

**TUGAS JURNAL  
KONSTRUKSI PERANGKAT LUNAK**

**MODUL V**

**GENERICICS**



**Disusun Oleh :**

**Andera Singgih Pratama / 2211104007**

**SE-06-01**

**Asisten Praktikum:**

**Naufal El Kamil Aditya Pratama Rahman**

**Imelda**

**Dosen Pengampu :**

**Yudha Islami Sulistya, S.Kom., M.Cs.**

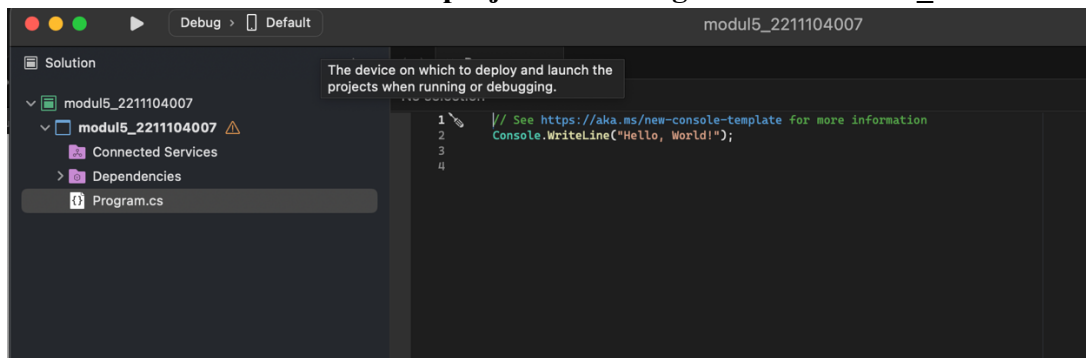
**PROGRAM STUDI S1 REKAYASA PERANGKAT LUNAK**

**FAKULTAS INFORMATIKA**

**TELKOM UNIVERSITY PURWOKERTO**

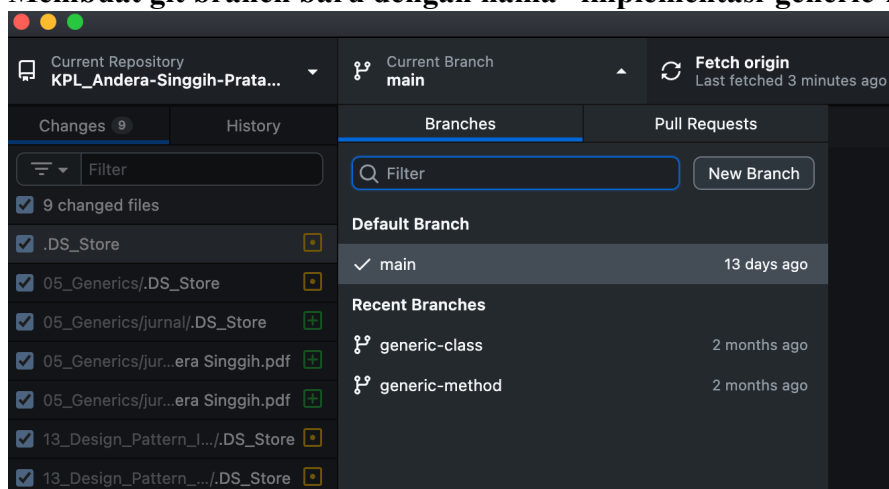
**2025**

## TUGAS JURNAL 1. Membuat project baru dengan nama modul5\_NIM



### 2. Melakukan git push ke github repo

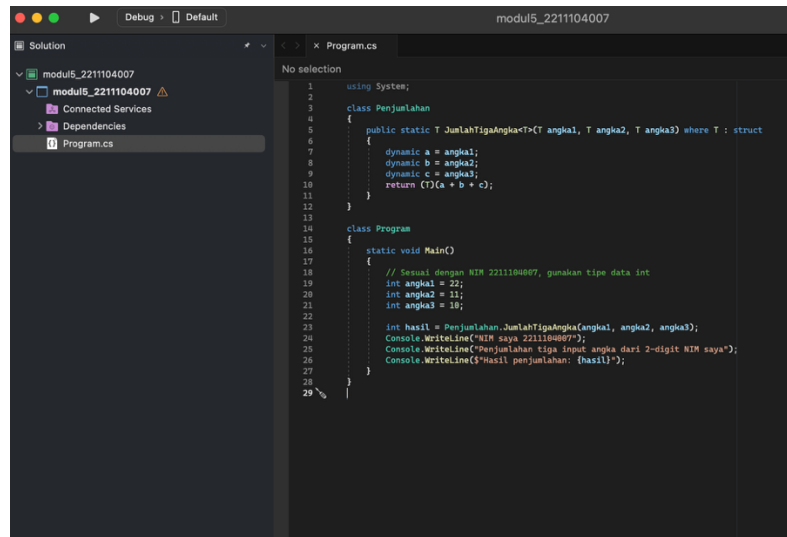
### 3. Membuat git branch baru dengan nama “implementasi-generic-method”



### 4. Menambahkan method dengan generic

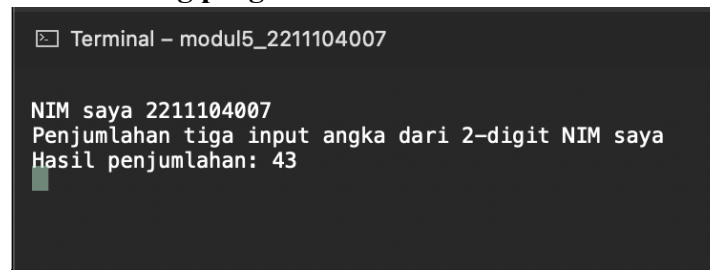
Tanpa membuat file baru (gunakan file yang dibuat saat membuat project):

- Buatlah sebuah class bernama “Penjumlahan”.
- Pada class tersebut, tambahkan sebuah method dengan nama “Jumlah Tiga Angka” yang memiliki tiga parameter generic yang sama yaitu “T”
- Method tersebut dapat melakukan penjumlahan dari tiga input/argument yang diberikan pada method tersebut.
- Hint: gunakan variabel sementara dengan tipe data dynamic untuk memungkinkan operasi matematis misalnya penjumlahan.
- Panggil method tersebut pada fungsi/method utama dengan tiga input angka yaitu 2-digit dari NIM. Misalnya NIM 12345678, maka tiga input angka yaitu “12”, “34” dan “56” dengan tipe data sebagai berikut:
  - NIM berakhiran 1 atau 2: tipe data input float
  - NIM berakhiran 3, 4 atau 5: tipe data input double
  - NIM berakhiran 6, 7 atau 8: tipe data input int - NIM berakhiran 9 atau 0: tipe data input long **Source code**



```
1 using System;
2
3 class Penjumlahan
4 {
5     public static T JumlahTigaAngka<T>(T angka1, T angka2, T angka3) where T : struct
6     {
7         dynamic a = angka1;
8         dynamic b = angka2;
9         dynamic c = angka3;
10        return (T)(a + b + c);
11    }
12 }
13
14 class Program
15 {
16     static void Main()
17     {
18         // Sesuai dengan NIM 2211104007, gunakan tipe data int
19         int angka1 = 22;
20         int angka2 = 11;
21         int angka3 = 10;
22
23         int hasil = Penjumlahan.JumlahTigaAngka(angka1, angka2, angka3);
24         Console.WriteLine("NIM saya 2211104007");
25         Console.WriteLine("Penjumlahan tiga input angka dari 2-digit NIM saya");
26         Console.WriteLine($"Hasil penjumlahan: {hasil}");
27     }
28 }
29 %
```

### Hasil running program



```
Terminal - modul5_2211104007

NIM saya 2211104007
Penjumlahan tiga input angka dari 2-digit NIM saya
Hasil penjumlahan: 43
```

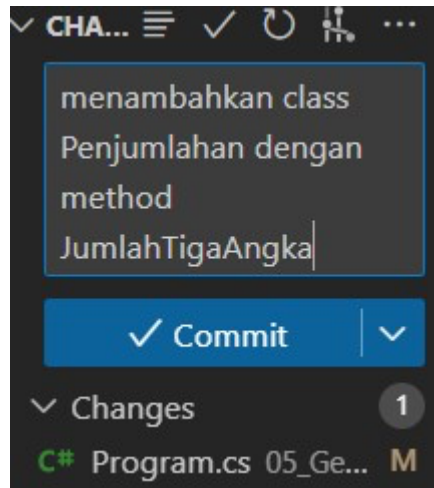
### Penjelasan program

Program ini terdiri dari dua kelas utama, yaitu kelas Penjumlahan dan kelas Program. Kelas Penjumlahan memiliki method generik bernama JumlahTigaAngka<T>, yang berfungsi untuk menjumlahkan tiga angka dengan tipe data yang sama. Dalam method ini, digunakan dynamic agar operasi penjumlahan dapat dilakukan untuk berbagai tipe data seperti float, double, int, dan long. Selain itu, terdapat batasan where T : struct yang memastikan hanya tipe data nilai (value type) yang dapat digunakan dalam method ini. Di dalam kelas Program, program diawali dengan menampilkan NIM pengguna, yaitu "2211104007". Selanjutnya, tiga angka diambil dari NIM, yakni 22, 11, dan 10. Pemilihan tipe data dilakukan berdasarkan digit terakhir NIM. Karena digit terakhir NIM adalah 2, maka tipe data yang digunakan adalah float. Kemudian, method JumlahTigaAngka dipanggil dengan tiga angka bertipe float, dan hasilnya ditampilkan di layar. Program ini memastikan bahwa operasi penjumlahan berjalan dengan benar serta mengikuti aturan penentuan tipe data sesuai dengan soal yang diberikan.

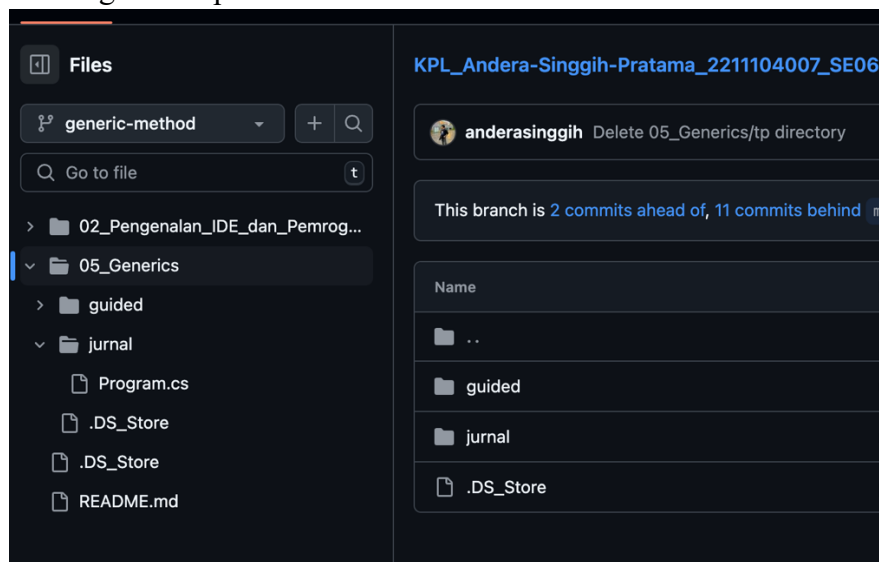
## 5. Melakukan commit, push, dan pindah ke branch

Pada branch yang sedang aktif saat ini (branch "implementasi-generic-method"):

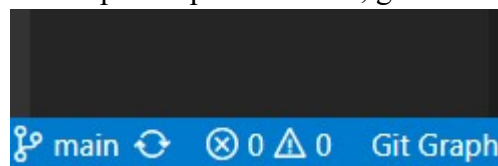
- Lakukan commit dengan pesan "menambahkan class Penjumlahan dengan method JumlahTigaAngka".



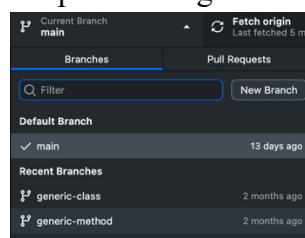
- b) Lakukan push ke github pada branch “implementasi-generic-method” di remote/github repo.



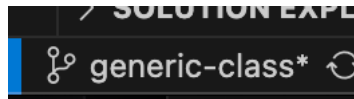
- c) Setelah proses push berhasil, ganti branch yang aktif ke master/main branch



- d) Pada branch master/main, buat branch baru dengan nama “implementasigeneric-class”.



- e) Setelah itu, lakukan pindah branch ke branch “implementasi-genericclass” yang sudah dibuat.



## 6. Menambahkan method dengan generic

Tanpa membuat file baru (gunakan file yang dibuat saat membuat project dan pastikan branch aktif adalah pada branch implementasi-generic-class):

- a) Buatlah sebuah class bernama “SimpleDataBase” dengan mengikuti class model yang ditunjukkan pada gambar/tabel di bawah ini.

SimpleDataBase
- storedData: List<T>
- inputDates: List<Date>
+ SimpleDataBase()
+ AddNewData(T)
+ PrintAllData(): void

b)

Class tersebut memiliki dua property yaitu:

- Property “storedData” yang merupakan suatu List (struktur data bawaan/default) yang berisi data bertipe generic “T”.
- Property “inputDates” yang bertipe List (atau tipe data List di C#) yang akan list dari waktu input

- c) Class tersebut juga memiliki beberapa method yaitu:

- Konstruktor SimpleDataBase() yang akan membuat property “storedData” berisi List kosong.
- Method AddNewData(T) yang akan menambahkan data baru bertipe T ke dalam list “storedData” dan waktu saat itu (Now) ke dalam list “inputDates”.
- Method PrintAllData() yang akan memberikan output console berupa teks yang menampilkan seluruh data yang tersimpan pada “storedData” dan “inputDates”, contohnya:
  - Data 1 berisi: 12, yang disimpan pada waktu UTC: 3/10/2022 5:32:01 AM
  - Data 2 berisi: 34, yang disimpan pada waktu UTC: 3/10/2022 5:32:02 AM
  - Data 2 berisi: 56, yang disimpan pada waktu UTC: 3/10/2022 5:32:02 AM

- d) Panggil method PrintAllData() pada fungsi/method utama setelah menambahkan tiga data yang berisi dan bertipe dua-digit NIM seperti pada bagian 4E.

## Source code:

### Hasil running program

```
NIM saya yaitu 2211104007
Hasil penjumlahan: 43
Data 1 berisi: 22, yang disimpan pada waktu UTC: 6/11/2025 12:10:52PM
Data 2 berisi: 11, yang disimpan pada waktu UTC: 6/11/2025 12:10:52PM
Data 3 berisi: 10, yang disimpan pada waktu UTC: 6/11/2025 12:10:52PM
```

### Penjelasan program

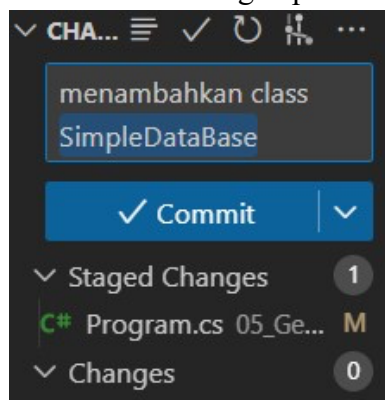
Program ini terdiri dari dua kelas utama, yaitu `SimpleDataBase<T>` sebagai kelas generik dan `Program` sebagai eksekutor utama. Kelas `SimpleDataBase<T>` memiliki dua atribut utama, yaitu `storedData`, yang berfungsi menyimpan data bertipe generik `T`, serta `inputDates`, yang digunakan untuk mencatat waktu saat data dimasukkan. Konstruktor dalam kelas ini akan menginisialisasi `storedData` dan `inputDates` sebagai list kosong. Method `AddNewData(T item)` digunakan untuk menambahkan data baru ke dalam `storedData`, sekaligus mencatat waktu saat data tersebut dimasukkan dalam format UTC dan menyimpannya ke dalam `inputDates`. Sementara itu, method `PrintAllData()` akan menampilkan seluruh data yang tersimpan beserta waktu penyimpanannya.

Pada kelas `Program`, program akan terlebih dahulu menampilkan NIM pengguna. Selanjutnya, objek `SimpleDataBase<float>` dibuat berdasarkan aturan bahwa jika digit terakhir dari NIM adalah 1 atau 2, maka tipe data yang digunakan adalah `float`. Setelah itu, tiga angka dari NIM, yaitu 22, 11, dan 10, dimasukkan ke dalam database menggunakan method `AddNewData()`. Terakhir, method `PrintAllData()` dipanggil untuk menampilkan seluruh data yang telah tersimpan beserta waktu penyimpanannya.

## 7. Melakukan commit, push, dan pindah ke branch bagian 2

Pada branch yang sedang aktif saat ini (branch “implementasi-generic-class”):

- Lakukan commit dengan pesan “menambahkan class `SimpleDataBase`”.



- Lakukan push ke github pada branch “implementasi-generic-class” di remote/github repo.

Name	Last commit message	Last commit date
..		
Program.cs	menambahkan class SimpleDataBase	1 minute ago

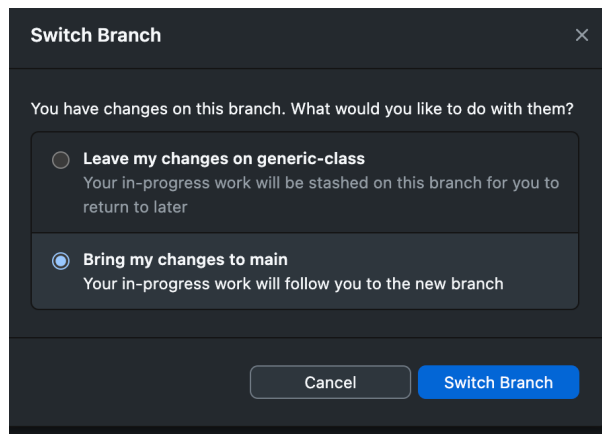
- c) Setelah proses push berhasil, ganti branch yang aktif ke master/main branch.



## 8. Melakukan git merge dari kedua branch baru

Pastikan branch aktif adalah branch master/main:

- a) Lakukan git merge branch “implementasi-generic-method” ke branch master/main.



- b) Lakukan git merge branch “implementasi-generic-class” ke branch master/main, dan jika terjadi merge conflict, pastikan semua baris yang conflict sudah diperbaiki.

- c) Lakukan git push untuk branch master/main ke github repository