

ゲーム好きが、つくる側になるまで

マインクラフトやRobloxは 何でできている？

子どもに人気のゲームから知る
プログラミング言語とゲーム開発

ゲームが大好きなお子さまから、「ゲームってどうやって作っているの?」「自分でもゲームを作れる?」と聞かれたことはないでしょうか。普段何気なく遊んでいるゲームの裏側では、さまざまな処理がプログラムによって動いています。本冊子では、子どもたちに身近なゲームや制作環境を例に、「ゲームとプログラミング」の関係を保護者の皆さまに向けて整理しています。



ゲームエンジン

Python・C#・Lua

Scratch

ゲーム制作

IoT × AI プログラミング専門スクール Swimmy

保護者向けガイド／2026年8月

01 そもそもゲームは何で作られている？

ゲームによって、使われるプログラミング言語や開発プラットフォームは異なります。まずは全体像から見ていきましょう。

ゲーム制作というと、「プログラミング言語を使って、すべてを一からコードで作る」というイメージを持つ方もいるかもしれませんが。現在のゲーム開発では、プログラミング言語だけでなく、**ゲームエンジン**と呼ばれる開発環境を組み合わせるのが一般的です。

ゲームエンジンには、キャラクターや物体の表示、3D空間の構築、物理演算、音声、アニメーション、ユーザーインターフェースなど、ゲーム制作に必要な機能があらかじめ用意されています。

たとえば代表的な開発環境の一つであるUnityでは、C#というプログラミング言語を使ってゲームの動きやルールを作ることができます。Unityはモバイル、PC、家庭用ゲーム機、XRなど幅広いプラットフォームへの開発に対応しています。



コードを書いた結果が、画面の中の世界に反映される。その体験が理解を深めます。

ゲーム制作の全体像

「ゲームエンジンや制作プラットフォーム」 + 「プログラミング言語」

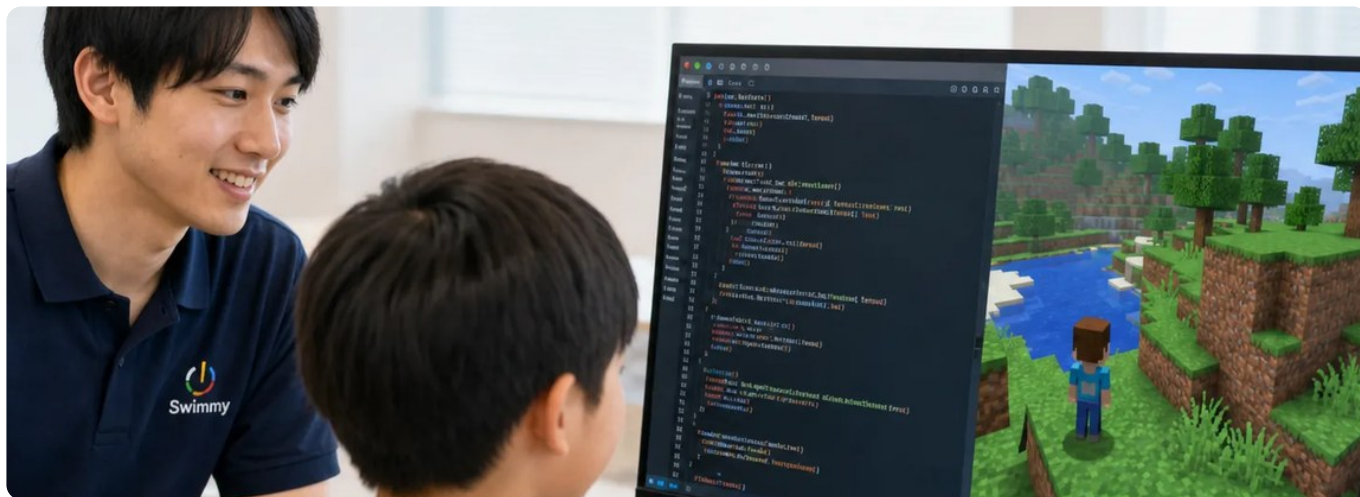
この組み合わせを理解すると、ゲーム制作の全体像がつかみやすくなります。

02 マインクラフトとプログラミング

ブロックを自由に配置し、自分だけの世界を作るMinecraftは、プログラミング教育との相性が非常に良いゲームの一つです。ただし、ここで知っておきたいのが、**「Minecraftそのものを開発するための技術」と「Minecraftを使ってプログラミングを学ぶための技術」は別物**だということです。

Minecraftにはエディションや開発方法によってさまざまな仕組みがあります。たとえばMinecraft Bedrock Editionでは、公式のCreator向け環境を使ってJavaScriptやTypeScriptによるスクリプトを作成し、ブロックやアイテム、エンティティの動作などをカスタマイズできます。Microsoftの公式ドキュメントでも、JavaScriptの基礎からTypeScriptを使ったより複雑なカスタマイズへ進める方法が紹介されています。

一方、教育の場ではMinecraftの世界を、プログラムの結果を目で確認できる「アウトプットの場」として活用する方法があります。



SwimmyではMinecraftとPythonを組み合わせて学習。コードを書いた結果が世界に反映されます。

プログラミングスクールSwimmyでは、MinecraftとPythonを組み合わせた学習を行っています。コードを書いた結果がMinecraftの世界に反映されるため、「このコードを書くと、なぜこう動くのか」を視覚的に理解しながら、テキストプログラミングへステップアップできる設計です。

普段遊んでいるMinecraftの世界が、そのままプログラミングを学ぶ場所になることで、「勉強している」という感覚だけではなく、**自分で世界を動かしている楽しさ**につながります。

03 Robloxは「遊ぶ」だけでなく「作れる」

Robloxは、一つのゲームというよりも、世界中のクリエイターが制作したさまざまなゲームやバーチャル空間を遊べるプラットフォームです。そしてRobloxには、ゲームを制作するための**Roblox Studio**という開発環境があります。

Roblox Studioで使われているプログラミング言語が**Luau（ルアウ）**です。Roblox公式ドキュメントでは、LuauはLua 5.1をベースに開発されたスクリプト言語であり、Roblox Studioでゲームの仕組みを作るために使用されると説明されています。

プログラムでつくられている「ゲームらしい仕組み」の例

- ✓ プレイヤーが床を踏んだら床が消える
- ✓ コインに触れたら得点が増える
- ✓ 一定時間ごとに敵が出現する
- ✓ ボタンを押すと扉が開く

ゲームを「遊ぶ側」から「作る側」の視点で見ると、「この仕掛けはどうやって作っているのだろう？」という新しい興味が生れます。

04 フォートナイトの世界でもプログラミングが使われている

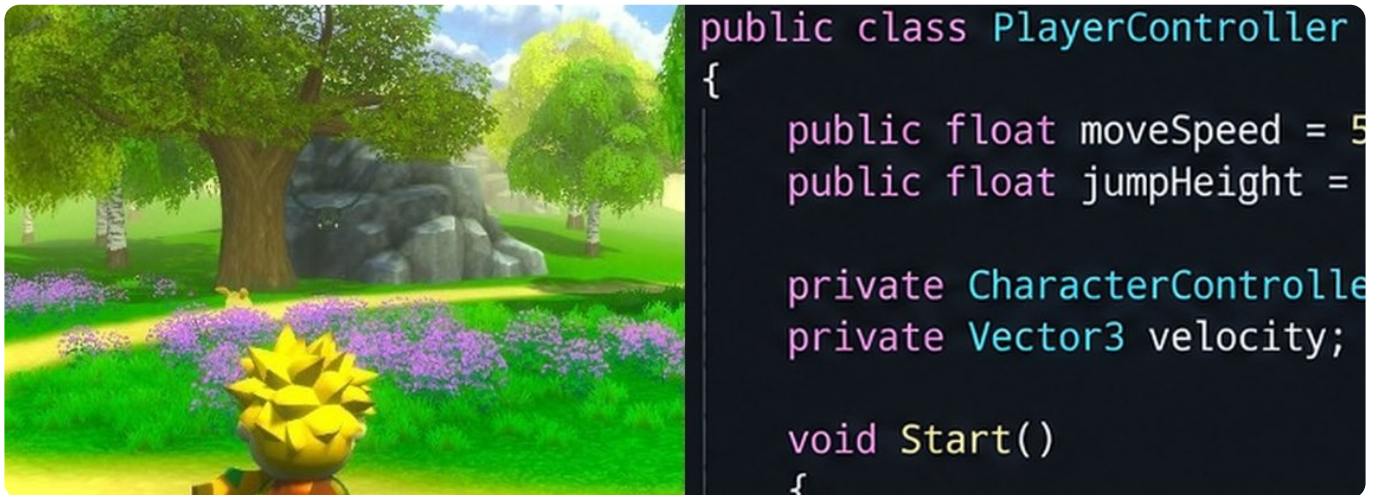
Fortniteにも、ユーザー自身がゲームや体験を制作できる環境があります。その一つが**Unreal Editor for Fortnite (UEFN)**です。

UEFNでは、Epic Gamesが開発するゲームエンジンUnreal Engineの技術を活用しながら、Fortniteの世界にオリジナルのステージやゲームルールを作ることができます。さらに、UEFNでは**Verse（ヴァース）**というプログラミング言語が使われています。Epic Gamesの公式ドキュメントでも、VerseによってNPCの動作やゲーム独自の仕組みをプログラミングできることが紹介されています。

「ゲームの中で遊ぶ」だけだった世界が、現在では「自分でゲームを作り、世界中の人に遊んでもらう」場所へと広がっています。

05 本格的なゲーム制作で広く使われる「Unity」とC#

子どもがゲーム制作に興味を持ったとき、さらに本格的な開発へ進む選択肢の一つがUnityです。Unityは2D・3Dゲームなどを制作できるゲームエンジンで、プログラミングには主に**C#（シーシャープ）**を使用します。Unity公式マニュアルでも、Unityのスクリプティング環境がC#に対応していることが明記されています。



Swimmy実践コースのゲームクリエイタークラス。UnityとC#でオリジナルゲームを制作します。

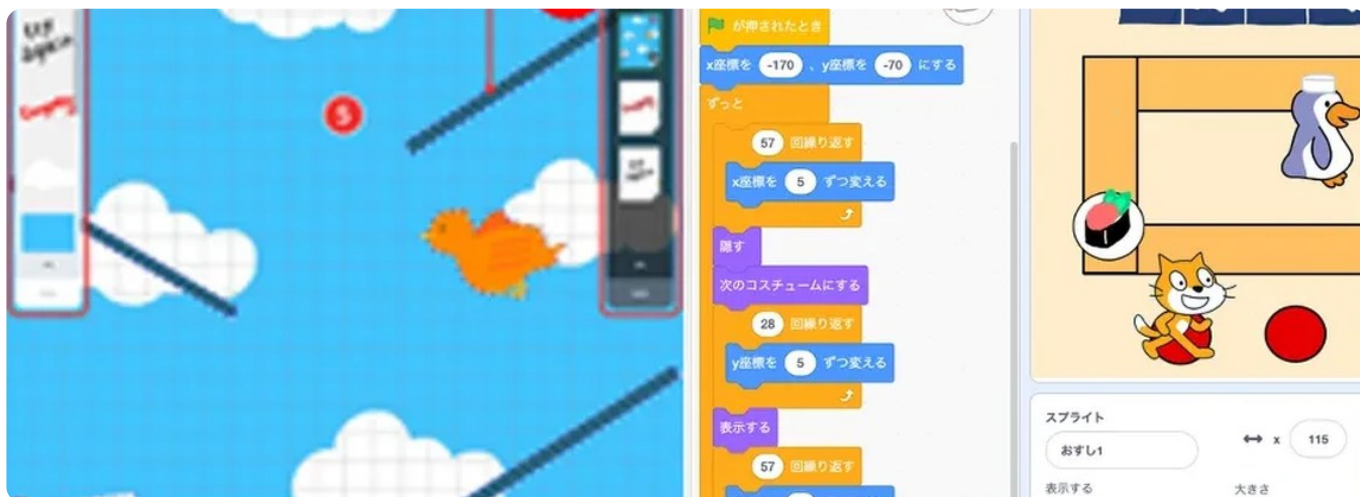
C#で組み立てられる「ゲームのルール」の例

- ✓ キャラクターの移動速度を設定する
- ✓ ジャンプボタンを押したら上方向に力を加える
- ✓ 敵に触れたらHPを減らす
- ✓ すべての敵を倒したら次のステージへ進む

ゲームを完成させるためには、単にコードを書く力だけではありません。「どんなゲームにするのか」を考える**企画力**、キャラクターやステージを作る**創造力**、バグの原因を調べる**問題解決力**、完成するまで試行錯誤する力など、さまざまな能力が必要になります。

「ゲーム制作には難しい英語のコードが必要なのでは？」と思われるかもしれませんが、しかし、最初からC#やPythonのようなテキストプログラミングを始める必要はありません。

小学生のプログラミング入門として広く利用されているScratchでは、「10歩動かす」「もし～なら」「ずっと繰り返す」といった命令ブロックを組み合わせながらゲームを作ることができます。実は、**ここで学んでいる考え方は本格的なプログラミングと大きく変わりません。**



Swimmyベーシックコースの制作例。ブロックを組み合わせながらゲームの仕組みを作ります。

ゲーム制作を通して自然に体験できる、プログラミングの基本概念

順番に処理する

▶ 順次処理

もし敵に触れたらゲームオーバー

▶ 条件分岐

ずっと敵を動かす

▶ 繰り返し処理

スコアを記録する

▶ 変数

Swimmyのベーシックコースでも、ScratchやSpringin'を使ったゲーム・作品制作を通して、創造力と論理的思考力を伸ばし、その後micro:bit、Minecraft×Pythonなどへ段階的に進むカリキュラムが用意されています。

保護者の方から見ると、ゲームは「遊び」という印象が強いかもしれませんが、しかし子ども自身が、「こんなゲームを作りたい」「もっと強い敵を作りたい」「自分だけのステージを作りたい」と思った瞬間、ゲームはプログラミングを学ぶ強力な入口になります。

プログラミングでは、思い通りに動かないことが何度も起こります。

子どもたちが自然と繰り返す問い

「なぜ動かないのだろう？」／「どこを直したらいいのだろう？」／「別の方法ならできるかもしれない」

と考えながらプログラムを修正する経験そのものが、論理的思考力や問題解決力につながります。Swimmyでも、プログラミングの正解は一つではなく、子ども自身が試行錯誤しながら答えを組み立てていくプロセスを重視しています。メンターが答えをすぐに教えるのではなく、子ども自身が解決へたどり着けるよう伴走する指導方針です。



「なぜ？」「どうすれば？」を子ども自身が考えられるよう、メンターが伴走します。

08

ゲーム制作から、本格的なプログラミングへ

ゲームを作るために使われる技術を整理すると、学びのステップが見えてきます。

ゲーム・制作環境	主な言語・技術	学べること
Scratch	ビジュアルプログラミング	条件分岐・繰り返し・変数などの基礎
Minecraftを活用した学習	Python など	テキストプログラミングへの入門
Roblox Studio	Luau	本格的なゲームロジック
Fortnite / UEFN	Verse	ゲームルールやインタラクションの制作
Unity	C#	2D・3Dゲーム、アプリなどの本格開発

大切なのは、「どの言語が一番優れているか」ではありません。**子どもの年齢や興味、プログラミング経験に合わせて、楽しく作れる環境から始め、少しずつ本格的な技術へ進んでいくこと**です。

プログラミングスクールSwimmyでは、プログラミング初心者のお子さまから、本格的なゲーム開発に挑戦したいお子さままで、段階的に学べるカリキュラムを用意しています。

ベーシック まずは「つくる楽しさ」から

MESH、Scratch、Springin'、micro:bitなどを活用しながら、プログラミングの基本的な考え方を身につけます。

エキスパート Minecraft × Python でテキストプログラミングへ

MinecraftとPythonを組み合わせ、本格的なテキストプログラミングへ進みます。

実践 ゲームクリエイタークラス (Unity × C#)

UnityとC#を使って、オリジナルゲームやアプリ制作に挑戦。ゲーム制作を通じてプログラム設計力や論理的思考力を高めます。



子どもにとって「大好きなゲーム」は、単なる遊びで終わるものとは限りません。「このゲームはどうやって動いているんだろう？」という疑問が、「自分でも作ってみたい」という好奇心に変わり、やがてプログラミングを学ぶきっかけになります。

ゲーム遊ぶ側から、自分のアイデアをプログラムで形にする側へ。

Swimmyでは、子どもたちの「好き」「やってみたい」という気持ちを出発点に、ゲーム制作を含むさまざまなプログラミング体験を通して、これからの時代に必要な論理的思考力、創造力、問題解決力を育てていきます。



体験授業では、お子さまの様子をご覧いただきながら、学び方や進め方をご相談いただけます。

無料体験授業を実施しています

対象	小学生～中学生（コースにより異なります）
コース	ベーシック（MESH／Scratch／Springin'／micro:bit）、エキスパート（Minecraft×Pythonほか）、実践（ゲームクリエイター／AI・データ分析／Web制作）
教室	高田馬場・三鷹・六本木・イオン柏・品川・津田沼 ほか、全国のフランチャイズ教室
ご予約	公式サイトフォームより承っています

プログラミング スイミー **検索**



IoT × AI プログラミング専門スクール swimmy-school.jp

出典・注記：技術情報は2026年8月時点の各社公式ドキュメント（Microsoft、Roblox Corporation、Epic Games, Inc.、Unity Technologies ほか）および「Swimmy スクール案内2026」をもとに記載しています。ゲームや開発環境で利用できる言語・機能は、バージョンや制作方法によって異なる場合がありますので、最新の情報は各社の公式サイトをご確認ください。

本冊子に記載されている会社名・製品名・サービス名は、各社の商標または登録商標です。本文中では™ および® を省略しています。本冊子はSwimmyが独自に作成した保護者向けの情報提供資料であり、各社が監修・提供するものではありません。掲載写真・画面はすべてSwimmyのイメージです。

発行：株式会社エス・イー・アイ プログラミングスクール Swimmy