



# Back End Development 1

## sesi 2

---



# C# Overview<sup>+</sup>

## *C# Overview - Sesi 02*

# C# Overview

Pertemuan sebelumnya kita telah belajar apa itu .net hingga cara kerja .net di akhiri dengan instalasi dan konfigurasi tools Visual Studio dan Mono SDK.

Pada pertemuan kali ini kita akan explore beberapa hal tentang C# ,yaitu :

1. Variabel
2. Tipe Data
3. Operator
4. Class
5. Method

Sebagai sesi awal kita akan membahas syntax C# yang kita buat di pertemuan sebelumnya dengan program Hello World.

## STRUKTUR PROGRAM

Pada C# ada 3 Bagian dasar Struktur Program , apa itu ?

1. Bagian Deklarasi Pustaka
2. Bagian Class
3. Bagian Fungsi

Sesi1 >  HelloWorld.cs

```
1  using System;  
2  
3  public class HelloWorld  
4  {  
5      public static void Main(string[] args)  
6      {  
7          Console.WriteLine("Hello World");  
8      }  
9  }
```

1. Deklarasi Pustaka

2. Class

3. Fungsi



HACKTIV8

## 1. Deklarasi Pustaka

Kita wajib menuliskan pustaka (library) yang dibutuhkan dalam program ini,

Yang dimana Pustaka/Library terdiri dari sekumpulan fungsi, method, object, const dan variabel yang kita bisa gunakan didalam program yang akan kita buat.

Question :

Pada Program HelloWorld kita menggunakan library apa ? `using System;`

Library system kita butuhkan untuk menggunakan class `console` dan method `WriteLine()`



**HACKTIV8**

## 2. Bagian Class

NICE TO KNOW...

C# Programming Languages menggunakan paradigma OOP ( Object Oriented Programming ), contohnya :

Setiap kali dalam membuat program C# kita harus membungkus fungsi dan variabel di dalam **class**.

Masih Menggunakan contoh program HelloWorld :

```
Sesi1 > C# HelloWorld.cs
1  using System;
2
3  public class HelloWorld
4  {
5      public static void Main(string[] args)
6      {
7          Console.WriteLine("Hello World");
8      }
9  }
```

Nama Program yang kita buat HelloWorld yang dimana :

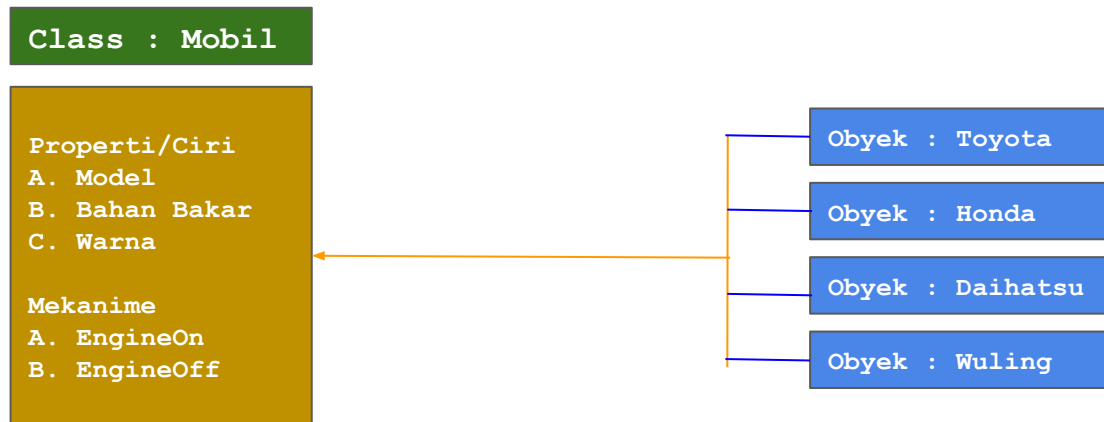
didalam class kita isi dengan fungsi dan variabel.

Seperti yang sudah di bahas sebelumnya, class yang akan membungkus fungsi dan variabel.

Lalu apa itu CLASS ?

Class disebut blueprint dari sebuah Objek.

Contoh kita punya :



Sekarang kita memiliki sebuah class bernama Mobil.

Class Mobil memiliki ciri/properti apa ?

Class Mobil memiliki model apa ?

Class Mobil memiliki bahan bakar apa ?

Class Mobil memiliki warna apa ?

Lalu Class ini memiliki mekanisme seperti Menghidupkan dan Mematikan Mesin.

Dengan kita men-define Class itu sama dengan kita membuat sebuah blueprint tentang :  
Bagaimana Sebuah Mobil nantinya mau dibuat.

Sehingga ketika Kita akan membuat beberapa Objek yang didapat dari blueprint yang kita buat.

Lalu ketika sudah sampai Objek , apa lagi yang bisa kita lakukan ?

Kita bisa membedakan 1 Objek dengan yang lain, contoh :

Kita membuat objek Toyota, kita tentukan properti model nya = hatchback dengan properti berwarna hijau sedangkan

Objek Honda, Properti Model = SUV dan properti berwarna merah.

Nah materi terkait objek akan kita bahas di sesi 5 dan 7 ya..



### 3. Fungsi / Method

Nah pada bagian ini kita bisa menuliskan semua fungsi dari program, dimana :  
Fungsi yang harus ada didalam setiap program adalah fungsi / method **Main()**.

Jika tidak ada method ini, maka program kita tidak akan bisa dijalankan, yang artinya Ketika Kita Menjalankan Program : **Main()** adalah method yang pertama kali dipanggil.

#### PERLU DIPERHATIKAN

Dalam setiap Penulisan Statement dan Ekspresi harus diakhiri dengan tanda semicolon (;).

```
Sesi2 > ProgramHello.cs > ...  
1  using System;  
2  
   0 references  
3  public class HelloWorld  
4  {  
   0 references  
5      public static void Main(string[] args)  
6      {  
7          Console.WriteLine("Hello World");  
8          Console.WriteLine("Hello nama Kalian");  
9          Console.WriteLine("Selamat Datang di Program Bootcamp OCBC");  
10         Console.WriteLine("Tahap 1 saya akan belajar C# ");  
11     }  
12 }
```



```
Maliks-MacBook-Air:Sesi2 swijaya$ cd "/Us  
mono ProgramHello  
Hello World  
Hello nama Kalian  
Selamat Datang di Program Bootcamp OCBC  
Tahap 1 saya akan belajar C#  
Maliks-MacBook-Air:Sesi2 swijaya$ █
```

Nah ini seputar STRUKTUR PROGRAM yang wajib kita ketahui.

## C# Overview - Sesi 02

# TIPE DATA

Tipe data = Jenis Data.

Contoh :

1234 adalah tipe data numerik atau bilangan bulat

“HelloWorld” adalah tipe data string atau chars

112,12 adalah tipe data float atau bilangan desimal.

Untuk Tipe data Numerik dan Tipe data Float ,

Q : Apakah komputer bisa membedakan mana entitas huruf dan angka ?

Answer : TIDAK BISA!

Lalu Bagaimana agar Komputer dapat membedakannya ?

Caranya : Siapkan sebuah Kotak yang sudah kita labeli lebih dulu dengan sebuah identitas, contoh :

**Kotak1 = 1234**

**Kotak2 = "Hello World"**

**Kotak3 = 112,12**

Dalam dunia programming : Kotak ini kita namakan dengan Variabel lalu Kotak1, Kotak2, dan Kotak3 disebut dengan Deklarasi Variabel.

Lalu dimana tipe datanya ????????????????

**BilanganBulat** kotak1 = 1234;

**Chars** kotak2 = "HelloWorld";

**BilanganDesimal** kotak3 = 112,12;

1234, "HelloWorld", 112,12 disebut dengan Nilai/Value yang kita simpan kedalam masing2 variabel yang sudah di deklarasikan.

## C# Overview - Sesi 02

# Variabel

PANDUAN dalam Deklarasi Variabel :

1. AWALAN variabel **TIDAK BOLEH** ANGKA
2. **TIDAK BOLEH** mengandung karakter khusus seperti \$#@!% dan sebagainya
3. **TIDAK BOLEH** ada variable yang memiliki spasi seperti : “na ma”

Gunakan Nama variable seperti : nama, nama1, nama\_ , nama\_depan, & nama\_belakang.

Format Deklarasi Variabel :

1. [tipe data] [identifier];  
=> `int jumlahSemuaBarang;`
2. [tipe data] [identifier] = [nilai];  
=> `string namaCustomer = "Foxi";`
3. [tipe data] [identifier 1], [identifier 2], [identifier N];  
=> `double harga1, harga2, harga3;`
4. [tipe data] [identifier 1] = [nilai 1], [identifier 2] = [nilai 2], [identifier N] = [nilai N]  
=> `int jumlah1 = 10, jumlah2 = 7, jumlah3 = 12;`



Format Deklarasi Variabel : 1, 2 dan 3 hanya menyatakan variabel saja artinya tidak memberikan nilai apapun untuk variabel tersebut.

Dalam Bahasa C#

Variabel Kosong akan terjadi runtime error, oleh karena itu Variabel Kosong seperti diatas harus diberi nilai sebelum dibaca.

## C# Overview - Sesi 02

# RESERVED KEYWORD

```
1  abstract  as  base  bool
2  break  byte  case  catch
   !!MISSING: command!!
3  char  checked  class  const
4  continue  decimal  default  delegate
   !!MISSING: command!!
5  do  double  else  enum
   !!MISSING: command!! | !!MISSING: command!!
6  event  explicit  extern  false
7  finally  fixed  float  for
8  foreach  goto  if  implicit
   !!MISSING: command!!
9  in  int  interface  internal
10 is  lock  long  namespace
11 new  null  object  operator
   !!MISSING: command!!
12 out  override  params  private
13 protected  public  readonly  ref
14 return  sbyte  sealed  short
15 sizeof  stackalloc  static  string
   !!MISSING: command!! | 0 references
16 struct  switch  this  throw
17 true  try  typeof  uint
18 ulong  unchecked  unsafe  ushort
19 using  using static  virtual  void
20 volatile  while
```

*Reserved Keywords* adalah kata kunci yang tidak boleh digunakan sebagai nama variabel.

Jadi Pada saat membuat Penamaan Variabel hindari nama variabel yang ada pada gambar disamping.

## C# Overview - Sesi 02

# Jenis Tipe Data

Ada 2 Macam tipe data yang disupport oleh C# yaitu :

### 1) Value Types = Tipe Nilai

Tipe data ini akan memuat nilai sebenarnya dari data yang tersimpan.

Kita punya variabel bilangan bulat yang digunakan untuk simpan angka 10, angka 10 ini lah yang disebut value type yang artinya tidak dapat dibiarkan kosong/null.

### 2) Reference Types = Tipe Referensi

Tipe data ini berbeda dengan value type, karena digunakan hanya untuk menyimpan alamat yang menunjuk sebuah lokasi dimana nilai / objek tersebut disimpan.

Kalau Value type tidak bisa menyimpan objek NULL, sedangkan reference memungkinkan untuk bernilai NULL.

2 Jenis Tipe Data ini akan kita bahas lebih detail setelah kita mempelajari Pengalokasian memori.

Script Data Type	C Data Type	.NET Framework class	Rentang Nilai
Number	int	Int32 (int)	-2,147,483,648 ~ 2,147,483,647
Number	long	Int64 (int)	-9,223,372,036,854,775,808 ~ 9,223,372,036,854,775,807
Number	double	Double (double)	+/-1.7 x 10 <sup>308</sup>
Number	float	Single (float)	+/-3.4 x 10 <sup>38</sup>
String	BSTR	String (string)	-
Boolean	VARIANT_BOOL	Boolean (bool)	True atau false

Sources : <https://docs.microsoft.com/en-us/windows/win32/wmp/about--net-framework-data-types>

Pada tabel tipe data di atas, kolom pertama adalah nama alias dari tipe data sedangkan kolom ketiga adalah nama tipe data .NET yang sebenarnya.



# Tipe Numerik

Tipe yang digunakan untuk menangani masalah seperti perhitungan angka. Di bawah ini adalah tipe data yang termasuk dalam tipe numerik.

Tipe Data	Nilai Minimal	Nilai Maksimal
Sbyte	-128	127
Byte	0	255
Short	-32.767	32.767
Ushort	0	65.535
Int	-2.147.483.648	2.147.483.647
UInt	0	4.294.967.295



## C# Overview - Sesi 02

### Tipe Numerik

<b>Long</b>	-9.223.3742.036.854. 775.808	9.223.3742.036.854. 775.807
<b>Ulong</b>	0	18.446.774.073. 709.551.615
<b>Float</b>	$1,5 \times 10^{-28}$	$3,4 \times 10^{38}$
<b>Double</b>	$5,0 \times 10^{-324}$	$1,7 \times 10^{308}$
<b>Decimal</b>	$1,0 \times 10^{-28}$	$7,9 \times 10^{28}$

## C# Overview - Sesi 02

### Project1

Untuk Kali ini kita akan explorasi seputar tipe data numerik melalui menghitung luas persegi panjang dimana panjang = 20 dan lebar = 2.3

Expektasi pada program kita nantinya memberikan nilai akhir Luas Persegi Panjang.

```
Hitung Luas Persegi Panjang
Luas = P x L
Luas = 20x2.3
Luas = 46
□
```

## C# Overview - Sesi 02

### Project1 - Live Code

```
1  using System;
2
3  0 references
4  class Project1
5  {
6      0 references
7      public static void Main(string[] args)
8      {
9          int panjang = 20;
10         double lebar = 2.3;
11
12         Console.WriteLine("Hitung Luas Persegi Panjang");
13         Console.WriteLine("Luas = P x L");
14         Console.WriteLine("Luas = " + panjang + "x" + lebar);
15         Console.WriteLine("Luas = " + panjang * lebar);
16         Console.ReadLine();
17     }
18 }
```



## C# Overview - Sesi 02

# Tipe Boolean

Tipe data boolean memiliki dua nilai, [yang pertama] nilai **true** berarti nilai benar, dan [yang kedua] nilai **false** berarti nilai salah.

Tipe boolean dalam bahasa C# dikenal dengan **bool**.

Contoh:

```
bool                                nilai1                                =                                true;
bool nilai2 = false;
```

Hasil :

```
Maliks-MacBook-Air:Sesi2 swijaya$ cd "/Users/swijaya/Downloads/C#/SourceCode/Sesi2/" && mcs Project2.cs -out:Project2 && mono Project2
Nilai ATrue
Nilai BFalse
```

## C# Overview - Sesi 02

### Project2 - Live Code

```
1  using System;
2
   0 references
3  class Project2
4  {
   0 references
5      public static void Main(string[] args)
6      {
7          bool nilai1 = true;
8          bool nilai2 = false;
9          Console.WriteLine("Nilai A" + nilai1);
10         Console.WriteLine("Nilai B" + nilai2);
11         Console.ReadLine();
12     }
13
14 }
```



## C# Overview - Sesi 02

# Tipe Data String

Tipe Data yang digunakan untuk mengolah sekumpulan karakter.

Contoh kali ini kita akan membuat program PendaftaranPenduduk sederhana

```
=== PROGRAM PENDAFTARAN PENDUDUK ===  
Masukan nama: foxy  
Masukan alamat: tangerang selatan  
Masukan umur: 18  
  
Terimakasih!  
Data Berikut  
Nama: foxy  
Alamat: tangerang selatan  
Umur: 18 tahun  
SUDAH DISIMPAN!
```



## C# Overview - Sesi 02

### Project3 - Live Code

```
1  using System;
2
3  class Project3
4  {
5      static void Main(String[] args)
6      {
7          // membuat variabel kosong
8          string nama;
9          int umur;
10
11          Console.WriteLine("=== PROGRAM PENDAFTARAN PENDUDUK ===");
12          Console.Write("Masukan nama: ");
13          nama = Console.ReadLine();
14          Console.Write("Masukan alamat: ");
15          var alamat = Console.ReadLine();
16          Console.Write("Masukan umur: ");
17          umur = int.Parse(Console.ReadLine());
18
19          Console.WriteLine();
20          Console.WriteLine("Terimakasih!");
21          Console.WriteLine("Data Berikut");
22          Console.WriteLine($"Nama: {nama}");
23          Console.WriteLine($"Alamat: {alamat}");
24          Console.WriteLine($"Umur: {umur} tahun");
25          Console.WriteLine("SUDAH DISIMPAN!");
26      }
27 }
```





## C# Overview - Sesi 02

# Operator

Operator = Symbol

Contoh :

**A \* B**

Q : Pada perkalian dari variabel A dan B diatas , mana yang disebut operator ?

A : \*

Ya Perkalian atau Simbol \* adalah operator.

A dan B disebut Operan

## C# Overview - Sesi 02

# Operator Aritmatika

Operator untuk melakukan operasi aritmatika seperti kali, bagi, tambah, kurang.

Nama Operator	Simbol
Penjumlahan	+
Pengurangan	-
Perkalian	*
Pembagian	/
Sisa Bagi	%
Increment	++
Decrement	--



## C# Overview - Sesi 02

# Operator Aritmatika

Untuk Peng-Aplikasian Operator Aritmatika sudah kita coba pada sesi sebelumnya maka pada praktik ini kita akan buat dengan beberapa penggabungan operator aritmatika

Task :

Buat Sebuah Program dimana User akan memasukkan nilai 1 , 2 & 3 yang kemudian Program bisa menghasilkan AVERAGE/Rata-Rata dari 3 Nilai yang disubmit oleh User.

```
===Menghitung Nilai===  
Masukan nilai PERTAMA: 25  
Masukan nilai KEDUA: 38  
Masukan nilai KETIGA: 72  
Total Nilai adalah: 135  
Rata Rata Nilai adalah: 45
```

## C# Overview - Sesi 02

### HitungNilai - Live Code

```
1  using System;
2
   0 references
3  class HitungNilai
4  {
   0 references
5      public static void Main(string[] args)
6      {
7          int pertama, kedua, ketiga, total;
8          double rata;
9          Console.WriteLine("===Menghitung Nilai===");
10         Console.Write("Masukan nilai PERTAMA: ");
11         pertama = int.Parse(Console.ReadLine());
12         Console.Write("Masukan nilai KEDUA: ");
13         kedua = int.Parse(Console.ReadLine());
14         Console.Write("Masukan nilai KETIGA: ");
15         ketiga = int.Parse(Console.ReadLine());
16
17         total = pertama + kedua + ketiga;
18         rata = total / 3.0;
19         Console.WriteLine("Total Nilai adalah: " + total);
20         Console.WriteLine("Rata Rata Nilai adalah: " + rata);
21         Console.Read();
22     }
23 }
```



## Operator Penugasan/Assignment

Nama Operator	Sombol
Pengisian Nilai	=
Pengisian dan Penambahan	+=
Pengisian dan Pengurangan	-=
Pengisian dan Perkalian	*=
Pengisian dan Pembagian	/=



## Operator Penugasan/Assignment

Pengisian dan Sisa bagi	<code>%=</code>
Pengisian dan shift left	<code>&lt;&lt;=</code>
Pengisian dan shift right	<code>&gt;&gt;=</code>
Pengisian dan bitwise AND	<code>&amp;=</code>
Pengisian dan bitwise OR	<code> =</code>
Pengisian dan bitwise XOR	<code>^=</code>



Operator untuk memberikan tugas pada variabel yang biasanya digunakan untuk mengisi nilai.

Pada Sesi Kali ini kita akan membuat 2 variabel nilai : nilai1 dan nilai2

Dimana nilai1 kita akan isi ulang variable nya sebanyak 5 sedangkan nilai2 kita tambahkan 6 dari nilai semula.

Contoh :

```
mono HitungNilai2
nilai1 = 15
nilai2 = 14
Maliks-MacBook-Air:Sesi2 swijaya$
```

## C# Overview - Sesi 02

### HitungNilai2 - Live Code

```
1 using System;
2
3 class HitungNilai2
4 {
5     public static void Main(String[] args)
6     {
7         // menggunakan operator = untuk mengisi nilai
8         int nilai1 = 10;
9         int nilai2 = 8;
10
11         // mengisi ulang nilai variabel mangga
12         nilai1 = 15;
13
14         Console.WriteLine($"nilai1 = {nilai1}");
15
16         // menggunakan += untuk mengisi dan menjumlahkan
17         nilai2 += 6;
18
19         Console.WriteLine($"nilai2 = {nilai2}");
20     }
21 }
```





## C# Overview - Sesi 02

# Operator Perbandingan

Nama Operator	Simbol
Lebih Besar	>
Lebih Kecil	<
Sama Dengan	==
Tidak Sama dengan	!=
Lebih Besar Sama dengan	>=
Lebih Kecil Sama dengan	<=



Operator Perbandingan ini juga dikenal dengan operator relasi.

Yang digunakan untuk membandingkan dua buah nilai dan hasil perbandingan akan berupa boolean : True dan False.

Contoh :

Praktik sesi ini kita hanya akan menggunakan semua perbandingan dengan menggunakan source-code HitungNilai2 dan menggunakan seluruh operator yang ada pada tabel diatas.

```
mono HitungNilai3
jumlah nilai1 = 10
jumlah nilai2 = 8
Hasil perbandingan:
nilai1 > nilai2 : True
nilai1 >= nilai2 : True
nilai1 < nilai2 : False
nilai1 <= nilai2 : False
nilai1 == nilai2 : False
nilai1 != nilai2 : True
```



## C# Overview - Sesi 02

### HitungNilai3 - Live Code

```
1  using System;
2
3  0 references
4  class HitungNilai3
5
6      0 references
7      public static void Main(String[] args)
8      {
9          int nilai1, nilai2 = 0;
10
11          Console.Write("jumlah nilai1 = ");
12          nilai1 = int.Parse(Console.ReadLine());
13          Console.Write("jumlah nilai2 = ");
14          nilai2 = int.Parse(Console.ReadLine());
15
16          Console.WriteLine("Hasil perbandingan: ");
17          Console.WriteLine($"nilai1 > nilai2 : {nilai1 > nilai2}");
18          Console.WriteLine($"nilai1 >= nilai2 : {nilai1 >= nilai2}");
19          Console.WriteLine($"nilai1 < nilai2 : {nilai1 < nilai2}");
20          Console.WriteLine($"nilai1 <= nilai2 : {nilai1 <= nilai2}");
21          Console.WriteLine($"nilai1 == nilai2 : {nilai1 == nilai2}");
22          Console.WriteLine($"nilai1 != nilai2 : {nilai1 != nilai2}");
23      }
```



## C# Overview - Sesi 02

# Operator Logika

Nama Operator	Simbol
Logika AND	&&
Logika OR	
Negasi/kebalikan	!

Hasil operasi dari operator logika sama seperti operator perbandingan, yakni boolean True dan False. Namun perlu diingat: Operan harus bertipe boolean. Karena jika tidak, maka akan terjadi error.

Operator Logika digunakan untuk membuat operasi logika.

Contoh :

Pernyataan: Nilai1 bernilai Genap  
Pernyataan: Nilai2 bernilai Genap

Pernyataan 1	Pernyataan 2	Hasil Logika AND
true	true	true
true	false	false
false	true	false
false	false	false

Kita coba implementasi dengan tabel  
Logic disamping :

Nilai1 dan Nilai2 :

Nilai1 atau Nilai2 :

Pernyataan 1	Pernyataan 2	Hasil Logika OR
true	true	true
true	false	true
false	true	true
false	false	false



Bukan Nilai1 :

Pernyataan 1	Hasil Logika NOT
true	false
false	true

Logika AND, OR, dan NOT nantinya akan banyak kita pakai dalam membuat logika di program.

Nah sekarang kita coba implementasikan dalam program C# dengan Logika AND

Dimana Kondisi Program akan mengizinkan user sign in jika :

Usia : > 18 dan Password : OCBC => Welcome To The OCBC

\*Disclaimer : Untuk Penggunaan case ini kita akan menggunakan percabangan IF-ELSE  
All detail terkait Logika Pemogramman akan kita bahas di sesi 3 ya. Untuk Sekarang cukup ikuti dulu.

## C# Overview - Sesi 02

### HitungNilai4 - Results

Benar

```
mono HitungNilai4
Enter your age: 20
Password: 0CBC
WELCOME TO THE CLUB!
Maliks-MacBook-Air:Sesi2 swijaya$
```

Salah

```
mono HitungNilai4
Enter your age: 17
Password: 0CBC
Sorry, try again!
Maliks-MacBook-Air:Sesi2 swijaya$
```



## C# Overview - Sesi 02

# HitungNilai4 - LIVE CODE

```
1 using System;
2
3 class HitungNilai4
4 {
5     public static void Main(String[] args)
6     {
7
8         Console.Write("Enter your age: ");
9         int age = int.Parse(Console.ReadLine());
10        Console.Write("Password: ");
11        string password = Console.ReadLine();
12
13        bool isAdult = age > 18 ; // pernyataan 1
14        bool isValidPassword = password == "0CBC"; // pernyataan 2
15
16        // menggunakan logika AND
17        if(isAdult && isValidPassword)
18        {
19            Console.WriteLine("WELCOME TO THE CLUB!");
20        }
21        else
22        {
23            Console.WriteLine("Sorry, try again!");
24        }
25    }
26 }
27
```

