

## **MODUL 2**

# **PENGENALAN ARDUINO IDE**

---

Arduino merupakan alat elektronik yang bersifat open-source dan mudah digunakan baik dari segi perangkat keras ataupun perangkat lunak. Arduino memiliki sebuah board yang digunakan sebagai input dan output. Input dapat berupa sensor, button, dan komunikasi data. Output dapat berupa motor, lampu led, dan komunikasi data.

Arduino dapat diprogram dengan menggunakan aplikasi Arduino Software (IDE) atau bisa juga secara Online menggunakan Arduino Web Editor.

### **Struktur Pemrograman Arduino**

Secara umum struktur program pada Arduino berbasis pada Bahasa C yang terdiri atas 2 (dua) bagian besar yaitu Fungsi setup() dan Fungsi loop().

### **Fungsi Setup()**

Fungsi setup() merupakan bagian program yang pertama kali dieksekusi ketika sketch berjalan. Inisialisasi yang ada pada fungsi setup() yaitu variabel, pin mode, library, serial. Setup() hanya dijalankan satu kali ketika program dijalankan atau ketika Arduino dalam kondisi reset.

```
int buttonPin = 3;

void setup()
{
  Serial.begin(9600);
  pinMode(buttonPin, INPUT);
}

void loop()
{
  // ...
}
```

### **Fungsi Loop()**

Fungsi loop() dijalankan setelah fungsi setup() dan fungsi loop() merupakan program yang akan terus dijalankan selama alat dalam kondisi hidup. Inisialisasi pada fungsi loop() diantaranya yaitu inisialisasi nilai variabel, pengendalian program, dan proses.

```

const int buttonPin = 3;

// setup initializes serial and the button pin
void setup()
{
  Serial.begin(9600);
  pinMode(buttonPin, INPUT);
}

// loop checks the button pin each time,
// and will send serial if it is pressed
void loop()
{
  if (digitalRead(buttonPin) == HIGH)
    Serial.write('H');
  else
    Serial.write('L');

  delay(1000);
}

```

## Komentar

Komentar diberikan untuk memberikan penjelasan program atau keterangan dari fungsi program. Penggunaan komentar didahului dengan tanda `/*` dan diakhiri dengan tanda `*/`, jika komentar lebih dari tiga baris maka setiap baris kecuali komentar pembuka `/*` dan komentar penutup `*/` didahului dengan tanda `*`.

```

/*
 * Program Arduino
 *
 * Contoh Arduino.
 * Komentar pada Arduino
 *
 * http://www.arduino.cc/
 */

```

Jenis komentar yang lain yaitu menggunakan komentar perbaris yang menggunakan tanda `//`.

```

int ledPin = 13;  //LED pada digital pin 13

```

## Variabel

Variabel merupakan sebuah ruang penyimpanan data yang memiliki nama variabel, jenis tipe data, dan nilai variabel. Pembuatan variabel sering disebut deklarasi.

```

int pin = 13;

```

Seperti deklarasi variabel diatas yang memiliki nama variabel `pin`, dengan tipe data integer (`int`), dan memiliki nilai 13 dimana 13 merupakan nilai pin yang ada

pada Arduino Board. Untuk nilai variabel tidak harus nilai pin pada Arduino Board tetapi bisa nilai apapun dengan menyesuaikan kebutuhan data dan tipe data.

## Fungsi

Fungsi adalah sekumpulan instruksi untuk mengerjakan suatu keperluan tertentu dengan hasil akhir pengembalian nilai dari keperluan tersebut.

```
void setup()
{
  pinMode(13, OUTPUT);
}
```

Nama dari fungsi diatas yaitu setup, badan fungsi dimulai dari tanda { dan diakhiri dengan }, sehingga program yang berada antara dua tanda tadi akan berjalan ketika fungsi dipanggil.

## Fungsi pinMode()

Pada program fungsi diatas terdapat sebuah fungsi pinMode() yaitu fungsi untuk mendeklarasikan sebuah pin pada Arduino Board berdasarkan dua parameter. Parameter yang ada dalam fungsi pinMode(13, OUTPUT) ; yaitu 13 dimana nilai dari pin yang ada dalam Arduino Board bisa juga diisi dengan nama variabel yang sudah dideklarasikan dan OUTPUT yaitu jenis dari kegunaan pin tersebut bisa diisi dengan INPUT atau OUTPUT.

```
int ledPin = 13;
void setup()
{
  pinMode(ledPin, OUTPUT);
}
```

atau,

```
void setup()
{
  pinMode(13, OUTPUT)
}
```

## Fungsi digitalWrite()

Fungsi digitalWrite() digunakan untuk memberikan nilai tegangan pada pin yang bersifat sebagai Output, jika tegangan diberi tegangan maka nilainya HIGH atau bisa dengan menggunakan nilai 1 sedangkan jika tidak diberi tegangan nilainya LOW atau bisa dengan menggunakan nilai 0.

```
digitalWrite(ledPin, HIGH);
```

### Fungsi delay()

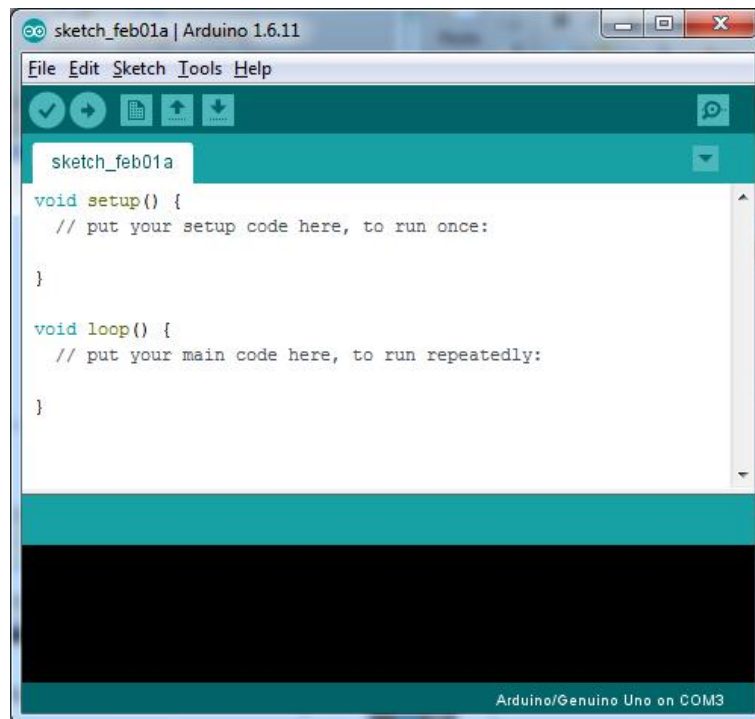
Fungsi delay() berfungsi untuk memberikan waktu jeda sebelum melanjutkan program selanjutnya. Delay yang diberikan satuannya yaitu miliseconds, jika dikonversikan ke second yaitu 1000 miliseconds.

```
delay(1000);
```

## PENGUNAAN SKETCH PADA ARDUINO IDE






Untuk memberikan gambaran tentang penggunaan sketch pada Arduino IDE, silahkan mengikuti langkah-langkah berikut.

1. Buka aplikasi **Arduino**, akan muncul tampilan awal sebagai berikut.

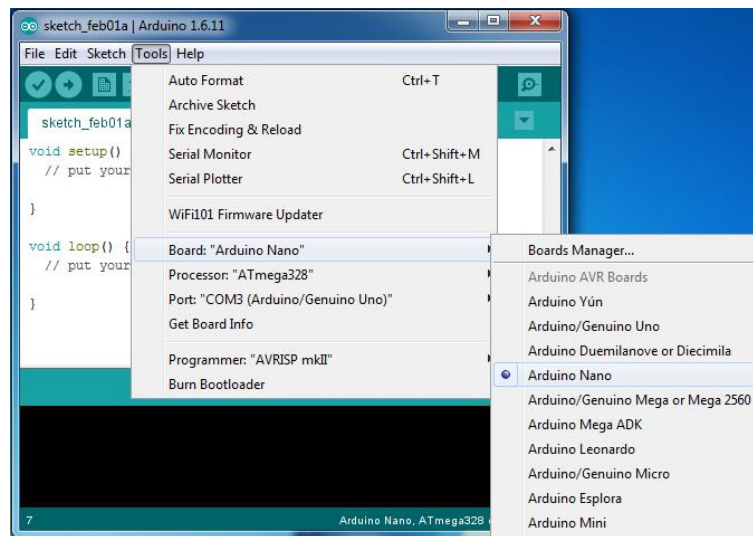


Gambar 2.1. Tampilan awal Arduino IDE

Pada tampilan diatas ada beberapa ikon yang akan sering digunakan ketika membuat program Arduino, diantaranya yaitu:

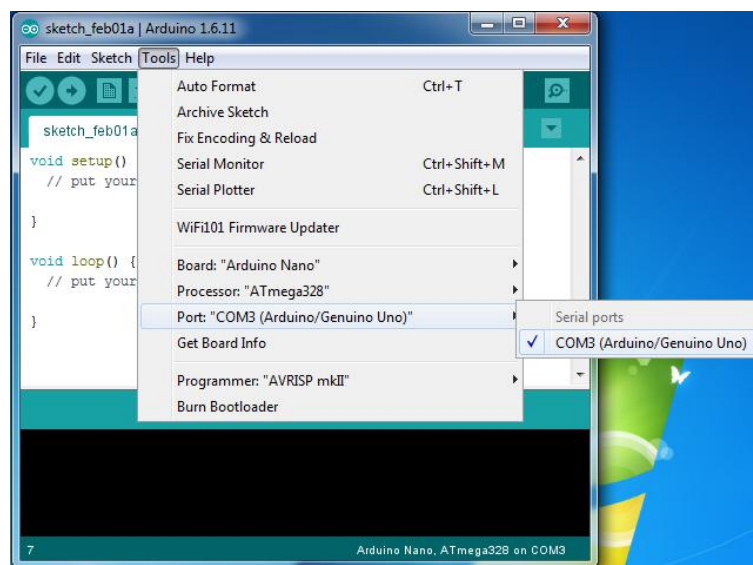
-  : Fungsi dari ikon disamping yaitu untuk compile sketch Arduino
-  : Fungsi dari ikon disamping yaitu untuk upload sketch ke Arduino
-  : Fungsi dari ikon disamping yaitu untuk membuat sketch baru
-  : Fungsi dari ikon disamping yaitu untuk membuka file sketch
-  : Fungsi dari ikon disamping yaitu untuk menyimpan file sketch

2. Lakukan pengaturan Boards pada sketch Arduino, sesuaikan dengan Arduino yang digunakan. Pilih **Tools** >> Klik **Board** >> Pilih Sub menu '**Arduino Nano**'.




Gambar 2.2. Tampilan menu pemilihan Board

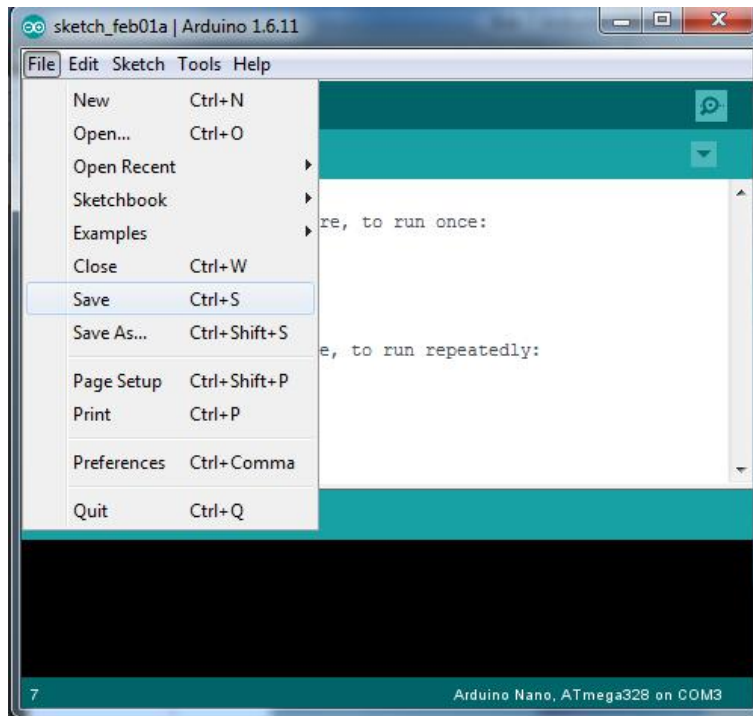
3. Kemudian atur Port yang digunakan pada Arduino, yaitu Port USB yang dipakai ke Arduino. Pilih **Tools** >> Klik **Port** >> **COMX**.



Gambar 2.3. Tampilan menu pemilihan Port

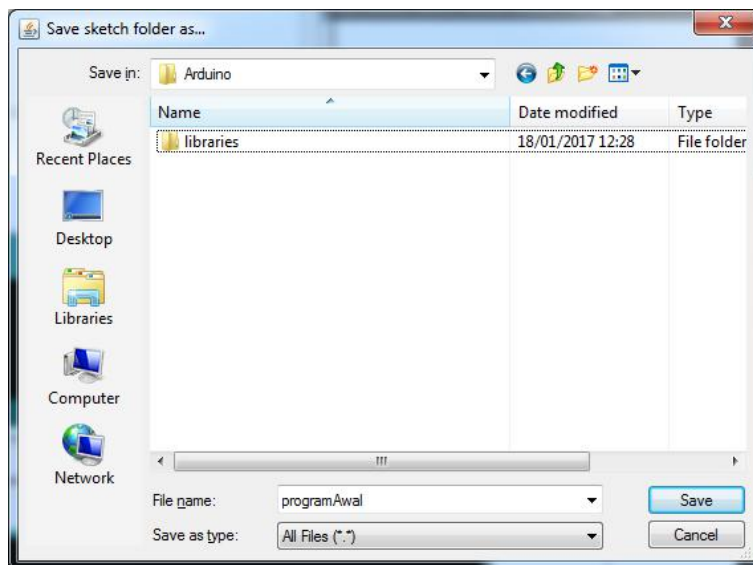
Pada port akan berbeda setiap komputer atau laptop, selain itu jenis Arduino juga berbeda.

4. Selanjutnya simpan sketch dengan menggunakan ikon  atau bisa juga dengan Pilih **File** >> Klik **Save**, selain itu dengan **Ctrl + S**.



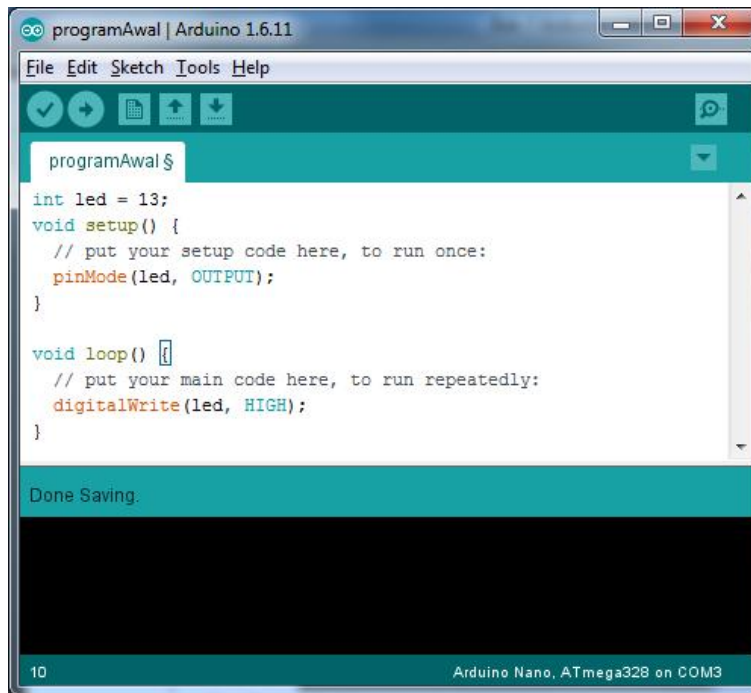
Gambar 2.4. Tampilan opsi Save

5. Setelah memilih Save maka akan muncul jendela Save, dan simpanlah pada drive yang disediakan sekaligus beri nama sketch tersebut. Seperti gambar berikut,



Gambar 2.5. Jendela Save file

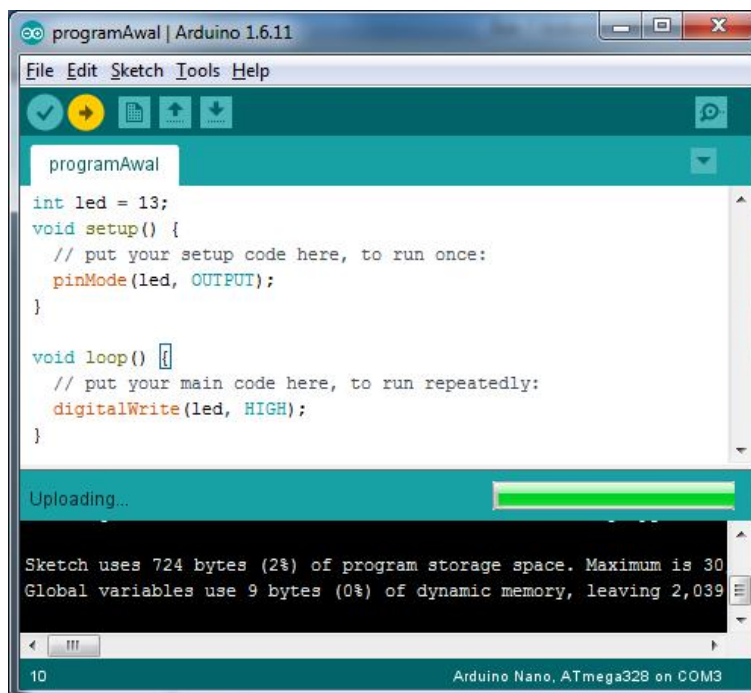
6. Setelah disimpan maka nama dari sketch akan berubah sesuai dengan nama sketch yang sudah disimpan. Lalu tuliskan kode dari program pada sketch tersebut, seperti gambar dibawah ini.



Gambar 2.6. Kode program dituliskan pada Sketch

7. Kemudian, **Compile** atau **Upload** program ke Arduino dengan memilih ikon

 untuk **Compile** dan  untuk **Upload** sketch ke Arduino.



Gambar 2.7. Proses kompilasi (compile) program Arduino

Jika terdapat error pastikan Arduino sesuai dengan jenis Board, selain itu syntax pada program juga sudah sesuai.

## PENAMBAHAN DAN PENGGUNAAN LIBRARY DAN BOARD MANAGER

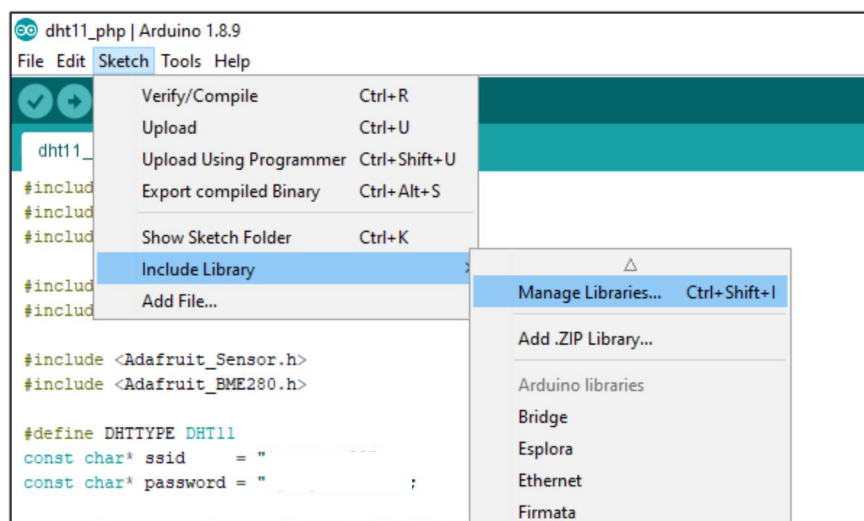
### TAMBAHAN

Arduino yang merupakan base board, pada penggunaannya akan banyak berinteraksi dengan modul-modul atau board lain, terutama modul atau board sensor. Beberapa sensor sudah langsung dapat dikenali dan diprogram secara default, akan tetapi sebagian sensor-sensor lainnya membutuhkan library tambahan supaya dapat diakses oleh board Arduino.

Apabila diketemukan sensor yang tidak dapat langsung digunakan maka kita bisa melakukan instalasi library tambahan, baik melalui library resmi atau library third party (pihak ketiga).

Langkah-langkah melakukan instalasi library tersebut adalah sebagai berikut.

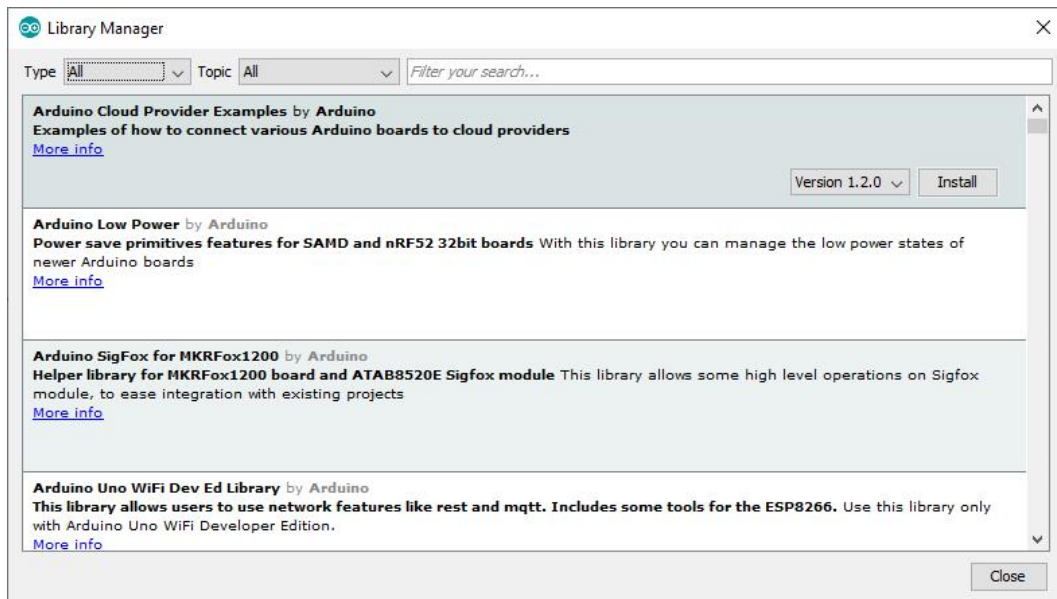
1. Semua proses selanjutnya membutuhkan koneksi internet, pastikan komputer dihubungkan dengan internet selama melakukan proses instalasi library.
2. Buka Arduino IDE kemudian akses opsi untuk manajemen library dengan salah satu cara sebagai berikut (pilih salah satu) :
  - a. pilih menu Sketch > Include Library > Manage Libraries,
  - b. pilih menu Tools > Manage Libraries,
  - c. atau dengan menggunakan shortcut Ctrl + Shift + I



Gambar 2.8. Mengakses Manage Libraries melalui menu Sketch

3. Arduino IDE akan melakukan pemutakhiran daftar library secara online.





Gambar 2.9. Tampilan Library Manager

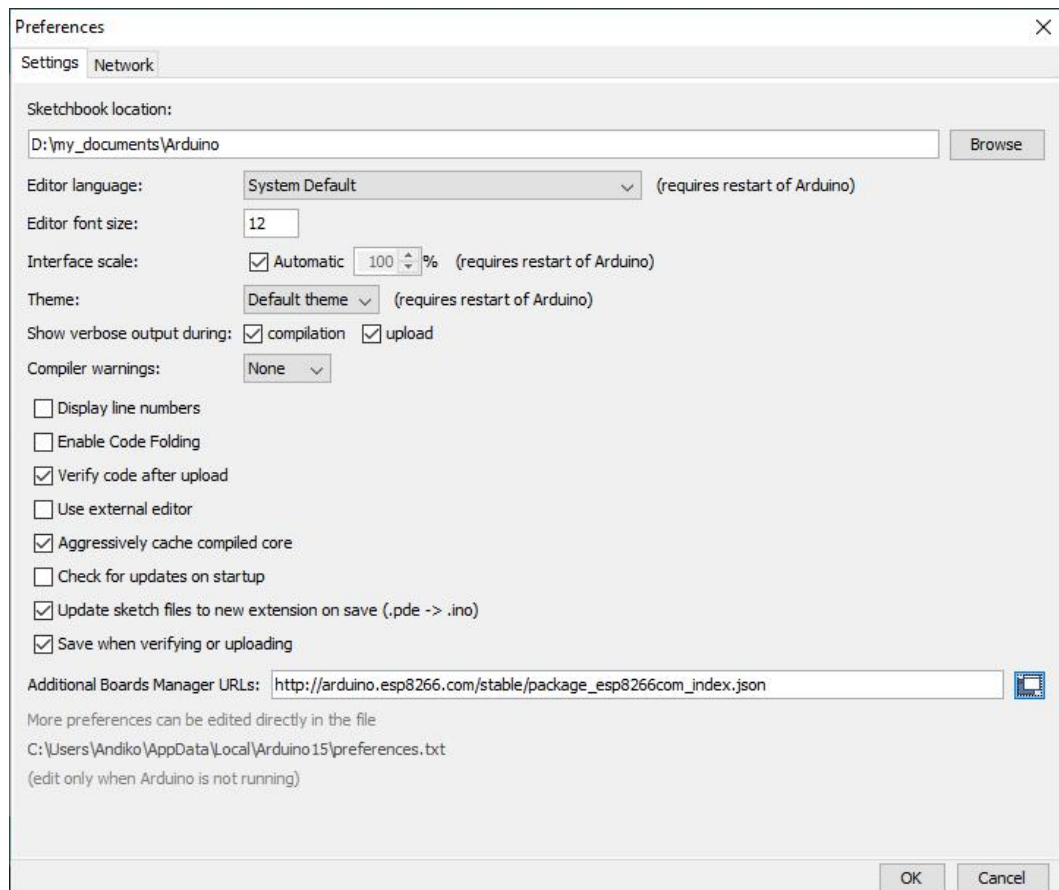
4. Setelah pemutakhiran selesai dilakukan maka akan muncul jendela Library Manger seperti terlihat pada gambar 2.2.
5. User dapat melakukan pencarian nama library yang diinginkan pada field “Filter your search” dan kemudian memilih library sesuai yang diinginkan untuk di-install. Arduino akan terlebih dahulu mengunduh berkas library yang akan di-install, pastikan komputer terhubung dengan internet seama proses instalasi berjalan.

Selain metode di atas, user juga bisa melakukan instalasi library dengan menggunakan berkas ZIP (yang didapat dari pihak ketiga) dengan cara memilih menu Sketch > Include Library > Add .ZIP Library, kemudian memilih berkas ZIP yang diinginkan.

Selain library, user juga bisa menambahkan board manager tambahan untuk menambahkan submenu Contributed Library pada menu Include Library. Seluruh library spesifik dari sebuah board yang bukan merupakan keluarga Arduino (tetapi masih merupakan keluarga AVR) akan tersedia dan dapat langsung di-install (dengan langkah install library tambahan di atas). Langkah-langkah menambahkan board manager adalah sebagai berikut.

1. Semua proses selanjutnya membutuhkan koneksi internet, pastikan komputer dihubungkan dengan internet selama melakukan proses instalasi board.
2. Pilih menu File > Preference,
3. Kemudian pada jendela yang muncul pilih tab Settings dan cari field isian dengan label “Additional Boards Manager URLs”,

4. Isikan URL dari board manager yang ingin di-install. User dapat melihat daftar board manager yang didukung oleh Arduino IDE melalui link berikut : <https://github.com/arduino/Arduino/wiki/Unofficial-list-of-3rd-party-boards-support-urls>



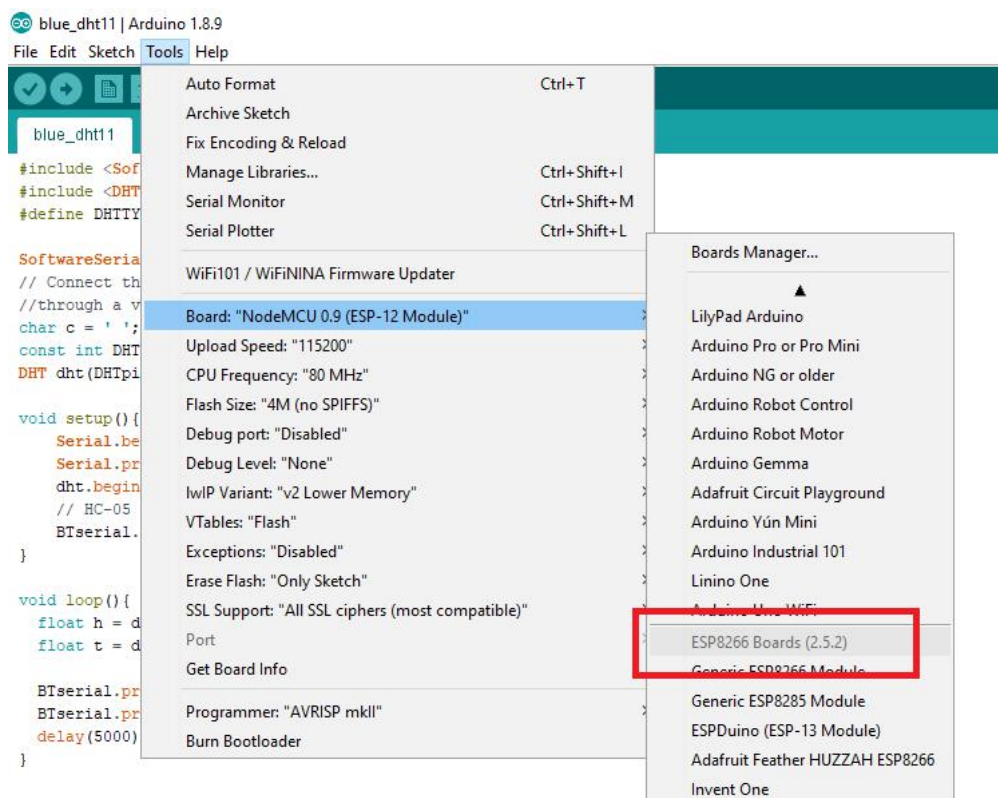
Gambar 2.10. Jendela Preferences untuk menambahkan Board Manager

5. Setelah menambahkan tautan/link di atas, langkah selanjutnya adalah melakukan instalasi dari board yang ditambahkan. Pilih menu Tools > Board > Boards Manager. Akan tampak jendela Boards Manager seperti gambar 2.11.



Gambar 2.11. Jendela Boards Manager

6. Lakukan instalasi board yang diinginkan dengan melakukan search pada field isian di bagian atas jendela. Tunggu hingga proses selesai dan lakukan pengecekan pada Tools > Board, boards baru akan muncul sebagai kelompok board terpisah seperti ditunjukkan pada gambar 2.12.



Gambar 2.12. Tampilan opsi board baru yang selesai di-install

7. Apabila akan melakukan proses compile dan upload pada board tertentu (termasuk ESP8266) untuk pertama kalinya kita harus melakukan penyesuaian setting board (Tools > Board). Untuk ESP8266 (NodeMCU) ada 2 macam jenis board yang tersedia yaitu “NodeMCU 0.9 (ESP-12 module)” atau “NodeMCU 1.0 (ESP-12E module)”. Jika menggunakan ESP-01 maka jenis board yang dipilih adalah “Generic ESP8266 module”