

LAPORAN TUGAS PERTEMUAN 11

PEMROGRAMAN



Nama :	Abdur Razak
NIM :	22.11.4862
Dosen Pengampu :	Abd. Mizwar A. Rahim, M.Kom
Asisten Dosen Koordinator :	Dimas Ariyanto
Tanggal Pengumpulan :	11 Juli 2023

S1-INFORMATIKA UNIVERSITAS AMIKOM YOGYAKARTA
2023.

POLYMORPHISM ABSTRACT

Program ini terdiri dari:

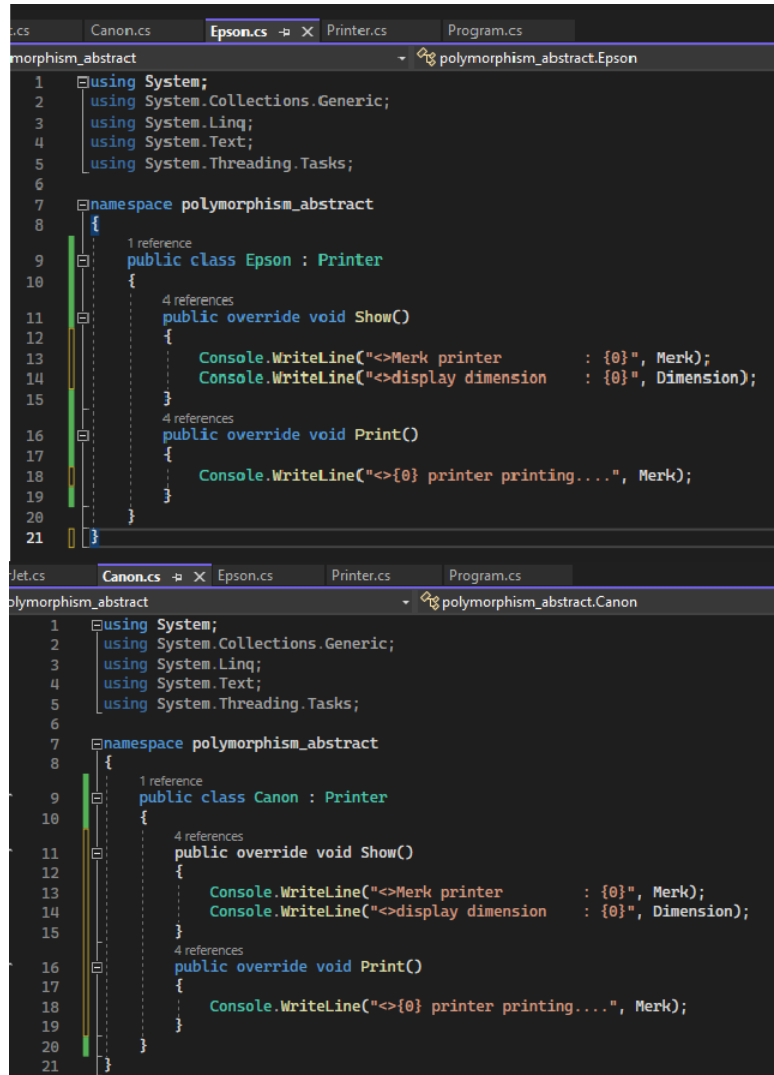
- class dimana berfungsi sebagai blueprint atau rancangan yang berisi rincian dari object, tempat pendeklarasian variabel dan method.
- object yang berfungsi sebagai bahan dari class baik itu object konkrit (yang ada di dunia nyata) maupun object abstract (yang ada tetapi secara fisik tidak ada)

Polymorphism terdiri dari 2 kata yaitu **poly** yang artinya **banyak** dan **morph** yang artinya **bentuk**. **Polymorphism** adalah kemampuan beberapa objek yang mengimplementasikan sesuatu hal yang berbeda dengan cara yang sama. Juga dapat dikatakan kemampuan beberapa objek untuk memutuskan method mana yang akan diterapkan. **abstract** merupakan kelas yang tidak diinstansiasi langsung dan digunakan sebagai kerangka kerja /cetakan untuk kelas turunannya. Dan kelas anaknya menggunakan **public override**. Sebenarnya, pada polymorphism juga terdapat 2 keyword yaitu **virtual**(untuk memberi tanda bahwa method pada class induk dapat di override/timpa oleh class anaknya) dan **override**(keyword untuk method pada class anak yang mengoverride), namun pada program kali ini tidak memakai keyword **virtual**. **Polymorphism** memiliki manfaat untuk membuat program yang memiliki objek yang berbeda-beda dengan lebih efisien.

Berikut penjelasan pada class:

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6
7  namespace polymorphism_abstract
8  {
9      public abstract class Printer
10     {
11         public string Dimension { get; set; }
12         public string Merk { get; set; }
13         public abstract void Show();
14         public abstract void Print();
15     }
16 }
```

1. membuat class dengan nama **Printer** dan menjadi *class induk* menggunakan keyword *abstract*.
2. Membuat method abstract berupa void (Show dan Print) yang nantinya digunakan kelas anak untuk menampung property berisi elemen-elemen/pesan yang akan ditampilkan.
3. membuat property bertipe string dengan nama (Dimension dan Merk) yang nantinya digunakan kelas anak untuk menampung elemen-elemen/pesan yang akan ditampilkan.



The image shows two screenshots of a Visual Studio code editor. The top screenshot displays the `Epson.cs` file, which implements the `Printer` abstract class. The bottom screenshot displays the `Canon.cs` file, which also implements the `Printer` abstract class. Both files are part of the `polymorphism_abstract` namespace and include the same using statements: `using System;`, `using System.Collections.Generic;`, `using System.Linq;`, `using System.Text;`, and `using System.Threading.Tasks;`. The `Printer` abstract class is defined in a separate file (not shown) and has two abstract methods: `Show()` and `Print()`. The `Epson` class overrides `Show()` to write the printer's name and dimension to the console, and `Print()` to write a message indicating the printer is printing. The `Canon` class also overrides `Show()` and `Print()` with similar logic.

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6
7 namespace polymorphism_abstract
8 {
9     1 reference
10     public class Epson : Printer
11     {
12         4 references
13         public override void Show()
14         {
15             Console.WriteLine("<>Merk printer      : {0}", Merk);
16             Console.WriteLine("<>display dimension  : {0}", Dimension);
17         }
18         4 references
19         public override void Print()
20         {
21             Console.WriteLine("<>{0} printer printing...", Merk);
22         }
23     }
24 }
```

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6
7 namespace polymorphism_abstract
8 {
9     1 reference
10     public class Canon : Printer
11     {
12         4 references
13         public override void Show()
14         {
15             Console.WriteLine("<>Merk printer      : {0}", Merk);
16             Console.WriteLine("<>display dimension  : {0}", Dimension);
17         }
18         4 references
19         public override void Print()
20         {
21             Console.WriteLine("<>{0} printer printing...", Merk);
22         }
23     }
24 }
```

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6
7 namespace polymorphism_abstract
8 {
9     1 reference
10     public class LaserJet : Printer
11     {
12         4 references
13         public override void Show()
14         {
15             Console.WriteLine("<>Merk printer : {0}", Merk);
16             Console.WriteLine("<>display dimension : {0}", Dimension);
17         }
18         4 references
19         public override void Print()
20         {
21             Console.WriteLine("<>{0} printer printing....", Merk);
22         }
23     }
24 }
```

1. Membuat class lagi bernama **Epson**, **Canon**, dan **LaserJet** sebagai public class.
2. Menggunakan sintaks dengan contoh *public class Epson : Printer* sebagai program bahwa class **Epson** adalah *Class* turunan dari **Printer** dan melakukan hal yang sama seperti itu pada **Canon** dan **LaserJet** agar menjadi class turunan.
4. Menuliskan sintaks *void* yang sama seperti pada class induk, namun pada class turunan (Epson, Canon, dan LaserJet) memakai keyword *override* dan di dalam void terdapat pesan yang ingin ditampilkan sesuai class turunannya masing-masing seperti contoh pada class Epson dalam **void Show** terdapat pesan "*Merk printer : {0}*" dan "*{0} display dimension : {0}*" dan dalam **void Print** terdapat pesan "*{0} printer printing....*" (*{0}* merupakan sintak untuk menampilkan isi dari property yang sudah di tentukan pada object).

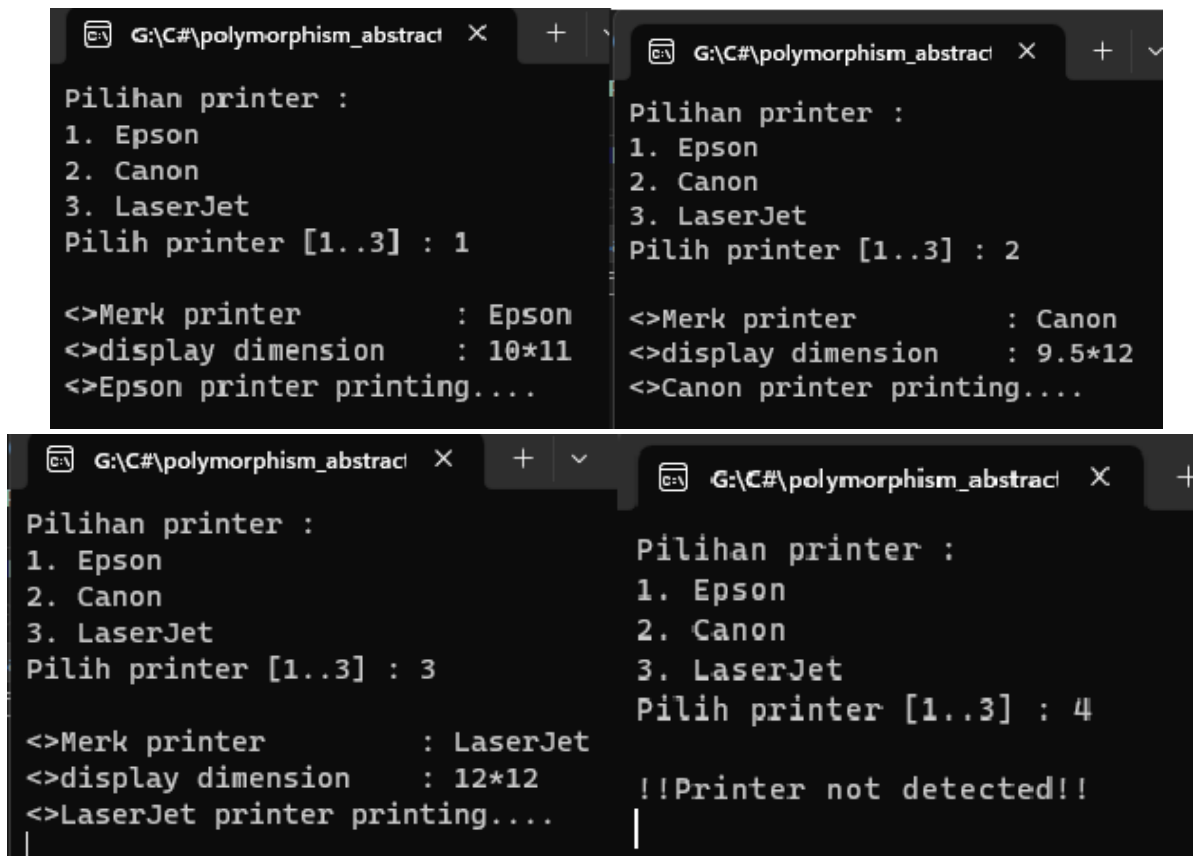
Berikut penjelasan pada object:

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6
7 namespace polymorphism_abstract
8 {
9     internal class Program
10     {
11         static void Main(string[] args)
12         {
13             Printer epson = new Epson();
14             Printer canon = new Canon();
15             Printer laserjet = new LaserJet();
16
17             epson.Dimension = "10*11";
18             epson.Merk = "Epson";
19
20             canon.Dimension = "9.5*12";
21             canon.Merk = "Canon";
22
23             laserjet.Dimension = "12*12";
24             laserjet.Merk = "LaserJet";
25
26             Console.WriteLine("Pilihan printer : ");
27             Console.WriteLine("1. Epson");
28             Console.WriteLine("2. Canon");
29             Console.WriteLine("3. LaserJet");
30             Console.Write("Pilih printer [1..3] : ");
31             int pilihan = Convert.ToInt32(Console.ReadLine());
32             Console.WriteLine();
33
34             //pengkondisian
35             if (pilihan == 1)
36             {
37                 epson.Show();
38                 epson.Print();
39             }
40             else if (pilihan == 2)
41             {
42                 canon.Show();
43                 canon.Print();
44             }
45             else if (pilihan == 3)
46             {
47                 laserjet.Show();
48                 laserjet.Print();
49             }
50             else
51                 Console.WriteLine("!!Printer not detected!!");
52
53             Console.ReadKey();
54         }
55     }
56 }
```

1. Pada object, membuat *nama object* yang ditujukan untuk *nama class* yang akan dituju, jadi programnya yaitu *nama_class_induk nama_object = new nama_class_anak();* seperti contoh program saya yaitu *Printer epson = new Epson();* .

2. Pada line 17-24, membuat elemen pada property yang isinya disesuaikan dengan informasi pada masing-masing kelas anaknya dengan sintaks *nama_object.nama_property = "isi informasi";*
3. Pada line 26-30, membuat tampilan awal saat program dijalankan berupa pilihan printer mana yang akan dipilih sesuai pada class turunannya Printer (Epson, Canon, dan LaserJet).
4. *Int pilihan = Convert.ToInt32(Console.ReadLine());* digunakan agar pada saat program dijalankan, dapat menginputkan nilai sesuai tipe data nya yaitu **int** dan inputan tersebut diinisialisasikan sebagai *pilihan*.
5. Pada line 25, melakukan pendeklarasian class IPrinterWindows sebagai cetak agar dapat mengambil fungsi void yang ada pada class tersebut.
6. Pada line 35-51, melakukan pengkondisian menggunakan if-else. Jika kita menginputkan angka 1, maka akan menampilkan pesan dari class anak **Epson**, jika 2, maka akan menampilkan pesan dari class anak **Canon**, jika 3, maka akan menampilkan pesan dari class anak **LaserJet**, dan jika menginputkan angka selain 1-3, maka akan menampilkan pesan *"!!Printer not detected!!"*.

Tampilannya sebagai berikut:



```
Pilihan printer :
1. Epson
2. Canon
3. LaserJet
Pilih printer [1..3] : 1

<>Merk printer      : Epson
<>display dimension  : 10*11
<>Epson printer printing....

Pilihan printer :
1. Epson
2. Canon
3. LaserJet
Pilih printer [1..3] : 2

<>Merk printer      : Canon
<>display dimension  : 9.5*12
<>Canon printer printing....

Pilihan printer :
1. Epson
2. Canon
3. LaserJet
Pilih printer [1..3] : 3

<>Merk printer      : LaserJet
<>display dimension  : 12*12
<>LaserJet printer printing....

Pilihan printer :
1. Epson
2. Canon
3. LaserJet
Pilih printer [1..3] : 4

!!Printer not detected!!
```