

迷わないプログラミングの つくりかた

——動き出すための10ページ

鷺野
文哉

お試し版

Made with GAMMA

まず何かを動かす

プログラミングを始めるとき、完璧な理解は必要ありません。最初に大切なのは「動いた」という事実です。コードが何をしているか全部わからなくても、画面に何か表示されたり、ボタンを押したら反応があったりすれば、それで十分なスタートです。

完璧でなくていい理由は明確です。プログラミングは、書いてみないと何が起きるかわからないことが多いからです。頭の中で考えているだけでは、実際の動きは見てきません。まずは動かして、その結果を見るのが次の一歩につながります。

最初の目標

何か一つを画面に表示させる

確認すること

エラーが出ても止まらない

次のステップ

動いた結果を見て考える

理解は後からついてきます。動かすことで、「こういうことか」という納得が生まれます。完璧を目指すより、まず一つ動かすことを優先してください。そこから学びが始まります。

作りたい気持ちは正しい

従来の学び方

- 基礎から順番に学ぶ
- 理論を理解してから実践
- 全体を把握してから作る

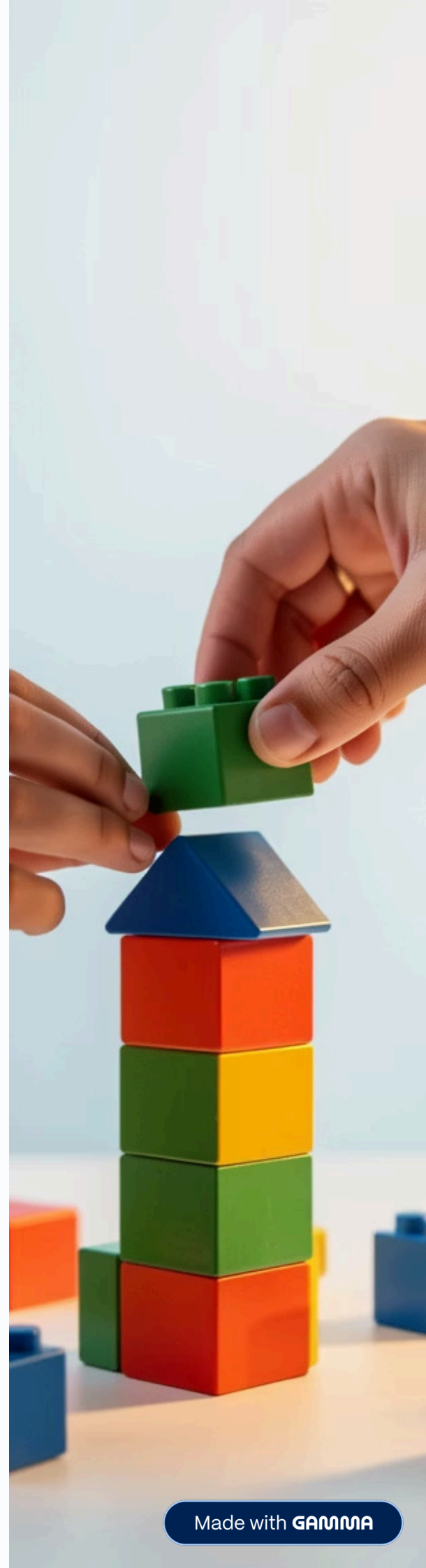
作りながら学ぶ

- 必要なことから学ぶ
- 実践しながら理解する
- 作りながら全体が見える

「作りたい」と思ったその気持ちは、プログラミングを続けるための一番大切なエネルギーです。教科書を最初から順番に読まなくても、基礎を完璧に理解していなくても、作りたいものがあればそれで十分です。

分からないまま進んでよい理由があります。プログラミングは、作りながら必要な知識を調べて、試して、動かして、という繰り返しで身につくものだからです。最初から全部知っている人はいません。誰もが作りながら学んできました。

あなたが「これを作りたい」と思ったなら、その気持ちを信じてください。わからない部分があっても、それは後で調べればいいことです。大切なのは、作りたい気持ちを持ち続けることです。



行動の前に一つだけ決める

プログラミングを始めるとき、全部を最初に決める必要はありません。今日やることを一つだけ決めれば、それで十分です。全体の設計、完成形、必要な機能、すべてを考えてから始めようとする、始める前に疲れてしまいます。

01

今日の一つを決める

今日動かしたいことを一つだけ選ぶ

02

それだけを作る

決めた一つに集中して作業する

03

明日のことは明日決める

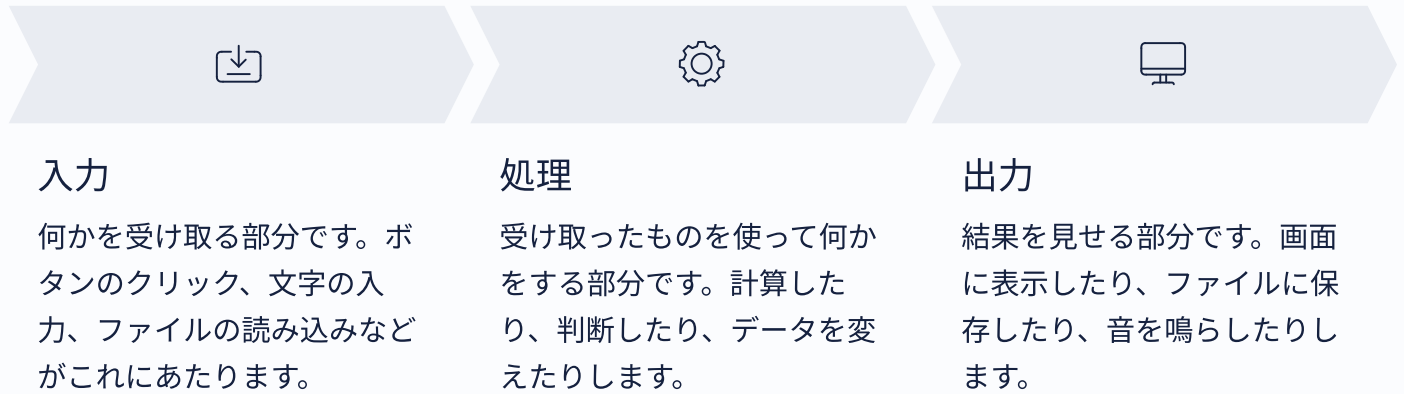
次のステップは作ってから考える

全部決めないメリットは明確です。作っている途中で「こうした方がいいかも」と気づくことがたくさんあるからです。最初に全部決めてしまうと、その気づきを活かせなくなります。柔軟に変えられる余地を残しておくことが、良いものを作るコツです。

今日の一つが決まったら、それだけに集中してください。他のことは後で考えればいいのです。軽くスタートすることで、続けやすくなります。

最小構成で考える

プログラミングで何かを作るとき、どんなに複雑なものでも、基本は同じ構造でできています。それが「入力」「処理」「出力」という三つの要素です。この考え方を使えば、作りたいものを整理しやすくなります。



最初は雑でよい理由があります。細かいところまで作り込もうとすると、時間がかかりすぎて、動くものができるまでに挫折してしまうからです。まずはこの三つが揃っていれば、それで動くものになります。

例えば「名前を入力したら挨拶を表示する」というシンプルなものでも、入力・処理・出力がすべて含まれています。この小さな構成で動かすことができれば、後から少しずつ機能を足していけます。最初は最小限で構いません。

【ワーク】 今日作るものを書く

ここで一度立ち止まって、今日作りたいものを一文で書いてみてください。完璧な文章である必要はありません。思いついたままでもかまいません。後で消してもいいし、変えてもいいという前提で書いてください。

□ 書く場所

この下の余白、メモ帳、紙のノート、どこでもかまいません。自分が書きやすい場所を使ってください。

例えば、こんな感じです。「ボタンを押したら今の時刻が表示されるものを作る」「入力した文字を逆さまにして表示する」「好きな色を選んだら画面の背景が変わる」。どれも一文で、動きがイメージできる内容です。

書くときのポイント

- 一文で表せる範囲にする
- 動きが具体的に想像できる
- 今日のうちに試せそうな大きさ

書かなくていいこと

- どの言語で作るか
- どうやって実現するか
- 完成後の拡張案

書いたら、それを今日の目標にしてください。途中で変わっても問題ありません。書いたことで、頭の中が少し整理されたはずですよ。それだけで、次に進む準備ができました。

止まることは想定内

プログラミングをしていると、必ずどこかで止まります。エラーメッセージが出る、思った通りに動かない、次に何をすればいいかわからなくなる。これは誰にでも起きることで、あなただけではありません。止まることは、作業の一部です。

エラーの扱い方

エラーメッセージは敵ではありません。何が起きているかを教えてくれる情報です。読んでみて、わからなければそのまま調べればいいのです。

停止の扱い方

動きが止まったら、最後にうまく動いた場所まで戻ってください。そこから一つずつ確認すれば、どこで止まったかがわかります。

止まった場所が次の作業になる理由は、そこに「わからないこと」や「うまくいかないこと」が集まっているからです。そこを調べたり、試したりすることが、次の学びになります。止まったことで、次にやることが明確になるのです。

大事なのは、止まったときに「自分には無理だ」と思わないことです。止まるのは当たり前で、そこから調べて進むのがプログラミングです。止まった回数が多いほど、あなたは学んでいます。



【ワーク】止まりそうな所を書く

次に進む前に、今から作り始めたときに「ここで止まりそうだな」と思う場所を書き出してみてください。不安な点、わからない部分、自信がないところ、何でもかまいません。先に書いておくことで、実際に止まったときに慌てずに済みます。

止まりそうな場所の例

- コードをどこに書けばいいかわからない
- エラーが出たときの対処法がわからない
- 必要な機能の作り方を知らない
- 動いているか確認する方法がわからない

戻る場所を決める意味は、迷子にならないためです。作業中に「どこまで進んだっけ」「何をやろうとしていたっけ」と忘れることがあります。先に書いておけば、そのメモが地図になります。

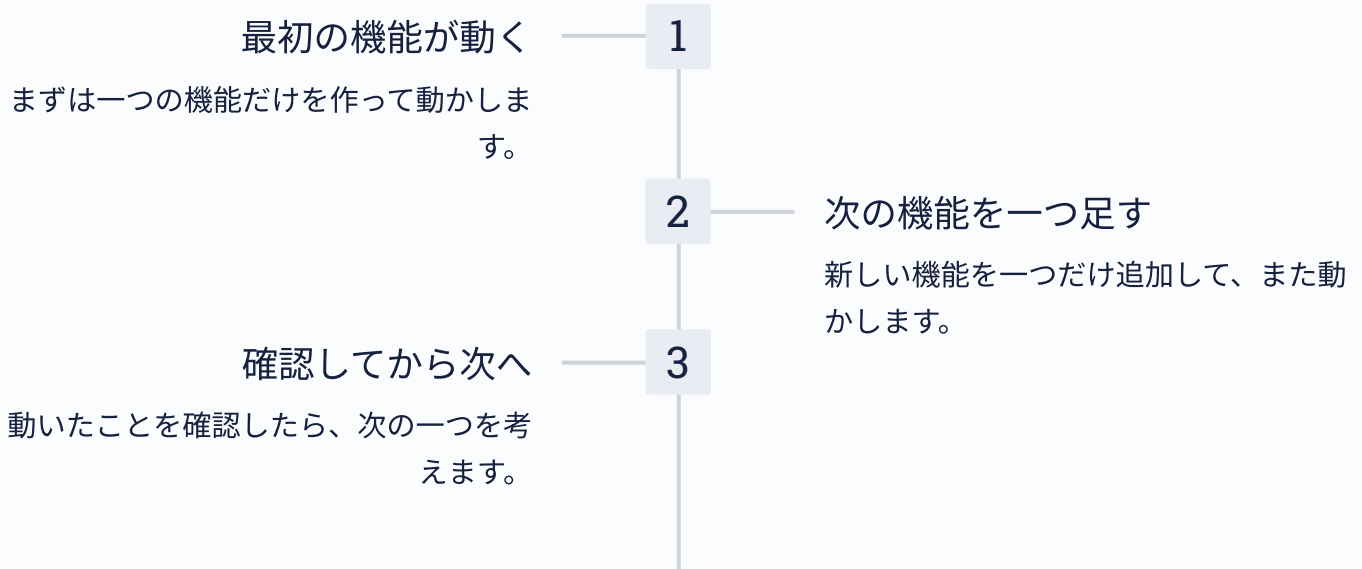
□ 書く内容

完璧な予測である必要はありません。今の時点で思いつくことを、箇条書きで書いてください。

書いたら、その中で一番不安なものを一つ選んでください。それを調べるか、試すかを決めれば、次にやることが明確になります。止まりそうな場所がわかっていれば、止まったときの対応も早くなります。

一つだけ広げる

何かが動いたら、次は機能を足したくなります。それは自然なことです。でも、一度にたくさん足そうとすると、どこで問題が起きたかわからなくなります。機能は一つずつ足してください。一つ足したら動かして確認する、これを繰り返します。



足さない判断も前進であることを覚えておいてください。「この機能は今はいらない」と気づいて、足さないと決めることも、作業が進んでいる証拠です。何でも足せばいいわけではありません。今必要なものだけを足すことで、シンプルで動きやすいものができます。

一つずつのメリット

- 問題が起きても原因がわかりやすい
- 動く状態を保ちながら作れる
- 途中でやめても使えるものが残る

複数同時のリスク

- どこで失敗したかわからない
- 全部直さないと動かない
- 途中で投げ出したくなる

【ワーク】次にやる一手を書く

ここまで読んで、あるいは少し作ってみて、次にやることを一つだけ決めてください。「次に足す機能」か「調べること」のどちらか一つを書いてみましょう。一つに絞ることで、迷わず動けるようになります。



次に足す機能

今動いているものに、一つだけ追加したい動きや表示を書きます。



調べること

わからないこと、試したいこと、確認したいことを一つだけ書きます。

例えば、こんな感じです。「ボタンの色を変える機能を足す」「エラーが出たときの処理を追加する」「入力欄を二つに増やす」。または、「ボタンの色を変える方法を調べる」「エラーメッセージの意味を確認する」「入力欄を増やすコードを探す」。どちらでもかまいません。

□ 書いたら

その一つだけをやってください。終わったら、また次の一つを書きます。この繰り返しが、作り続けるリズムになります。

一つずつ進めることで、少しずつ形になっていきます。焦らなくていいのです。今日の一つができれば、それで十分です。明日また次の一つを考えればいいのです。

この進め方を続ける

ここまで読んだあなたは、もう「まず動かす」「一つずつ進める」「止まることを想定する」という考え方を知っています。この進め方を続けることが、作り続けるための基本になります。迷ったときは、この教科書に戻ってきてください。



迷ったら戻る考え方は、この教科書全体に共通しています。うまくいかなかったら、最後にうまくいった場所まで戻ってください。そこから一つずつやり直せば、必ず前に進めます。戻ることは後退ではなく、確実に進むための方法です。

作り続けるための姿勢は、完璧を目指さないことです。今日できることをやる、わからないことは調べる、止まったら戻る、一つずつ足す。これを繰り返すだけで、あなたは作り続けられます。

この教科書の役割は、あなたが止まったときに戻ってこられる場所であることです。完璧な答えは書いてありませんが、続けるためのヒントは書いてあります。あなたの「作りたい」という気持ちを、これからも大切にしてください。