

**TUGAS PENDAHULUAN  
KONSTRUKSI PERANGKAT LUNAK**

**MODUL V  
GENERICS**



**Disusun Oleh :**

Atika Aji Hadiyani

2211104003

SE-06-01

**Dosen Pengampu :**

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

**PROGRAM STUDI S1 SOFTWARE ENGINEERING FAKULTAS**

**INFORMATIKA**

**TELKOM UNIVERSITY PURWOKERTO**

**2025**

## TUGAS PENDAHULUAN

### 1. Membuat project baru dengan nama tpmodul5\_NIM.

Project name  
tpmodul5\_2211104003

Location  
C:\Users\ASUS\OneDrive\Documents\Kuliah\Semester 6\Praktikum KPL\KPL\_AtikaAjiHadiyani\_2211

Solution  
Create new solution

Solution name ⓘ  
tpmodul5\_2211104003

☐ Place solution and project in the same directory

Project will be created in "C:\Users\ASUS\OneDrive\Documents\Kuliah\Semester 6\Praktikum KPL\KPL\_AtikaAjiHadiyani\_2211104003\_SE0601\tpmodul5\_2211104003\tpmodul5\_2211104003\"

### 2. Melakukan git push ke github repo

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

```
PS C:\Users\ASUS\OneDrive\Documents\Kuliah\Semester 6\Praktikum KPL\KPL_AtikaAjiHadiyani_2211104003_SE0601\05_Generics> git checkout -b generic-method  
Switched to a new branch 'generic-method'  
PS C:\Users\ASUS\OneDrive\Documents\Kuliah\Semester 6\Praktikum KPL\KPL_AtikaAjiHadiyani_2211104003_SE0601\05_Generics> |
```

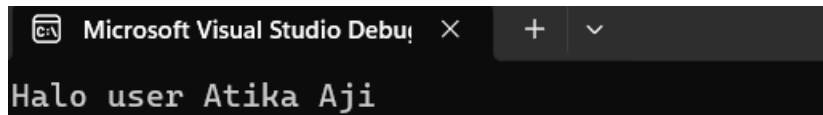
### 4. Menambahkan Method dengan Generic

- 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

```
using System;  
  
1 reference  
public class HaloGeneric  
{  
    1 reference  
    public static void SapaUser<T>(T user)  
    {  
        Console.WriteLine($"Halo user {user}");  
    }  
}  
  
class Program  
{  
    0 references  
    static void Main()  
    {  
        HaloGeneric.SapaUser("Atika Aji");  
    }  
}
```

- **Screenshot hasil run**




- **Penjelasan**

Program ini menggunakan konsep generic method. Pada class HaloGeneric, terdapat method SapaUser<T>(T user), <T> adalah tipe data generic sehingga bisa menerima berbagai jenis data (string, angka, atau objek lain). Saat method ini dipanggil, maka akan mencetak teks "Halo user X", di mana X memanggil SapaUser("Hanifaa"), sehingga output yang dihasilkan adalah "Halo user Hanifaa". Dalam method Main(), program menjalankan method SapaUser dengan input "Hanifaa", yang diproses oleh SapaUser<T>.

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

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

- Lakukan commit dengan pesan “menambahkan class HaloGeneric dengan method SapaUser”.
- Lakukan push ke github pada branch “generic-method” di remote/github repo.

 tpmodul5.cs

Uploud file tp dan jurnal

- Setelah proses push berhasil, ganti branch yang aktif ke master/main branch.
- Pada branch master/main, buat branch baru dengan nama “generic-class”.

```
PS C:\Users\ASUS\OneDrive\Documents\Kuliah\Semester 6\Praktikum KPL\KPL_AtikaAjiHadiyani_2211104003_SE0601\05_Generics> git checkout -b generic-class
Switched to a new branch 'generic-class'
PS C:\Users\ASUS\OneDrive\Documents\Kuliah\Semester 6\Praktikum KPL\KPL_AtikaAjiHadiyani_2211104003_SE0601\05_Generics> |
```

- Setelah itu, lakukan pindah branch ke branch “generic-class” yang sudah dibuat.

## 6. Menambahkan method dengan generic

- 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.

- b. 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.
- c. Panggil method PrintData() setelah mengisi “data” dengan NIM pada fungsi/method utama

- **Source code**

```
using System;

1 reference
public class HaloGeneric
{
    1 reference
    public static void SapaUser<T>(T user)
    {
        Console.WriteLine($"Halo user {user}");
    }
}

3 references
public class DataGeneric<T>
{
    private T data;

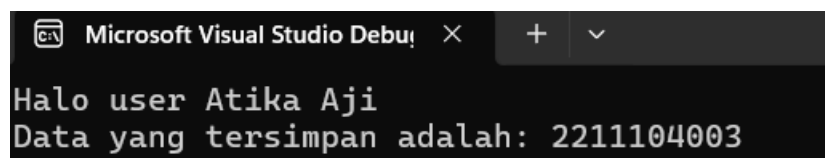
    1 reference
    public DataGeneric(T data)
    {
        this.data = data;
    }

    1 reference
    public void PrintData()
    {
        Console.WriteLine($"Data yang tersimpan adalah: {data}");
    }
}

0 references
class Program
{
    0 references
    static void Main()
    {
        HaloGeneric.SapaUser("Atika Aji");

        // Membuat objek DataGeneric dengan NIM
        DataGeneric<string> data = new DataGeneric<string>("2211104003");
        data.PrintData();
    }
}
```

- **Screenshot hasil run**



The screenshot shows the Microsoft Visual Studio Debug Console window. The title bar reads "Microsoft Visual Studio Debug Console". The console output displays two lines: "Halo user Atika Aji" and "Data yang tersimpan adalah: 2211104003".

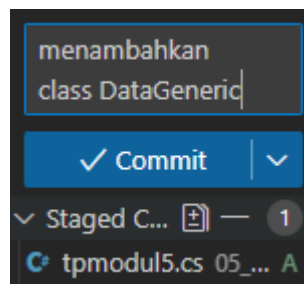
## - Penjelasan

Program ini menggunakan konsep generic class di C#. Class `DataGeneric<T>` digunakan untuk menyimpan data dengan tipe yang fleksibel. Class ini memiliki properti data bertipe `T`, yang bisa diisi dengan berbagai jenis data saat objek dibuat. Konstruktor `DataGeneric(T data)` digunakan untuk menginisialisasi nilai data, sementara method `PrintData()` berfungsi untuk mencetak teks "Data yang tersimpan adalah: Y", di mana Y adalah nilai yang diberikan ke data. Didalam method `Main()`, program membuat objek `DataGeneric<string>` dengan nilai "2211104017". Kemudian, method `PrintData()` akan mencetak "Data yang tersimpan adalah: 2211104017" di layar.

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

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" di remote/github repo.

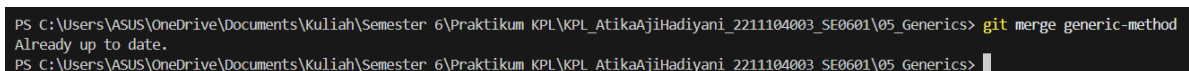


- Setelah proses push berhasil, ganti branch yang aktif ke master/main branch.

## 8. Melakukan git merge dari kedua branch baru

Pastikan branch aktif adalah branch master/main :

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



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

```
PS C:\Users\ASUS\OneDrive\Documents\Kuliah\Semester 6\Praktikum KPL\KPL_AtikaAjiHadiyani_2211104003_SE0601\05_Generics> git merge generic-class
Already up to date.
PS C:\Users\ASUS\OneDrive\Documents\Kuliah\Semester 6\Praktikum KPL\KPL_AtikaAjiHadiyani_2211104003_SE0601\05_Generics> |
```

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

```
PS C:\Users\ASUS\OneDrive\Documents\Kuliah\Semester 6\Praktikum KPL\KPL_AtikaAjiHadiyani_2211104003_SE0601\05_Generics> git add .
PS C:\Users\ASUS\OneDrive\Documents\Kuliah\Semester 6\Praktikum KPL\KPL_AtikaAjiHadiyani_2211104003_SE0601\05_Generics> git commit -m "Menggabungkan generic-method dan generic-class"
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean
PS C:\Users\ASUS\OneDrive\Documents\Kuliah\Semester 6\Praktikum KPL\KPL_AtikaAjiHadiyani_2211104003_SE0601\05_Generics> git push origin main
Everything up-to-date
```