

# کتابخانه ها در آردوینو

تالیف

ایمان آسمانی

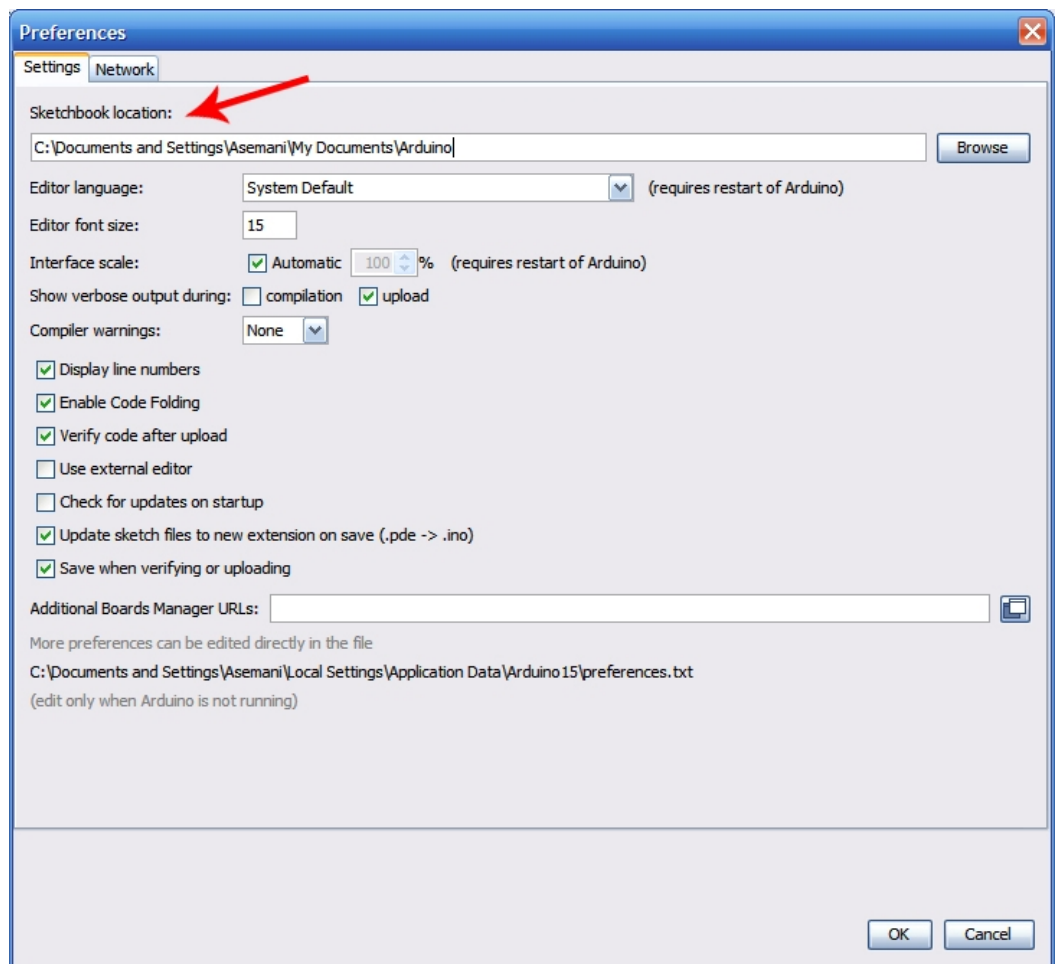
## طراحی کتابخانه برای آردوینو

شناخت و برنامه نویسی کتابخانه ها در آردوینو یکی از مهمترین بخش های کار با آردوینوست که هر علاقمندی به این شاخه میبایست به خوبی با آن آشنا باشد، البته جدای از شناخت کتابخانه، برنامه نویسی کتابخانه ها مربوط به سطح پیشرفته ی کار با آردوینوست. با این حال پیشنهاد میکنم کار با کتابخانه ها در آردوینو را به خوبی یاد بگیرید، این باعث میشود تا درک بهتری نسبت به ساختار کاری آردوینو چه در بخش سخت افزار و چه نرم افزار پیدا کنید. این آموزش را با مثالی پیش میبرم.

در این مثال قصد دارم یک کتابخانه با نام **Fade** تولید کنم. همانطور که از نامش پیداست برای ساختارهای محوسازی مورد استفاده قرار میگیرد. از آنجایی که کتابخانه های آردوینو با زبان **C++** نوشته میشوند، ما نیاز به ساخت دو فایل مجزا داریم، **Fader.h** و **Fader.cpp**. این دو شامل "سرآمد **Header**" و "بدنه **Body**" ی کلاس **Fader** میشوند. **Class** در واقع ساختار مادری یک قالب است که از آن میشود چندین شمایل خارج کرد. همچون قالب ساخت یک مجسمه که از آن مجسمه های فراوانی تولید میشود. که باعث میشود شما یک بار قالب را بسازید و ده ها بار از آن برای ساخت نمونه ها استفاده کنید. از آنجایی که محیط برنامه نویسی آردوینو بر پایه ی زبان **C++** است، دیگر نیازی به ابزار دیگری برای ساخت کتابخانه ندارید، اما من برای اینکه تمامی ورقه های کد همزمان در دسترس باشند و سرعت تولید کتابخانه بیشتر گردد و کارتان ساده تر گردد، نرم افزار همراه این مقاله را طراحی کرده ام و به صورت کاملاً رایگان و متن باز در دسترس قرار میدهم.

### • گام اول: تنظیمات

قبل از شروع به نوشتن کدها، باید پوشه ای را که آردوینو برای استفاده از کتابخانه هایش از آن استفاده میکند پیدا کنیم. شما میتوانید این مسیر را با توجه به بخش **Sketchbook location** در پنجره ی **Preferences** نرم افزار آردوینو، واقع در منوی **File** پیدا کنید.



ما باید پوشه ی کتابخانه های خود را به درون این پوشه کپی کنیم یا در مسیر اصلی نرم افزار،

**C:\Program Files\Arduino\libraries**

مثلاً:

پوشه ی کتابخانه باید با نام کلاس ما همخوانی داشته باشد و نباید از فاصله و کاراکترهای اضافی در آن استفاده کرد.

## • گام دوم: فایل سرآمد یا سرآغاز (Header.h)

اگر شما با زبان C++ آشنایی قبلی داشته باشید متوجه میشوید که فایل Header شامل خلاصه ای از محتوای هر کتابخانه است. فایل سرآمد یک برگه ی مهم در شناسایی هر کتابخانه توسط کامپایلر های مختلف است. هر زمان که شما مایل به استفاده از یک کتابخانه باشید، می بایست آن را با معرفی فایل سرآمد به پروژه ی خود وارد کنید. با این کار کامپایلر (مترجم زبان ما بین سخت افزار و نرم افزار) متوجه میشود که چه توابع، متغیر و ثبات هایی در کتابخانه ی مورد نظر وجود دارد.

پسوند فایل سرآمد ".h" است و نام فایل هم نام با کتابخانه ی آن است؛ و تقریباً هر فایل سرآمد شمایی شبیه به زیر دارد:

```
1 #ifndef Fader_h
2 #define Fader_h
3
4 #if ARDUINO >= 100
5   #include "Arduino.h"
6 #else
7   #include "WProgram.h"
8   #include "pins_arduino.h"
9   #include "WConstants.h"
10 #endif
11
12 // Your class header here...
13
14 #endif
```

البته این فقط بدنه ی اصلی و خام فایل Header میباشد. خطوط 1 و 2 و 14 موجب جلوگیری از بروز خطا در زمانی که کاربری دوبار فایل Header را فراخوانی کند می شود؛ که در زبان C++ این یک امر طبیعی ست. خطوط 4 تا 10 در زمانی که شما میخواهید از توابع و ثبات ها در کتابخانه ی خود استفاده کنید لازم و ضروریست، من توصیه میکنم همیشه آن را در ابتدای فایل Header قرار دهید.

توجه کنید که در خطوط 1 و 2 همیشه بعد از دو کلمه ی: `ifndef` , `define` نام کتابخانه با الحاق "\_h" در انتهای آن آورده میشود.

## کدنویسی سرآمد

در واقع فایل Header برای معرفی کامل کلاس هر کتابخانه مورد استفاده است، که در اینجا این کتابخانه به نام Fader می باشد. و مشخص میکند که چه متدها، صفات، توابع و متغیرهایی در کلاس مورد استفاده قرار میگیرد.

```
1 class Fader {
2   public:
3     Fader();
4
5     void init(float initial);
6     void fadeTo(float value, unsigned long duration);
7     float getFade();
8
9   private:
10     unsigned long _startTime;
11     float _startValue; // Start from this value
12
13     unsigned long _stopTime;
14     float _stopValue;  // Stop at this value
15
16     float lerp(float m1, float M1, float m2, float M2, float v1);
17 };
```

در این قسمت باید توجه کنید که دو متد اصلی با نام های Public و Private موجود است. خطوطی که زیر هر متد وجود دارد نشان دهنده ی محدوده ی عملیاتی هر کدام است. همانطور که از نام آنها نیز پیداست، هر دستوری که در بخش متد Public قرار میگیرد، توسط کاربر یا برنامه نویسان دیگر در محیط کدنویسی آردوینو قابل دسترس است، و بالعکس هرآنچه در بخش متد Private قرار میگیرد، تنها برای خود کلاس و کتابخانه قابل دسترسی و نمایش است.

همانطور که در تصویر بالا مشاهده میکنید، متد Public "عمومی"، دارای سه تابع و پروسیژر است: `init`, `fadeTo`, `getFade`. نکته ای که در بخش متد Private "خصوصی"، باید به آن اشاره کرد قرار گیری علامت "—" قبل از نام هر متغیر است، و این به شما یاد آوری میکند که این متغیر جزء متغیرهای متد Private است. بدین شکل شما در زمان برنامه نویسی کلاس تان کمتر دچار اشتباه میشوید.

## • گام سوم: بدنه کلاس (Fader.cpp)

فایل بدنه ي کتابخانه، شامل کدهاي اصلي ميشود، که با فراخواني فايل سرآمد: `#include "Fader.h"` آغاز ميشود. و شامل تمامي بخش هاي متدهايي که در فايل سرآمد از آن ياد شد ميشود. توجه داشته باشيد با قيد کردن نام هر تابع بدین شکل: `Fader::fadeTo` ما به کامپایلر مي‌فهمانيم که مي‌خواهيم بدنه ي تابع مثلاً "fadeTo" را کدنويسي کنيم، که در کلاس `Fader` واقع است. نترسيد!، کدنويسي ها همانند هرآنچه که تا کنون در کدنويسي آردوينو آموخته ايد است. فقط چند چيز کوچک به آن افزوده شده است.

```
1 #include "Fader.h"
2
3 Fader::Fader()
4 {
5 }
6
7 Fader::init(float initial)
8 {
9     _startTime = millis() - 1;
10    _startValue = initial - 1;
11
12    _stopTime = _startTime;
13    _stopValue = initial;
14 }
15
16 void Fader::fadeTo(float value, unsigned long duration)
17 {
18     _startTime = millis();
19     _startValue = getFade();
20
21     _stopTime = _startTime + duration;
22     _stopValue = value;
23 }
24
25 float Fader::getFade()
26 {
27     unsigned long currentTime = millis();
28     float currentValue = lerp(_startTime, _startValue, _stopTime, _stopValue, currentTime);
29     currentValue = constrain(currentValue, _startValue, _stopValue);
30     return currentValue;
31 }
32
33 // Linear interpolation
34 float Fader::lerp (float m1, float M1, float m2, float M2, float v1)
35 {
36     float d = M1 - m1;
37     float c = (M2 - m2) / d;
38     float o = ( -(M2 * m1) + (m2 * m1) + m2 * d) / d;
39     return v1 * c + o;
40 }
```

بايد توجه داشت که فايل بدنه داراي پسوند "cpp" است که نشان دهنده ي فايل سورس کد زبان C++ است. و هميشه با نام کتابخانه همراه است. و تنها تفاوتش با نام فايل سرآمد، در پسوندشان است.

## • گام چهارم: کلمات کلیدی (Keywords.txt)

يکي از نقص هاي مشهود نرم افزار آردوينو، عدم توانايي در تشخيص کلمات کليدي در هر کد است (به جهت رنگين کردن و متمايز کردن آنها از ديگر کلمات)، و شما با افزودن فايل "Keywords.txt" و کدنويسي آن، به نرم افزار آردوينو در شناسايي اين کلمات کمک بسزايي ميکنيد. البته بايد گفت که افزودن اين فايل کاملاً اختياريست، اما در سامان دهی کردن کتابخانه ي شما، و تشخيص کلمات کليدي هر کتابخانه به برنامه نويس کمک شاياني ميکند، و از بروز خطاهاي تاثيري در نامري هر کلمه کليدي جلوگیری ميکند.



```

1 #####
2 # Syntax Coloring Map For Fader
3 #####
4
5 #####
6 # Datatypes (KEYWORD1)
7 #####
8
9 Fader    KEYWORD1
10
11 #####
12 # Methods and Functions (KEYWORD2)
13 #####
14
15 init     KEYWORD2
16 fadeTo   KEYWORD2
17 getFade  KEYWORD2
18
19 #####
20 # Constants (LITERAL1)
21 #####

```

این فایل شامل معرفی نام هر تابع عمومی و هر متغیر عمومی و همچنین تئبات های عمومیست.

باید توجه داشت که فاصله ای که مابین کلمات قرار گرفته است مثل:

Fader      KEYWORD1

با استفاده از کلید [Tab] موجود بر روی کیبورد ایجاد شده است و نه کلید [Space] !!!

البته بخش هایی که در تصویر روبرو مشاهده میکند که با رنگ نارنجی تیره نمایش داده شده است، تنها بخش اطلاعاتیست و تاثیری در ساختار عملیاتی این فایل ندارد. و استفاده از آنها کاملاً اختیاریست. دقت کنید که پسوند این فایل "txt" میباشد.

### • گام پنجم: استفاده از کتابخانه و تولید فایل مثال

گام نهمی استفاده از کتابخانه و فراخوانی آن در نرم افزار آردوینوست. توصیه میکنم زمانی که کتابخانه ای را تولید میکنید، حتماً یک یا چند مثال برای نحوه ی استفاده از تمام اجزاء کتابخانه تولید کنید و در کنار کتابخانه ی خود قرار دهید تا زمانی که آن کتابخانه را به شخصی بدهید بدانند که چگونه باید به توابع عملیاتی آن دسترسی داشته باشد و چگونه از آن استفاده کند. شما باید برای هر جنبه از کتابخانه ی خود یک مثال جدا گانه درست کنید و در آن بخش های کد را توضیحات نویسی کنید (در تصویر با خطوط نارنجی مشخص شده است) تا برای برنامه نویسان دیگر قابل فهم باشد.

```

1 #include "Fader.h"
2 Fader fader = Fader();
3
4 int led = 9;           // the pin that the LED is attached to
5 int fadeDuration = 1000; // 1 second
6
7 // the setup routine runs once when you press reset:
8 void setup() {
9     // declare pin 9 to be an output:
10    pinMode(led, OUTPUT);
11    fader.init(255);
12 }
13
14 // the loop routine runs over and over again forever:
15 void loop() {
16     // Get the current fade
17    int fade = (int) fader.getFade();
18    analogWrite(led, fade);
19
20    // Restarts the fade, if necessary
21    if (fade == 0)
22        fader.fadeTo(255, fadeDuration);
23    if (fade == 255)
24        fader.fadeTo(0, fadeDuration);
25 }

```

فایل های مثال با پسوند "ino". در پوشه ای به نام examples، در درون پوشه ی کتابخانه ی شما قرار میگیرند.

همانطور که بیان شد، نام پوشه ی کتابخانه همنام با خود کتابخانه میباشد.

### • گام ششم: نصب کتابخانه

حال که تمامی فایل ها تولید شدند و

در پوشه ی مادر قرار گرفتند، شما برای استفاده از کتابخانه می بایست آن را به پوشه ی libraries اضافه کنید (کپی کنید). و یا، میتوانید پوشه ی کتابخانه را با نرم افزار Winzip به صورت پیش فرض با پسوند ".zip" ساختار بندی کنید و از طریق خود نرم افزار آردوینو، منوی Sketch و زیر منوی import library یا include library به بخش کتابخانه های آردوینو اضافه کنید.

