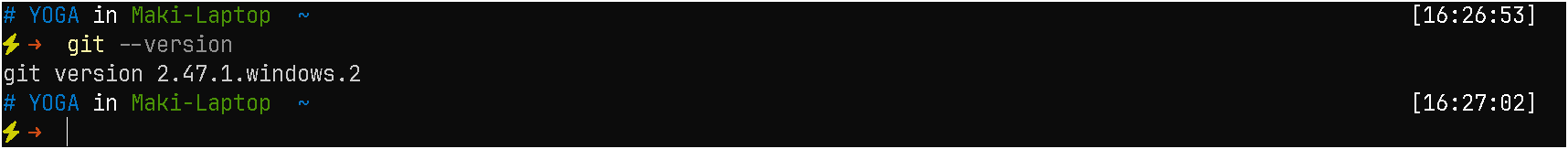
Jawaban Soal 2

Ada Beberapa Tahapan untuk instalasi dan pengaturan Git:

1. Install Git Dan Setup
2. Register Account GitHub dan Setup SSH
3. Membuat Project
4. Sync Local Project ke Repository GitHub

Pada sesi kali ini env : windows x64, menggunakan windows terminal | powershell

Install Git Dan Setup Github

1. Download(pilih sesuai Environment):
   1. Download dan Install <https://git-scm.com/downloads>
   2. Jika Menggunakan Scoop, buka windows terminal “scoop Install Git”
   3. Jika Menggunakan winget “winget install --id Git.Git -e --source winget
   4. Check Installasi dengan “git –version” jika muncul versi , maka git sudah terinstall
2. Setup Git
   1. Masukan nama Dan Email dengan command Berikut
      1. git config --global user.name "Nama\_Lengkap"
      2. git config --global user.email [email@example.com](mailto:email@example.com)
   2. Check Hasil Konfigurasi dengan command berikut:
      1. A black rectangle with red border

         AI-generated content may be incorrect.git config --global –list
3. Registrasi Di Github
   1. Make Sure menggunakan Email yang sama
   2. Verifikasi sampai selesai
4. Setup SSH ke Github
   1. Buka terminal dan open directory :
      1. cd ~/.ssh/
      2. ssh-keygen -t rsa
      3. masukan nama / identifier untuk id
      4. enter saja saja di minta **passphrase**
      5. hasil dari command tadi adalah 2 file:
         1. .pub – public key
         2. Nama (file) – private key
      6. Buat file config dengan command ini:
         1. New-Item -Path $HOME\.ssh\config -ItemType File
         2. Masukan setup ini :

Host github.com

HostName github.com

User git

IdentityFile ~/.ssh/{private key anda}

AddKeysToAgent yes

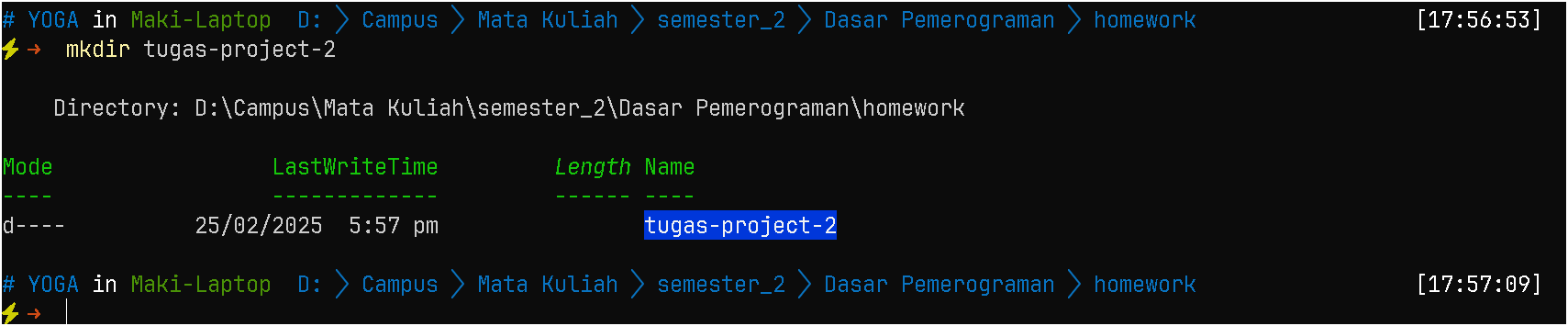
* 1. Buka Github.com
     1. Copy dulu key dengan perintah ini di terminal
        1. cat ~/.ssh/id\_{publickeyanda}.pub
        2. Buka Github → Settings and Gpg Keys
        3. New ssh key
        4. Title : isi missal nama laptop / lain nya
        5. Key Type: Auth key
        6. Key: Paste semua hasil copy dari terminal
  2. Uji Coba Koneksi
     1. Lakukan Uji Coba Koneksi Dengan perintah :
        1. ssh-add {privatekeyanda}
        2. ssh -T [git@github.com](mailto:git@github.com)

Jika Muncul seperti di bawah ini , berarti setup sukses



* + 1. setup branch utama default di terminal:
       1. git config --global init.defaultBranch main

1. Buat Project
   1. A screenshot of a computer screen

      AI-generated content may be incorrect.Navigasikan Dari Terminal ke folder Project dam buat folder (check gambar)
2. Setelah kita membuka Vscode dan ada di project
   1. Inisilisasi Local Repo:
      1. git init

A black background with text

AI-generated content may be incorrect.

1. buat file baru project misal readme.md

A black screen with white text

AI-generated content may be incorrect.

1. Tambahkan semua File ke staging
   1. git add .
2. Lakukan Commit pertama
   1. git commit -m "Initial commit"

A screenshot of a computer

AI-generated content may be incorrect.

1. Buka Github.com dan buat repository, lalu copy url ssh nya dan lakukan push

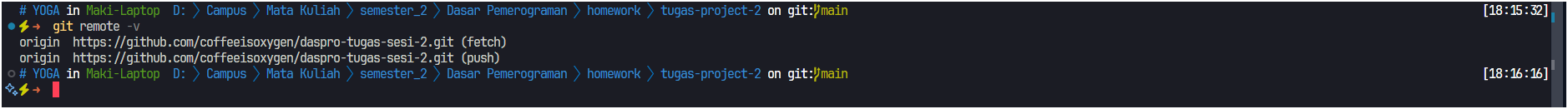
A screenshot of a computer program

AI-generated content may be incorrect.

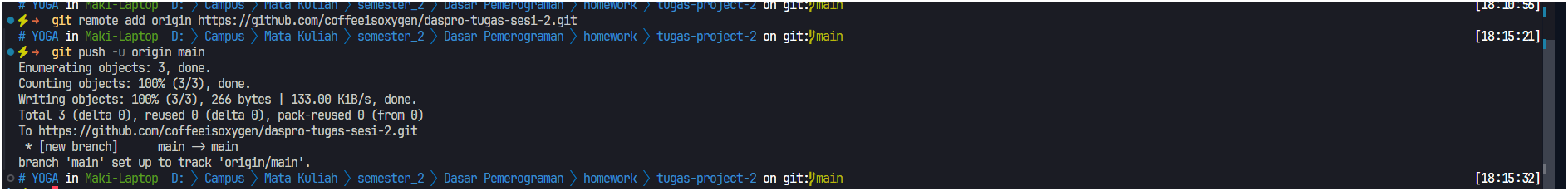
Paste ke terminal



Setelah enter coba test remote



Coba lakukan push



Repository sudah ada di github

A screenshot of a computer

AI-generated content may be incorrect.

Selesai   
selanjut nya hanya tinggal git push saja

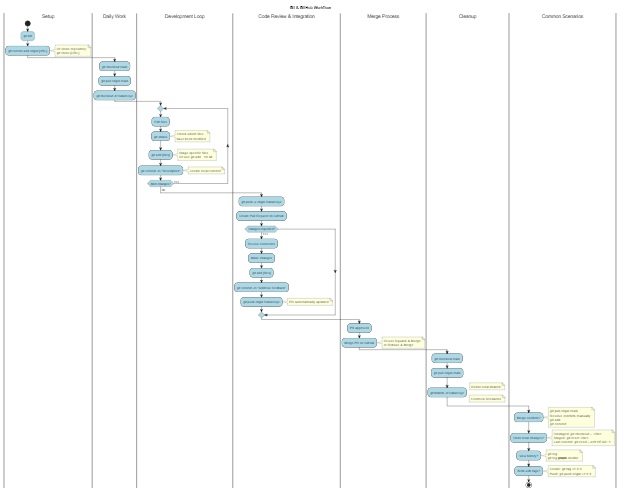
Di vscode sendiri sudah integrated untuk setup github, jadi bisa menghindari CLI (jika suka GUI)

Ada banyak sekali command command untuk git, dan workflow nya   
  
berikut adalah best practice nya

1. Lakukan Branching → Jangan di Main terus , jadi setiap development fitu baru buat branch, ini meminimalisir conflict di main
2. Hati Hati saat meger → pastikan sudah lulus unit test , dan selesaikan conflict dengan hati hati

A diagram of a git

AI-generated content may be incorrect.



1. @startuml "git-workflow"

2. skinparam backgroundColor white

3. skinparam activityBorderColor black

4. skinparam activityBackgroundColor lightBlue

5. skinparam arrowColor black

6.

7. title Git & GitHub Workflow

8.

9. |Setup|

10. start

11. :git init;

12. :git remote add origin [URL];

13. note right: Or clone repository:\ngit clone [URL]

14.

15. |Daily Work|

16. :git checkout main;

17. :git pull origin main;

18. :git checkout -b feature/xyz;

19.

20. |Development Loop|

21. repeat

22.   :Edit files;

23.   :git status;

24.   note right: Check which files\nhave been modified

25.   :git add [files];

26.   note right: Stage specific files\nOr use git add . for all

27.   :git commit -m "Description";

28.   note right: Create local commit

29. repeat while (More changes?) is (Yes)

30. ->No;

31.

32. |Code Review & Integration|

33. :git push -u origin feature/xyz;

34. :Create Pull Request on GitHub;

35. if (Changes requested?) then (Yes)

36.   :Review comments;

37.   :Make changes;

38.   :git add [files];

39.   :git commit -m "Address feedback";

40.   :git push origin feature/xyz;

41.   note right: PR automatically updated

42. endif

43.

44. |Merge Process|

45. :PR approved;

46. :Merge PR on GitHub;

47. note right: Or use Squash & Merge\nor Rebase & Merge

48.

49. |Cleanup|

50. :git checkout main;

51. :git pull origin main;

52. :git branch -d feature/xyz;

53. note right: Delete local branch

54.

55. |Common Scenarios|

56. floating note right: Common Scenarios

57. :Merge conflicts?;

58. note right

59.   git pull origin main

60.   Resolve conflicts manually

61.   git add .

62.   git commit

63. end note

64.

65. :Undo local changes?;

66. note right

67.   Unstaged: git checkout -- <file>

68.   Staged: git reset <file>

69.   Last commit: git reset --soft HEAD~1

70. end note

71.

72. :View history?;

73. note right

74.   git log

75.   git log --graph --oneline

76. end note

77.

78. :Work with tags?;

79. note right

80.   Create: git tag v1.0.0

81.   Push: git push origin v1.0.0

82. end note

83.

84. stop

85. @enduml