

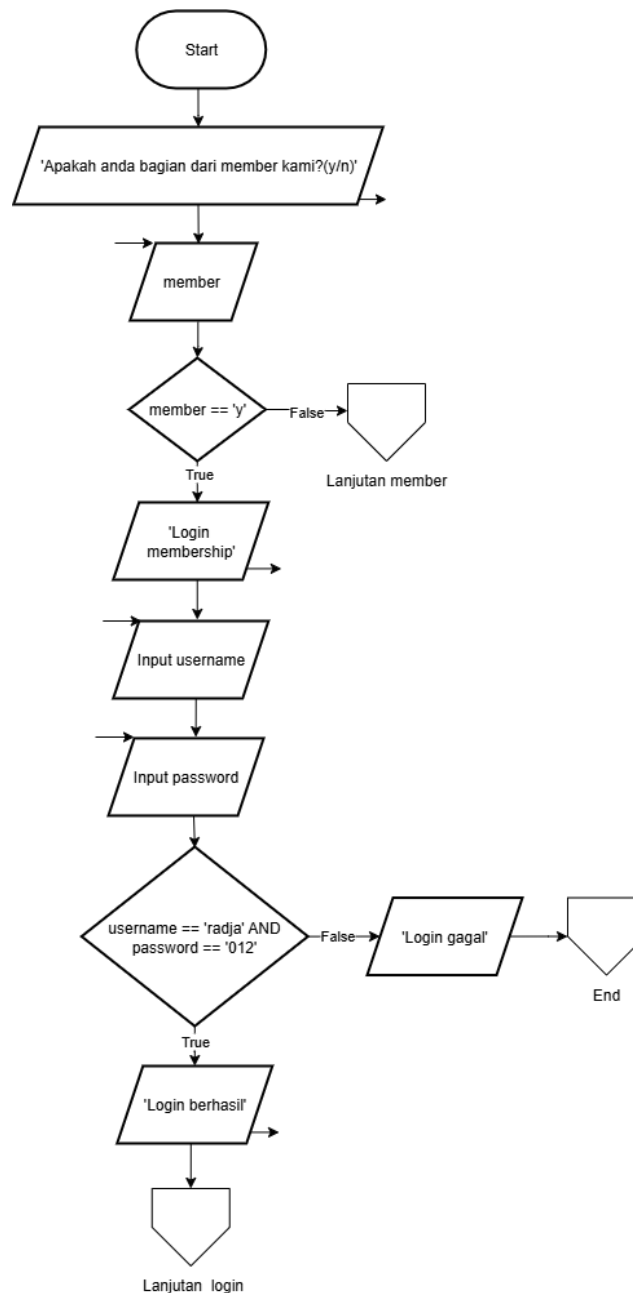
**LAPORAN PRAKTIKUM**  
**POSTTEST 3**  
**ALGORITMA PEMROGRAMAN DASAR**



**Disusun oleh:**  
**Muhamad Radja Nur Akbar (2509106012)**  
**Kelas (A1'25)**

**PROGRAM STUDI INFORMATIKA**  
**UNIVERSITAS MULAWARMAN**  
**SAMARINDA**  
**2025**

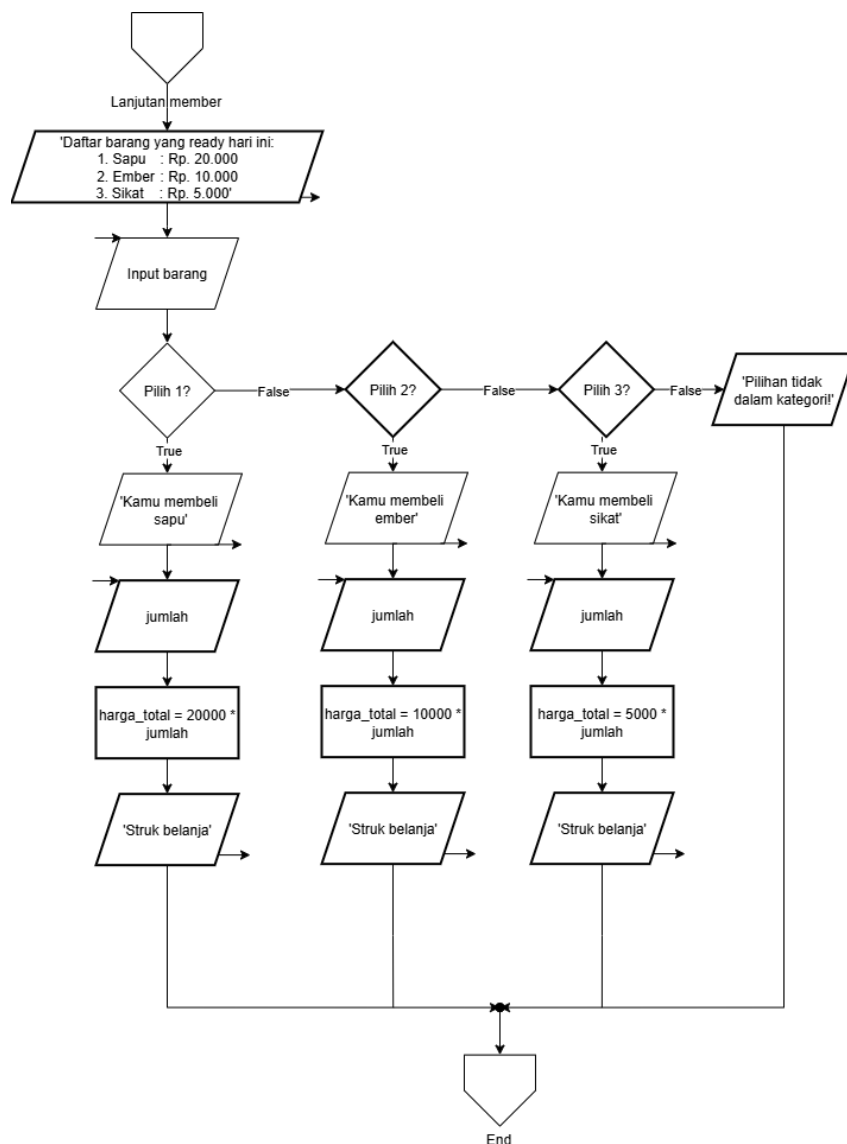
## 1. Flowchart



Gambar 1.1 Flowchart Bagian Membership dan Login

*Flowchart* menampilkan pesan “apakah pengguna adalah bagian dari membership?” Jika pengguna mengetik ‘y’ maka program akan berlanjut ke “login membership”. Jika tidak, pengguna akan berbelanja sebagai pelanggan biasa tanpa adanya diskon. Pengguna akan diminta memasukkan *username* dan *password*, jika sesuai maka program berlanjut namun jika salah satu *input* salah maka program langsung berakhir. *Flowchart* ini menggunakan simbol *Off-Page Reference* yaitu simbol yang menghubungkan dua halaman atau lembar

*flowchart* yang berbeda. Setiap simbol *Off-Page Reference* harus diberi nama dan *flowchart* yang dilanjutkan harus dimulai dengan nama dari *Off-Page Reference* tersebut.

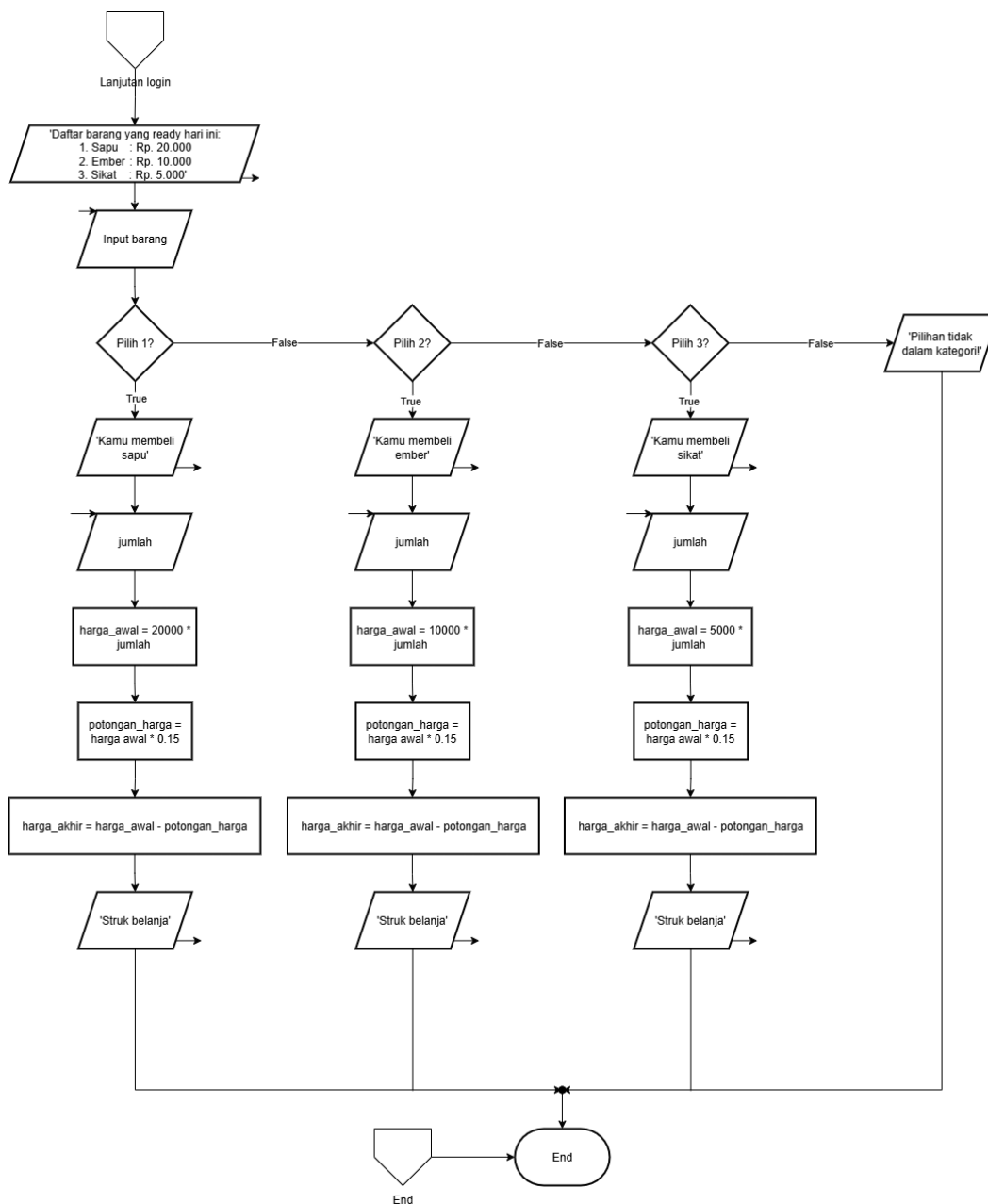


Gambar 1.2 Flowchart Sebagai Pelanggan Biasa

Gambar 1.2 melanjutkan *flowchart* jika dalam percabangan pertama pengguna bukan bagian dari *membership*, di sini sudah disiapkan tiga jenis barang dengan harga yang berbeda-beda. Sesudah memilih salah satu jenis barang, pengguna diminta untuk memasukkan jumlah barang yang ingin dibeli, lalu jumlah dan harga barang yang dipilih sebelumnya akan dikali untuk mendapat harga total. Sesudah itu struk belanja akan dikeluarkan dengan menampilkan jenis barang, jumlah (*quantity*), dan harga total barang.

Begitu juga dengan jenis barang yang lain, setiap barang memiliki proses yang sama, ketika struk sudah dicetak maka program berakhir, hanya harga barang saja yang menjadi

pembeda suatu barang dengan barang lain. Jika pengguna memasukkan angka yang di luar kategori maka *output* akan berupa “Pilihan tidak dalam kategori!” dan program berakhir.



Gambar 1.3 Flowchart Membership

Gambar 1.3 melanjutkan fitur login yang *username* dan *password*-nya sesuai, perbedaan antara *membership* dan pelanggan biasa yaitu *membership* akan mendapat diskon sebesar 15% untuk semua jenis barang yang dibeli. Pengguna diminta untuk memilih salah satu dari tiga jenis barang yang tersedia, lalu jumlah barang yang ingin dibeli. Setelah itu jumlah barang dan harga satuan dikali untuk mendapat harga awal.

Harga awal tadi akan mendapat diskon 15% ( $\text{harga awal} - (\text{harga awal} \times 15\%)$ ) dan terakhir struk akan dicetak dan program berakhir. Ini juga berlaku untuk setiap jenis barang yang lain, tentunya setiap barang memiliki harga yang berbeda-beda. Jika pengguna memasukkan pilihan selain 1, 2, dan 3 akan menampilkan *output* “Pilihan tidak dalam kategori” dan program berakhir.

## 2. Deskripsi Singkat Program

Tujuan dari program ini untuk membuat fitur login bagi para *membership* sehingga tidak bergabung dengan mereka yang adalah pelanggan biasa. Program ini juga memberi kelebihan bagi para *membership* yaitu diskon untuk setiap jenis barang yang dibeli.

## 3. Source Code

### A. Fitur Pelanggan Membership

```
username = 'radja'
password = '012'
print(''---TOKO CERIA---
Selamat datang di Toko Ceria'')

print('Anda akan mendapat diskon 15% jika bagian dari member kami')
member = input('Apakah anda bagian dari membership?(y/n):')
```

Gambar 3.1 Fitur Pelanggan Membership

Dalam fitur ini isi dari variabel *username* dan *password* sudah ditetapkan yang nantinya akan digunakan dalam fitur *login*, pengguna meminta pengguna untuk memasukkan apakah mereka bagian dari *membership* atau bukan dengan mengetikkan ‘y’ atau ‘n’.

### B. Fitur Login

```
if member == 'y':
    print('Login Membership')
    username_kamu = input('Username\t:')
    password_kamu = input('Password\t:')
```

Gambar 3.2 Fitur Login

Jika pengguna mengetik 'y' pada fitur sebelumnya, menu *login* akan ditampilkan, pengguna diminta untuk memasukkan *username* dan *password* yang sesuai dengan isi dari kedua variabel tersebut sesuai dengan yang ditampilkan di fitur sebelumnya.

### C. Ternary Operator

```
status_login = True if username == username_kamu and password == password_kamu
else False
```

Gambar 3.3 Ternary Operator

*Ternary Operator* merupakan bentuk lain dari percabangan yang diketik dalam satu baris, di sini kita juga bisa memasukkan lebih dari dua variabel, namun karena fungsi dari bentuk ini untuk penyederhanaan, dua variabel sudah cukup untuk menyusun bentuk ini. Alasan kenapa menggunakan *boolean* yaitu agar terlihat lebih sederhana lagi dan kita hanya perlu memasukkan *True/False* pada percabangan berikutnya.

### D. Percabangan Bersarang

```
if status_login == True:
    print('Login berhasil')
    print('''Daftar barang yang ready hari ini:
1.Sapu\t: Rp. 20.000
2.Ember\t: Rp. 10.000
3.Sikat\t: Rp. 5.000''')
    barang = input('Barang apa yang ingin anda beli?:')

    if barang == '1':
        print('Kamu membeli sapu')
        jumlah = int(input('Berapa banyak barang yang ingin dibeli?:'))
        harga_awal = 20000 * jumlah
        potongan_harga = harga_awal * 0.15
        harga_akhir = harga_awal - potongan_harga
        print(f'''---STRUK BELANJA---
Barang\t      : Sapu
Harga satuan\t: Rp. 20.000/unit
Jumlah\t      : {jumlah}
Harga awal\t: Rp. {harga_awal}
Diskon\t      : 15%
Potongan harga\t: Rp. {potongan_harga}
```

```

Harga akhir\t: Rp. {harga_akhir}
---TERIMA KASIH TELAH BERBELANJA---')

    elif barang == '2':
        print('Kamu membeli ember')
        jumlah = int(input('Berapa banyak barang yang ingin dibeli?'))
        harga_awal = 10000 * jumlah
        potongan_harga = harga_awal * 0.15
        harga_akhir = harga_awal - potongan_harga
        print(f'---STRUK BELANJA---

Barang\t      : Ember
Harga satuan\t: Rp. 10.000/unit
Jumlah\t       : {jumlah}
Harga awal\t   : Rp. {harga_awal}
Diskon\t       : 15%
Potongan harga\t: Rp. {potongan_harga}
Harga akhir\t  : Rp. {harga_akhir}
---TERIMA KASIH TELAH BERBELANJA---')

    elif barang == '3':
        print('Kamu membeli sikat')
        jumlah = int(input('Berapa banyak barang yang ingin dibeli?'))
        harga_awal = 5000 * jumlah
        potongan_harga = harga_awal * 0.15
        harga_akhir = harga_awal - potongan_harga
        print(f'---STRUK BELANJA---

Barang\t      : Sikat
Harga satuan\t: Rp. 5.000/unit
Jumlah\t       : {jumlah}
Harga awal\t   : Rp. {harga_awal}
Diskon\t       : 15%
Potongan harga\t: Rp. {potongan_harga}
Harga akhir\t  : Rp. {harga_akhir}
---TERIMA KASIH TELAH BERBELANJA---')

    else:
        print('Pilihan tidak dalam kategori!')
else:
    print('Login gagal')

```

Gambar 3.4 Percabangan Bersarang

Percabangan bersarang dalam Python adalah kondisi di mana blok *if/elif/else*, berada di dalam blok *if/elif/else* lainnya. Seperti pada *Gambar 3.4* setelah status *login* adalah *True*, daftar barang yang tersedia akan ditampilkan beserta harga satuannya, Di sini terdapat 3 jenis barang yang masing-masing memiliki harga berbeda, program meminta pengguna memilih salah satu barang dengan mengetik angka '1', '2', atau '3'.

Setelah memilih salah satu, proses berikutnya yaitu permintaan *input* jumlah barang yang ingin dibeli. Lanjut dari harga satuan dan jumlah tadi didapat harga awal dengan mengalikan keduanya. Harga awal tadi kemudian dikali dengan 15% yaitu diskon sebagai seorang *membership* dan diperoleh potongan harga, lalu ada harga akhir yang diperoleh dengan mengurangi harga awal dengan potongan harga. Terakhir ada percetakan struk yang menampilkan jenis barang, harga satuan, jumlah barang, harga awal, diskon, potongan harga, dan harga akhir, setelah itu program berakhir. Jika pengguna memasukkan pilihan yang tidak dalam daftar, program akan berakhir dengan pesan "Pilihan tidak dalam kategori!".

```
if status_login == True:
```

```
else:
    print('Login gagal')
```

*Gambar 3.5 Login Gagal*

Sedikit kembali ke *login*, jika *username* dan *password* yang dimasukkan salah, *login* akan gagal dan program berakhir.

## E. Program Tanpa Membership

```
else:
    print('Anda berbelanja sebagai pelanggan biasa!')
    print('Daftar barang yang ready hari ini:
1.Sapu\t: Rp. 20.000
2.Ember\t: Rp. 10.000
3.Sikat\t: Rp. 5.000')
    barang = input('Barang apa yang ingin anda beli?:')

    if barang == '1':
        print('Kamu membeli sapu')
        jumlah = int(input('Berapa banyak barang yang ingin dibeli?:'))
        harga_total = 20000 * jumlah
```



```

        print(f'''---STRUK BELANJA---
Barang\t      : Sapu
Harga satuan\t: Rp. 20.000/unit
Jumlah\t       : {jumlah}
Harga total\t: Rp. {harga_total}
---TERIMA KASIH TELAH BERBELANJA---''')

    elif barang == '2':
        print('Kamu membeli ember')
        jumlah = int(input('Berapa banyak barang yang ingin dibeli?:'))
        harga_total = 10000 * jumlah
        print(f'''---STRUK BELANJA---
Barang\t      : Ember
Harga satuan\t: Rp. 10.000/unit
Jumlah\t       : {jumlah}
Harga total\t: Rp. {harga_total}
---TERIMA KASIH TELAH BERBELANJA---''')

    elif barang == '3':
        print('Kamu membeli sikat')
        jumlah = int(input('Berapa banyak barang yang ingin dibeli?:'))
        harga_total = 5000 * jumlah
        print(f'''---STRUK BELANJA---
Barang\t      : Sikat
Harga satuan\t: Rp. 5.000/unit
Jumlah\t       : {jumlah}
Harga total\t: Rp. {harga_total}
---TERIMA KASIH TELAH BERBELANJA---''')

    else:
        print('Pilihan tidak dalam kategori!')

```

Gambar 3.6 Program Tanpa Membership

Gambar 3.6 merupakan kelanjutan program jika pengguna bukan bagian dari *membership*, prosesnya sama saja dengan yang *membership*, bedanya tidak menggunakan diskon, setelah memproses harga total struk akan langsung dicetak dan program berakhir.

#### 4. Hasil Output

```
---TOKO CERIA---
Selamat datang di Toko Ceria
Anda akan mendapat diskon 15% jika bagian dari member kami
Apakah anda bagian dari membership?(y/n):y
Login Membership
Username      :radja
Password      :012
Login berhasil
Daftar barang yang ready hari ini:
1.Sapu   : Rp. 20.000
2.Ember  : Rp. 10.000
3.Sikat  : Rp. 5.000
Barang apa yang ingin anda beli?:2
Kamu membeli ember
Berapa banyak barang yang ingin dibeli?: 4
---STRUK BELANJA---
Barang      : Ember
Harga satuan : Rp. 10.000/unit
Jumlah      : 4
Harga awal  : Rp. 40000
Diskon      : 15%
Potongan harga : Rp. 6000.0
Harga akhir  : Rp. 34000.0
---TERIMA KASIH TELAH BERBELANJA---
PS D:\Praktikum APD>
```

*Gambar 4.1 Output Jika Bagian dari Membership dan Berhasil Login*

```
---TOKO CERIA---
Selamat datang di Toko Ceria
Anda akan mendapat diskon 15% jika bagian dari member kami
Apakah anda bagian dari membership?(y/n):y
Login Membership
Username      :radja
Password      :radja
Login gagal
```

*Gambar 4.2 Output Jika Salah Memasukkan Input Login*

```

---TOKO CERIA---
Selamat datang di Toko Ceria
Anda akan mendapat diskon 15% jika bagian dari member kami
Apakah anda bagian dari membership?(y/n):Y
Anda berbelanja sebagai pelanggan biasa!
Daftar barang yang ready hari ini:
1.Sapu   : Rp. 20.000
2.Ember  : Rp. 10.000
3.Sikat  : Rp. 5.000
Barang apa yang ingin anda beli?:4
Pilihan tidak dalam kategori!

```

*Gambar 4.3 Output Jika Jenis Barang tidak dalam Kategori*

```

---TOKO CERIA---
Selamat datang di Toko Ceria
Anda akan mendapat diskon 15% jika bagian dari member kami
Apakah anda bagian dari membership?(y/n):n
Anda berbelanja sebagai pelanggan biasa!
Daftar barang yang ready hari ini:
1.Sapu   : Rp. 20.000
2.Ember  : Rp. 10.000
3.Sikat  : Rp. 5.000
Barang apa yang ingin anda beli?:3
Kamu membeli sikat
Berapa banyak barang yang ingin dibeli?:3
---STRUK BELANJA---
Barang           : Sikat
Harga satuan     : Rp. 5.000/unit
Jumlah           : 3
Harga total      : Rp. 15000
---TERIMA KASIH TELAH BERBELANJA---

```

*Gambar 4.4 Output Jika Pengguna tidak Bagian dari Member*

## 5. Langkah-Langkah GIT

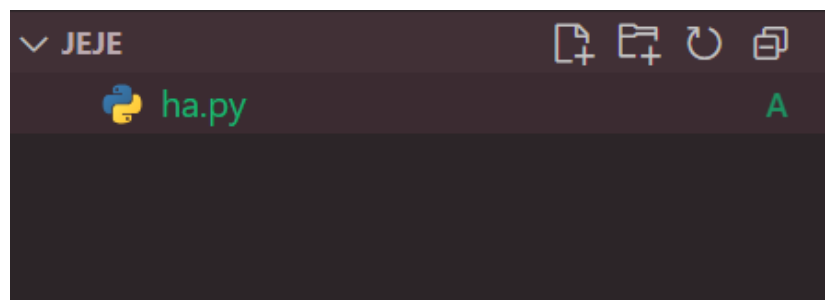
Jika dalam suatu folder kita sudah pernah melakukan *git init* dan *git remote* maka kita tidak perlu lagi melakukannya dan langsung ke *git add*.

### 5.1 GIT Add

```
PS D:\jeje> git add ha.py
PS D:\jeje> git add .
PS D:\jeje> 
```

Gambar 5.1 Proses Git Add

*Git add* berfungsi untuk memindah semua perubahan di area kerja ke indeks, ada dua cara untuk melakukan *git add*, yang pertama dengan mengetikkan nama *file* yang ingin di *add*. Jika kalian ingin menambahkan banyak *file* sekaligus, ketik “*git add .*”



Gambar 5.2 Sebuah File yang Telah di Add

*File* yang telah di *add* akan menampilkan huruf ‘A’ tepat di sebelah nama *file*, ini menandakan *file* sudah ditambahkan.

### 5.2 GIT Commit

```
PS D:\jeje> git commit -m "hellothere"
[main (root-commit) 55b666b] hellothere
1 file changed, 1 insertion(+)
create mode 100644 ha.py
PS D:\jeje> 
```

Gambar 5.3 Proses Git Commit

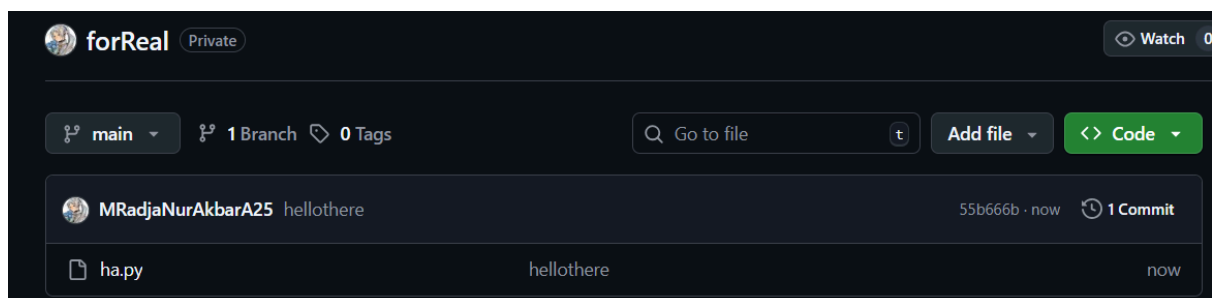
*Git commit* berfungsi untuk mengkonfirmasi setiap perubahan pada *repository* kalian dengan mengetikkan “`git commit -m ‘pesan yang ingin ditulis’`”

### 5.3 GIT Push

```
PS D:\jeje> git push -u origin main
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 235 bytes | 235.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/MRadjaNurAkbarA25/forReal.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
PS D:\jeje> █
```

*Gambar 5.4 Repository Lokal diunggah ke Github*

*Git push* berfungsi untuk mengunggah *file* lokal tadi ke GitHub kalian dengan mengetikkan “`git push -u origin main`”.



*Gambar 5.5 File sudah terunggah di Akun Github*