

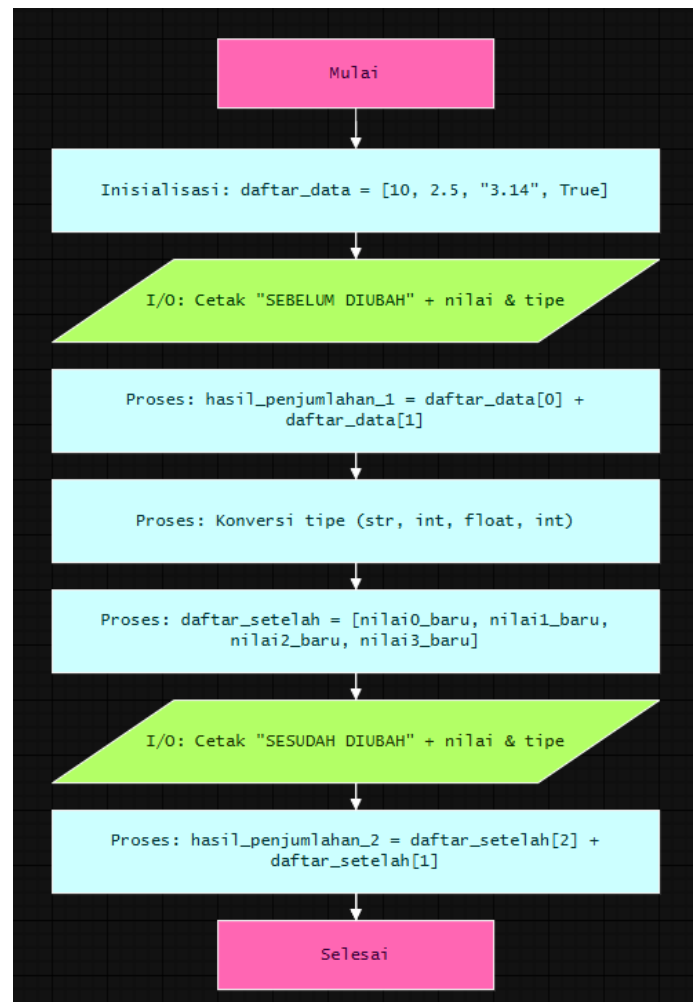
**LAPORAN PRAKTIKUM**  
**POSTTEST (2)**  
**ALGORITMA PEMROGRAMAN DASAR**



**Disusun oleh:**  
**Mutia Rahmah (2509106040)**  
**Kelas (A2 ' 25)**

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

## 1. Flowchart



Gambar 1.1 Flowchart

*Flowchart di atas menjelaskan alur program yang dimulai dengan membuat sebuah list berisi data dengan tipe berbeda, lalu menampilkan nilai dan tipe data sebelum diubah. Setelah itu, program melakukan penjumlahan dua elemen pertama dari list, kemudian mengubah tipe data setiap elemen sesuai kebutuhan (string, integer, float, dan integer). Hasil konversi disimpan dalam list baru dan ditampilkan kembali beserta tipe datanya. Terakhir, program melakukan penjumlahan ulang pada elemen tertentu dari list yang sudah diubah, lalu program selesai dijalankan.*

## 2. Deskripsi Singkat Program

*Tujuan saya membuat program ini untuk untuk memperlihatkan bagaimana data dalam sebuah list yang berisi berbagai tipe (integer, float, string, dan boolean) dapat ditampilkan, dikonversi ke tipe data lain (type casting), lalu digunakan dalam operasi perhitungan. Akhirnya, program ini membantu memahami perbedaan hasil operasi sebelum dan sesudah konversi tipe data.*

## 3. Source Code

```
daftar_data = [10, 2.5, "3.14", True]

print("=== SEBELUM DIUBAH ===")
print("1.", "nilai =", daftar_data[0], "| tipe =",
type(daftar_data[0]))
print("2.", "nilai =", daftar_data[1], "| tipe =",
type(daftar_data[1]))
print("3.", "nilai =", daftar_data[2], "| tipe =",
type(daftar_data[2]))
print("4.", "nilai =", daftar_data[3], "| tipe =",
type(daftar_data[3]))
hasil_penjumlahan_1 = daftar_data[0] + daftar_data[1]
print ("hasil penjumlahan = ",hasil_penjumlahan_1)

nilai0_baru = str(daftar_data[0])
nilai1_baru = int(daftar_data[1])
nilai2_baru = float(daftar_data[2])
nilai3_baru = int(daftar_data[3])

daftar_setelah = [nilai0_baru, nilai1_baru, nilai2_baru, nilai3_baru]
```

```
print("\n=== SESUDAH DIUBAH ===")
print("1.", "nilai =", daftar_setelah[0], "| tipe =",
type(daftar_setelah[0]))
print("2.", "nilai =", daftar_setelah[1], "| tipe =",
type(daftar_setelah[1]))
print("3.", "nilai =", daftar_setelah[2], "| tipe =",
type(daftar_setelah[2]))
print("4.", "nilai =", daftar_setelah[3], "| tipe =",
type(daftar_setelah[3]))
hasil_penjumlahan_2 = daftar_setelah[2] + daftar_setelah[1]
print ("hasil penjumlahan = ",hasil_penjumlahan_2)
```

## 4. Hasil Output

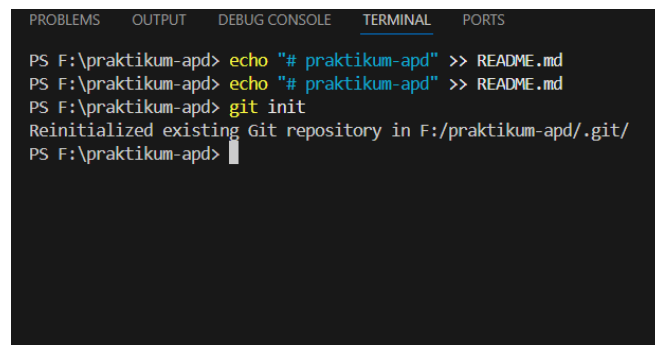
```
=== SEBELUM DIUBAH ===
1. nilai = 10 | tipe = <class 'int'>
2. nilai = 2.5 | tipe = <class 'float'>
3. nilai = 3.14 | tipe = <class 'str'>
4. nilai = True | tipe = <class 'bool'>
hasil penjumlahan = 12.5

=== SESUDAH DIUBAH ===
1. nilai = 10 | tipe = <class 'str'>
2. nilai = 2 | tipe = <class 'int'>
3. nilai = 3.14 | tipe = <class 'float'>
4. nilai = 1 | tipe = <class 'int'>
hasil penjumlahan = 5.1400000000000001
```

Gambar 4.1 Hasil Output

## 5. Langkah-langkah GIT

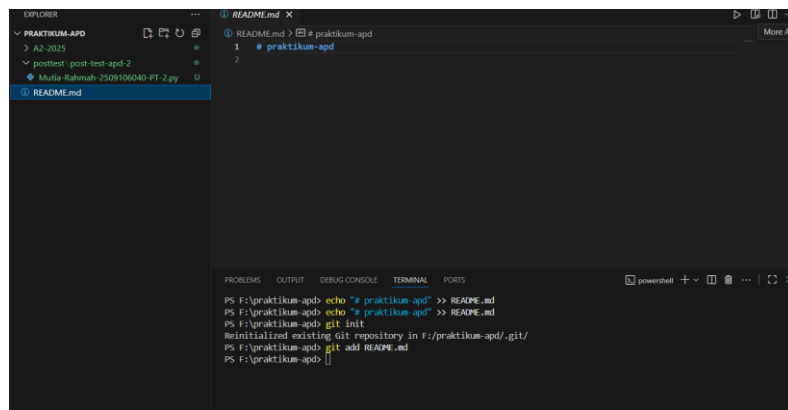
### 5.1 GIT Init



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS F:\praktikum-apd> echo "# praktikum-apd" >> README.md
PS F:\praktikum-apd> echo "# praktikum-apd" >> README.md
PS F:\praktikum-apd> git init
Reinitialized existing Git repository in F:/praktikum-apd/.git/
PS F:\praktikum-apd>
```

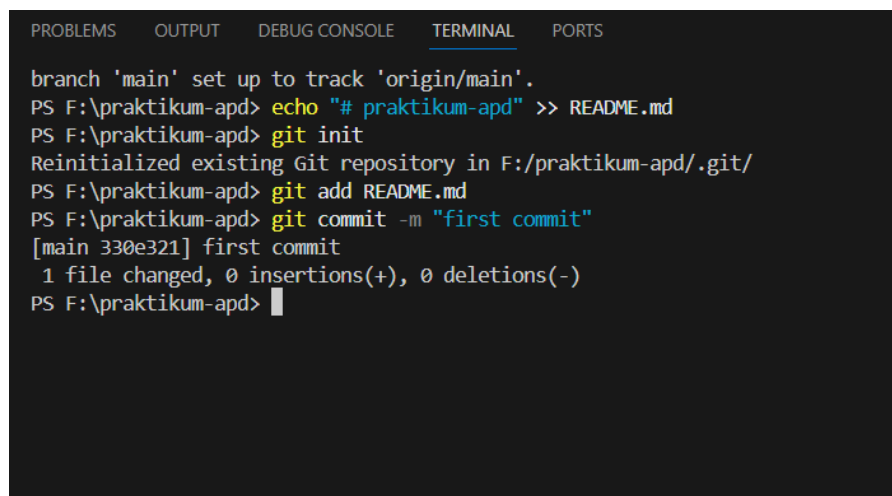
fungsi dari GIT Init ini adalah agar kita bisa ngetrack progress yang ada di folder tersebut jika misalnya ada perubahan atau tambahan.

## 5.2 GIT Add



Fungsi dari GIT Add ini adalah untuk menambah file contohnya dengan menulis git add (nama file) setelah melakukan GIT Init.

## 5.3 GIT Commit



Fungsi dari GIT Commit adalah untuk mengkonfirmasi perubahan pada file, dan jika di github akan terlihat bahwa perubahan apa saja yang sudah diubah.

## 5.4 GIT Remote

```
PS F:\praktikum-apd> git remote add origin https://github.com/mouriiuu/praktikum-apd.git
PS F:\praktikum-apd>
```

Fungsi dari GIT Remote ini adalah untuk menyambungkan repository yang ada di git dengan Github.

## 5.5 GIT Push

```
PS F:\praktikum-apd> git push -u origin main
Enumerating objects: 11, done.
Counting objects: 100% (11/11), done.
Delta compression using up to 12 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (8/8), 898 bytes | 898.00 KiB/s, done.
Total 8 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/mouriiuu/praktikum-apd.git
   330e321..5cc2da5  main -> main
branch 'main' set up to track 'origin/main'.
PS F:\praktikum-apd>
```

Fungsi dari GIT Push ini adalah untuk mengupload file yang awalnya di komputer ke Github.