

**TUGAS PENDAHULUAN
KONSTRUKSI PERANGKAT LUNAK**

**PERTEMUAN 3
GENERIC**



Disusun Oleh :
Muhammad Abdul Aziz
2211104026
SE0601

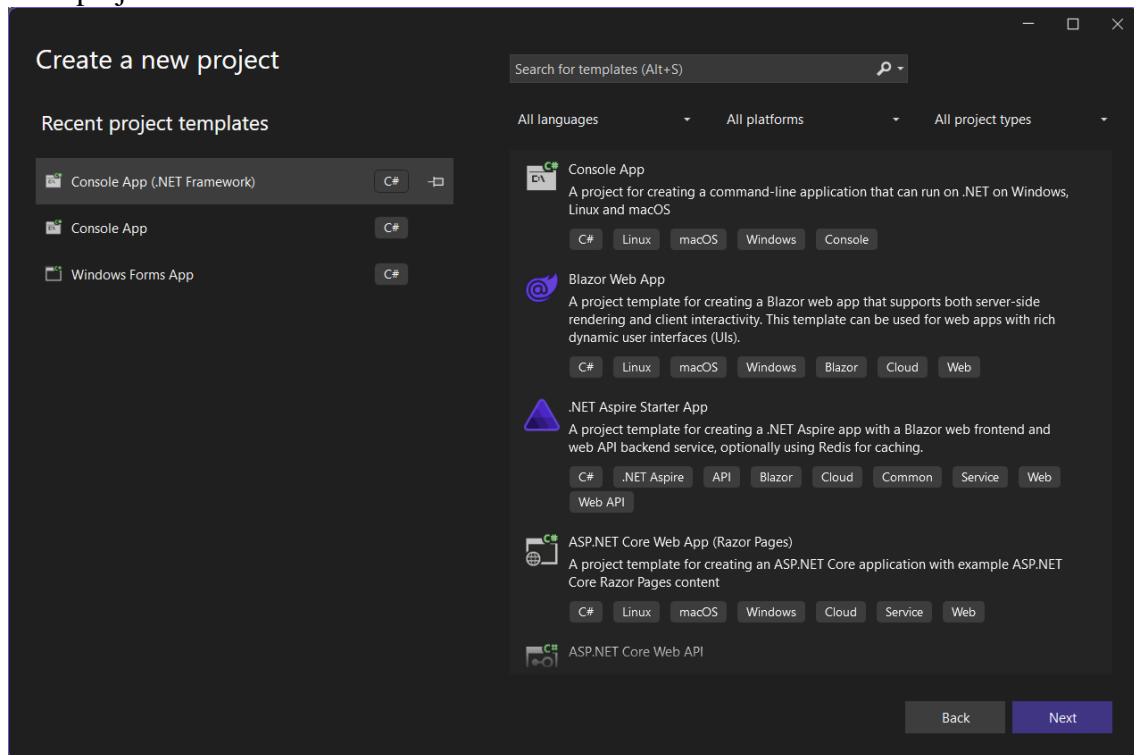
Asisten Praktikum :
Naufal El Kamil Aditya Pratama Rahman
Imelda

Dosen Pengampu :
Yudha Islami Sulistya, S.Kom., M.Cs.

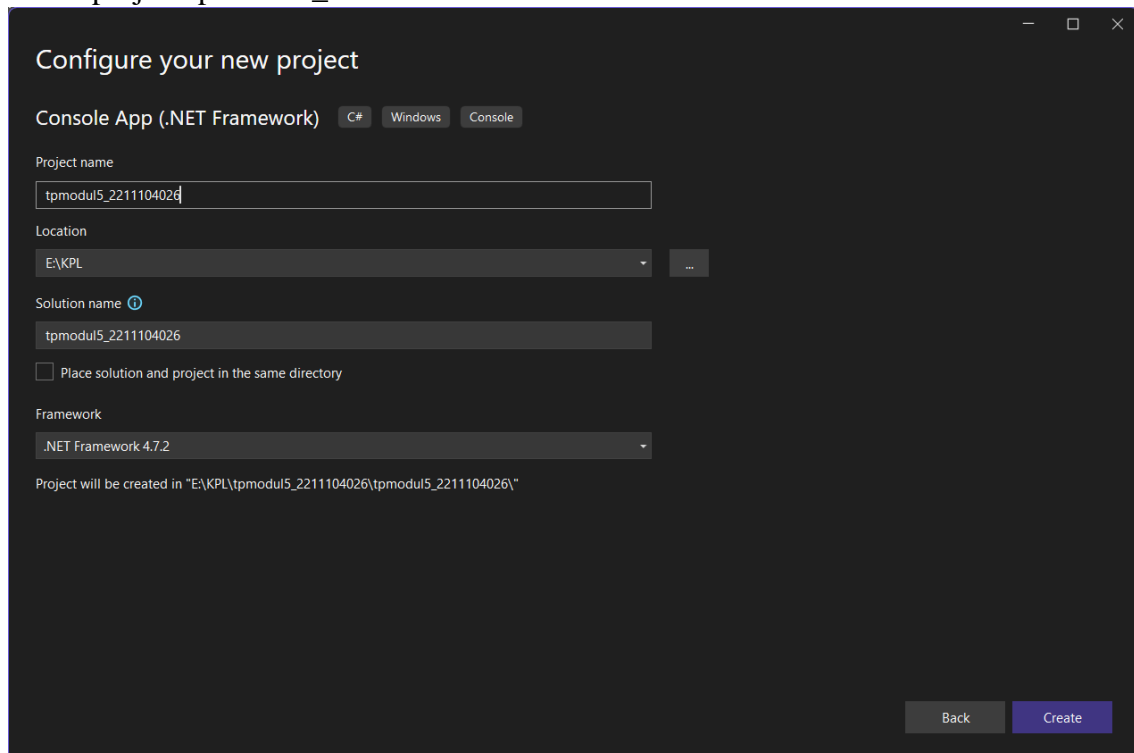
PROGRAM STUDI S1 SOFTWARE ENGINEERING
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2025

1. Membuat Project Console/Tanpa GUI

a. Buat project baru



b. Nama project tpmodul5_NIM



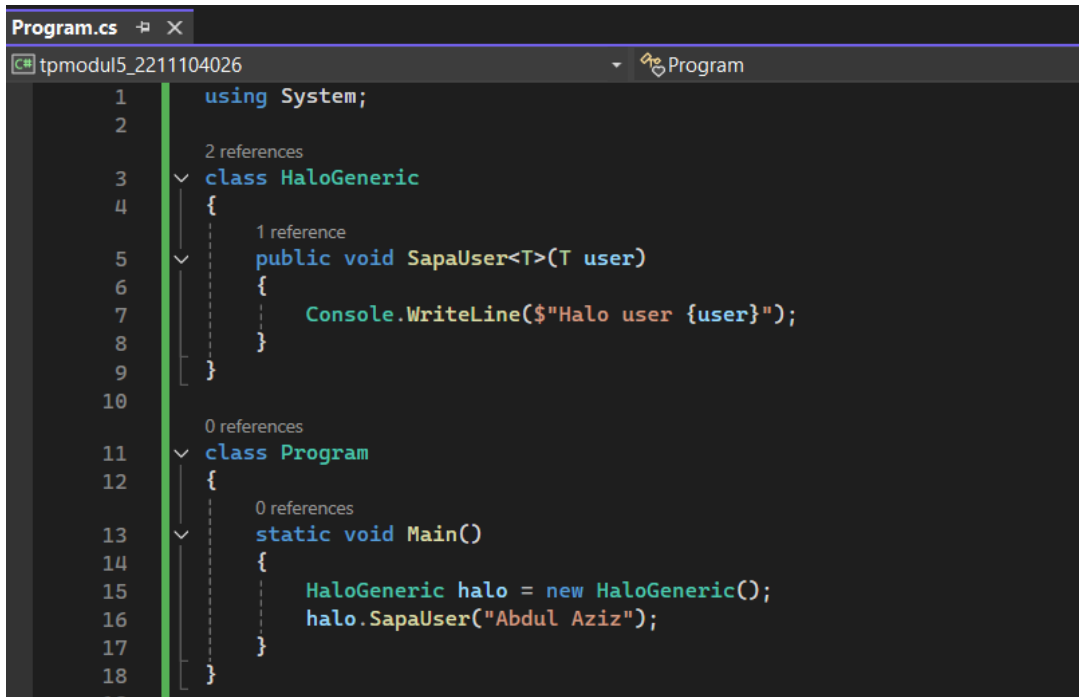
2. Membuat branch baru pada Git project dan pindah ke branch tersebut

```
PS E:\KPL_MUHAMMAD-ABDUL-AZIZ_2211104026_SE0601> git checkout -b generic-method
Switched to a new branch 'generic-method'
PS E:\KPL_MUHAMMAD-ABDUL-AZIZ_2211104026_SE0601> |
```

3. Menambahkan method dengan generic

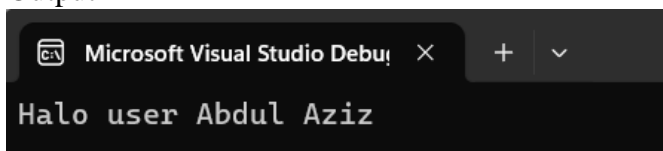
- Buatlah sebuah class bernama "HaloGeneric".
- Pada class tersebut, tambahkan sebuah method dengan nama "SapaUser" yang memiliki
- generic parameter yang akan melakukan print "Halo user X" dimana X adalah input/nilai argument yang diberikan pada method tersebut.
- Panggil method tersebut pada fungsi/method utama dengan input String dengan isi nilai nama panggilan praktikan.

- Source Code



```
Program.cs
1 using System;
2
3 class HaloGeneric
4 {
5     public void SapaUser<T>(T user)
6     {
7         Console.WriteLine($"Halo user {user}");
8     }
9 }
10
11 class Program
12 {
13     static void Main()
14     {
15         HaloGeneric halo = new HaloGeneric();
16         halo.SapaUser("Abdul Aziz");
17     }
18 }
```

- Output



```
Microsoft Visual Studio Debug Console
Halo user Abdul Aziz
```

- Penjelasan

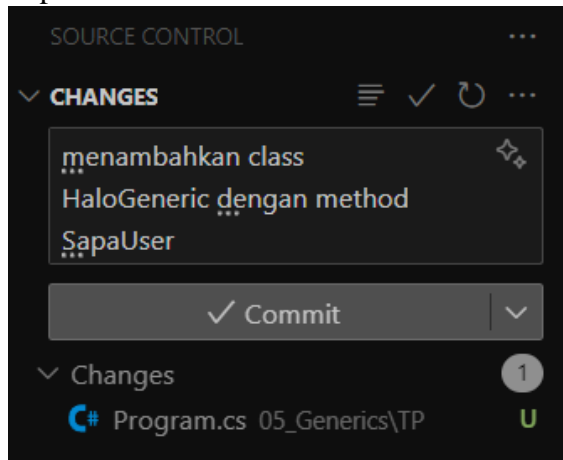
Kode ini merupakan implementasi Generic Method dalam bahasa C# untuk menyapa pengguna dengan fleksibilitas tipe data. Program ini terdiri dari dua class utama, yaitu HaloGeneric dan Program.

Class HaloGeneric memiliki method SapaUser<T>(T user), di mana T adalah tipe data generic yang memungkinkan method ini menerima berbagai tipe data saat dipanggil. Method ini mencetak teks "Halo user X", di mana X adalah nilai dari parameter user. Dalam contoh ini, method dipanggil dengan string "Abdul Aziz", sehingga output yang dihasilkan adalah "Halo user Abdul Aziz".

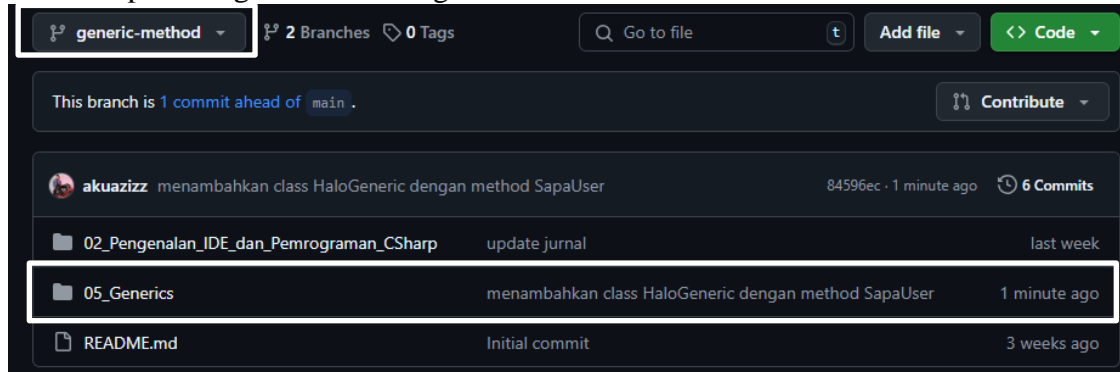
Class Program berisi method utama (Main), yang berfungsi sebagai titik awal eksekusi program. Di dalamnya, dibuat objek HaloGeneric yang kemudian digunakan untuk memanggil SapaUser dengan input "Abdul Aziz". Dengan penggunaan **Generic**, method ini dapat digunakan untuk berbagai tipe data tanpa perlu membuat method baru, sehingga meningkatkan efisiensi dan fleksibilitas kode.

4. Melakukan commit push dan pindah ke branch

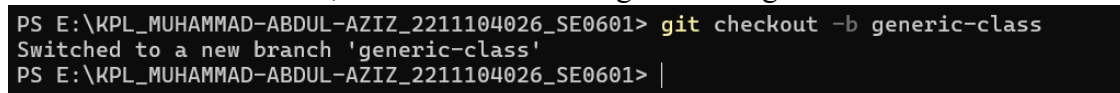
- Melakukan commit dengan pesan “menambahkan class haloGeneric dengan method SapaUser”



- Lakukan push ke github branch “generic-method”



- Pada branch master/main, buat branch baru dengan nama “generic-class”.



5. Menambahkan method dengan Generic

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

- Buatlah sebuah class bernama “DataGeneric” dengan mengikuti class model yang ditunjukkan pada gambar/tabel di bawah ini. Class tersebut memiliki property “Data” yang bertipe generic “T” dan memiliki konstruktor dengan parameter data.

DataGeneric
- data: T
+ DataGeneric(T)
+ PrintData(): void

- Class tersebut juga memiliki method bernama PrintData yang melakukan print di console dengan output “Data yang tersimpan adalah: Y”, dengan “Y” adalah nilai dari property “data” dari kelas tersebut.
- Panggil method PrintData() setelah mengisi “data” dengan NIM pada fungsi/method utama.

- Source Code

```

1  using System;
2
3  1 reference
4  public class HaloGeneric
5  {
6      1 reference
7      public static void SapaUser<T>(T user)
8      {
9          Console.WriteLine($"Halo user {user}");
10     }
11
12     3 references
13     public class DataGeneric<T>
14     {
15         private T data;
16         1 reference
17         public DataGeneric(T data)
18         {
19             this.data = data;
20         }
21         1 reference
22         public void PrintData()
23         {
24             Console.WriteLine($"Data yang tersimpan adalah: {data}");
25         }
26     }
27
28     0 references
29     class Program
30     {
31         0 references
32         static void Main()
33         {
34             HaloGeneric.SapaUser("Abdul Aziz");
35             DataGeneric<string> data = new DataGeneric<string>("2211104026");
36             data.PrintData();
37         }
38     }

```

- Output

```

Halo user Abdul Aziz
Data yang tersimpan adalah: 2211104026

```

- Penjelasan

Kode ini merupakan implementasi Generic Method dan Generic Class dalam bahasa C#. Class `HaloGeneric` memiliki method statis `SapaUser<T>(T user)`, yang menggunakan Generic Parameter `T` sehingga dapat menerima berbagai jenis data sebagai input. Method ini mencetak pesan "Halo user X", di mana X adalah nilai yang diberikan sebagai parameter. Dalam program ini, method dipanggil dengan input string "Abdul Aziz", sehingga outputnya adalah "Halo user Abdul Aziz".

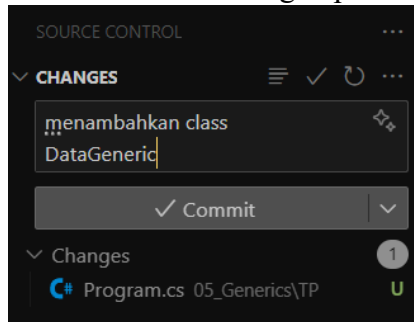
Selain itu, terdapat class `DataGeneric<T>` yang menggunakan Generic Class, memungkinkan penyimpanan data dengan tipe fleksibel. Class ini memiliki atribut `private data` bertipe `T`, yang diinisialisasi melalui konstruktor. Method `PrintData()` digunakan untuk mencetak nilai data yang tersimpan dalam format "Data yang tersimpan adalah: Y", di mana Y adalah nilai dari `data`. Dalam method `Main()`, objek `DataGeneric<string>` dibuat dengan nilai "2211104026" (NIM mahasiswa), sehingga hasil outputnya adalah "Data yang tersimpan adalah: 2211104026".

Penggunaan Generic Programming dalam kode ini memberikan fleksibilitas dalam menangani berbagai tipe data tanpa perlu menduplikasi kode, meningkatkan efisiensi, serta mempermudah pemeliharaan sistem.

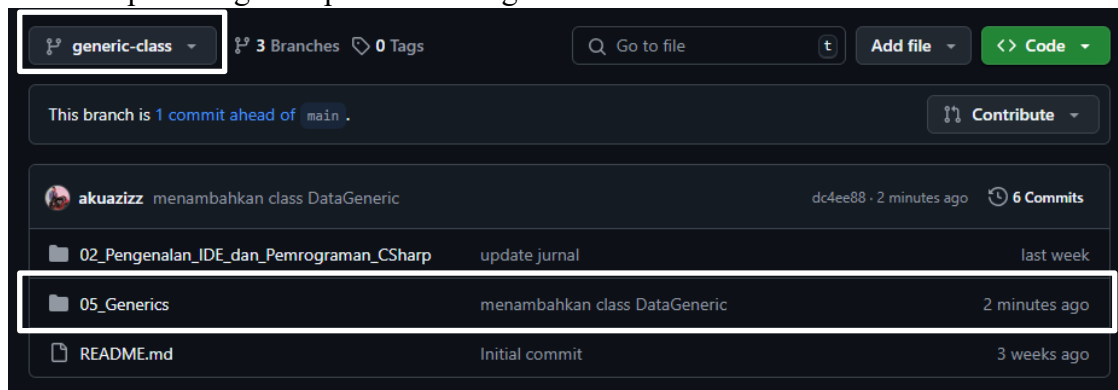
6. Melakukan commit, push, dan pindah ke branch bagian kedua

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

- Lakukan commit dengan pesan “menambahkan class DataGeneric”.



- Lakukan push ke github pada branch “generic-class”



7. Melakukan git merge dari kedua branch baru

Pastikan branch aktif adalah branch master/main :

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

```
PS E:\KPL_MUHAMMAD-ABDUL-AZIZ_2211104026_SE0601> git merge generic-method
Updating 8c15700..84596ec
Fast-forward
 05_Generics/TP/Program.cs | 19 ++++++
 1 file changed, 19 insertions(+)
 create mode 100644 05_Generics/TP/Program.cs
PS E:\KPL_MUHAMMAD-ABDUL-AZIZ_2211104026_SE0601> |
```

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

```
PS E:\KPL_MUHAMMAD-ABDUL-AZIZ_2211104026_SE0601> git merge generic-class
Auto-merging 05_Generics/TP/Program.cs
```

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

```
PS E:\KPL_MUHAMMAD-ABDUL-AZIZ_2211104026_SE0601> git add .
PS E:\KPL_MUHAMMAD-ABDUL-AZIZ_2211104026_SE0601> git commit -m "menggabungkan generic-method dan generic-class"
[main 1ce70d2] menggabungkan generic-method dan generic-class
PS E:\KPL_MUHAMMAD-ABDUL-AZIZ_2211104026_SE0601> git push origin main
Enumerating objects: 1, done.
Counting objects: 100% (1/1), done.
Writing objects: 100% (1/1), 246 bytes | 246.00 KiB/s, done.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/akuazizz/KPL_MUHAMMAD-ABDUL-AZIZ_2211104026_SE0601.git
 84596ec..1ce70d2  main -> main
PS E:\KPL_MUHAMMAD-ABDUL-AZIZ_2211104026_SE0601> |
```