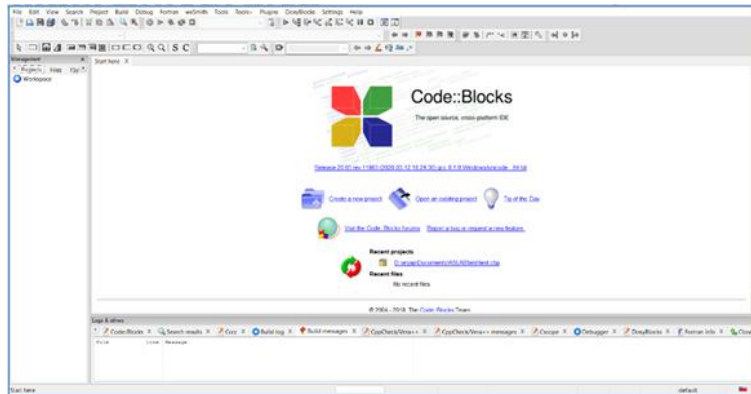


I. Guided

➤ Instalasi Code Blocks

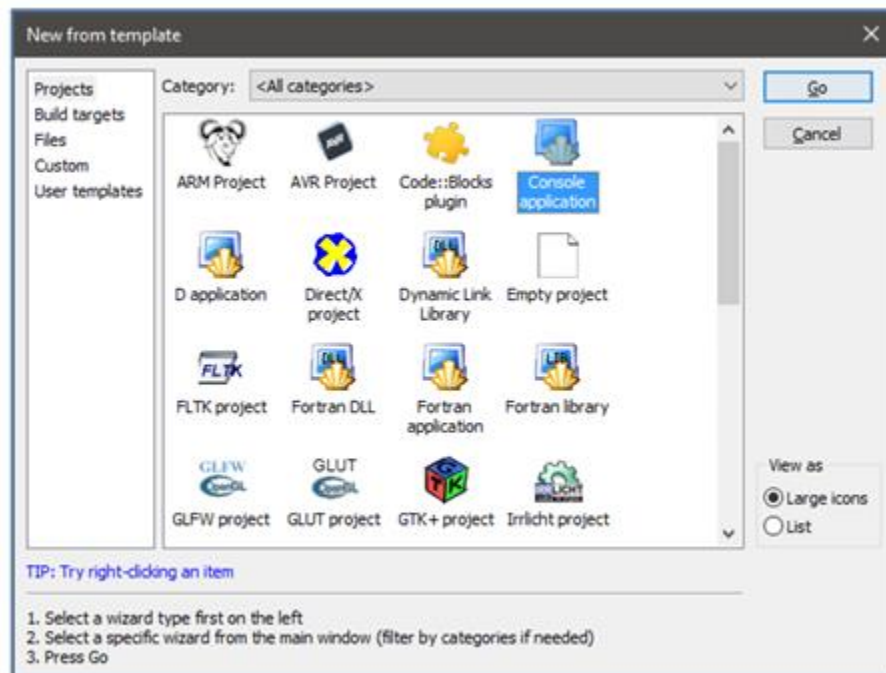
Adapun cara menginstall Code Blocks adalah sebagai berikut.

1. Download terlebih dahulu file exe pada <http://www.codeblocks.org/downloads>. Pilih Download the binary release kemudian pilih file yang menggunakan mingw-setup (e.g. codeblocks-20.03mingw-setup.exe).
2. Setelah itu install file tersebut, akan muncul tampilan seperti pada gambar ini:



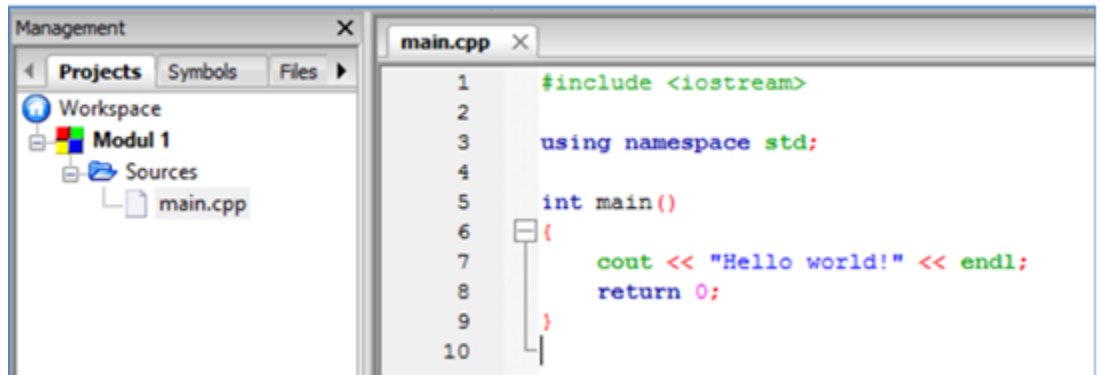
➤ Cara Menggunakan Code Blocks

1. Membuat Project Baru dengan cara memilih File > New > Projects. Kemudian pada panel kiri pilih Project, pada panel kanan pilih Console application kemudian klik Go seperti pada gambar ini

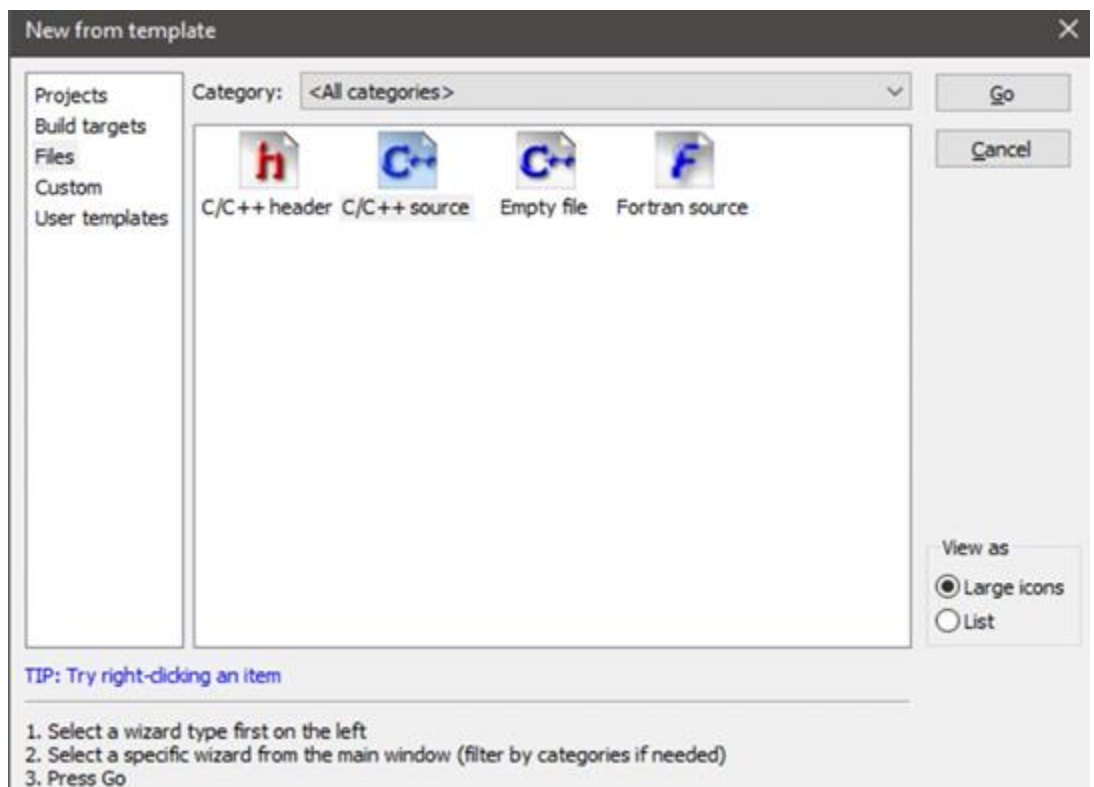


Setelah itu klik Next > Pilih Bahasa Pemrograman yang akan digunakan > Isi Project title dan Folder to create project in (tempat menyimpan project) > Klik Finish.

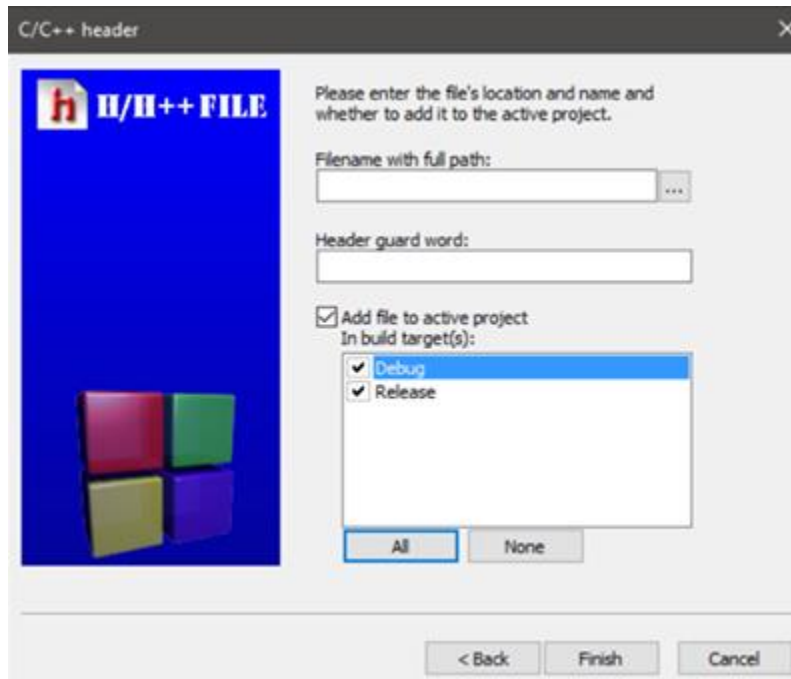
2. Menulis Sintak pada editor seperti pada Error! Reference source not found.



3. Membuat Class Baru dengan cara klik File > New > File. Pada panel kiri pilih Files, dan pada panel kanan pilih C/C++ source Kemudian Klik Go seperti pada Error! Reference source not found.



Kemudian klik Next > Pilih bahasa pemrograman > Isi Filename with full path > Centang all in build target > Finish, seperti pada Error! Reference source not found.



Jika anda lupa mencentang build target, dapat dilakukan setting manual dengan cara klik kanan pada project > properties > Build targets > Debug > Centang semua target files seperti pada Error! Reference source not found..

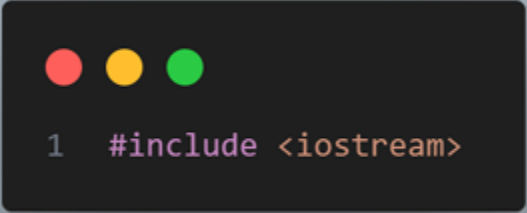
```

1  #include <iostream>
2  #include <conio.h>
3
4  using namespace std;
5  int main(){
6      cout<<"Hello World!"<<endl;
7      getch();
8      return 0;
9  }
```

Setelah di-compile (F9) dan di-run (Ctrl+F9) maka hasil keluaran programnya adalah sebagai berikut:


➤ Dasar Pemograman Bahasa C++

1. Pendeklarasian library yang akan digunakan di dalam program menggunakan “#include ” digunakan untuk mendecklarasikan fungsi c++.




```
1  #include <iostream>
```

2. Pendekralasian namespace “using namespace std;” digunakan untuk menyatakan bahwa program akan menggunakan semua entitas (seperti kelas, fungsi, dan objek) yang berada dalam namespace std tanpa harus menyebutkan namespace tersebut secara eksplisit setiap kali.




```
1  using namespace std;
```

3. Pendekralasian namespace “using namespace std;” digunakan untuk menyatakan bahwa program akan menggunakan semua entitas (seperti kelas, fungsi, dan objek) yang berada dalam namespace std tanpa harus menyebutkan namespace tersebut secara eksplisit setiap kali.



```
1  int main(){
```

4. Pendekralasian variabel variabel dengan tipe data dasar di C++, seperti “int”, “float”, “double”, “string”, “char”, dan “bool”, yang digunakan untuk menyimpan berbagai jenis nilai seperti bilangan bulat, bilangan pecahan, teks, karakter, dan nilai logika.




```

1  int main(){
2      int angka = 10;
3      float desimal = 10.5;
4      string kalimat = "aldi";
5      double tinggi = 10.4;
6      char jenis_kelamin = "laki-laki";
7      bool isSunny = true;
8

```

Dari salah satu variable yaitu “angka akan dicetak menggunakan fungsi “cout” digunakan untuk menghasilkan output pada variable pada angka.



```

1  cout << "Angka;" << angka << endl;

```

Maka akan menghasilkan output sebagai berikut:

```
Angka:10
```

```

Process returned 0 (0x0)   execution time : 0.060 s
Press any key to continue.

```

5. Pendekralasian fungsi input agar menghasilkan output yang kita mau menggunakan fungsi “cout << "Masukkan angka: ";”

```

1  int main(){
2      int angka;
3      cout << "Masukkan angka: ";
4      cin >> angka;
5
6      cout << "Angka: " << angka << endl;
7
8  }

```

Maka akan menghasilkan output sebagai berikut:

```

masukan angka: 10
Angka 10
Zhafiravel

Process returned 0 (0x0)   execution time : 1.877 s
Press any key to continue.
|

```

6. Pendekralasian fungsi operator digunakan untuk mengoperasikan hasil perhitungan pada variabel.

```

#include <iostream>
using namespace std;

int main(){
    int angka1 = 8;
    int angka2 = 3;
    int hasil = angka1 + angka2;
    cout << "Hasilnya adalah " << hasil << endl;
    cout << "Zhafiravel" << endl;
}

```

Dari hasil input diatas akan menghasilkan output sebagai berikut:

```

Hasilnya adalah 11
Zhafiravel

Process returned 0 (0x0)   execution time : 0.061 s
Press any key to continue.
|

```

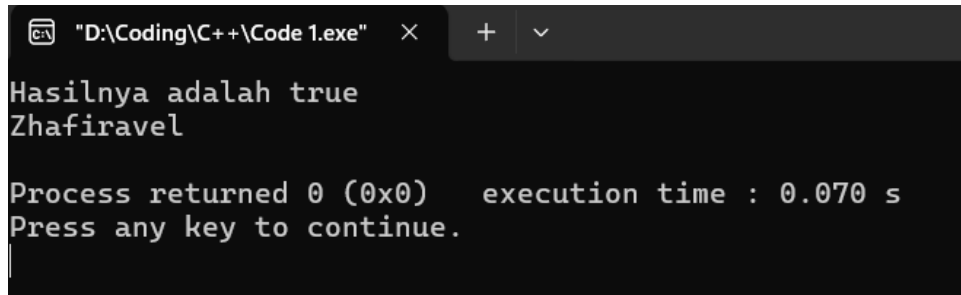
7. Pendekralasian fungsi perbandingan digunakan untuk membandingkan dua angka yang nilainya sama. Hasil perbandingan tersebut disimpan dalam variabel boolean hasil, yang kemudian dicetak ke layar. Dengan menggunakan boolalpha, nilai boolean ditampilkan dalam bentuk teks "true" atau "false".

```
#include <iostream>
using namespace std;

int main(){
    int angka1 = 5;
    int angka2 = 5;

    bool hasil = (angka1 == angka2);
    cout << "Hasilnya adalah " << boolalpha << hasil << endl;
    cout << "Zhafiravel" << endl;
}
```

Output:



```
"D:\Coding\C++\Code 1.exe" x + v
Hasilnya adalah true
Zhafiravel

Process returned 0 (0x0)   execution time : 0.070 s
Press any key to continue.
```

8. Pendekralasian fungsi operator logika berfungsi sebagai menunjukkan perubahan pada kondisi1 dan kondisi2 berupa true atau false.

```
#include <iostream>
using namespace std;

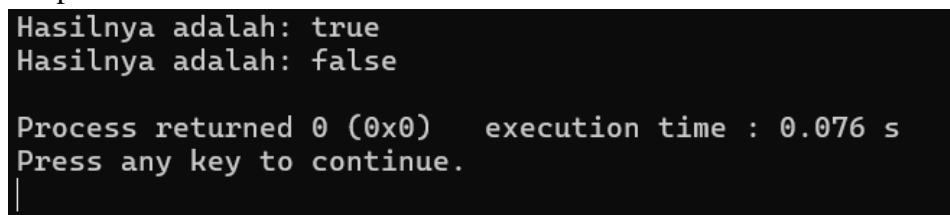
int main(){
    bool kondisi1 = true;
    bool kondisi2 = true;

    bool hasil = (kondisi1 && kondisi2);
    cout << "Hasilnya adalah: " << boolalpha << hasil << endl;

    bool kondisi1 = false;
    bool hasil = kondisi1;
    cout << "Hasilnya adalah: " << boolalpha << hasil << endl;

    cout << "Zhafiravel" << endl;
}
```

Output:



```
Hasilnya adalah: true
Hasilnya adalah: false

Process returned 0 (0x0)   execution time : 0.076 s
Press any key to continue.
```

9. Pendekralasian fungsi percabangan berfungsi untuk menerima input berupa kata dari pengguna. Jika kata yang dimasukkan oleh pengguna sesuai dengan

"HALO", maka program akan menampilkan pesan "Kata sesuai". Namun, jika kata yang dimasukkan berbeda, program akan menampilkan pesan "Kata tidak sesuai".

```
#include <iostream>
#include <string>

using namespace std;

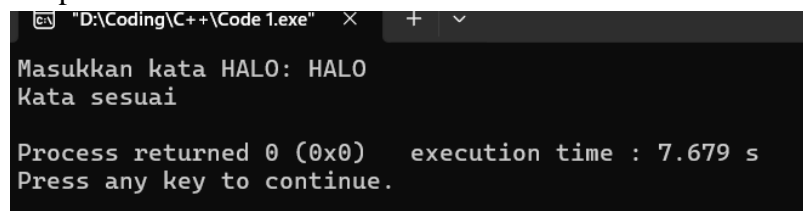
int main() {
    string kata;

    cout << "Masukkan kata HALO: ";
    cin >> kata;

    if (kata == "HALO") {
        cout << "Kata sesuai" << endl;
    } else {
        cout << "Kata tidak sesuai" << endl;
    }

    return 0;
}
```

Output:

A screenshot of a Windows command prompt window titled "D:\Coding\C++\Code 1.exe". The window shows the output of a C++ program. The first line is "Masukkan kata HALO: HALO", followed by "Kata sesuai" on the next line. At the bottom, it says "Process returned 0 (0x0) execution time : 7.679 s" and "Press any key to continue.".

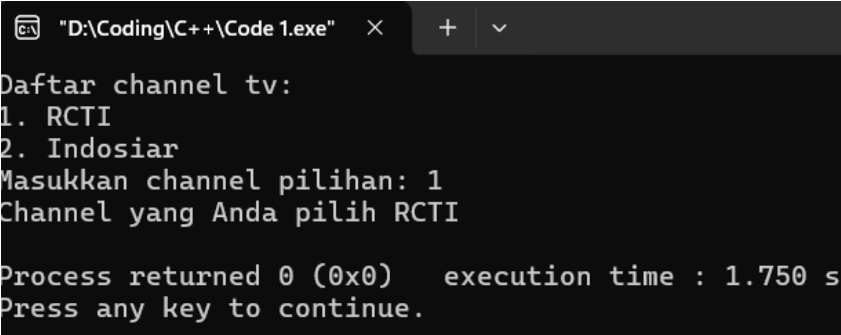
```
"D:\Coding\C++\Code 1.exe" X + v
Masukkan kata HALO: HALO
Kata sesuai

Process returned 0 (0x0)   execution time : 7.679 s
Press any key to continue.
```


Berikut contoh kedua pada percabangan yang menerima input dari pengguna apabila nomor list yang dimasukan pengguna sesuai daftar pada inputan maka program akan memunculkan sebagai contoh “channel yang anda pilih RCTI”

```
int main() {  
    int tv;  
  
    cout << "Daftar channel tv:" << endl;  
    cout << "1. RCTI" << endl;  
    cout << "2. Indosiar" << endl;  
  
    cout << "Masukkan channel pilihan: ";  
    cin >> tv;  
  
    switch (tv) {  
        case 1:  
            cout << "Channel yang Anda pilih RCTI" << endl;  
            break;  
        case 2:  
            cout << "Channel yang Anda pilih Indosiar" << endl;  
            break;  
        default:  
            cout << "Channel tidak tersedia" << endl;  
    }  
  
    return 0;  
}
```

Output:



```
"D:\Coding\C++\Code 1.exe" × + ▾  
Daftar channel tv:  
1. RCTI  
2. Indosiar  
Masukkan channel pilihan: 1  
Channel yang Anda pilih RCTI  
  
Process returned 0 (0x0)   execution time : 1.750 s  
Press any key to continue.
```

10. Pendekralasian fungsi perulangan digunakan sebagai mencetak teks secara berulang, dengan variasi pada cara perulangan dilakukan,

```

int main() {
    int i;

    // Perulangan pertama
    for(i = 0; i < 5; i++) {
        cout << "hello world" << endl;
    }

    // Perulangan kedua
    for(int i = 0; i < 5; i++) {
        cout << "hello world" << endl;
    }

    // Perulangan ketiga
    for(int i = 6; i > 0; i--) {
        cout << i - 1 << " hello world" << endl;
    }

    return 0;
}

```

Output:

```

5 hello world
4 hello world
3 hello world
2 hello world
1 hello world
0 hello world
hello world
hello world
hello world
hello world
hello world

```