

## IchigoJam BASICのコマンド解説(初級編)

2015/10/05 第1.1版  
ファームウェアVer. 1.0.2対応

IchigoJamにはたくさんのコマンドがある。下の一覧は初めての人でもすぐ使えるものばかり。

これだけあれば立派なゲームを組み立てることができるよ！

まずは色んなゲームのプログラムを読んで、コマンドの意味をひとつひとつ確かめたり、自分でも書いて、動かして、試してみよう。

No	分類	コマンド	読み方	意味	使い方
1	プログラミングする時に使う	CLS	クリアスクリーン、シーエルエス	画面の文字を全部消す。 書いたプログラムは覚えているので、あとでLISTで呼び出せる。	CLS
2		NEW	ニュー	書いたプログラムを全部消す(SAVEしたプログラムは消えない)。 新しいプログラムを書き始めるときや、書いていたプログラムが気に入らなくなったときに・・・ 消したプログラムは二度と戻らないので注意！	NEW
3		LIST	リスト	書いたプログラムを画面に表示しなおす。 [LIST 行番号 1,行番号 2]で行番号 1 から 2 までのプログラムを表示することもできる。	LIST 10, 100
4		RUN	ラン	書いたプログラムを動かす。一番ワクワクするコマンド。 プログラムを止めたい時はキーボードの「esc」キーで！	RUN
5		SAVE	セーブ	書いたプログラムを別の場所に記録する。 「SAVE 0」で 0 番目の場所に記録。場所は 0 から 3 まで。 場所を決めずに「SAVE」とだけ書いた場合は、1 回もLOADしていなければ 0 番目、前にLOADしていればLOADした場所に記録。 NEWしたばかりでプログラムがない時にSAVEをすると、SAVEしてあったプログラムを消すこともできるよ。	SAVE 1
6		LOAD	ロード	別の場所に記録したプログラムを呼び出す。 「LOAD 0」で 0 番目の場所に記録したプログラムを呼び出す。 「LOAD」とだけ書いた場合は、1 回もLOADしていなければ 0 番目、前にLOADしていればLOADした場所のプログラムをもう一度呼び出す。 注意！まだSAVEしていない書きかけのプログラムがあるところに別のプログラムをLOADすると、書きかけのプログラムは消えてしまう。	LOAD 1
7		FILES	ファイルズ	これまでに記録したプログラムの一覧を表示する。 どこに何を記録したっけ？という時に・・・ プログラムを記録した場所とプログラムの1行目を表示するので、プログラムの1行目には「REM xxxのプログラム」と分かりやすくコメントを書いておくと便利。	FILES
8		行番号	ぎょうばんごう	いらなくなった行の行番号を指定すると、その行のプログラムを消す。 画面に表示していた文字は消えないけど、だいじょうぶ。「CLS」と「LIST」で画面を表示しなおしてみれば、ちゃんと行が消えているのが分かるよ。	20
9	行番号 コマンド	ぎょうばんごう	行の先頭に行番号をつけることで、その行はプログラムになる。 RUNで実行すると、行番号が小さい行から順番にプログラムが進んでいく。	10 PRINT "HELLO!"	
10	REM	リマーク、レム	「REM」と書くと、その先から行の終わりまでがプログラムとして実行されなくなる。プログラムの中に説明(コメント)を書いておきたい時や、今は動かしたくないプログラムがある時に使うと便利。 「REM」の代わりに「!」と書いてもいい。	REM START ' START	
11	プログラムの流れを作る:制御(せいぎょ)する	LET	レット	「LET 変数, 数」で変数(数を入れておくハコ)に数を入れる。変数にはアルファベットの1文字が使える。数は-32768~32767まで使える。 「変数=数」と書いてもいい。	LET A, 10 A=10
12	:	コロン	コマンドとコマンドをつなぐ。 つないだコマンドは、同じ行の中で続けて動く。	WAIT 60:LED 1	
13	GOTO	ゴートウ	「GOTO 行番号」で次にその行番号へジャンプする。	GOTO 10	
14	WAIT	ウェイト	「WAIT 数」で、指定した数の分プログラムの動作を待たせる。数は60でだいたい1秒くらい。1から32767まで指定できる。	WAIT 120	

IchigoJam BASICのコマンド解説(初級編)

No	分類	コマンド	読み方	意味	使い方
15	プログラムの流れを作る:制御(せいぎよ)する(続き)	IF THEN ELSE	イフ ゼン エルス	プログラムに枝分かれの流れを作る。 「IF 条件式 コマンド」で条件式の結果が0でなければ(つまり、条件に合っていれば)その後のコマンドを動かす。 「IF 条件式 THEN コマンド」という書き方でもOK。 また、「IF 条件式 THEN コマンド1 ELSE コマンド2」と書くと、条件式が0以外を返す時はコマンド1、0を返す時はコマンド2を動かすことができる。 条件式については次ページの「条件式を作る」にくわしく書いてあるよ。	IF A=1 LED 1 IF A=1 THEN LED 1 IF A=1 THEN LED 1 ELSE LED 0
16		FOR TO STEP NEXT	フォア トゥ ステップ ネクスト	プログラムにくりかえしの流れを作る。 「FOR 変数=数1 TO 数2 STEP 数3」で、まず最初に変数(数を入れておくハコ)に数1を入れ、NEXTまでのプログラムを動かす。NEXTでFORに戻る。この時に変数に数3を足して、数2より小さければまたNEXTまでのプログラムを動かす。変数が数2になったらくりかえしをやめる。 数3=1の時は、「STEP 数3」の部分を書かなくてもいい。	FOR A=1 TO 10 STEP 1 PRINT A NEXT
17		END	エンド	プログラムを終わらせる。	END
18	プログラムから信号を出す:出力する	LOCATE LC	ロケート	「LOCATE 数1, 数2」で画面でPRINTする場所(ヨコ、タテ)を決める。数1がヨコで、画面の一番左はじが0、右はじが35。数2はタテで画面の一番上が0、下が26。 「LOCATE」の代わりに「LC」と書いてもいい。	LOCATE 17, 13 LC 17, 13
19		PRINT ?	プリント	画面に文字や数字を表示する。文字の場合は「"」で文字の両側を囲む。「;」(セミコロン)で文字をつないで表示することもできる。 「PRINT」の代わりに、「?」と書いてもいい。	PRINT "HELLO!" PRINT "TIME: ";A ? 1000
20		SCROLL	スクロール	「SCROLL 方向」で画面を上下左右に1文字分ずらす(スクロールする)。方向は上が0、右が1、下が2、左が3。またはUP、RIGHT、DOWN、LEFTと書いてもいい。	SCROLL 2 SCROLL DOWN
21		BEEP	ビーブ	IchigoJamに取り付けた圧電(あつでん)ブザーから音を鳴らす。「BEEP 数1 数2」と書くと、数1で音の高さ(周期)、数2で音の長さを決めることができる。数1は1から255までで、小さい方が高い音になる。数2は60でだいたい1秒くらい。	BEEP BEEP 100, 180
22		LED	エルイーディー	IchigoJamにのっているLEDを、「LED 1」で光らせることができる。消すときは「LED 0」。	LED 1
23		CLS	クリアスクリーン、シーエルエス	「プログラミングする時に使う」の時に出てきたCLSと同じ。プログラムの中でも画面を書き直すときに使える。	CLS
24	外から信号をもらう:入力する	INKEY()	インキー	キーボードのキーが押されていれば、1文字分の文字コードをもらう。何も押されていない時は0をもらう。 文字コードの値については、別紙の文字コード表を見てね。	PRINT INKEY() IF INKEY()=28 GOTO 10
25		BTN(数)	ボタン	ボタンが押されていれば1、押されていない場合は0をもらう。 「BTN(数)」の数はどのボタンかを定めるもので、0(IchigoJamにのっているボタン)、UP/DOWN/RIGHT/LEFT/SPACE(キーボードの↑↓←→キーとスペースキー)のどれかを書く。 「BTN()」と書くと、「BTN(0)」と書いた時と同じ動きをする。	IF BTN(SPACE) LED 1 LED BTN(0)
26	計算する	INPUT	インプット	「INPUT 変数」と書くと、画面に「?」と表示されてキーボードのキーが押されるのを待つ。キーボードから数字のキーとENTERキーを押すと、受け取った数字を変数のハコに入れる。最初に押したキーが数字じゃなかったり、ただENTERキーを押しただけだと変数には0が入るよ。 「INPUT "文字", 変数」と書くと「?」の代わりに文字を表示する。	INPUT A INPUT "ANS?", A
27		+	たす	「数1+数2」で足し算した答えを返す。	PRINT 2+1
28		-	ひく	「数1-数2」で引き算した答えが返ってくる。	PRINT 2-1
29		*	かける	「数1*数2」でかけ算した答えを返す。「*」は算数の「×」といっしょ。	PRINT 2*1
30		/	わる	「数1/数2」でわり算した答えを返す。「/」は算数の「÷」といっしょ。	PRINT 2/1
31		%	剰余(じょうよ)	「数1%数2」でわり算のあまりを出す。「5%2」は「5÷2=2あまり1」だから、「1」を返すよ。	PRINT 2%1
32		()	かっこ	()で囲まれた式の方が先に計算される。例えば「2*(1+3)」は、1+3を先に計算した答えに2をかけるから、「8」を返す。「(2*1)+3」だと「5」を返す。	PRINT 2*(1+3)

IchigoJam BASICのコマンド解説(初級編)

No	分類	コマンド	読み方	意味	使い方
33		数1=数2 数1 == 数2	一致(いっち)、 イコール	数1と数2を比べて、同じであれば1を返す。 それ以外は0を返す。	IF A=B LED 1 IF A==B LED 1
34		数1!=数2 数1<>数2	不一致(ふいっち)、 ノットイコール	数1と数2を比べて、同じでなければ1を返す。 それ以外は0を返す。	IF A<>B LED 1 IF A!=B LED 1
35		数1<数2	小なり(しょうなり)	数1と数2を比べて、数1が数2より小さければ1を返す。 それ以外は0を返す。	IF A<B LED 1
36		数1<=数2	以下、 小なりイコール	数1と数2を比べて、数1が数2と同じか数2より小さければ1を返す。 それ以外は0を返す。	IF A<=B LED 1
37	条件式(じょうけんしき)を作る	数1>数2	大なり(だいなり)	数1と数2を比べて、数1が数2より大きければ1を返す。 それ以外は0を返す。	IF A>B LED 1
38		数1>=数2	以上、 大なりイコール	数1と数2を比べて、数1が数2と同じか数2より大きければ1を返す。 それ以外は0を返す。	IF A>=B LED 1
39		条件式1 AND 条件式2 条件式1 & 条件式2	アンド	条件式1も条件式2も 両方とも 1を返すときに1を返す。 それ以外は0を返す。	IF A=1 AND B=1 LED 1
40		条件式1 OR 条件式2 条件式1   条件式2	オア	条件式1と条件式2のどちらかが1を返すときに1を返す。 それ以外は0を返す。	IF A=1 OR B=1 LED 1
41		NOT 条件式 ! 条件式	ノット	条件式が0を返すときに1を返す。 それ以外は0を返す。	IF NOT A=1 LED 1 IF !A=1 LED 1
42	その他、便利なコマンドたち	RND(数)	ランダム	0から数-1までの数字をランダムに返す。「RND(6)」と書いたら、0から5までの数字のうちどれかが返ってくる。サイコロのようなもの。	FOR A=1 TO 10 PRINT RND(5) NEXT
43		TICK()	ティック	IchigoJamをONにした時、またはCLTで時間をリセットした時からどれだけ時間がたったかを数字で返す。数は60で大体1秒くらい。32767まで数えたら、また0に戻って数え始める。	PRINT TICK()
44		CLT	クリアティック	時間のカウントを0にリセットする。	CLT
45		CHR\$(数)	キャラ	数で示した文字コードにあたる文字を返す。文字コードを「,」で区切ると続けて書くこともできる。 文字コードの値については、別紙の文字コード表を見てね。	PRINT CHR\$(65) PRINT CHR\$(65, 66, 67)
46		ASC("文字")	アスキー	"文字"にあたる文字コード(数)を返す。 文字コードの値については、別紙の文字コード表を見てね。	PRINT ASC("A")
47		VPEEK(数1, 数2)	バイピーク	「VPEEK(数1, 数2)」で画面のある場所に表示されている文字の文字コードを返す。数1がヨコで、画面の一番左はじが0、右はじが35。数2はタテで画面の一番上が0、下が26。 ( )の中に数を書かなければ、今いる場所の文字コードを返す。	VPEEK(3, 10) VPEEK()
48		SCR(数1, 数2)	スクリーン	VPEEKと同じ。	SCR(3, 10) SCR()

※この解説は 初級編として書かれているため、すべてのコマンドあるいはコマンド内の機能を解説するものではありません。

※この解説は、以下にCC BYで公開されているリファレンスを元に、より子どもに読みやすい形を目指して書かれています。

BASICリファレンス: CC BY 福野泰介 jig.jp (<http://ichigojam.net/IchigoJam-BASIC-reference.pdf>)

イチゴジャム レシピ: CC BY 志賀 慶一 (<http://15jamrecipe.jimdo.com/>)

※また、本書は CC BY として 公開します。 CC BY ライセンスに基づいて複製/改変/再配布可能です。(著者表示は必須)

CC BY PCN 金沢 (<http://kanazawa.pcn.club/>)