

楽しく学ぶプログラミング

ピーケー

PKジュニアで3ステップ学習

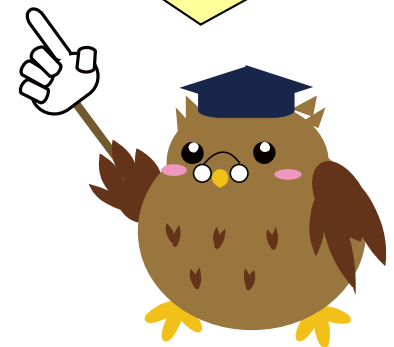
いろいろな体験→こんなことしてみたい→できた！

```
① while(1){           //以下、くりかえし
② led(on,on,on,on);  // 4つのLED点灯
③ delay_ms(500);     // 0.5秒そのまま
④ led(off,off,off,off); // 4つのLED消灯
⑤ delay_ms(500);     // 0.5秒そのまま
⑥ }                  //くりかえし、ここまで
```

言語編は、必要に応じて
辞書のように使おう

言語編

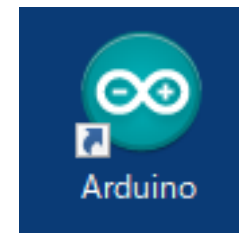
//(コメント記号)から左がプログラムで、
右はその説明



1 PK-ジュニアの言語仕様

基本仕様

- 基本的な仕様は、**Arduino**（アルドゥイーノ）によります。
- 詳細は、「**日本語リファレンス**」参照（ネット接続必要）



拡張仕様

- 入出力命令をPK-ジュニア専用**に設定しています。
- 例として、「0番の^{エルイーディー}**LEDを点灯**」を2通りで書くと次のようになります。

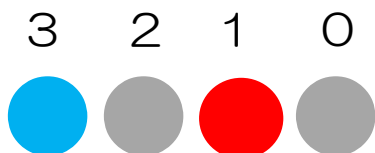
区分	命令文	説明
Arduino	デジタルライト ロー digitalWrite(2,LOW);	2は、LEDの0が接続されているピン番号 LOWは、そのピンの電圧を0Vにすること この結果、回路のしくみでLEDが点灯
PK-ジュニア	led0(on);	Offの場合は、led0(off);

同じ動作をさせるにも、**PK-ジュニアの方がわかりやすい**と言えます。もちろんArduino仕様で記述しても動作します。

2

LEDの点滅

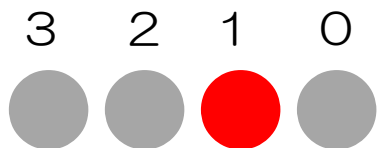
●4つのLEDをひとつおきに点滅指示する例



```
led(on,off,on,off);
```

命令文の最後には、
;(セミコロン)をつける
約束

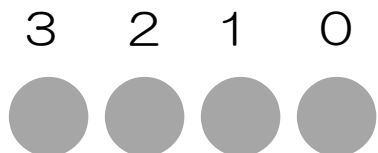
●1番のLEDを点灯する例（他のLEDの点滅は不定）



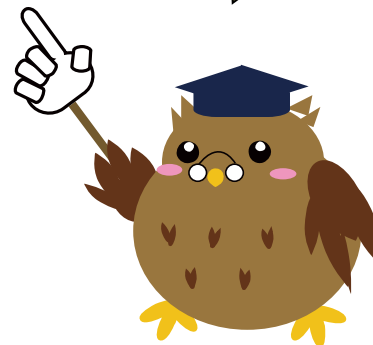
```
led1(on);
```

4つのLEDを同時に点滅
指示する方法と個別に
指示する方法があるんだ

●1番のLEDを消灯する例（他のLEDの点滅は不定）



```
led1(off);
```



24

キーワード(予約語)

- キーワードは、マイコンシステムが使用している文字列なので、自作のプログラムの名前や変数には使わないようにします。
- PK-ジュニアのおもなキーワードとして、次のようなものがあります。

分類	一例
型宣言	byte、int、long、float
各種文	while、if、for、switch、case、break、return、void
時間	delay、sec、ms、us、millis
LED	led、led0、led1、led2、led3、b_led、y_led、r_led、g_led
ポート	pa、pb、pc、pd、pa0～pa3、pb0～pb3、pc0～pc3、pd0～pd3
ブザー等	pi、bz、melody、relay
端子の状態	HIGH、LOW、on、off
スイッチ	sw0、sw1、sw2、sw3、sw4、sw5、sw6、sw7
センサー	distance、temp、photo
音階	dl、re、mi、fa、so、ra、si、dh、ya

このようなキーワードは、マイコンシステムによって異なります。

25 コメント

- プログラムの内容を説明するための文章を**コメント**（注釈）と言います。
- コメントは、マイコンのプログラムとは無関係です。**
- 複数行のコメント**には、**/* */**を使います。

```
/* エンジョイ プログラミング  
   enjoy_programming           // プログラム名、enjoy : 楽しむ  
*/
```

- //**は、これから**右側をコメント**扱いしたい場合に使用します。

```
led(on,on,on,on); // すべてのLEDを点灯
```

```
//melody(on); // メロディーを鳴らす
```

このプログラムは実行されません

26

#include(インクルード、取り込み)

エルシーディー

- LCDなどの高機能部品を動かすには、数十行のプログラムが必要です。
- 「Hello」と表示するのに、都度、数十行のプログラムを書くのは面倒です。
- この面倒を解消するのが#includeです。

```
ワイヤー  
#include <Wire.h>  
アイツシー リクウィッドクリスタル  
#include <I2CLiquideCrystal.h>
```

LiquideCrystalは液晶という意味

- 上記で、PK-ジュニアで使用しているLCDを使うためのWire.hとI2CLiquideCrystal.hの2つのプログラムファイルが取り込まれます。
- これによって、「Hello」表示が1行の命令文で済みます。

```
lcd_out_msg(0,0," Hello "); // 左端基準、上の行にHelloを表示
```

表示文は、空白を含む半角英数字8文字

27

#define(ディファイン、定義・置換)

- #defineは、プログラム中の指定した文字列を別の文字列に置換したい場合に使います。
- 例えば、下記によって、プログラム中の文字列のonが、コンパイル（作成したプログラムをマイコンに書き込むプログラムに変換すること）時に、LOWに置き換えられます。

```
#define on LOW // プログラム中のonをLOWに置換
```

- LEDは、端子をLOW（低電圧→0V）にすると点灯する回路になっているので、例えば、1番のLEDを点灯させるには、本来ledO(LOW);と書きます。しかし、ledO(LOW);では、回路のことがわかっていないと、点灯か消灯かがわかりません。ledO(on);と書くと、点灯とわかります。

ledO(on);

onがLOWに置き換えられる

ledO(LOW);

- PK-ジュニアがベースにしているArduinoのプログラムでは、初期設定として`setup()`を使う約束です。
- 初期設定以外に、最初の1回だけ実行させたい動作もここに書きます。

```
void setup( ) { // 初期設定
  エルシーディー lcd_start( ); スタート // LCDの使用宣言
  led(off,off,off,off); // LEDすべて消灯
} // 初期設定ここまで
```

- 初期設定がない場合でも、次のように書いておきます。

```
void setup( ) { // 初期設定
} // 初期設定ここまで
```

- プログラムのどの位置に書くかは、[30] プログラムの基本構成 を参照

29 loop() (ループ、くりかえし)

- 一般的に、プログラムは、あることをし続けるように書きます。
- PK-ジュニアがベースにしているArduinoのプログラムでは、これをloop()で実現します。

```
void loop(){ // 以下、くりかえし
  lcd_out_msg(0,0," Hello "); // LCDにHello表示
} // くりかえし、ここまで
```

- くりかえしとして、while(1){ };がありますが、loop()の代替えとして使うことはできません（約束事）。loop()の中で使うことは構いません。

```
void loop(){ // 以下、くりかえし
  while(1){ // 以下、くりかえし
    lcd_out_msg(0,0," Hello "); // LCDにHello表示
  } // while(1){ ここまで
} // loop(){ ここまで
```

- プログラムのどの位置に書くかは、[30] プログラムの基本構成 を参照

- プログラムの基本的な構成例を次に示します（No.4とNo.5は必須）。

No	プログラム例	説明
1	<pre>/* enjoy_programming */</pre>	/*と*/ の間は コメント扱い (実行されない) タイトルや日付、作成者名など
2	<pre>#include <Wire.h> #include <I2CLiquidCrystal.h></pre>	I2C(アイ ツー シー)ライブラリ の読み込み I2C駆動LCDライブラリ の読み込み
3	<pre>#define LCD_ADRS 0x3E</pre>	プログラム中のLCD_ADRSを0x3Eに 置換
4	<pre>void setup(){ lcd_start(); }</pre>	初期設定 LCDの使用宣言
5	<pre>void loop(){ lcd_out_msg(0,0,"Hello! "); }</pre>	くりかえし(メインプログラム) Hello!を表示
6	<pre>lcd_start(){ lcd.begin(8,2); lcd.clear(); }</pre>	LCD使用宣言 使用しているLCDは8文字×2行のもの LCDの表示をクリア
7	<pre>void lcd_out_msg(char x,char y,msg[]){ lcd.setCursor(x,y); lcd.print(msg); }</pre>	LCDにメッセージを表示 表示位置を指定 メッセージを表示