

# MODUL 1

## PENGENALAN BAHASA C#

### A. Kompetensi Dasar

- Perkembangan dan sejarah C#
- Instal dan konfigurasi C# di visual studio code
- Penamaan file dan class dalam C#
- Sintaks dasar C#

### B. Alokasi Waktu

6 JP (6 x 45 menit)

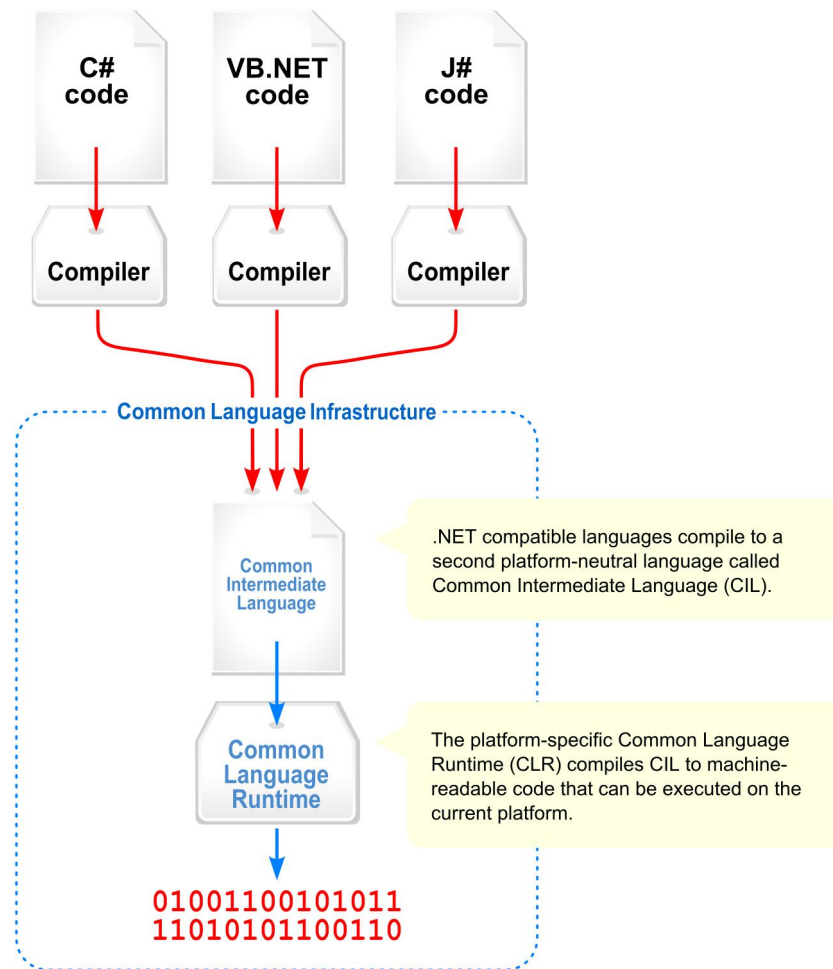
### C. Petunjuk

- Awali setiap aktivitas dengan doa, semoga berkah dan mendapat kemudahan
- Pahami tujuan dasar teori dan latihan praktikum dengan baik dan benar
- Kerjakan tugas dengan baik, sabar, teliti dan jujur
- Tanyakan pada gurru apabila mengalami kesulitan dan ada hal yang kurang jelas

### D. Dasar Teori

C# (baca : C sharp) adalah bahasa pemrograman sederhana, modern, berorientasi objek dan bekerja dengan berbagai compiler, serta platform khusus seperti Microsoft .NET Framework yang dapat dikembangkan di berbagai aplikasi seperti web, mobile, desktop dan game. Dotnet itu semacam mesin virtual yang tugasnya untuk menjalankan program C#, F#, VB.NET dan program lainnya. Selain itu, Dotnet juga menyediakan tools, libary, dan API yang kita butuhkan untuk membuat program C#. Jika dotnet tidak ada, maka kita tidak isa menjalankan program yang dibuat dengan C#.

Syntax bahasa pemrograman C# mirip dengan bahasa pemrograman lainnya, seperti bahasa pemrograman Java. Sebagai contoh, kode program terdiri dari pernyataan yang diakhiri dengan titik koma (;) dan kurung kurawal ({dan}) digunakan untuk mengelompokkan pernyataan, seperti dalam perulangan foreach.



**Gambar 1. Compiler dalam C#**

- Sejarah singkat C#

Tahun 1999 Anders Hejlsberg membentuk tim di Microsoft untuk memuat bahasa pemrograman baru yang diberi nama **Cool (C-Like Object Oriented Language)**, karena melanggar trademark dari produk lain maka pada tahun 2000 nama Cool diubah menjadi C#. Bahasa C# lebih direkomendasikan untuk membuat Game dibandingkan bahasa Java. Bahkan tidak hanya di Game saja, C# juga banyak dipakai untuk membuat aplikasi desktop dan Web.

Berikut ini versi C# dan tahun rilisnya:

- ◆ C# 1.0 (January 2002) – .NET Framework 1.0
- ◆ C# 1.1 (April 2003) – .NET Framework 1.1
- ◆ C# 1.2 (April 2003) – .NET Framework 1.1
- ◆ C# 2.0 (November 2005) – .NET Framework 2.0, .NET Framework 3.0
- ◆ C# 3.0 (Agustus 2007) – .NET Framework 2.0 (Except LINQ), .NET Framework 3.0 (Except LINQ), .NET Framework 3.5
- ◆ C# 4.0 (April 2010) – .NET Framework 4
- ◆ C# 5.0 (Agustus 2012) – .NET Framework 4.5
- ◆ C# 6.0 (Juli 2015) – .NET Framework 4.6, .NET Core 1.0, .NET Core 1.1
- ◆ C# 7.0 (Maret 2017) – .NET Framework 4.7
- ◆ C# 7.1 (Agustus 2017) – .NET Core 2.0

- ◆ C# 7.2 (November 2017) – .NET Core 2.0
- ◆ C# 7.3 (Mei 2018) – .NET Core 2.1, .NET Core 2.2, .NET Framework 4.8
- ◆ C# 8.0 (September 2019) – .NET Core 3.0

- Struktur Dasar Program C#

Struktur program C# yang paling dasar, terdiri dari tiga bagian:

1. Bagian deklarasi pustaka

Ini adalah bagian paling atas dari program C#. Pada bagian ini, kita menuliskan pustaka (library) yang dibutuhkan dalam program. Pustaka berisi sekumpulan fungsi, method, class, objek, konstanta, dan variabel yang bisa kita gunakan ulang di dalam program.

2. Bagian Class

Class adalah nama program, sehingga nama class harus sama dengan nama program. Blok class dapat diisi dengan fungsi dan variabel.

3. Bagian Fungsi atau Method

Pada bagian ini, kita bisa menuliskan fungsi-fungsi dari program. Fungsi yang harus ada di dalam setiap program adalah fungsi Main().

Kalau tidak ada fungsi ini, program tidak akan bisa dijalankan. Karena fungsi Main() merupakan fungsi utama yang akan dieksekusi pertama kali. Oleh sebab itu, kita biasanya akan banyak menulis kode program di dalam fungsi Main().

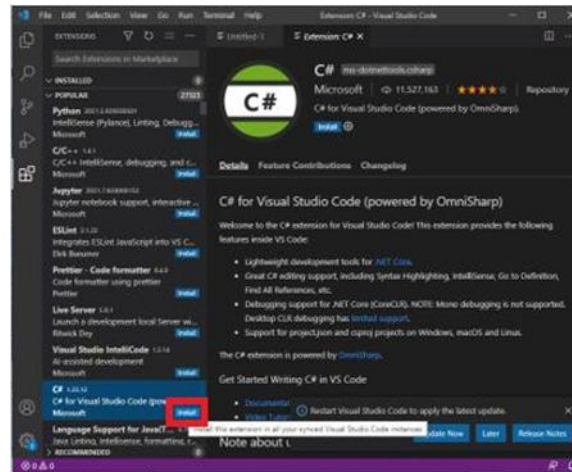
```
// 1. Deklarasi pustaka
using System;

// 2. Bagian Class
class ProgramHello
{
    // 3. Bagian Fungsi
    static void Main(String[] args)
    {
        Console.WriteLine("Hello World!");
    }
}
```

**Gambar 2. Sintaks Dasar Bahasa Pemrograman C#**

## E. Instalasi dan Konfigurasi

- Untuk siswa yang sudah memiliki laptop:
  1. Langkah pertama dimulai dengan *install .NET Framework SDK* dengan *download file installer* di <https://dotnet.microsoft.com/download>.
  2. Buka Visual Studio Code
  3. Lik tab Extensions yang ada di sebelah kiri, kemudian cari C#
  4. Kemudian klik install untuk extensions C#

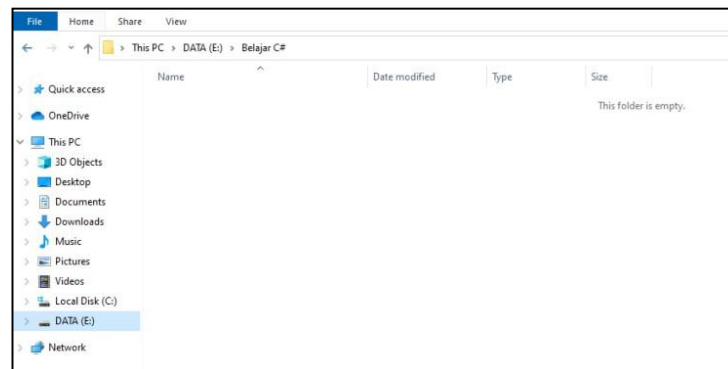


**Gambar 3. Extensions C#**

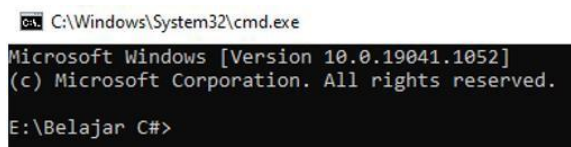
5. Jika extensions sudah selesai diinstal, maka tutup dulu VSCode nya.

6. Menjalankan aplikasi VS Code

- a. Buat sebuah folder di lokasi manapun di dalam direktori komputer anda. Jika kalian menggunakan sistem operasi *Windows*, buka folder tersebut, lalu ketik *cmd* di *directory bar* pada *Windows Explorer* kemudian tekan enter untuk membuka *command prompt* pada direktori tersebut.



**Gambar 4. Pembuatan Folder**



**Gambar 5. Tampilan Command Prompt di Windows**

- b. Ketik **dotnet new console** pada *command prompt* untuk membuat aplikasi *console* di .NET

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19041.1052]
(c) Microsoft Corporation. All rights reserved.

E:\Belajar C#>dotnet new console
The template "Console Application" was created successfully.

Processing post-creation actions...
Running 'dotnet restore' on E:\Belajar C#\Belajar C#.csproj...
Determining projects to restore...
Restored E:\Belajar C#\Belajar C#.csproj (in 180 ms).
Restore succeeded.
```

**Gambar 6. Tampilan Perintah Dotnet New Console**

- c. **Setelah** selesai ketik **code .** pada jendela *command prompt* untuk membuka Visual Studio Code

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19041.1052]
(c) Microsoft Corporation. All rights reserved.

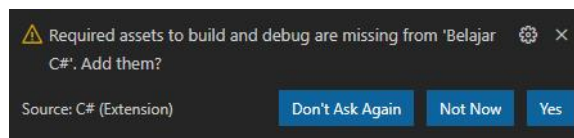
E:\Belajar C#>dotnet new console
The template "Console Application" was created successfully.

Processing post-creation actions...
Running 'dotnet restore' on E:\Belajar C#\Belajar C#.csproj...
Determining projects to restore...
Restored E:\Belajar C#\Belajar C#.csproj (in 180 ms).
Restore succeeded.

E:\Belajar C#>code .
E:\Belajar C#>
```

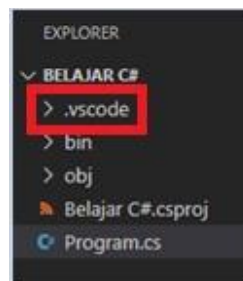
**Gambar 7. Tampilan Perintah Code**

- d. Setelah VS Code terbuka, maka muncul pesan agar kalian menambahkan aset yang dibutuhkan untuk debugging seperti gambar di bawah ini. Tekan tombol “Yes”.



**Gambar 8. Tampilan Required Assets**

- e. Akan muncul folder `.vscode` di panel *Explorer* di sisi kiri editor.



**Gambar 9. Tampilan folder .vscode**

- f. Menulis kode program

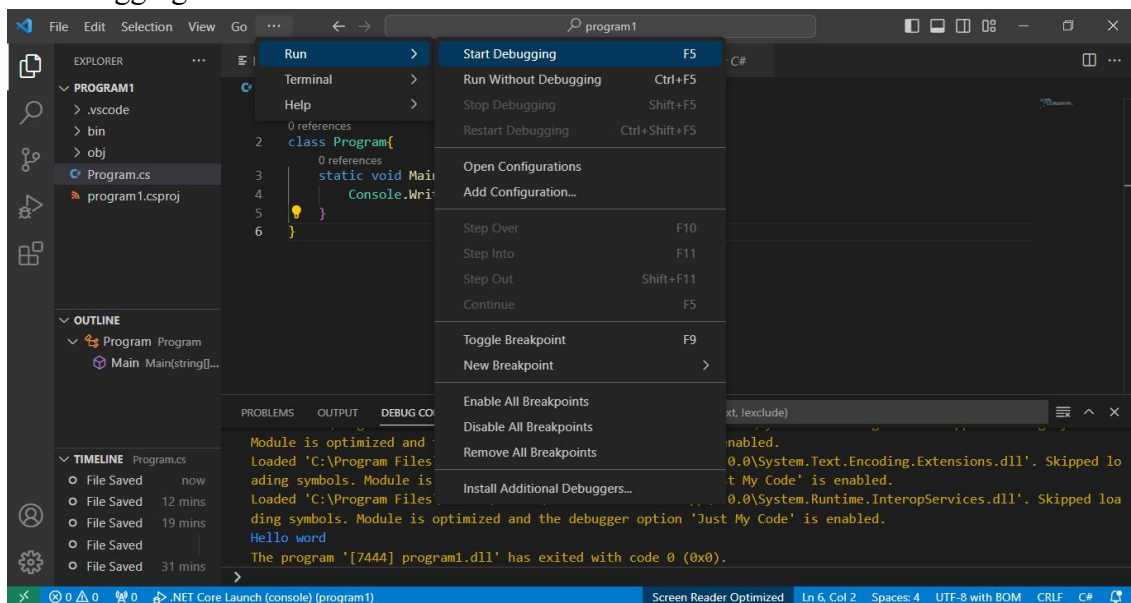
Pilih file `Program.cs` di panel *Explorer*. File tersebut berisi

kode dasar untuk mencetak kalimat “*Hello World*”.

```
Program.cs x
Program.cs > ...
1  using System;
2
3  namespace Belajar_C_
4  {
5      0 references
6      class Program
7      {
8          0 references
9          static void Main(string[] args)
10         {
11             Console.WriteLine("Hello World!");
12         }
13     }
14 }
```

**Gambar 10. Sintaks HelloWorld**

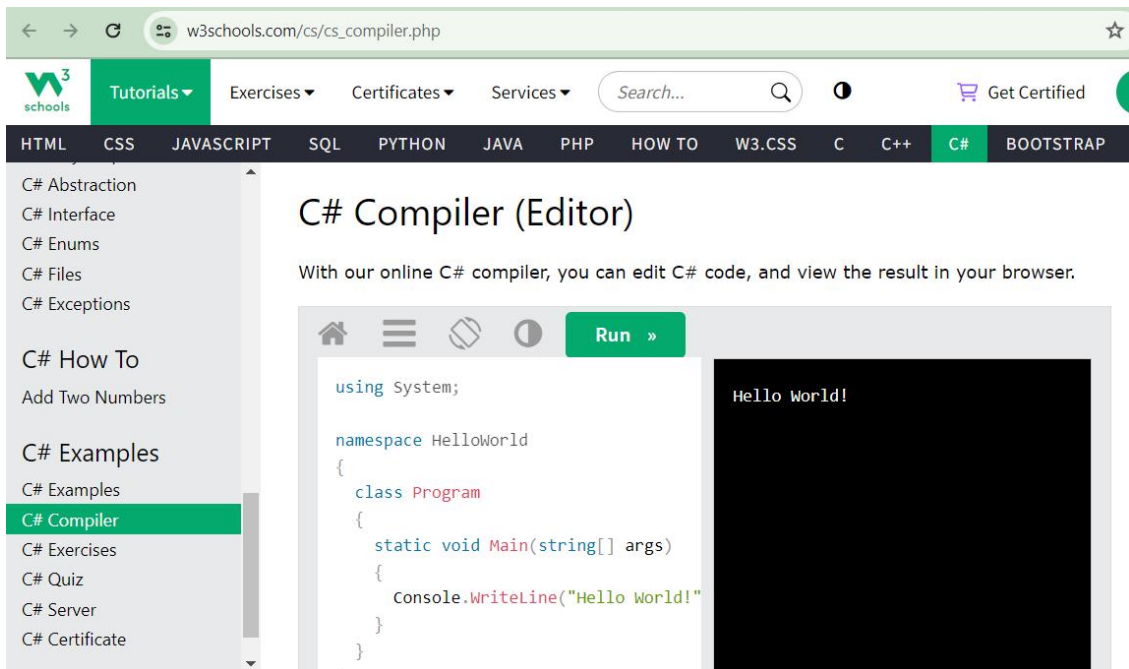
7. Untuk running program, klik titik tiga di atas, pilih run, pilih start debugging



**Gambar 11. Running Program**

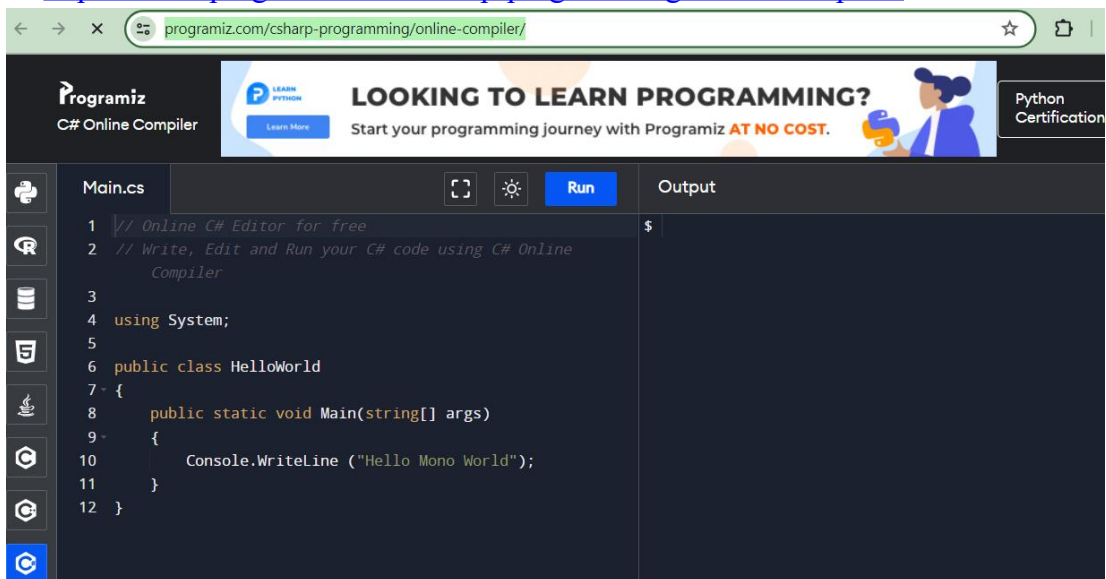
8. Output program akan tampil.

- Untuk siswa yang belum memiliki laptop:
  1. Silakan buka browser melalui HP kalian
  2. Kalian bisa menggunakan salah satu C# compiler online berikut:
    - A. [https://www.w3schools.com/cs/cs\\_compiler.php](https://www.w3schools.com/cs/cs_compiler.php)



**Gambar 12. Tampilan C# Compiler di w3school**

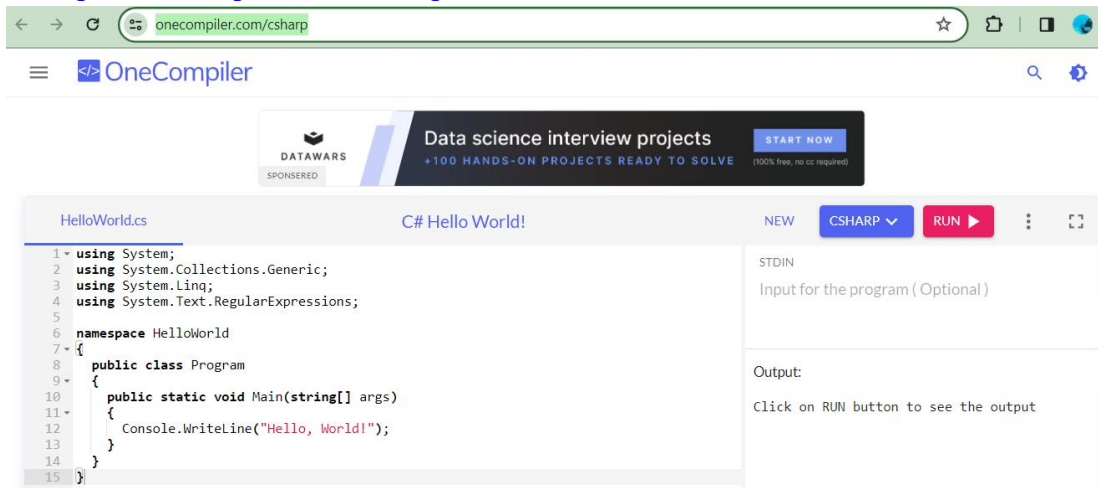
B. <https://www.programiz.com/csharp-programming/online-compiler/>



**Gambar 13. Tampilan C# Compiler di programiz**



C. <https://onecompiler.com/csharp>



**Gambar 14. Tampilan C# Compiler di onecompiler**

3. Kelemahannya jika kalian menggunakan compiler online, maka sintaksnya tidak bisa disimpan.
4. Solusinya kalian harus mengcopy semua sintaks dan mengopynya ke penyimpanan lain.

**F. Tugas Mandiri**

1. Buatlah program C# untuk menampilkan nama, kelas, tempat dan tanggal lahirmu!
2. Carilah karakter khusus dalam pemrograman C# dan fungsinya!
3. Carilah macam-macam tipe data dalam pemrograman C#!
4. Carilah operator dalam pemrograman C#!
5. Jelaskan yang kalian ketahui tentang operator increment dan decrement!