



創造的コンピューティング

カレン・ブレナン | クリントン・バルチ | ミシェル・チャン
ハーバード大学 教育学大学院

目次

バックグラウンド.	1
“創造的コンピューティング”とは	1
Scratchとは	2
ガイドについて	3
ガイドの対象者	3
ガイドを用いるために必要な物	4
ガイドに記述されていること	4
ガイドの利用方法	5
ガイドを作成した所	5
ユニット0 - はじめに	7
Scratchの紹介	10
Scratchアカウント	12
デザイン日誌	14
Scratchサブライズ	16
Scratchスタジオ	18
批評グループ	20
ユニット1 - 探究しよう	23
ダンスをプログラム	26
ステップ・バイ・ステップ	28
10ブロック	30
マイスタジオ	32
デバッグしよう！	34
自己紹介	36
ユニット2 - アニメーション	39
スクリプトを演じる	42
バンドを作ろう	44
色付きの四角と丸	46
生きてる！	48
デバッグしよう！	50
ミュージックビデオ	52

ユニット3 - 物語	55
登場人物	58
会話	60
場面	62
デバッグしよう！	64
創造物の制作	66
続けて創ろう	68
ユニット4 - ゲーム	71
理想のゲームリスト	74
入門用ゲーム	76
スコア機能	80
拡張機能	82
対話処理	84
デバッグしよう！	86
ユニット5 - もっと探究しよう	89
学びたいことを知ろう	92
第2ラウンド	94
高度な概念	96
ハードウェアと拡張機能	100
アクティビティのデザイン	102
マイデバッグ！	106
ユニット6 - ハッカソン	109
プロジェクト・ピッチ	114
プロジェクト・プランニング	116
デザインスプリント	120
プロジェクト・フィードバック	122
プロジェクト・チェックイン	124
アンフォーカスグループ	126
ショーケースの準備	128
ショーケース	130
付録	133
用語集	135
スタンダード	139
コンピュータ処理の概念考	141
参考書籍、論文、レポート	147
リンク	149



バックグラウンド

できるだけ早く、“創造的コンピューティング”の世界にダイブできるよう、よくある8つの質問にまとめて答えていきます。

1. “創造的コンピューティング”とは
2. Scratchとは
3. ガイドについて
4. ガイドの対象者
5. ガイドを用いるために必要な物
6. ガイドに記述されていること
7. ガイドの利用方法
8. ガイドを作成した所

創造的
コンピューティング
カリキュラムガイド
によろこそ！

“創造的コンピューティング”とは



“創造的コンピューティング”とは、
創造性に関する事です。

コンピュータサイエンスやコンピュータに関連する分野は、子供たちの興味や価値観とは別物として、長らく紹介されてきました。また、創造的な可能性よりも、技術的記述だけが重要視されてきました。“創造的コンピューティング”は、子供たちの創造性、想像力、関心を引き出すことで、コンピュータとの関わり方を成熟させます。



“創造的コンピューティング”とは、
主体的な行為に関する事です。

コンピュータにアクセスする多くの子供たちは、デザイナーやクリエイターとしてではなく単なる消費者として関与しています。“創造的コンピューティング”は、若者が日常生活で楽しんでいるさまざまな動的でインタラクティブなコンピュータ・メディア[訳注1]を創作するのに必要な知識、実践、および基本的なリテラシーを重要視しています。



“創造的コンピューティング”とは、
コンピュータ利用に関する事です。

子供たちは、コンピュータ作品の創作に関わることで、コンピュータサイエンティストやプログラマーとしてのキャリア以外の、より多くのキャリアを身に付けます。“創造的コンピューティング”は、子供たちを“Computational Thinker (コンピュータ利用による思考者)”に成長させてくれます。すなわち、分野や背景を超えて、コンピュータ処理の概念、実践、そして人生の展望を描ける子供にしてくれます。

訳注1: PCやスマホのゲームやアプリのようなコンピュータ技術を使った情報の媒体
訳注2: 原文はAgency. 自ら結果をもたらす主体的な行為

Scratchとは



創造的コンピューティングのために利用可能なさまざまなツールがありますが、このガイドでは、<http://scratch.mit.edu>で利用可能なフリーのプログラミング言語、Scratchを用います。Scratchを使うと、多種多様な双方向メディアのプロジェクト（アニメ、物語、ゲーム、その他）を創作し、オンライン・コミュニティでこれらのプロジェクトを友達と共有することができます。Scratchは、2007年5月にリリースされて以来、世界中の何十万人もの人々が、600万以上のプロジェクトを創作し、共有しています。

ガイドについて

このガイドは、Scratchプログラミング言語を使った入門的な創造的コンピューティングの体験のためのアイデア、戦略、およびアクティビティの集大成です。このアクティビティは、親しみながら、コンピュータ利用による創造性や計算機論的思考[訳注1]を使いこなす能力を高めるよう、デザインされています。特に、アクティビティは、計算機論的思考の主要な概念（順次処理、繰り返し処理、並列処理、イベント処理、条件分岐、演算子、データ）と思考の主要な実践（実験と反復、テストとデバッグ、再利用とリミックス、抽象化とモジュール化）の探究を促します。計算機論的思考についての詳細は付録のリソースや<http://scratched.gse.harvard.edu/ct>を参照下さい。

学びに対する構築主義者のアプローチ[訳注2]によって影響を受けており、このガイドのアクティビティでは、以下の原則が重視されています。

原則#1 創作

単に聴いたり観たり使うだけでなく、デザイン活動や制作活動に関与する機会を学習者にもたらしめます。

原則#2 パーソナライズ

自分にとって意義があり、意味あるアクティビティに関与させる機会を学習者にもたらしめます。

原則#3 共有

聴衆、コーチ、そして共同創作者のような人と交流する機会を学習者にもたらしめます。

原則#4 振り返り

創造的な実践を振り返り、再考する機会を学習者にもたらしめます。

[訳注1] “Jeannette M. Wing : Computational Thinking, Communications of the ACM, Vol.49, No.3, pp.33-35(Mar. 2006)”の翻訳版も参照下さい。 <https://www.cs.cmu.edu/afs/cs/usr/wing/www/ct-japanese.pdf>

[訳注2] シーモア・パパートが提唱している効果的学習のアプローチ

ガイドの対象者

あなたの現在の状況や経験に関係なく、このガイドは幅広い学習者と教育者を念頭に置いてデザインされています。ガイドの利用者とその活用例をいくつか示します。

小・中・高等学校(K-12)の先生

Scratchは、世界中の何千もの小・中・高等学校の教室で利用されています。このガイドは、学期を通じたコンピュータ教育コースや他のカリキュラムの一部の選択コースとして使用することができます。

多くの教育者が、放課後プログラムや昼食時プログラムとして創造的コンピューティングを紹介しており、生徒のオープンエンドの探究[訳注1]のための「きっかけ」と「足がかり」としてこのアクティビティを使っています。

博物館または図書館の教育者

Scratchは、教室などの公式な学習環境に加えて、博物館や図書館などの形式ばらない学習空間でも使用されています。体系化されたワークショップ体験会や一時利用スペースかどうかにかかわらず、博物館や図書館の学習環境は、従来の環境（旧来の教室）に存在する制限無しに、創造的コンピューティングの探究をサポートするのに理想的です。

保護者

保護者はこのガイドを幅広い方法で使用可能です。我々は、在宅学習の活動のサポートから、学校での創造的コンピュータクラブの立ち上げ、地域コミュニティセンターでのワークショップ開催に至るまで、保護者が、若い学習者の創造的な体験をサポートするためのガイドの活用方法について良く考えることを奨励しています。

創造的コンピューティングは、みんなのものです！

大学の講師

Scratchは、基本的なコンピュータの概念と実践の入門用として役立ち、多くの場合、コンピューターサイエンス・コースの従来のテキストベースのプログラミング言語に引き継がれていきます。例えば、ハーバード大学のCS50コースでは、Cプログラミング言語に移行する前の入門的なプログラミング経験としてScratchが使われています。このアクティビティは、大学レベルの教育、芸術、メディアリテラシーのコースの一部としても、活用されています。

若い学習者

Scratchの発表以来7年間、さまざまな場面において、若い学習者は、創造的コンピューティングに対する情熱的な擁護者でした。彼らの両親や先生にプログラミングを紹介することから、仲間の学習機会創出に至るまで、創造的コンピューティングは、受け身ではなく、学習者が一緒に自分達で成し遂げる何かとなる可能性があります。

ガイドを用いるために必要な物

アドベンチャーのための時間とオープンさに加えて、以下の重要なリソースが必要となります。

- + **スピーカー付きのコンピュータ**（マイクとウェブカメラは任意）：コンピュータを使ったデザイン・アクティビティのために使用。
- + **ネットワーク接続**：Scratchオンライン[訳注2]に接続するため。
ネットワーク接続ができない場合は、ダウンロード可能なScratchのバージョンが利用可能です。
- + **プロジェクターやスピーカー付きインタラクティブ・ホワイトボード**[訳注3]：進行中の作品の共有とデモンストレーションのために使用。
- + **デザインノート**（紙またはデジタル）：アイデアとプランを書いたり、スケッチしたり、ブレインストーミング[訳注4]するために使用。

[訳注1] 生徒の興味に任せ終りのない探究

[訳注2] MITメディアラボが公開しているScratchウェブサイト(<http://scratch.mit.edu/>)

[訳注3] PCやタブレットのデータを映したり、直接思いついたアイデアを書き込めるホワイトボード

[訳注4] グループでアイデアを自由に出し合うことで多くの発想を誘発する手法

ガイドに記述されていること

このガイドは、最初の予備的なユニットから最終的なプロジェクトベースのユニットまでの7ユニットで構成され、各ユニットには通常アクティビティが6つ含まれています。各ユニットの概要は以下の通りです。

ユニット0

ユニット0【はじめに】

可能性を探ることによって創造的コンピューティングのカルチャーに備えて、技術的インフラ（Scratchアカウントの作成、デザイン日誌のスタートなど）と社会的インフラ（批評グループの設立など）を準備します。Scratchのキャラクターにサプライズを起こすことによって、最初の創造的な体験にダイブします。

ユニット1

ユニット1【探究しよう】

さまざまなレベルの仕組み（ステップ・バイ・ステップのチュートリアル、限られた数のブロックを使う創造的なチャレンジ、自己紹介プロジェクトの制作を通じたオープンエンドの探究）を提供する一連のアクティビティを通して、主要なコンピュータ処理の概念を使いこなせるようにします。

ユニット2

ユニット2【アニメーション】

アニメーション、芸術、音楽にフォーカスしたアクティビティでは、映像やオーディオで遊びます。独自のバンドを作成し、生きもののアニメーションをデザインし、好きな曲のミュージックビデオの制作することで、Scratchが重点を置いているメディアと主要なコンピュータ処理の概念（繰り返し処理、イベント処理、並列処理）を探究します。

ユニット3

ユニット3【物語】

合作の物語を通じて新たなインタラクティブ・ワールドを創作します。登場人物を制作することから始め、会話のコーディングを学び、登場人物や会話を場面の变化に合わせてます。登場人物、会話、場面を、より大きな物語プロジェクトに組み込みます。そのプロジェクトは、他のクリエイターに引き継がれ、さらに発展します。おそらく、すっかり違ったものにイメージし直されます。

ユニット4

ユニット4【ゲーム】

スコアやレベル付けなどの基本的なゲームの仕組みを、主要なコンピュータ処理の概念（変数、演算子、条件分岐など）に結びつけます。ピンポンゲームのようなクラシックなゲームを作成（し、拡張）することで、お気に入りのゲームを分析し、新しいゲームをイメージし、ゲームのデザインを実践します。

ユニット5

ユニット5【もっと探究しよう】

最終的なユニットの前に、以前のユニットの作業を再び訪れる時間をとります。さらに高度な概念を探究したり、新しいアクティビティやデバッグの課題をデザインすることで友達を手助けします。

ユニット6

ユニット6【ハッカソン】

計画・作成・共有の反復的なサイクルを通して、自分自身のプロジェクトをデザインし、開発することによって、コンピュータ処理の概念と実践のすべてをアクションに落とししていきます。

評価の戦略はガイド全体で説明されています。さらにガイドの付録にいくつかの評価ツールが掲載されています。我々は、評価のアプローチとして、生徒が自分自身（と他人）の作品や創作の実践について話し合う機会を作り出すことに焦点を当てたプロセス指向を採用しています。プロセス指向の収集可能なデータのフォームがあり、さまざまな戦略はガイドを通して提案されています。例えば、以下です。

- + プロジェクトに関する生徒との会話や生徒間の会話のサポート、記録された音声・ビデオ・テキスト
- + プロジェクトのポートフォリオ^[訳注1]の調査
- + デザイン日誌の維持管理

生徒が「既に知っていることは何か」や「更に学びたいことは何か」を理解するのを支援するために、我々は、評価を「生徒と一緒にやっているもの」と見ています。クリエイター、仲間、先生、保護者などを含むさまざまな参加者を巻き込んで評価することが可能です。

[訳注1]ポートフォリオとは、戦略目標を達成するためにグループとしてマネジメントされるプロジェクト、プログラム（関連プロジェクトを協調的にマネジメント）、サブポートフォリオおよび定常業務の集合。ここではポートフォリオの戦略目標を調査しその一部を担うプロジェクトの個別目標を確認する。

ガイドの利用方法

好きなだけ
使おう

新しい
アクティビティを
デザインしよう

アクティビティを
リミックスしよう

あなた自身の
アドベンチャー
を選択しよう！

我々は、新しいアクティビティをデザインしたり、アクティビティをリミックスするために、このガイドを好きなだけ使うことを勧めています。

我々は、過去の経験や専門知識とは無関係に、教育者は全員、創造的コンピューティングの体験の共同デザイナーであると考えています。

我々は、みなさんが行っている事について知りたいと思っています。そのため、[ScratchEdコミュニティー](#)を介して、我々や他の教育者とのあなたの経験をドキュメントにして共有することを奨励しています。

我々は、このガイドを Creative Commons Attribution-ShareAlike ライセンスの下でリリースしています。つまり、適切な帰属を示し、他者が二次著作物にアクセスできる限り、この著作物を自由に利用、改変、共有することができます。

ガイドを作成した所

このガイドは、ハーバード大学 教育学大学院のScratchEd研究チームのメンバー（Christan Balch氏、Michelle Chung氏、Karen Brennan氏）によって作成されました。Jeff Hawson氏は編集のサポートと疲れを知らない熱意を提供してくれました。

ガイドの内容は、Creative Computing Guide（2011年出版）の以前のバージョンとCreative Computing Online Workshop（2013年開催）を引用しています。これらは、補助金DRL-1019396、Google CS4HSプログラム、およびCode-to-Learn財団を通じて国立科学財団の支援を得て可能になりました。

我々は、このガイドの以前のバージョンを使用したワークショップに参加した数多くの教育者に大変感謝しています。特に、最初のガイド（Russell Clough氏、Judy Hoffman氏、Kara Kestner氏、Alvin Kroon氏、Melissa Nordmann氏、Tyson Spraul氏）および現在のガイドを幅広く見直した教育者（Ingrid Gustafson氏、Megan Haddadi氏、Keledy Kenkel氏、Adam Scharfenberger氏、LeeAnn Wells氏）に感謝しています。

我々は、共同研究者にも大変感謝しております。教育開発センターの児童・技術センターのWendy Martin氏、Francisco Cervantes氏、Bill Tally氏、MIT Media LabのMitch Resnick氏の計算論的思考フレームワークとリソースの開発の多大な貢献に感謝しております。

2011年の初版以来、数年間にわたり、ガイド開発に貢献してきた数多くのハーバード大学 教育学大学院のインターン（Vanity Gee氏、Vanessa Gennarelli氏、Mylo Lam氏、Tomoko Matsukawa氏、Aaron Morris氏、Matthew Ong氏、Roshanak Razavi氏、Mary Jo Madda氏、Eric Schilling氏、Elizabeth Woodbury氏）にも感謝しております。

ユニット0 はじめに



インデックス

内容

0

1

2

3

4

5

6

Scratchの紹介	10
Scratchアカウント	12
デザイン日誌	14
Scratchサプライズ	16
Scratchスタジオ	18
批評グループ	20

ユニット0

概要

“ビッグ・アイデア”

我々が、このガイドのドラフトを教師に共有したとき、最初の反響は「ユニット0?」、「なぜ、0なの?」でした。

我々は、「ユニット0は準備のユニットであり、創作、パーソナライズ、共有、振り返りの原則を通して、創造的コンピューティングのカルチャーの確立をサポートしています。」と伝えたいと願っていました。学びのカルチャーをサポートするという我々の望みを、ガイドを通して明らかにしていきます。

創造的コンピューティングのカルチャーには、以下の側面があります。

- コンピュータ処理の概念と実践に取り組む知的な側面
- 机、椅子、コンピュータの配置を通して他者との対話を促進する物理的な側面
- 信頼感と不安のない安心感を養う最も重要な感情的な側面

学習の目的

- 生徒は、
- + Scratch環境の中で、コンピュータ創作の概念を紹介されます。
 - + 独自のScratchベースのコンピュータ創作の可能性をイメージできるようになります。
 - + コンピュータ創作をサポートするリソース[訳注1]に詳しくなります。
 - + Scratchプロジェクトを創作する準備を整えます。
(Scratchアカウントの作成、Scratchスタジオ[訳注2]の検索、デザイン日誌の作成、および批評グループの結成)

訳注1: 参考になる配布用プリント(配布資料)、プロジェクト、ビデオ、カードなど
訳注2: Scratchの作品(プロジェクト)を収録・公開するためのスペース

あなたが持っているクラスにある種の文化や風潮があるなら、本当に役に立つでしょう。子供たちに「ミスをしなさい」、「私が何か難しいことをさせようとしている」ということを理解してもらうのまましておきます。私はいつもそうすると子供たちは初めは上手いきません。彼らはただ何かをクリアしたいだけなのです。大人ですら失敗したりミスをしたりすることを好みませんが、重要なことだと私は思います。困難との遭遇は、諦めたり泣いたりするための時間ではなく、課題を解く方策を考えたり、助けを求めたりする作戦を考えるための時間です。その調子で続けましょう。

小学校教師TS氏

キーワード、概念、実践

- + プロフィールエディタ
- + 批評グループ
- + プロジェクトページ
- + レッド、イエロー、グリーン
- + スタジオ

メモ

- + IT部門と調整して、PCからScratch Webサイトにアクセス可能か確認します。
- + インターネットにアクセスできない場合、Scratchオフライン版が利用可能です。
<https://scratch.mit.edu/download>

アドベンチャーを選択しよう



始める準備はできましたか？

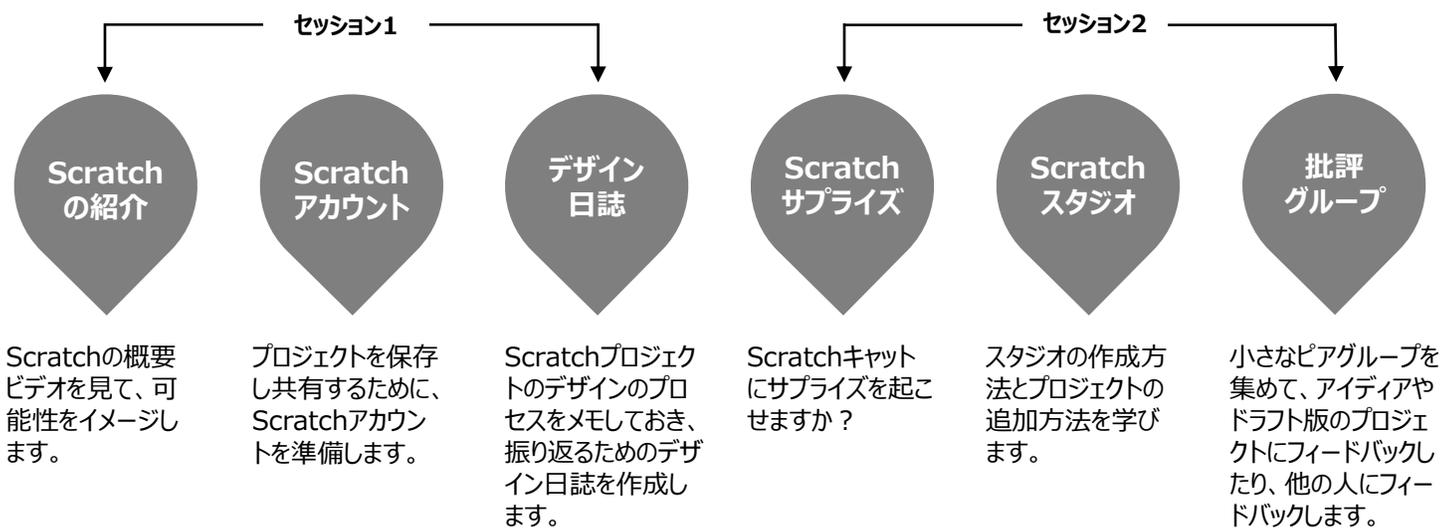
このユニットはScratchの初心者向けにデザインされています。各アクティビティは感動的なプロジェクトを探究することから、Scratchアカウントを作成すること、最初のプロジェクト・エディタの体験に至るまで、Scratchをスタートするプロセスを通して、あなたや生徒をガイドするようデザインされています。

我々は、各ユニットにはアクティビティの選択肢を1つ、準備しています。しかし、アクティビティの選択や順序をあれこれ変更してみることをお勧めします。さまざまな背景と参加者は、多様な体験を引き起こしてくれます。あなたや学習者にとって説得力ある方法で、アクティビティをミックスしたり、マッチングして、あなた自身のアドベンチャー[訳注1]を選択して下さい。

始める場所がわからないなら、下記の可能なルートをチェックしましょう。

訳注1: 想定外のことが起こることを表現している

可能なルート



Scratch の紹介

 **推奨時間**
5-15分

目的

- このアクティビティを完了すると、生徒は、
- + Scratchの概要ビデオを視聴したり、サンプルプロジェクトを調べることで、Scratchプログラミング環境でのコンピュータの創作品が紹介されます。
 - + 生徒自身のScratchベースのコンピュータ作品の可能性をイメージできるようになります。

アクティビティの説明

- 「振り返りを促す質問」を活用して、生徒にコンピュータでの体験について質問します。
- 生徒にScratchを使った「創造的コンピューティング」とさまざまなプロジェクトを紹介します。魅力と感動を感じるScratchの概要ビデオとサンプルプロジェクトを見せることによって、生徒は創作できるようになります。次のいくつかのセッションで、生徒がScratchでのインタラクティブなコンピュータ・メディアを創作する予定であると説明します。
- あなたは何を創作したいですか？
生徒に、Scratchでどんなプロジェクトを創りたいかイメージするよう促します。

リソース

- Scratchの概要ビデオを見せるためのプロジェクター（任意）
- Scratchの概要ビデオ
<http://vimeo.com/65583694>
<http://youtu.be/-SjuiaWRMU4>
- Sample Scratch Projectsスタジオ
<http://scratch.mit.edu/studios/137903>

振り返りを促す質問

- + コンピュータとやりとりするいろいろな方法を上げてください？
- + これらの方法のうち、コンピュータで創造的になれる方法はいくつありますか？

作業の確認ポイント

- + 生徒はさまざまなプロジェクトのアイデアをブレインストーミングしたか？
そうでないなら、生徒に可能性を感じてもらおうよう、多種多様なプロジェクトを示します。

メモ

- + インターネットがアクセスできない場合、授業が始まる前に、VimeoからScratchの概要ビデオをダウンロードして下さい。<http://vimeo.com/65583694>から入手可能です。
- + 「振り返りを促す質問」の回答を書き留める代わりに、生徒に回答を絵に描いてクリエイティブになるよう勧めます。（例、“コンピュータと対話するさまざまな方法を絵に描く”）

追加メモ記入欄

- _____
- _____
- _____
- _____

がクリックされたとき

10 回繰り返す

10 歩動かす

色 ▾ の効果を 25 ずつ変える

終わるまで ニャー ▾ の音を鳴らす

Scratchによようこそ！ という



Scratch アカウント

 推奨時間
5-15分

目的

- このアクティビティを完了すると、生徒は、
- + Scratchアカウント[訳注1]を作成します。
 - + Scratchオンライン・コミュニティを検索し、Scratchコミュニティ・ガイドラインを確認します。

アクティビティの説明

- Scratchオンラインのアカウントには、Eメールアドレスが必要です。生徒が個人や学校のEメールアドレスを用意できない場合、教師や両親/保護者のEメールアドレスを使用します。オンラインアカウントの許可証を収集する必要がある場合は事前に計画しておいて下さい。
- 生徒を、Scratchウェブサイトナビゲートし、“Scratchに参加しよう”をクリックして、Scratchアカウント作成をガイドします。必要に応じて、生徒をガイドする「Scratchアカウント」配布用プリントを用意します。生徒に時間を与え、アカウントの登録・プロフィールの更新・オンライン・コミュニティの検索をさせます。生徒に自分のアカウントへのサインインとサインアウトを練習するよう促します。
- クラスのメンバーがお互いのScratchプロフィールを見つけてフォローしやすいよう、ユーザー名や名前の一覧を作成することを検討して下さい。
- 敬意を表す行動と建設的な行動についてディスカッションするために、グループでScratchコミュニティ・ガイドライン[訳注2]を調べておきます。ウェブサイトでの不適切な投稿を報告する方法を確認しておきます。

リソース

- 「Scratchアカウント」配布用プリント
- Scratchコミュニティ・ガイドライン
http://scratch.mit.edu/community_guidelines

振り返りを促す質問

- + Scratchアカウントのユーザー名とは何ですか？
- + パスワードを覚え易くするためのヒントは何ですか？

作業の確認ポイント

- + 生徒は、Scratchアカウントを作成し、Scratchウェブサイトにもうまくサインイン/サインアウトすることができたか？

訳注1: Scratchウェブサイトを利用するためのユーザーアカウント。プロジェクトをオンライン上に保管し、共有するために必要。

訳注2: https://scratch.mit.edu/community_guidelines

メモ

- + Scratchウェブサイト上での不適切な行動の通知は、アカウントに登録されているメールに送られます。このため、教師は、生徒にEメールを提出させたり、クラス専用のEメールアドレスを作成するのを希望するかもしれません。
- + 既にオンラインアカウントを持っている生徒がいるか確認しておきます。
- + プライバシーを維持しつつ、パスワードを覚えておくために、生徒にユーザー名とパスワードを書き留めさせた紙を封筒に入れて、安全な場所に保管しておきます。

追加メモ記入欄

- _____
- _____
- _____
- _____

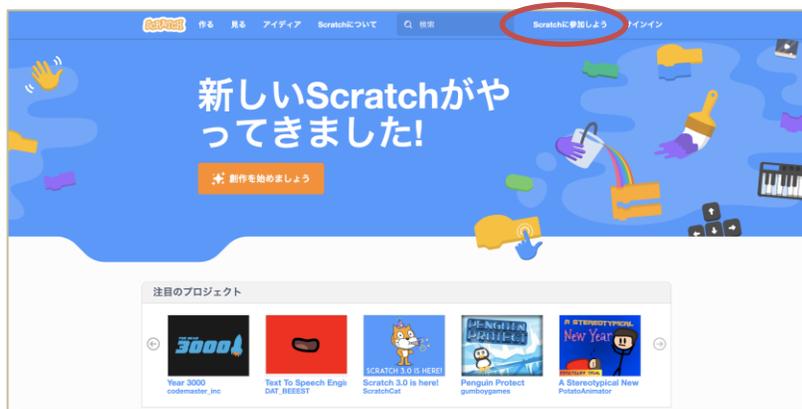
Scratchアカウント

初めての方は、Scratchアカウントを作成することから始めましょう！

Scratchプロジェクトを保存して共有するには、Scratchアカウントが必要です。
新しいアカウントを作成し、プロフィールを設定する手順を以下に示します。

ここからスタート

- ウェブブラウザを立ち上げ、Scratchウェブサイト (<http://scratch.mit.edu>)を開きます。
- ホームページ上で、右上のメニューバーあるいは青い丸の中の“Scratchに参加する”をクリックします。
- 3ステップの入力を完了させて、あなた自身のScratchアカウントを作成します！



Scratchに参加しよう

Scratchのアカウント作成はとても簡単！（そして無料です）

Scratchで使うユーザー名を入力 本名は使わないでね

パスワードを入力

パスワードの確認



1 2 3 4

デザイン日誌

 **推奨時間**
15-30分

目的

このアクティビティを完了すると、生徒は、
+ 自分のデザインのプロセスと振り返りの記入用にパーソナライズされたデザイン日誌を付け始めます。

アクティビティの説明

- 生徒にデザイン日誌のアイデアを紹介します。デザイン日誌は、個人的な日誌や日記と同じように、アイデアをブレインストーミングしたり、ひとりで振り返りができる物理的なノートやデジタルのノートです。生徒に、Scratchプログラミングのアドベンチャー全体を通して、デザイン日誌の更新を奨励していることを説明します。プロジェクトをデザインしている時はいつでも、思い付いたこと（アイデア、インスピレーション、メモ、スケッチ、質問、うまくいかなかった事、うまくいった事など）を自分の日誌に追加するよう勧めます。
- どのタイプのデザイン日誌（紙またはデジタル）が、生徒にとって最も機能するかのヒントを得るため、サンプルのデザイン日誌に目を通します。生徒に時間を与え、デザイン日誌を準備し、カスタマイズさせます。
- 生徒に、「振り返りを促す質問（右記）」に回答してもらい、最初のデザイン日誌の記録を作成してもらいます。
- 生徒に、デザイン日誌と最初の振り返りを友達と共有するよう促します。

リソース

- デザイン日誌のサンプル
<http://bit.ly/designjournal-paper>
<http://bit.ly/design-journal-digital>
<http://bit.ly/designjournal-blog>
- ペーパークラフトのマテリアル（紙の日誌）

振り返りを促す質問

- + どのように友達にScratchを説明しますか？
- + 創作してみたいScratchプロジェクトのアイデアを3つ書き出すか、絵にしてください。

作業の確認ポイント

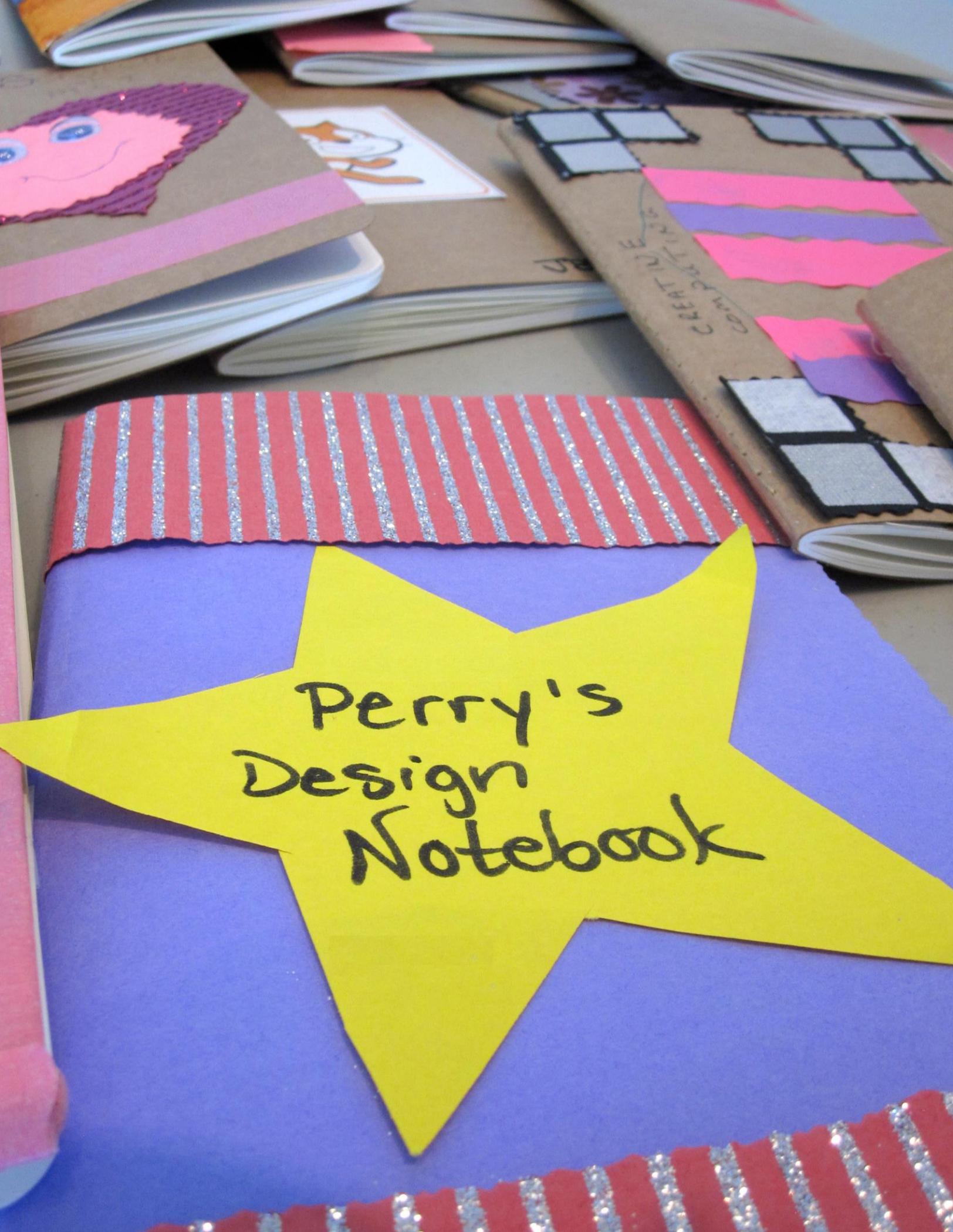
- + 生徒が追求することに関心があるプロジェクトに関して、振り返りから何がわかるか？
- + 生徒の反応に基づくと、このガイドのどのユニットが色々な生徒にアピールする可能性があるか？

メモ

- + 他のガイドのアクティビティで、「振り返りを促す質問」についてのグループ討議をファシリテートします。
- + デザイン日誌をプライベートにするかパブリックにするか、決定します。例えば、
 プライベートな日誌を通して生徒との1対1のフィードバックを保持しておきます。
 あるいは、
 生徒に共有された日誌に仲間のためのコメントを残してもらいます。
 それぞれの選択肢の長所と短所についてよく考えて下さい。

追加メモ記入欄

- _____
- _____
- _____
- _____



Perry's
Design
Notebook

Scratch サプライズ

 推奨時間
15-30分

目的

このアクティビティを完了すると、生徒は、
+ Scratchを使って予備的なハンズオンの体験
をします。

アクティビティの説明

- Scratchウェブサイトにナビゲートして、Scratchアカウントにサインインし、ページの上部の“作る”をクリックして、生徒がScratchプロジェクトのエディタをオープンできるようガイドします。
必要に応じて、生徒をガイドする「Scratchサプライ」配布用プリントとScratchカードを用意します。
- 生徒に10分間与え、オープンエンドのやり方でScratchのユーザ・インタフェースを探究させます。「10分間でScratchキャットに何かサプライズを起こしましょう。」、あるいは「10分間、怖がらずにインタフェースを調べましょう。何か気づきましたか？」と生徒に促します。生徒に、一緒に作業し、お互いに助け合い、何を理解したか共有するよう促します。
- 3,4人、ボランティアを頼み、気付いた事をひとつ、グループ全体に共有してもらいます。
必要に応じて、ボランティアが共有した後、生徒にいくつかの課題を示します。
 - 誰か、サウンドの追加方法がわかりますか？
 - 誰か、背景の変更方法がわかりますか？
 - 誰か、ブロックのヘルプの表示方法がわかりますか？

リソース

- 「Scratchサプライズ」配布用プリント
- Scratchカード
<http://scratch.mit.edu/info/cards/>

振り返りを促す質問

- + 何を理解しましたか？
- + 何について、もっと知りたいですか？

作業の確認ポイント

- + 生徒は、新しいプロジェクトを開始する方法を知っているか？
- + 生徒は、Scratchブロックを一緒にスナップする基本的なメカニズムを理解しているか？

メモ

- + このアクティビティの主なゴールは、大胆さ、探究心、そして友達とのコラボレーションのカルチャーを確立することです。
生徒（その教師も）が、すべてことを事前に知らなくても良いこと、そして環境がみんな一緒に学ぶ空間になることが、期待されています。

追加メモ記入欄

- _____
- _____
- _____
- _____

Scratchサプライズ

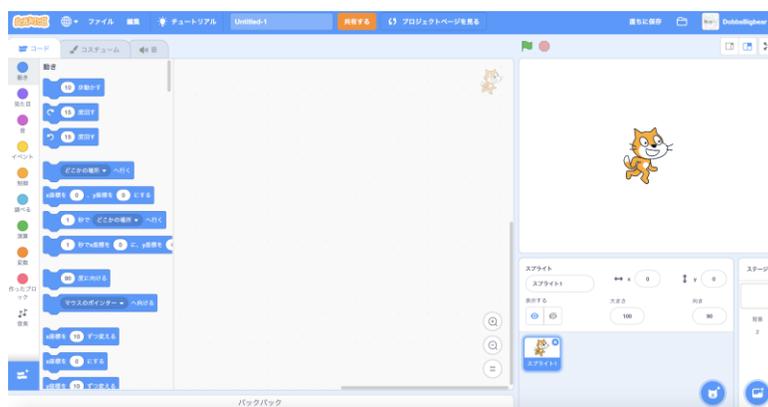
Scratchキャットにサプライズを起こせますか？

このアクティビティでは、Scratchで新しいプロジェクトを作成し、そしてScratchキャットにサプライズを起こすよう、色々なScratchブロックを試します。何を作成しますか？



ここからスタート

- ❑ Scratchウェブサイトを開きます。：
<http://scratch.mit.edu>
- ❑ 自分のアカウントでサインインします。
- ❑ ブラウザ上部の左にある“作る”タブをクリックして、新しいプロジェクトを作成します。
- ❑ さあ、見つけてみよう！ Scratchインタフェース [訳注1]の色々なパーツをクリックして、何が起こるか見てみます。
- ❑ 色々なScratchブロックで遊みましょう！ スクリプト・エリアにScratchブロックをドラッグ & ドロップします。各ブロックをクリックして何をするか確認したり、複数のブロックをスナップしてみます。 [訳注2]



訳注1: Scratch開発環境のユーザーインターフェイスのこと

訳注2: くっつけるブロックの凸（でっぱり）に、くっつけたいブロックの凹（へっこみ）を近づけると、複数のブロックがくっつきます。

Scratch スタジオ

 **推奨時間**
5-15分

目的

- このアクティビティを完了すると、生徒は、
- + スタジオにプロジェクトを追加できるようになります。
 - + 他のScratchプロジェクトにコメントを投稿できるようになります。

アクティビティの説明

- Scratchスタジオは、Scratchオンラインでプロジェクトを集めてまとめておくための方法の1つです。
このアクティビティでは、生徒が、「スタジオとは何か」、「どうしたらスタジオにプロジェクトを追加できるのか」を理解するのに役立ちます。
必要に応じて、生徒をガイドする「Scratchスタジオ」配布用プリントを用意します。
- まず、生徒をScratchウェブサイトにナビゲートし、自分のアカウントでサインインさせます。次にScratch Surpriseスタジオか、あらかじめ作成しておいたクラス・スタジオを見つけられるようガイドします。生徒に、自分のプログラムをスタジオに追加してもらいます。
続けてScratch Surpriseでやったことを友達と共有してもらいましょう。
- 生徒に、スタジオにある他の生徒のプロジェクトを調べて、特に興味を引いたプロジェクトや、感激したプロジェクトの中から2つ選んで、プロジェクト・ページにコメントを追加するよう促します。
適切で目的にあったフィードバックの仕方についてグループで討議するよう促します。
- 生徒に、デザイン日誌やグループ討議の中で振り返りを促す質問に回答させて創造的な探究を振り返ってもらいます。

リソース

- 「Scratchスタジオ」配布用プリント
- Scratch Surpriseスタジオ
<http://scratch.mit.edu/studios/460431>

振り返りを促す質問

- + Scratchスタジオの目的は何ですか？
- + 他のプロジェクトを見て、どんな興味や感動を見つけましたか？
- + どんな2つのコメントを共有しましたか？
- + どんなフィードバックが「良い」フィードバックですか？

作業の確認ポイント

- + 生徒は、自分のプロジェクトをうまくスタジオに追加したか？
- + 生徒は、適切に他の人の作品についてコメントしたか？

メモ

- + 生徒の作品を収集するスタジオを作成します。
Scratchアカウントを使ってクラスのScratchスタジオをスタートさせてから、生徒にスタジオへのリンクを教えてプロジェクトを“提出”してもらいます。
専用スタジオをひとつ作成してすべてのクラスプロジェクトを収集するか、スタジオ全体にアクティビティを配布して、生徒の進捗状況を追跡します。

追加メモ記入欄

- _____
- _____
- _____
- _____

Scratchスタジオ

サプライズ

Scratchスタジオにあなたのプロジェクトを追加する方法を学びましょう！

Scratchプロジェクトのコレクションをスタジオと呼びます。以下のステップに沿って、あなたのScratch SurpriseプログラムをScratch Surpriseスタジオに追加してください。



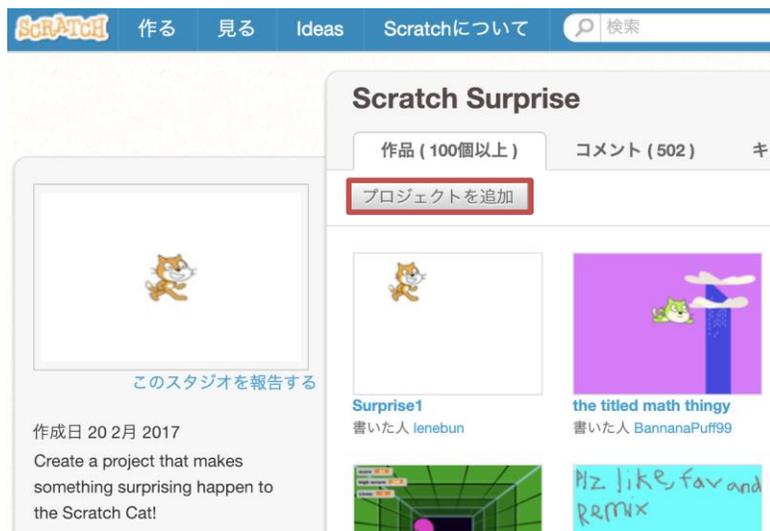
ここからスタート

- 次のリンクを開いてScratch Surpriseスタジオを開きます。

<http://scratch.mit.edu/studios/460431>

- 自分のアカウントでサインインします。
- “プロジェクトを追加”をクリックしてください。すると、ページの下に自分のプロジェクト、お気に入りのプロジェクト、最近、見たプロジェクトが表示されます。

- スタジオにプロジェクトを追加するには、左右の矢印を使って、自分のScratch Surpriseプロジェクトを見つけて “Add +”をクリックしてします。



Scratch Surprise

作品 (100個以上)

コメント (502)

プロジェクトを追加

批評グループ°

 **推奨時間**
15-30分

目的

このアクティビティを完了すると、生徒は、
+ 少人数の批評グループに分かれ、デザインのアイデアや進捗について、フィードバックしたり、フィードバックを受けたりします。

アクティビティの説明

- 生徒に、批評グループ（デザイナーの小グループ）のアイデアを紹介します。
批評グループとは、さらなる成長のためのフィードバックや提言を受けるために、アイデアやプロジェクトの進行状況を共有する人達のことです。
- 必要に応じて、フィードバック時に生徒をガイドする「批評グループ」配布用プリントを用意します。
- 生徒を3-4人の少人数の批評グループに分けます。
生徒に、批評グループで、アイデア、ドラフト、プロトタイプ（例えば、Scratchサブライズ・プロジェクト）を交互に共有し合ってもらいます。
- 批評グループのメンバーに、「レッド、イエロー、グリーン」の振り返りを促す質問に答えてもらうか、「批評グループ」配布用プリントを使って、生徒にフィードバックを集めてもらいます。生徒に、友達のコメント、フィードバック、提案を自分のデザイン日記に記録するよう促します。

リソース

- 「批評グループ」配布用プリント

振り返りを促す質問

- + ピンク：うまくいかないことは何ですか？
改善可能ですか？
- + イエロー：混乱していることは何ですか？
別の方法でできそうですか？
- + グリーン：うまくいっていることは何ですか？
プロジェクトについて本当に好きなことは何ですか？

作業の確認ポイント

- + 生徒全員が、自分の作品を共有してフィードバックをもらうチャンスがあったか？

メモ

- + 教育者であるあなたに刺激を与え、デザインの反復[訳注1]に対するフィードバックするために、専用の仲間のグループを持つことは有益なことです。
ユニット1-6の中で、生徒に批評グループとミーティングを続けるためのチャンスを与えてください！

自身に対する注釈

- _____
- _____
- _____
- _____

訳注1: 原文は、Design Iterations。さまざまなアクティビティには、デザインの反復が含まれており、フィードバックを元に、気づき、改善していく。

批評グループ

フィードバック対象: _____

プロジェクト名: _____

フィードバックくれた人	[レッド] うまくいかないことは何ですか？ 改善可能ですか？	[イエロー] 混乱していることは何ですか？ 別の方法でできそうですか？	[グリーン] うまくいっていることは何ですか？ プロジェクトについて本当に好きなことは何ですか？

プロジェクトに関するフィードバックを考えるのに役に立つヒント：

- + 明快さ：プロジェクトがやるはずだったことが、わかりましたか？
- + 機能性：プロジェクトにはどんな機能がありますか？期待どおりにプロジェクトは動作しますか？
- + アピール力：プロジェクトは、どんな風に魅力的ですか？
インタラクティブですか？独創的ですか？洗練されていますか？面白いですか？興味深いですか？
対話したとき、どう感じましたか？

ユニット1

探究しよう

がクリックされたとき

はい！ジェイミーだよ。 と 2 秒言う

私のこと、ちょっとだけ、教えてあげるね。 と 3 秒言う

私のことをもっと知るには、アイテムをクリックしてね！ と 3 秒言う

インデックス

内容



ダンスをプログラム	26
ステップ・バイ・ステップ	28
10ブロック	30
マイスタジオ	32
デバッグしよう！	34
自己紹介	36

ユニット1

概要

“ビッグ・アイディア”

何年もの経験がある多数の教育者でも、創造的コンピューティングを始める時に2つの疑問に苦しみます。学習者が始めるのに役立つ最善の方法は何か？教師として、自分は何を知る必要があるのか？これらの質問を考えるために、シーモア・パパート（Logoプログラミング言語を通してScratchの開発に大きな影響を与えた有名な数学者であり教育者）の執筆が役立ちます。

第1の疑問に関して、2つの極端な意見が取り上げられる傾向があります。学習者は、すべき事を教えられ、さらに高度に体系化された体験をする必要があります。一方で、学習者は、自分自身の考えのもとで自ら探究できるよう自由である必要があります。パパートは、若い学習者が自身の思考と学習を支持し、探究する者として行動すべきとの考えを支持しており、教師が教える事と学習する事のバランスをとるよう、勧めています。このガイド全体を通して、我々は、アクティビティの構成の分量に変化をつけて、バランスを取ろうとしています。

第2の疑問に関して、教育者は、時折、他人を助けられるほど、よくScratchを「知らない」と心配しています。我々は、Scratchを「知っている」ことが何を意味するのか、幅広い視点を持つよう勧めています。教師は、Scratchインターフェース^[訳注1]や学習者が遭遇するすべての問題を解決する方法についてすべてを知る必要はありません。パパートが指摘するように、教育者は、質問したり、問題を扱い易いように細かくブレークダウンするのを手助けして、認知のガイド役として貢献することができるのです。

訳注1: Scratch開発環境のユーザーインターフェイスのこと

学習の目的

生徒は：

- + インタラクティブなScratchプロジェクトを創作することで、Scratch環境の最初の探究の土台を作ります。
- + より多くのScratchブロックが紹介されます。
- + 順次処理の概念を理解できるようになります。
- + プロジェクトの創作中に、実験と反復を実践します。

子供と一緒に困惑しつつ、意外な事実に気づきました。「本当に修正方法がわからないの？」と彼は言いました。子供は、まだうまく説明できませんでしたが、明らかだったことは、彼と先生と一緒に研究プロジェクトに取り組んでいたということでした。この出来事は感動的です。ずっとコラボレーションがフィクションであると知りつつ、いつもこの子供は「一緒にやろう」という教師の目論見に乗っていたということです。発見は用意されるものでもなければ、発明は予定できるものでもありません。
(パパート著「Mind Storms」115頁、1980年)

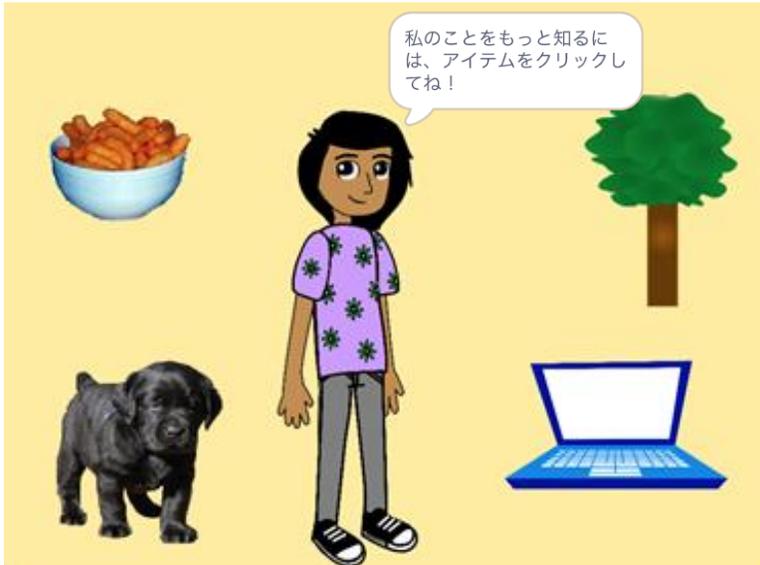
キーワード、概念、実践

- | | | |
|------------|-------------|-------------|
| + 実験と反復 | + 見た目 | + リミックス |
| + テストとデバッグ | + 音 | + インタラクティブ・ |
| + 順次処理 | + コスチューム | コラージュ |
| + スプライト | + バックドロップ | + ペアシェア |
| + 動き | + Tipsウィンドウ | |

メモ

- + オンラインでプロジェクトを保存したり、共有するために、生徒が既にScratchアカウントを持っているかを確認めます。
- + 生徒の作品にアクセスする方法を考えておきます。例えば、プロジェクトを収集するためのクラス・スタジオを作成して、生徒のプロジェクトのリンクをEメールで送信させたり、クラスのブログを始めます。

アドベンチャーを選択しよう



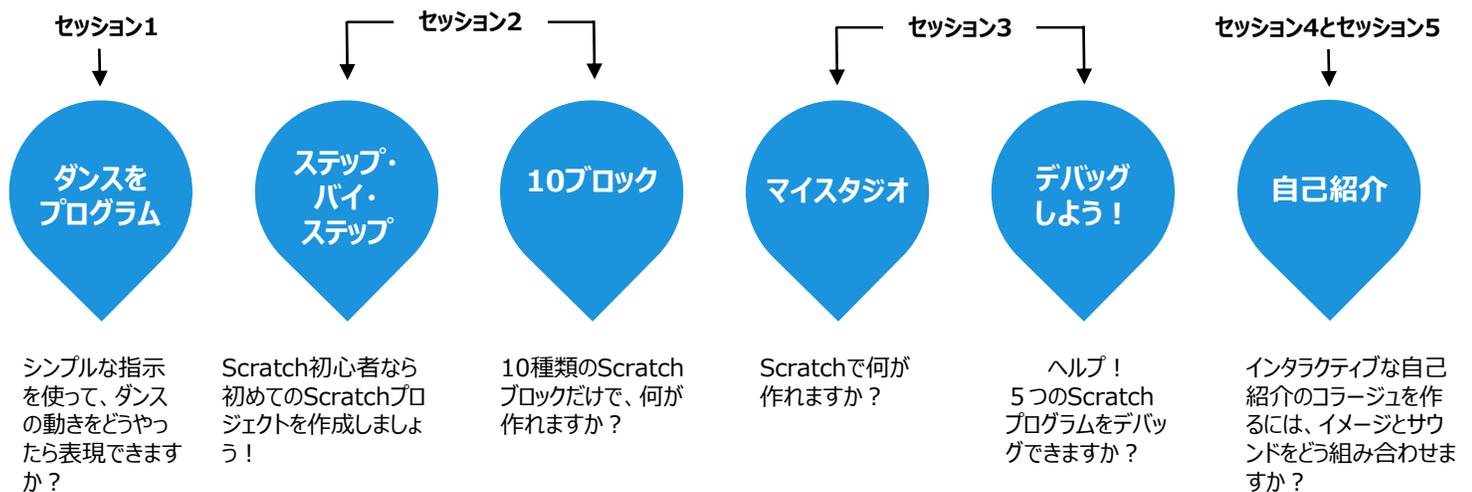
このユニットは、主要な概念である順次処理の探究に生徒を惹きつけるアクティビティの組み合わせ（体系化されたアクティビティとオープンエンドなアクティビティ）で構成されています。具体的には、順序付けられた手順を確認し、指示しています。これは、多くの場合、生徒にとって効果的な時間です。生徒らは、自分のアイデアをコンピュータのコードであるブロックに翻訳して、コンピュータに何をすべきかを命じているのです。

ステップ・バイ・ステップのチュートリアルから、制限されたブロック数でのプレイやデバッグの課題に至るまで、各アクティビティは、学習者が最後のセッションで「自己紹介」のプロジェクトを作成するのに必要なスキルを身につけるのに役立ちます。最後のプロジェクトでは、学習者は、スプライト、コスチューム、見た目、背景や音を探究し、体験して、Scratch上でパーソナライズされたインタラクティブ・コラージュ[訳注1]を作成します。

すべてのアクティビティを活用しても、生徒の具体的なニーズと興味に応えるものをいくつか選んでもかまいません。選択はあなた次第です。どこから始めたらいいかわからない場合は、アクティビティのために可能な順序を以下に示します。

訳注1: 双方向なやりとりができるデジタル表現された切り絵

可能なルート



ダンスを プログラム

 **推奨時間**
45-60分

目的

このアクティビティを完了すると、生徒は、
+ 一連のシンプルな指示で複雑なアクティビティを表現することを学びます。

アクティビティの説明

- 4人のコーチ役になってくれる人と、4人のダンサー役になってくれる人、計8人にボランティアを頼みます。コーチ役とダンサー役は2人一組のペアを作ります。必要に応じて、「ダンスをプログラム」のビデオを示すためにプロジェクターを用意します。[訳注1]
- コーチ役/ダンサー役のペア、それぞれに対して：
 1. ディスプレイが見えないようダンサー役を遠ざけて、コーチ役（と残り生徒）にはディスプレイの方を向いてもらいます。
 2. コーチ役と残り生徒にはビデオを見せますが、ダンサー役には見せません。
 3. コーチ役に、ビデオに表示されている一連のダンスの動き方を言葉だけでペアのダンサー役に指示するよう頼みます。
 4. ダンサー役は指示どおりの順番でダンスの動きをしてみせます。[訳注1]
- このアクティビティを使って、一連の指示を具体的に説明して、順序処理の重要性についてディスカッションを開始します。生徒に、各自のデザイン日誌に振り返りを書かせたり、別のコーチ役/ダンサー役のペアやオブザーバー[訳注2]を招待してグループ討議をして、彼らの考えを共有してもらいます。

訳注1: 理解し易くするため、意訳および補記。

原文のBossyをコーチ役と、Bossをダンサー役をと表記。

訳注2: 感想や意見を発言する第三者のこと。コーチ役/ダンサー役以外の人。

メモ

- + これは、コンピュータを使わないで可能なアクティビティの1つです。コンピューターから離れる事は、新鮮な観点に立たせ、コンピュータ処理の概念、実践、考え方の新たな理解をサポートします。
- + 生徒にダンスの1つをステップ・バイ・ステップで書き出させましょう。プログラミング上、これは「疑似コード」と呼ばれます。

リソース

- プロジェクター(任意)
- 「ダンスをプログラム」のビデオ
 - ・ <http://vimeo.com/28612347>
 - ・ <http://vimeo.com/28612585>
 - ・ <http://vimeo.com/28612800>
 - ・ <http://vimeo.com/28612970>

振り返りを促す質問

- + コーチ役をやってみて何が簡単でしたか/何が難しかったですか？
- + ダンサー役をやってみて何が簡単でしたか/何が難しかったですか？
- + 観察していて、何が簡単でしたか/何が難しかったですか？
- + このアクティビティは、Scratchで行っている事にどのように関連していますか？

作業の確認ポイント

- + 生徒は、指示する時、順次処理の何が重要か説明できるか。

追加メモ記入欄

- _____
- _____
- _____
- _____



ステップ・バイ・ステップ

▶ 推奨時間
15-30分

アクティビティの説明

- 生徒がScratchアカウントにサインインするのを手助けして、Scratchウェブサイトの上部の“作る”ボタンをクリックし、プロジェクト・エディタを開くようガイドします。必要に応じて、生徒をガイドする「ステップ・バイ・ステップ」配布用プリントとScratchカードを用意します。
- 生徒に、Tipsウィンドウ（画面右上の“？”をクリック）を開いて、ステップ・バイ・ステップ”タブにあるチュートリアル”Scratchをはじめよう”に従って、ダンスするネコのプログラムを作成してもらいます。プロジェクトを独自のものにするために、他のブロックを加えて、動き、スプライト、見た目、コスチューム、音、背景を実験してみるよう促します。
- 生徒に、初めてのScratch作品をお互いで共有させます。必要に応じて、生徒がStep-By-Stepスタジオやクラス・スタジオに自分のプロジェクトを共有したり、追加するようガイドします。
- 生徒に、デザイン日誌やグループ討議の中で振り返りを促す質問に回答させてデザインのプロセスを振り返ってもらいます。

目的

- このアクティビティを完了すると、生徒は、
- + ステップ・バイ・ステップのチュートリアルに従ってScratchでダンスするネコを作成します。
 - + 実験したり反復しながら、プログラムを作り上げる体験をします。

リソース

- 「ステップ・バイ・ステップ」配布用プリント
- Step-by-Stepスタジオ：
<http://scratch.mit.edu/studios/475476>
- Scratchカード：
<http://scratch.mit.edu/info/cards/>

振り返りを促す質問

- + アクティビティについて驚いた事は何ですか？
- + ステップ・バイ・ステップの手順どおりに進めてみて、どう感じましたか？
- + 最もクリエイティブと感じるのはどんな時ですか？

作業の確認ポイント

- + 生徒は、Scratchを開いてTipsのウィンドウを見つけたか？
- + 生徒は、ダンスするネコを作成できたか？
- + 生徒は、プロジェクトを保存し、共有できたか？

メモ

- + 生徒がアカウントをひとつも持っていない場合は、ユニット0の“Scratchアカウント”のアクティビティを使用して、Scratchアカウントを作成するのを手助けします。初めて作ったScratchプロジェクトを保存して友達や家族と共有できるようになります。
- + 生徒にユニット0の“Scratchスタジオ”のアクティビティや配布用プリントを使って、スタジオへのプロジェクト追加方法を復習してもらいます。

追加メモ記入欄

- _____
- _____
- _____
- _____

ステップ・バイ・ステップ

Scratchは初めて？
初めてのScratchプロジェクトを作成
しよう！

このアクティビティでは、“チュートリアル”（画面上左より）をクリックして、“さあ、始めましょう”のチュートリアルに従って、動くネコを作成します。このステップが完了したら、他のScratchブロックを加えることによって、プロジェクトを独自のものにする実験をします。

ここからスタート

- “チュートリアル”（画面上左より）をクリックして“さあ、始めましょう”のビデオとチュートリアルを見ましょう。
- 以下のプロジェクトのステップに従ってプログラムを作成します。
<https://scratch.mit.edu/projects/281820501/>
- ブロックをいくつか追加してみます。
- プログラムを自分好みにする体験をしてみよう。



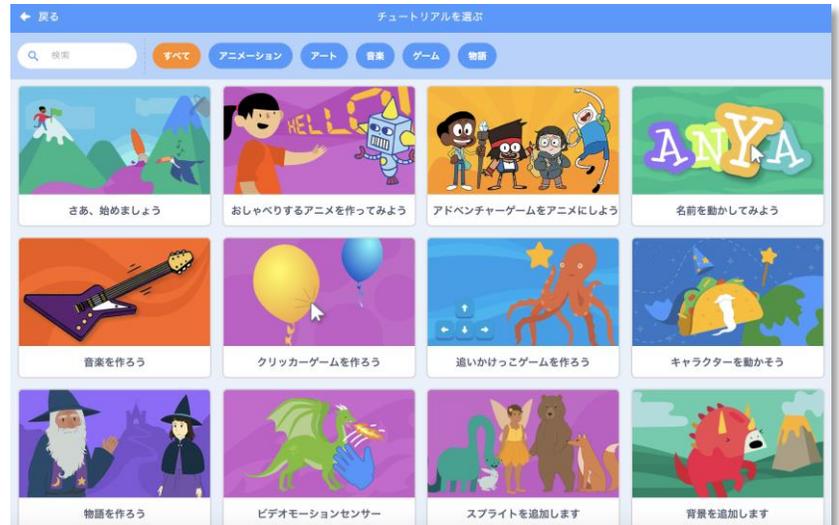
15 度回す

15 度回す

1 秒でx座標を 0 に、y座標を 0 に変える

大きさを 10 ずつ変える

どのブロックで実験したいですか？



試してみよう

- 自分自身の声を録音してみます。
- 異なる背景を作成します。
- ダンスするスプライトをもっと増やして、プロジェクトをダンスパーティにしてみましょう。
- スプライト用の新しいコスチュームをデザインしてみます。

完成した？

- + Step-by-Stepスタジオにプロジェクトを追加しましょう。
<http://scratch.mit.edu/studios/475476>
- + あなた自身にもっとチャレンジしましょう！
新しいブロック、音、動きを加えて遊びましょう。
- + 友達を手助けしましょう！
- + いくつかの新しいブロックを選んで実験します。新しいブロックを試してみましょう！

10ブロック

 **推奨時間**
15-30分

目的

このアクティビティを完了すると、生徒は、
+ 10種類のブロックだけしか使えないという制約の下でプロジェクトを作成します。

アクティビティの説明

- 生徒がScratchアカウントにサインインするのを手助けして、Scratchウェブサイトの上部の“作る”ボタンをクリックし、新しいプロジェクトを作成します。
必要に応じて、アクティビティ中に生徒をガイドする「10ブロック」配布用プリントを用意します。
- 生徒に時間を与え、10種類のブロックだけを使ったプロジェクトを作成してもらいます。:
x, y座標を〜にする、〜秒でx, y座標を〜に変える、〜と〜秒言う、表示する、隠す、大きさを〜%にする、終わるまで〜の音を鳴らす、〜秒待つ、このスプライトがクリックされたとき、〜回繰り返す、の10種類です。

生徒に、プロジェクトの中で少なくとも1回は各ブロックを使うよう伝え、色々なスプライト、コスチューム、背景を試すよう促します。
- 生徒に、彼らの批評グループの中でプロジェクトの共有を促します（ユニット0の“批評グループ”のアクティビティを参照）。必要に応じて、生徒に10Blockスタジオまたはクラス・スタジオにプロジェクトを追加してもらいます。
- 生徒に、デザイン日誌やグループ討議の中で振り返りを促す質問に回答させて自分のデザインのプロセスを振り返ってもらいます。

リソース

- 「10ブロック」配布用プリント
- 10Blocksスタジオ
<http://scratch.mit.edu/studios/475480>

振り返りを促す質問

- + 10種類のブロックしか使えないせいで、難しかったことは何ですか？
- + 10種類のブロックしか使えないおかげで、簡単だったことは何ですか？
- + どう違うと思いましたか？

作業の確認ポイント

- + プロジェクトには10種類のブロックがすべて含まれているか？
- + 制約付きで作成するアイデアに対して、生徒はどんな反応しているか？
この生徒がどう学んだかについて、何に気付いたか？

メモ

- + わずか10種類のブロックで、こんなに多くのことができるのは驚きです！
この機会に、数人の生徒にクラスの前でプロジェクトを発表してもらったり、10Blocksスタジオの他のプロジェクトを調べることで、色々なアイデアを促し、創造性を称賛します。

追加メモ記入欄

- _____
- _____
- _____
- _____

10ブロック

10種類のブロックだけで何が作成できますか？

これらの10種類のブロックだけを使って、プロジェクトを作成してください。
何回使ってもいいですが、少なくとも1回は使用してください。

ここからスタート

- 各ブロックを実験して、アイデアをテストします。
- さまざまな方法でブロックをうまく組み合わせます。
- 繰り返しましょう！

行き詰まりましたか？

大丈夫！ 次のことを試してみてください...

- 色々なブロックの組み合わせを試して、アイデアをテストします。興味のあるものが見つかるまで、ブロックをうまく組み合わせてみましょう！
- 隣の友達とアイデアをブレインストーミングしましょう！
- 他のプロジェクトを検索して、他の人が何をしているのかを参照します。ひらめきましたか？他のプロジェクト事例を参照してみるの、行き詰まった時に役立つ素晴らしいアプローチです！

1 秒でx座標を 0 に、y座標を 0 に変える

x座標を 0 、y座標を 0 にする

こんにちは! と 2 秒言う

表示する

隠す

大きさを 100 %にする

終わるまで Meow の音を鳴らす

このスプライトがクリックされたとき

1 秒待つ

10 回繰り返す

完成した？

- + 10 Blocksスタジオにあなたのプロジェクトを追加しましょう。
<http://scratch.mit.edu/studios/475480>
- + 色々なスプライト、コスチューム、背景で遊びましょう。
- + あなた自身にもっとチャレンジしましょう！これらの10ブロックで作成可能なプロジェクトをいくつ作成できるか考察してみてください。
- + パートナーとプロジェクトを交換し、お互いの作品をリミックスしましょう。

マイスタジオ

 **推奨時間**
15-30分

目的

このアクティビティを完了すると、生徒は、
+ Scratchウェブサイト上の何百万ものプロジェクトのいくつかを検索して、Scratchを使ったさまざまなクリエイティブな可能性を探ります。
+ Scratchスタジオで3つ以上のScratchプロジェクトのコレクションをキュレーション^[訳注1]します。

アクティビティの説明

- 必要に応じて、新しいスタジオを作成する方法を説明したり、生徒をガイドする「マイスタジオ」配布用プリントを用意します。
- 必要に応じて、提供されたリンクを使って発想の源となるサンプルスタジオを示します。生徒に10分間与え、Scratchのホームページで共有済みのScratchプロジェクトを閲覧したり、“見る”のページを使って興味のあるプログラムを検索してもらいます。
- 生徒に情報を与え、自分のプロジェクトに参考になりそうなScratchプロジェクトを3つ以上特定してもらいます。生徒が“私の作品”のページで新しいスタジオを作成し、感動したプロジェクトをスタジオに追加できるようガイドします。
- 生徒に、感動的なプログラムを見つけ出すアプローチを共有するよう促します。ペアシェアがお勧めです。生徒にペアになってもらい、スタジオを共有しあい、検索の戦略をディスカッションしてもらいます。
- 生徒に、デザイン日誌やグループ討議の中で振り返りを促す質問に回答させて発見の過程を振り返ってもらいます。

リソース

- 「マイスタジオ」配布用プリント
- サンプルスタジオ
<http://scratch.mit.edu/studios/211580>
<http://scratch.mit.edu/studios/138296>
<http://scratch.mit.edu/studios/138297>
<http://scratch.mit.edu/studios/138298>

振り返りを促す質問

- + 面白いプロジェクトを見つけるために、どんな方法で検索しましたか？（例：検索キー、カテゴリ）
- + 各サンプルプロジェクトは、それぞれどんな風に将来の作品に役立つでしょうか？
- + インスピレーションを受けた情報源に謝辞を示すことは重要なことです。サンプルプロジェクトから受けたインスピレーションに対して、どんな謝辞を示せましたか？

作業の確認ポイント

- + スタジオに3つ以上、プロジェクトがあるか？
- + 生徒のデザインに関する興味について、これらのプロジェクトから何を気づいたか？

メモ

- + 生徒がScratchの個人アカウントを持っていない場合は、生徒がキュレートできるクラス・スタジオを作成します。
- + さまざまなスタジオを作成可能です。将来の創作に組み込むための技術や資産^[訳注2]を含むプログラムを作成したり集めるために、生徒はテーマやトピックが欲しいものと類似しているScratchプロジェクトを収集できます。

訳注1: お気に入りのプロジェクトを集めて公開し、共有すること
 訳注2: 技術にはコード、定義、資産にはスプライト、コスチューム、サウンドなどがある。

追加メモ記入欄

- _____
- _____
- _____
- _____

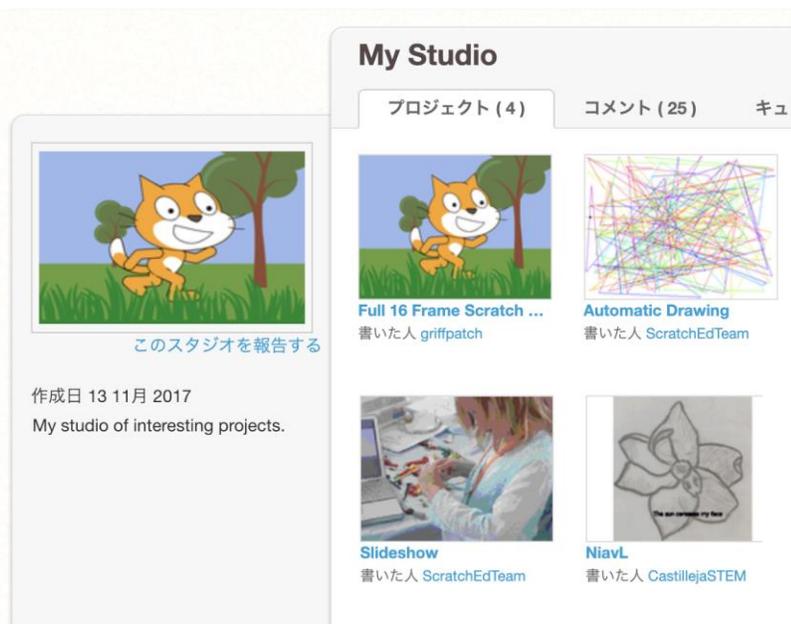
マイスタジオ

Scratchで何ができますか？

Scratchウェブサイト上の何百万ものプロジェクトのいくつかを検索して、Scratchを使ったさまざまなクリエイティブな可能性を探ります。さあ、Scratchスタジオでお気に入りのコレクションを始めましょう。

ここからスタート

- ❑ Scratchのホームページでプロジェクトを検索するか、“見る”をクリックして特定のタイプのプロジェクトを検索します。
- ❑ “私の作品”のページで“新しいスタジオ”を作成します。
- ❑ 3つ以上の感動的なプロジェクトをスタジオに加えてください。



試してみよう

- ❑ 検索バーを使って、あなたの興味に関連するプロジェクトを探し出します。
- ❑ “見る”のページのアニメーション、アート、ゲーム、音楽、物語、チュートリアルのカテゴリを検索しましょう。
- ❑ アイディアを求めて、ホームページ[訳注1]の「注目のスタジオ」を見てみましょう。

完成した？

- + あなた自身にもっとチャレンジしてください！
- + より多くのScratchプロジェクトを調べるほど、Scratchで何が達成できるのかをより多く知ることができます！
- + あなたが興味深いと思うScratcher達が作成したスタジオを見つけましょう！
- + 興味深いプロジェクトを見つけるために友達がどのような方法をとったのか、質問してみましょう。
- + あなたの新しく作ったスタジオを友達に公開しましょう！

[訳注1]ホームページは、オレンジのScratchボタン(※)をクリックする。

デバッグしよう！

 **推奨時間**
15-30分

目的

- このアクティビティを完了すると、生徒は、
- + 5つの課題に対する問題を調査してデバッグし、解決策を見つけ出します。
 - + テストとデバッグの実践を通してさまざまな概念（順次処理を含む）を探究します。
 - + プロジェクトをデバッグするための戦略リストを作成します。

アクティビティの説明

- 必要に応じて、アクティビティ中で、生徒をガイドするユニット1「デバッグしよう！」配布用プリントを用意します。
- ユニット1「デバッグしよう！」配布用プリントに記述されているプロジェクトへのリンクをたどってみせ、生徒が、Unit 1 Debug It!スタジオのプログラムを開くのをガイドします。生徒に、“中を見る”ボタンをクリックするよう促して、バグのあるプログラムを調べて、コードをあちこち変えてみて、可能性のある解決方法を試してもらいます。
- 生徒に時間を与え、それぞれの「デバッグしよう！」の問題をテストし、デバッグしてもらいます。必要に応じて、生徒にScratchのリミックス機能を使って、バグを解決し修正済みのプログラムを保存してもらいます。
- 生徒に、デザイン日誌やグループ討議の中で振り返りを促す質問に回答させてテストやデバッグの体験を振り返ってもらいます。
- 生徒の問題発見と問題解決のアプローチを集めて、クラス全体のデバッグ戦略リストを作成します。

リソース

- ユニット1「デバッグしよう！」配布用プリント
- Unit 1 Debug It!スタジオ
<http://scratch.mit.edu/studios/475483>

振り返りを促す質問

- + 何が問題でしたか？
- + どうやって問題を特定しましたか？
- + どうやって問題を解決しましたか？
- + 他の人は、問題を解決するのに別のアプローチを取りましたか？

作業の確認ポイント

- + 生徒は、バグをすべて解決できたか？
そうでない場合は、解決できなかったプログラムで示されている概念をどうやったら明確にできるか？
- + 生徒は、どんなテストとデバッグの戦略を採用したのか？

メモ

- + このアクティビティはグループでうまくいきます！
2~4人のチームで作業する生徒に、集団で問題解決し、デバッグの戦略を共有するよう促します。
- + テストとデバッグはおそらくプログラムの最も一般的な仕事です。物事は計画どおりに動くことはめったにないので、一連のテストとデバッグの戦略を身に付ける事は、どんなコンピュータ・クリエイターにとっても有益な事です。

追加メモ記入欄

- _____
- _____
- _____
- _____

デバッグしよう！

ヘルプ・ミー！

5つのScratchプログラムをデバッグできますか？

このアクティビティでは、「デバッグしよう！」の各問題について、何がうまくいっていないのかを調査し、解決策を見つけ出します。

ここからスタート

- ユニット 1 用のDebug It!スタジオを開きます。
<http://scratch.mit.edu/studios/475483>
- スタジオにある 5 つのデバッグの問題をテストし、デバッグします。
- 解決策を書き出したり、あなたの解決策を使ってバグのあるプログラムをリミックスしてみましょう。

行き詰まりましたか？

大丈夫！ 次のことを試してみてください...

- プログラム中で起こりうるバグを一覧にします。
- あなたの作業の経過を追ってみましょう！
この一覧は、すでに試したことを思い出すのに役立ち、次に試すべきことを示してくれます。
- 役立つものを発見するまで、友達と気づいた問題点と問題解決のアプローチを共有し、比較してみましょう！

□ デバッグしよう！ 1.1

<http://scratch.mit.edu/projects/10437040>

緑の旗がクリックされた時、Gobo[訳注1]とScratchキヤットが両方ともダンスを開始するはずですが、Scratchキヤットしかダンスを始めません。どうプログラムを修正しますか？

□ デバッグしよう！ 1.2

<http://scratch.mit.edu/projects/10437249>

緑の旗がクリックされた時、Scratchキヤットは、左側にいると言ってステージの左側からスタートし、ステージの右サイドにゆっくり移動し、右サイドにいると言うはずですが、最初に緑の旗がクリックされた時は正常に動作しますが、2度目は正常に動作しません。どうプログラムを修正しますか？

□ デバッグしよう！ 1.3

<http://scratch.mit.edu/projects/10437366>

スペースキーを押した時、Scratchキヤットが回転するはずですが、スペースキーを押しても、何も起こりません！どうプログラムを修正しますか？

□ デバッグしよう！ 1.4

<http://scratch.mit.edu/projects/10437439>

Scratchキヤットがクリックされた時、ステージを左右に行ったりきたりするはずですが、Scratchキヤットが反転せず、さかさまで歩きます。どうプログラムを修正しますか？

□ デバッグしよう！ 1.5

<http://scratch.mit.edu/projects/10437476>

緑の旗がクリックされた時、Scratchキヤットが、音を鳴らしながら、フキダシで“Meow, meow, meow!”と言うはずなのに、音の前にフキダシが出てきます。また、Scratchキヤットは、“Meow”の音を1度だけしか鳴らしません。どうプログラムを修正しますか？

終わった？

- + テストとデバッグの実践に関して、パートナーと話し合しましょう。あなたの戦略との類似点と相違点を書き出してみましょう。
- + スクリプト内のブロックを右クリックしてコードにコメントを追加します。コメントを追加する事は、友達がプログラムの色々な部分を理解するのに役立ちます！
- + 友達を手助けしましょう！

自己紹介

 **推奨時間**
45-60分

目的

このアクティビティを完了すると、生徒は、

- + 多数のScratchブロックに詳しくなります。
- + オープンエンド[訳注1]なScratchプロジェクトを創作できるようになります。それは生徒が興味があることをインタラクティブなデジタル表現することに他なりません。

アクティビティの説明

- 生徒に、スプライトをクリックすると自己紹介するインタラクティブ・コラージュ[訳注2]のコンセプトを紹介します。必要に応じて、About Meスタジオのインタラクティブなプロジェクトのサンプルを示します。
- 生徒に、Scratchアカウントにサインインして新しいプロジェクトを開いてもらいます。必要に応じて、ガイダンスのための「自己紹介」配布用プリントやScratchカードを用意します。生徒に時間を与え、インタラクティブなコラージュを作成してもらいます。プログラムを完成させるために実験と反復の実践を勧めましょう。
- 生徒が作業中の作品を他の人と共有できるようにします。ペアシェアがお勧めです。ペアシェアとは、ペアになってプロジェクトを共有し合うことで、ディスカッションを促してくれます。必要に応じて、生徒に、About Meスタジオまたはクラス・スタジオに自分のプロジェクトを追加してもらいます。
- 生徒に、デザイン日誌やグループ討議の中で振り返りを促す質問に回答させてデザインのプロセスを振り返ってもらいます。

訳注1: 生徒の興味に任せ終わりのないプロジェクト
訳注2: 双方向なやりとりができるデジタル表現された切り絵

メモ

- + サンプルプロジェクトから、感動と威圧[訳注3]を同時にもらうことができます。クリエイティブな空間を広げてそれを限定します。[訳注4]幅広い創作を奨励しましょう。多様性は素晴らしいことです！
- + 生徒は、プロジェクトに画像を取り込むためのカメラやウェブカメラを使って、さらにプロジェクトをパーソナライズできます。

訳注3: 威圧とは、「サンプルを見て、こんな事ができるんだ」と自分自身にプレッシャーを感じる」と解釈。

訳注4: 類似の例：デザインプリントは、アイデアを拡散して、絞り込む手法

リソース

- 「自己紹介」配布用プリント
- About Meスタジオ
<http://scratch.mit.edu/studios/475470>
- Scratchカード
<http://scratch.mit.edu/help/cards>

振り返りを促す質問

- + 最も誇りに思っているのは何ですか？ 理由は？
- + あなたは何に行き詰まりましたか？ どうやって抜け出しましたか？
- + 次に何をしたいですか？
- + 友達「自己紹介」プロジェクトを見て、あなたは何に気づきましたか？

作業の確認ポイント

- + プロジェクトは、スプライト、コスチューム、見た目、背景、音を創造的に使っているか？
- + プロジェクトはインタラクティブか？ 利用者はプロジェクト内のさまざまな要素とやり取りできるか？

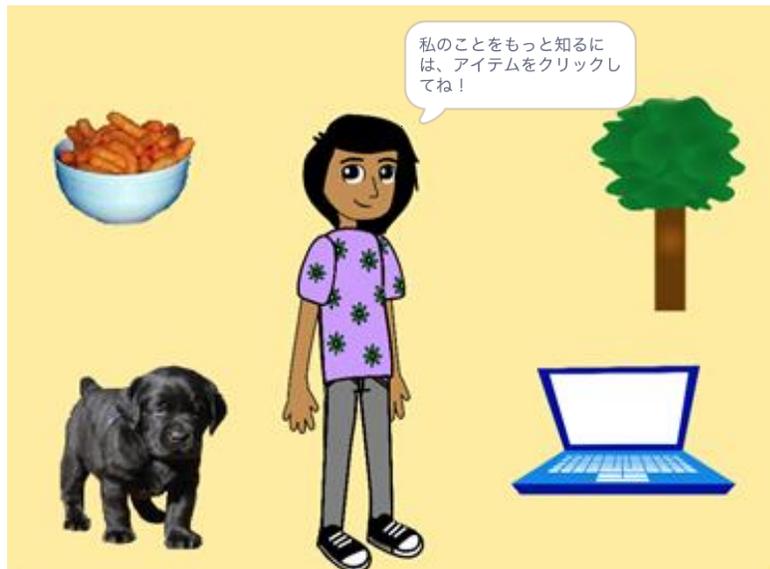
追加メモ記入欄

- _____
- _____
- _____
- _____

自己紹介

自己紹介用のインタラクティブなコラージュを作るために、興味のある画像と音を、どう組み合わせますか？

インタラクティブなScratchプロジェクトを創作するために、スプライト、コスチューム、背景、見た目、音を実験してみましょう。プロジェクトは、他の人が自分のこと（あなた自身、アイデア、アクティビティ、気に入っている人々）を良く知るのに役立ちます。



ここからスタート

- スプライトを作成します。
- インタラクティブにしてください。
- 繰り返そう！

クリックやキー入力などに反応させるスクリプトを追加して、あなたのスプライトをインタラクティブにしましょう！



試してみる

- スプライトの見た目を変えるには、複数のコスチュームを使います。
- 色々な背景を作成します。
- プロジェクトに音を加えてみましょう。
- あなたのコラージュに動きを加えてみましょう。

使用するブロック

完成した？



- + About Meスタジオにプロジェクトを追加しましょう。
<http://scratch.mit.edu/studios/475470>
- + あなた自身にもっとチャレンジしましょう！新しいブロック、音、動きを追加して遊びましょう！
- + 友達を手助けしましょう！

ユニット2 アニメーション

Turn up the music!



インデックス

内容



スクリプトを演じる	42
バンドを作ろう	44
色付きの四角と丸!	46
生きてる!	48
デバッグしよう!	50
ミュージック・ビデオ	52

ユニット2

概要

“ビッグ・アイディア”

子供たちは、Scratchオンライン・コミュニティで600万以上のプロジェクト（アニメーション、物語、ゲーム、他）を共有しています。

このガイドの我々のゴールの1つは、この莫大な創作品の多様性を振り返ることです。アクティビティの中で、我々はパーソナライズedの機会をサポートしつつ、「正しい」答えが1つしかないような課題を提示するのを避けています。我々は、アクティビティ全体で学習者をさまざまなジャンルに取り組みさせます。このユニットでは、アニメーション、アート、ミュージックにより深く関わってこの創造的な多様性の探究をスタートさせます。

しばしば、Scratchの創造的な多様性は、学習者によって強調されています。学習者から寄せられた質問をいくつか引用します。「友人の一人にScratchとは何かを説明しなければならない場合、どう説明しますか？」

みんなのイメージネーションを探究できるのが、プログラムなんだ。みんなが望むものだったら、何でもできるんだ。何でも作れるよ。作れる事に、制限なんて無いのさ。みんな、自分のものをデザインする。一度始めると、学ぶにつれてやめたくなくなるでしょ。もっと可能性があると思うと、さらにできることが出てきて、学んだばかりの事をもっと広げたいよな。

ブラッドリー、12歳

創造的に自分自身を表現することは本当に素晴らしいことだよ。君がこのプロジェクトを作れるという事じゃなくて、作れる事すべてがプロジェクトなのさ。

ネヴィン、9歳

あなた自身で創造性を表現できるなんて、本当に素晴らしいことよ。どんなこともできるわ。ビデオゲーム、ミュージック、アート、ビデオ、何でも作ることができるの。可能性はエンドレスで無制限なの、本当よ。

リンジー、12歳

そうですね、僕は、みんなが、Scratchでちよっぴり、何でもやれることが好き。みんなが望むものは何でもできるって事だよ。みんなが望むようにクリエイティブになれるんだよ。

アーロン、10歳

学習の目的

- 生徒は、
- + コンピュータ処理で考慮されている概念（繰り返し処理、イベント処理、並列処理）を紹介されます。
 - + 順次処理の概念をより深く理解します。
 - + “イベント”、“制御”、“音”、“見た目”のカテゴリーのブロックを新たに体験します。
 - + アートをテーマにしたさまざまなScratchプログラムを調べます。
 - + 動きのあるミュージック・ビデオのプロジェクトを制作します。

キーワード、概念、実践

- + 繰り返し処理 + スクリプト + アニメーション
- + イベント処理 + プレゼンテーション・モード + ギャラリーウォーク
- + 並列処理
- + 制御 + ビットマップ
- + メッセージ送信 + ベクター

メモ

- + このユニットの多くのアクティビティには、音とミュージックの要素が含まれています。生徒のために、ヘッドフォンをすぐに利用できるようにすることをお勧めします。

アドベンチャーを選択しよう



Scratchのプログラミングは、演劇を監督するのに似ています。演劇では、Scratchと同じように、登場人物（スクラッチ用語ではスプライト）、コスチューム、スクリプト、ステージがあります。

Scratchプログラミングは、“イベント”と呼ばれる「きっかけ」を利用しています。それは、プロジェクト中でいつ事を起こすかの合図を意味しています。例えば、プロジェクトの起動は、“緑の旗をクリックしたとき”を、スプライトの動きのトリガーは、“このスプライトをクリックされたとき”を、内部的な「きっかけ」[訳注1]をスプライトや背景に送るには、“～を送る”を使います。

演劇のメタファーに影響を受けている、このユニットのアートをテーマにしたアクティビティは、生徒がコンピュータ処理の概念（繰り返し処理、イベント処理、並列処理）を探究するのに役立つようデザインされています。結果としてパーソナライズされたミュージック・ビデオのデザインして結実します。

訳注1: 意識。原文は、silent cue。

外部から働きかけではなく、内部的な働きかけのため。

可能なルート



スクリプトを演じる

 **推奨時間**
30-45分

アクティビティの説明

- 必要に応じて、ブロックとスクリプトを演じるのを見せるため、Scratchとコンピュータに接続しているプロジェクターを立ち上げます。
- 2人のボランティアを頼みます。
- 2人のボランティアに、一連の指示を演じるよう促します。Scratchインターフェイス[訳注2]か、印刷済みの物理的なScratchブロックを使って、ボランティアを「プログラミング」します。
 - 1人に、1つのこと（「部屋の散歩」など）をしてもらいます。
 - その人に、「リセット」してもらいます。
 - その人に、同時に2つのこと（「部屋の散歩」と「おしゃべり」など）をしてもらいます。
 - 2人目を加えて、2人目に（それぞれで）同時に1つのタスク（「おしゃべり」など）を実行してもらいます。
 - 2人目の人に依存したタスク（「おしゃべり」の代わりに、1人目の人に「返事する」など）をしてもらいます。
- 右の「振り返りを促す質問」を使って、グループとしての経験を振り返って、イベント処理と並列処理の概念について、ディスカッションします。

目的

- このアクティビティを完了すると、生徒は、
- + パフォーマンス[訳注1]を通して、イベント処理と（別のことを引き起こすこと）と並列処理（同時に起こること）の概念が紹介されます。
 - + Scratchにどんなイベント処理があり、どう動くのかを説明できるようになります。
 - + Scratchにどんな並列処理があり、どう動くかを説明できるようになります。

リソース

- プロジェクター（オプション）
- 物理的なScratchブロック（オプション）

振り返りを促す質問

- + アクションを起こすには、他にどんな方法がありますか？
- + Scratch上のイベント処理のメカニズムとは何ですか？
- + どんなことが同時に起こっていましたか？
- + Scratchで並列処理を可能にするメカニズムとは何ですか？

作業の確認ポイント

- + 生徒は、Scratchにどんなイベント処理や並列処理があって、どう動くのか説明できるか？

訳注1: スプライトの動きを演じること。

生徒はScratchのブロックを台本に、Scratchの動きを演じる。

訳注2: Scratchのユーザインターフェイスで表示されるブロックを拡大してプロジェクターに投影して支持を出す。

メモ

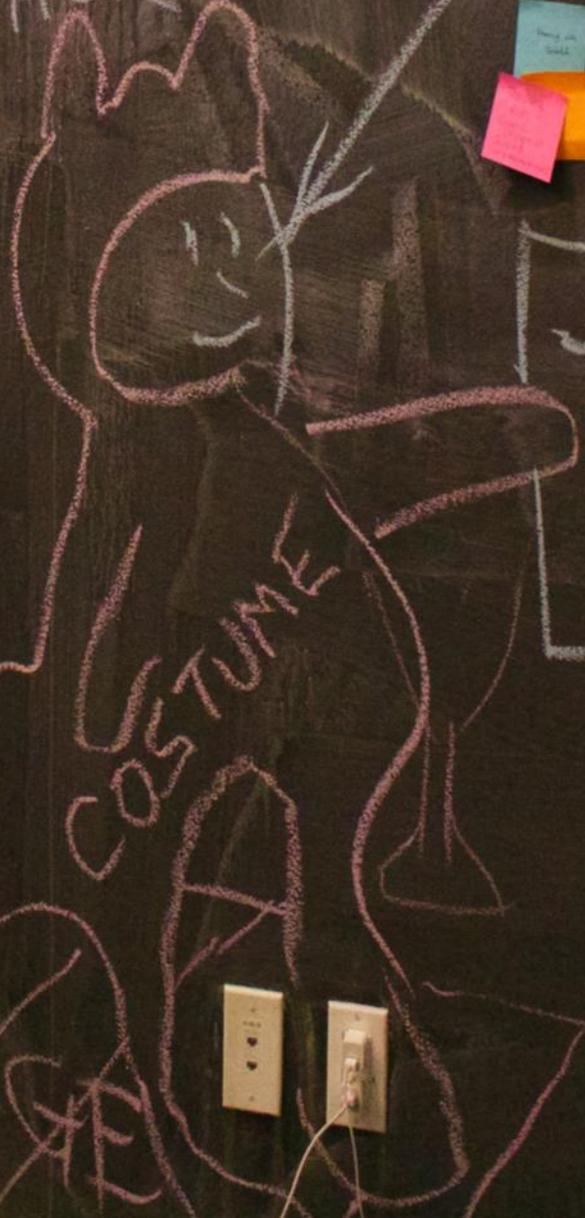
- + このアクティビティでは、「リセット」という概念が強調されています。これは、よくScratcher達が始めるとすぐに苦戦することです。特定の場所や特定の見た目などで物事を開始させたい場合、これらのセットアップ手順のプログラミングにあらゆる点で自分に責任があるということを生徒は理解する必要があります。
- + このアクティビティは、「～を送る」と「～を受け取ったとき」ブロックの組み合わせを説明するのに役立ちます。

追加メモ記入欄

- _____
- _____
- _____
- _____

SCRATCH'S
CENTRAL
THEATRICAL
METAPHOR

SPRITE



SCRATCH

~~STAGE~~

A dark wooden wall covered in various items: several colorful sticky notes (yellow, pink, blue, orange) scattered across the surface; a large chalk drawing of Scratch the mouse character in a costume; the words 'SCRATCH'S CENTRAL THEATRICAL METAPHOR' and 'SPRITE' written in white chalk; a rectangular box containing the word 'SCRATCH' written vertically; and the word 'COSTUME' written vertically on the character's body. At the bottom, there are two electrical outlets with a white cord plugged into the right one. A small wooden box with a pink object inside is on the right side. A small heart is drawn in the bottom right corner.

バンドを作ろう

 **推奨時間**
30-45分

目的

- このアクティビティを完了すると、生徒は、
- + インタラクティブなスプライトと面白いサウンドを組み合わせたプログラムを作成します。
 - + 順次処理、繰り返し処理、イベント処理、並列処理を使いこなす能力を身に付けます。
 - + プロジェクトの創作品を作り上げる際、実験と反復を実践します。

アクティビティの説明

- 必要に応じて、Build-a-Bandスタジオのサンプルプロジェクトを見せたり、生徒をガイドする「バンドを作ろう」配布用プリントを用意します。
- 生徒に時間を与え、スプライトに音を組み合わせ、インタラクティブな楽器を創作してもらいます。音カテゴリの他のブロックを調べたり、音タブの中にある編集ツール[訳注1]を使うことによって、Scratch上でサウンドを表現する方法を色々試すよう促します。
- 生徒に、お互いに自分達のバンドをデモンストレーションすること[訳注2]ことや、歩き回ってクラスメイトの楽器を操作すること[訳注3]を認めます。ギャラリーウォークがお勧めです。生徒に、自分のプロジェクトをプレゼンテーションモードにしてもらい、続いて友達を招待して、歩き回ってお互いのプロジェクトを調べ合います。[訳注4]必要に応じて、生徒にBuild-a-Bandスタジオ、またはクラス・スタジオに自分のプロジェクトを追加してもらいます。
- 生徒に、デザイン日誌やグループ討議の中で振り返りを促す質問に回答させてデザインのプロセスを振り返ってもらいます。

訳注1: Scratchに内蔵されているサウンド・エディタのこと。

訳注2: リアルではバンドは演奏するが、Scratch上ではバンドをデモする。

訳注3: リアルでは楽器は演奏するが、Scratch上では操作する。

訳注4: プロジェクトを動かしたり、コードやスクリプトなどを調べる。

メモ

- + グループ全体で共有するために、生徒にScratchの楽器と一緒に演奏し、クラスのバンドを結成してもらいます！

リソース

- 「バンドを作ろう」配布用プリント
- Build-a-Bandスタジオ
<http://scratch.mit.edu/studios/475523>

振り返りを促す質問

- + 最初に何をしましたか？
- + 次に何をしましたか？
- + 最後に何をしましたか？

作業の確認ポイント

- + プロジェクトはサウンドをクリエイティブに利用しているか？
- + プロジェクトのスプライトはインタラクティブか？

追加メモ記入欄

- _____
- _____
- _____
- _____

バンドを作ろう

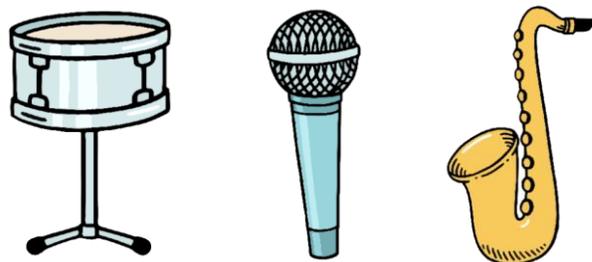
サウンド、楽器、バンドや最も愛する音楽を表現する音楽のスタイルを制作するには、Scratchをどう活かしますか？

このアクティビティでは、インタラクティブな楽器をデザインするために、スプライトとサウンドを組み合わせ、独自の感動的なミュージックのScratchプロジェクトを制作します。



ここからスタート

- スプライトを作成します。
- 音のブロックを加えます。
緑のブロックが見当たらないときは、“拡張機能を追加”ボタン（画面左下）を押し、“音楽”の拡張機能を選択してください。
- 楽器をインタラクティブにする方法を試します。



スプライトライブラリーから楽器を選ぶか、独自の楽器を作成します。

このスプライトがクリックされたとき

10 回繰り返す

🎵 (1) スネアドラム のドラムを 0.25 拍鳴らす

🎵 0.25 拍休む

このスプライトがクリックされたとき

8 回繰り返す

🎵 (8) 手拍子 のドラムを 0.25 拍鳴らす

🎵 1 拍休む

試してみよう

- “~回繰り返す”ブロックを使って、サウンドを数回、再生させます。
- あなた自身のサウンドをインポートしたり、録音します。またサウンドエディタを試してみよう。
- テンポのブロックを使って演奏してみて、リズムをスピードアップしたり、スローダウンしてみよう。

完成した？

- + Build-a-Bandスタジオにプロジェクトを追加しましょう。
<http://scratch.mit.edu/studios/475523>
- + あなた自身にもっとチャレンジしましょう！
新しい楽器を選択したり、独自のサウンドを録音しましょう。
- + 友達を手助けしましょう！

色付きの四角と丸

 **推奨時間**
30-45分

目的

- このアクティビティを完了すると、生徒は、
- + アートをテーマにした課題を完成させることで自身の創造性を表現します。
 - + “見た目”のブロックとペイントエディタを自在に使いこなします。

アクティビティの説明

- 必要に応じて、Orange Square, Purple Circleスタジオのサンプルプロジェクトを見せ、生徒をガイドする「色付きの四角と丸」配布用プリントを用意します。
- 生徒に時間を与え、オレンジ色の四角と紫色の丸を含んだプロジェクトを作成してもらいます。自分の芸術的な才能を探求するため、生徒を招待して“見た目”のブロックとペイント・エディタ[訳注1]を試してみます。
- 生徒に創作活動を友達と分かち合うよう促します。ギャラリーウォークがお勧めです。生徒に、自分のプロジェクトをプレゼンテーションモードにしてもらい、続いて友達を招待して、歩き回りってお互いのプロジェクトを調べ合います。
[訳注2]
必要に応じて、生徒にOrange Square, Purple Circleスタジオ、またはクラス・スタジオにプロジェクトを追加してもらいます。
- 生徒に、デザイン日誌やグループ討議の中で振り返りを促す質問に回答させてデザインのプロセスを振り返ってもらいます。

リソース

- 「色付きの四角と丸」配布用プリント
- Orange Square, Purple Circleスタジオ
<http://scratch.mit.edu/studios/475527>

振り返りを促す質問

- + どうやって、オレンジ色の四角形と紫色の円をプロジェクトに組み込みましたか？このアイデアはどこから来ましたか？
- + このアクティビティに関して、何がチャレンジでしたか？
- + このアクティビティに関して、何がサプライズでしたか？

作業の確認ポイント

- + プロジェクトがオレンジ色の四角形と紫色の円を含んでいるか？

メモ

- + 生徒が疑問を持っている場合は、Tipsウィンドウを開くと、ブロックやScratchエディタの色々なパーツについて知ることができることを教えてください。
- + Scratchは、ビットマップ[訳注3]とベクター・グラフィックス[訳注4]の両方をサポートしています。さまざまな種類の画像やテキストをデザインしたり操作するため、生徒がペイントエディタのベクターモードまたはビットマップモードのポタンにナビゲートするのを助けます。

訳注1: コスチュームや背景を編集する機能を使って、図形を描いてみる。

訳注2: プロジェクトを動かしたり、コードやスプライトなどを調べる。

訳注3: 点の集まりで構成される画像。拡大すると不鮮明になる。

訳注4: 点と線の情報を数値化した画像。拡大・縮小しても鮮明。

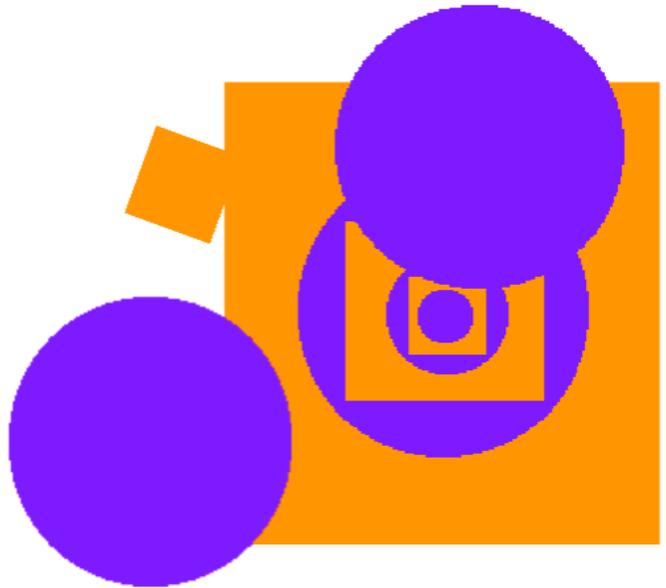
追加メモ記入欄

- _____
- _____
- _____
- _____

色付きの四角と丸

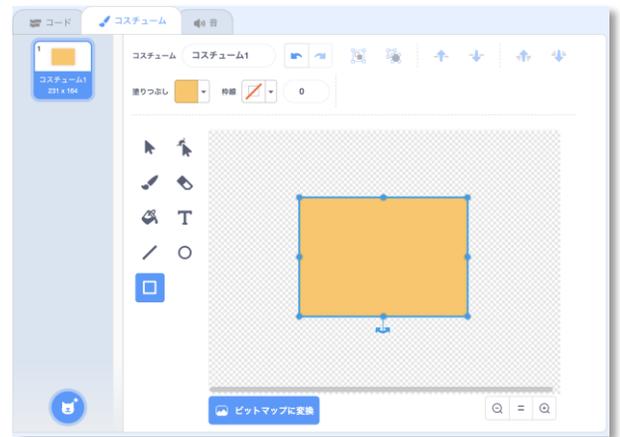
どんなオレンジ色の四角と紫色の丸を含んだプロジェクトを制作できますか？

この課題では、オレンジ色の四角と紫色の丸を含んだプロジェクトを制作します。
何を作りますか？



ここからスタート

- ペイント・エディタを使ってスプライトを描きます。
- さまざまな“見た目”カテゴリーや“動き”カテゴリーのブロックを加えて、スプライトを生き生きとさせます。
- 繰り返しましょう！



行き詰まりましたか？

大丈夫！ 次のことを試してみてください...

完成した？

- 隣の友達とブレインストーミングしましょう！
- Scratchでプロジェクトを作り始める前に、試してみたいことの一覧を作りましょう！
- 他の人のプロジェクトを検索して、Scratchで他の人がやっていることを調べてみましょう。これはインスピレーションを発見する素晴らしい方法です！

- + Orange Square, Purple Circleスタジオにプロジェクトを追加しましょう。 <http://scratch.mit.edu/studios/475527>
- + ビットマップモードとベクターモードの違いを調べましょう。切替えボタンがペイント・エディタの下に配置されています。
- + あなた自身にもっとチャレンジしましょう！ その他のシェイプや色を加えてみましょう。
- + パートナーとプロジェクトを交換して、お互いの作品をリミックスしましょう。
- + 友達を手助けしましょう！

生きてる！

 **推奨時間**
30-45分

目的

- このアクティビティを完了すると、生徒は、
- + 制御のブロックを試すことで、コンピュータ処理の概念（順次処理と繰り返し処理）をより深く理解します。
 - + スプライトとコスチュームとの違いを説明できるようになります。
 - + アニメーションのプロジェクトの制作を通して、実験と反復を実践します。

アクティビティの説明

- 必要に応じて、It's Alive!スタジオのサンプルプロジェクトを見せて、生徒をガイドする「生きてる！」配布用プリントを用意します。
- 一連の一つずつ異なる画像[訳注1]をループさせるという、アニメーションの概念を紹介します。アニメーションを作成するために、生徒に、コスチュームや背景を変化させて、繰り返し処理を探究するよう促します。
- 生徒に、ギャラリーウォークを開催して他の人と作品を共有するよう勧めます。
生徒に、プロジェクトをプレゼンテーションモードにもらい、続いて友達を招待して、歩き回ってお互いのプロジェクトを調べ合います。[訳注2]
必要に応じて、生徒にIt's Alive!スタジオまたはクラス・スタジオに自分のプロジェクトを追加させます。
- 生徒に、デザイン日誌やグループ討議の中で振り返りを促す質問に回答させてデザインのプロセスを振り返ってもらいます。

訳注1: バラバラ漫画や粘土のアニメのような画像。

訳注2: プロジェクトを動かしたり、コードやスプライトなどを調べる。

リソース

- 「生きてる！」配布用プリント
- It's Alive!スタジオ
<http://scratch.mit.edu/studios/475529>

振り返りを促す質問

- + スプライトとコスチュームの違いは何ですか？
- + アニメーションとは何ですか？
- + あなたが実生活で経験している繰り返し処理を3つリストアップしてください。（例えば、毎晩寝る）

作業の確認ポイント

- + 生徒はスプライトとコスチュームを区別できるか？
- + Scratcher達は、アニメーションのプロジェクトの制作に特に興味を持ち、スプライト、コスチューム、背景を描いたり、デザインするのに時間を費やすのを好みます。プロジェクトの美的な側面とテクニカルな側面の両方にどうやって生徒を取り組ませますか？

メモ

- + スプライトとコスチュームの違いは、しばしばScratcher達の混乱の原因となります。複数のコスチュームを着用して登場する人物のメタファーは、その違いを明確にするのに役立ちます。
- + 生徒は、カメラやウェブカメラを使って自分の写真を撮ることで、自分のイメージをアニメ化できます。

追加メモ記入欄

- _____
- _____
- _____
- _____

生きてる！

どうやって画像や写真を利用して、生き生きとさせることができますか？

このアクティビティでは、連続したコスチュームの変化をプログラミングすることによって、アニメーションのように、スプライトや画像に生命を吹き込むアイデアを探します。

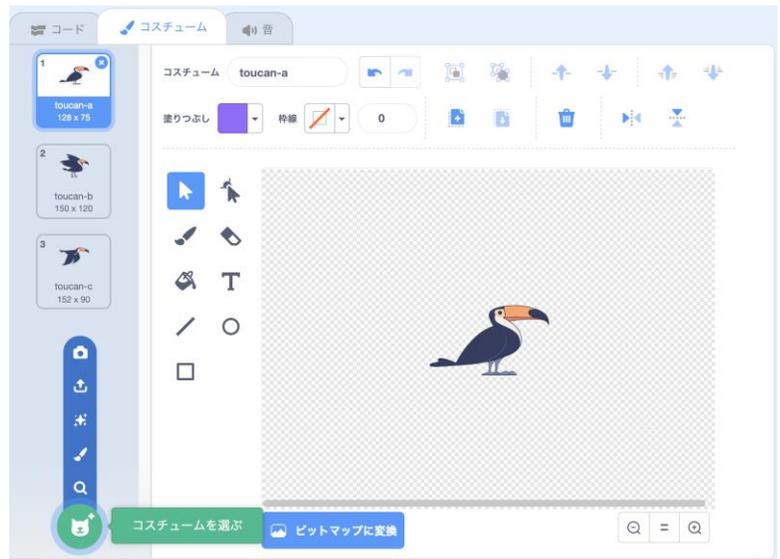


ここからスタート

- スプライトを選びます。
- 異なるコスチュームを追加します。
- ブロックを加えて、画像を生き生きとさせます。
- 繰り返しましょう！

このスプライトがクリックされたとき

- 10 回繰り返す
- 0.1 秒待つ
- 10 歩動かす
- 次のコスチュームにする



試してみよう

- まず、アニメーションのアイデアを（パラパラ漫画のように）紙にスケッチしてみましょう。
- 楽しめるものが見つかるまで、色々なブロックとコスチュームを試してみます。
- インスピレーションが必要ですか？ “見る”のページの“アニメーション”のセクションでプロジェクトを探しましょう。

完成した？

- + It's Aliveスタジオにプロジェクトを追加しましょう。
<http://scratch.mit.edu/studios/475529>
- + あなた自身にもっとチャレンジしましょう！
- + もっと多くの機能をプロジェクトに追加して、アニメーションをもっとリアルにしてみましょう。
- + 友達を手助けしましょう！
- + 感動的なアニメーションのプロジェクトを見つけて、リミックスしましょう！

デバッグしよう！

 **推奨時間**
15-30分

目的

- このアクティビティを完了すると、生徒は、
- + 5つの課題に対する問題を調査してデバッグし、解決策を見つけ出します。
 - + テストとデバッグの実践を通して、さまざまな概念（順次処理と繰り返し処理を含む）を探究します。
 - + プロジェクトをデバッグするための戦略のリストを作成します。

アクティビティの説明

- 必要に応じて、アクティビティの中で、生徒をガイドするユニット2「デバッグしよう！」配布用プリントを用意しておきます。
- ユニット2「デバッグしよう！」配布用プリントに記述されているプロジェクトへのリンクをたどってみせ、生徒が、Unit 2 Debug It!スタジオのプログラムを開くのをガイドします。生徒に、“中を見る”ボタンをクリックするよう促して、バグのあるプログラムを調べて、コードをあちこち変えてみて、可能性のある解決方法を試してもらいます。
- 生徒に時間を与え、それぞれの「デバッグしよう！」の問題をテストし、デバッグしてもらいます。必要に応じて、生徒にScratchのリミックス機能を使って、バグを解決し修正済みのプログラムを保存してもらいます。
- 生徒に、デザイン日誌やグループ討議の中で振り返りを促す質問に回答させてテストやデバッグの体験を振り返ってもらいます。
- 生徒の問題発見と問題解決のアプローチを集めて、クラス全体のデバッグ戦略リストを作成します。

リソース

- ユニット2「デバッグしよう！」配布用プリント
- ユニット2 Debug It!スタジオ
<http://scratch.mit.edu/studios/475539>

振り返りを促す質問

- + 何が問題でしたか？
- + どうやって問題を特定しましたか？
- + どうやって問題を解決しましたか？
- + 他の人は、問題を解決する別のアプローチを取りましたか？

作業の確認ポイント

- + 生徒は、バグをすべて解決できたか？
そうでない場合は、解決できなかったプログラムで示されている概念をどうやったら明確にできるか？
- + 生徒は、どんなテストとデバッグの戦略を採用したのか？

メモ

- + 全グループでこのアクティビティを、“スクリプトを演じる”のアクティビティと同様に、生徒に「デバッグしよう！」のプログラムを演じさせるようファシリテートします。
あるいは、プロジェクトのテストとデバッグのために、新たな戦略として、“スクリプトを演じる”のアクティビティを紹介します。

追加メモ記入欄

- _____
- _____
- _____
- _____

デバッグしよう！

ヘルプ・ミー！

5つのScratchプログラムをデバッグできますか？

このアクティビティでは、「デバッグしよう！」の各問題について、何がうまくいっていないのかを調査し、解決策を見つけ出します。

ここからスタート

- ユニット2用のDebug It!スタジオを開きます。
<http://scratch.mit.edu/studios/475483>
- スタジオの5つのデバッグの問題をテストし、デバッグします。
- 解決策を書き出したり、あなたの解決策を使ってバグのあるプログラムをリミックスしましょう。

行き詰まりましたか？

大丈夫！ 次のことを試してみてください...

- プログラム中で起こりうるバグを一覧にします。
- あなたの作業の経過を追ってみましょう！
この一覧は、すでに試したことを思い出すのに役立ち、次に試すべきことを示してくれます。
- 役立つものを発見するまで、友達と気づいた問題点と問題解決のアプローチを共有し、比較してみましょう！

[訳注1] MITが商標登録しているScratch上のキャラクタ

□ デバッグしよう！ 2.1

<http://scratch.mit.edu/projects/23266426>

このプロジェクトでは、Scratchキャットがダンスを見せたがっています。Scratchキャットをクリックすると、ドラムビートが鳴っている間、Scratchキャットは調子をあわせてダンスするはずですが、しかし、Scratchキャットはダンスし始めるとすぐに止まってしまう、ドラム演奏が続いてしまいます！ どうプログラムを修正しますか？

□ デバッグしよう！ 2.2

<http://scratch.mit.edu/projects/24268476>

このプロジェクトでは、緑の旗をクリックされた時、Pico[訳注1]が、Nano [訳注1]の方に移動して、Nanoのところに着いた時に、「Tag, you're it! (捕まえた!）」と言います。するとNanoが、「My turn! (私の番だ!）」と言うはずですが、しかし、何かがおかしい！ PicoはNanoのところに行っても何も言いません。どうプログラムを修正しますか？

□ デバッグしよう！ 2.3

<http://scratch.mit.edu/projects/24268506>

このプロジェクトでは、笑顔を描くようプログラムされていますが、何かが正しくありません！ ペンは、目の片方から笑っている口まで線をつながないはずなのに、線を描き続けてしまいます。どうプログラムを修正しますか？

□ デバッグしよう！ 2.4

<http://scratch.mit.edu/projects/23267140>

このプロジェクトでは、緑の旗をクリックされた時、花の成長のアニメーションが始まり、満開になったら花びらだけ停止するはずですが、しかし、何かが正しくありません！ すべての花びらが開いた時に止まるのではなく、アニメーション全てが止まってしまう。どうプログラムを修正しますか？

□ デバッグしよう！ 2.5

<http://scratch.mit.edu/projects/23267245>

このプロジェクトでは、緑の旗をクリックされると「Happy Birthday(ハッピーバースデー)」の曲の演奏がスタートして曲が終わったら、「click to blow out the candles! (クリックして、ロウソクを吹き消して!）」と言う掛け声が表示されるはずですが、しかし、何か正しく動作しません！ 曲が終わった後ではなく、曲が演奏されている時にロウソクを吹き消す掛け声が表示されてしまいます。どうプログラムを修正しますか？

終わった？

- + スクリプト内のブロックを右クリックしてコードにコメントを追加します。コメントを追加することは、友達がプログラムの色々な部分を理解するのに役立ちます。
- + テストとデバッグの実践についてパートナーと話し合しましょう。あなたの戦略との類似点と相違点を書き出しましょう！
- + 友達を手助けしましょう！

ミュージック ビデオ

 **推奨時間**
45-60分

目的

- このアクティビティを完了すると、生徒は、
- + 自らで方向性を決めたミュージックビデオ・プロジェクトに取り組んで、アニメーションと音楽を組み合わせたプロジェクトを創作できます。
 - + スプライト、コスチューム、サウンドの理解がさらに深まります。

アクティビティの説明

- 生徒に、Scratch上でサウンドとアニメーションを組み合わせたミュージックビデオ制作のアイデアを紹介します。必要に応じて、Music Videoスタジオからプロジェクトのサンプルをいくつか見せます。
- 生徒に、ガイダンスとインスピレーションのために提供可能な「ミュージックビデオ」配布用プリントを使って、プロジェクトに取り組むための自由時間を与えます。生徒に、他の人のアイデア、ミュージック、コードの利用に対して、プロジェクトページ上に謝辞を示すよう勧めます。
- 生徒が、プロジェクトを制作している間に、フィードバックしたり、フィードバックをもらったりするのをガイドします。隣の友達とチェックイン[訳注1]するのがお勧めです。フィードバックを求めるために、生徒に中断してもらい、作業中の作品を他の人や批評グループ（ユニット0の“批評グループ”のアクティビティを参照）に共有してもらいます。必要に応じて、生徒にMusic Videoスタジオまたはクラス・スタジオにプロジェクトを追加してもらいます。
- 生徒に、デザイン日誌やグループ討議の中で振り返りを促す質問に回答させてデザインのプロセスを振り返ってもらいます。

[訳注1] 参加者がお互いの率直な気持ち、気がかりな事、期待などを共有する時間をとること

メモ

- + プロジェクトをさらにパーソナライズするには、生徒が音タブの機能を使って、好きな曲を挿入したり、彼ら自身の歌や楽器の演奏を録音するのをガイドします。
- + アクティビティ中に、リミックスや盗作に関する質問が発生することがあります。リミックスに関するScratchのFAQを使って、謝辞と帰属を示すことに関するディスカッションをファシリテートする機会を持ちます。
<https://scratch.mit.edu/info/faq/#remix>

リソース

- 「ミュージックビデオ」配布用プリント
- Music Videoスタジオ
<http://scratch.mit.edu/studios/475517>

振り返りを促す質問

- + 克服したチャレンジは、どんなものでしたか？ どうやってそのチャレンジを克服しましたか？
- + さらに理解したいことは何ですか？
- + プロジェクトで使うために借りたアイデア、ミュージック、コードに対する謝辞をどう示しましたか？

作業の確認ポイント

- + プロジェクトは、スプライトとサウンドを組み合わせましたか？
- + 生徒はプロジェクトのどの部分をアニメ化することにしたのか？
- + これまでに紹介されたもので、生徒がまだ行き詰っている可能性のあるブロックや概念があるか？ どのように手助けできそうか？

追加メモ記入欄

- _____
- _____
- _____
- _____

ミュージックビデオ

Scratchの素晴らしいミュージックビデオを制作するには、どうやってミュージックとアニメーションを組み合わせますか？

このプロジェクトでは、パーソナライズされたミュージックビデオを制作するために、演劇、歌、ダンス、ミュージック、スケッチ、イラスト、写真撮影、アニメーションに関連するアイデアを探究します。

ここからスタート

- 音を加えます。
緑のブロックが見当たらないときは、“拡張機能を追加”ボタン（画面左下）を押し、“音楽”の拡張機能を選択してください。
- スプライトを作成して、動きを加えます。
- スプライトと音を相互に作用させます。



試してみることに

- コスチュームを使ってアニメーションを生き生きとさせます
- スプライトがクリックやキー入力などに反応するスクリプトを追加して、スプライトをインタラクティブにします。
- プログラムの操作方法を説明するために、プロジェクトのページに説明を追加します。

[訳注1] コンピュータがランダムに選んで驚かせる。

使用するブロック



完成した？

- + Music Videoスタジオにプロジェクトを追加しましょう。
<http://scratch.mit.edu/studios/475517>
- + プロジェクトページ使ったミュージック、コード、作品であれば、どんなものでも謝辞を示すことを忘れないようにしましょう。
- + あなた自身にもっとチャレンジしましょう！あなた自身のスプライト、音、コスチュームを創作しましょう。

ユニット3 物語



インデックス

内容



登場人物	58
会話	60
場面	62
デバッグしよう!	64
創造物の制作	66
続けて創ろう	68

ユニット3

概要

“ビッグ・アイデア”

アンドレス・モンローイ・エルナンデス氏（初期バージョンのScratchオンライン・コミュニティのリードデザイナー）が調査した「リミックス・カルチャー」の博士論文のイントロダクションには、3つ引用文[訳注1]が含まれています。

他人の作品を土台に構築することは、プログラミングにおいて長年の習慣です。そして他人の作品に幅広いアクセスを可能にするネットワーク技術によって、わずかに増幅されています。創造的コンピューティングの重要なゴールのひとつは、再利用とリミックスによる学習者間のつながりをサポートすることです。

Scratchオーサリング環境とオンライン・コミュニティは、このキーとなるコンピュータ上の実践の際に、学習者がアイデアやコードを見つけ出すのを手助けして築き上げたり、1人で制作するよりもっと複雑なプロジェクトを制作することを可能にすることで、若いデザイナーをサポートしています。

再利用とリミックスをサポートするカルチャーを育てるために、このユニットのアクティビティでは、最初のアイデアと戦略を示します。共有する事とつながる事を、さらにサポートするにはどうすれば良いでしょうか？

[訳注1] 右の3つのフキダシ

私たちは巨人の肩の上に立つ小人のようなもので、古代人よりも、より多く、より遠くを見ることができる。
シャルトルのベルナルド、
およそ1130年

私が、かなたを見渡せたのだとしたら、それはひとえに巨人の肩の上に乗っていたからです。
アイザック・ニュートン、
1676年

巨人に乗った小人の方が、巨人より遠くまで見える。
ジョージ・ハーバート、
1651年

学習の目的

- 生徒は、
- + デザイン中に、再利用とリミックスの利点に親しみを覚え、深く理解します。
 - + コンピュータ処理の概念（イベント処理と並列処理）と実践（実験と反復、テストとデバッグ、再利用と再リミックス）を使いこなす能力を身につけます。
 - + コラボレイティブな物語をデザインすることによって、物語というジャンルのコンピュータ創作品を探究します。

キーワード、概念、実践

- + 再利用とリミックス
- + Pass-it-onストーリー
- + Scratchスクリーン
- + ブロック作成
- + リー
- + バックバック
- + ペアプログラミング
- + デザインデモ
- + ステージ

メモ

- + 再利用とリミックスは、コードを読み取る極めて重要な能力を高め、所有権と著作権に関する疑問を抱かせます。共同的で協調的な作業をどうやって促進し、ディスカッションし、評価するのかについての戦略を色々検討してみよう。

アドベンチャーを選択しよう



このユニットでは、さまざまなハンズオンとコンピュータを離れたデザイン・アクティビティを通して、生徒がストーリーテリング[訳注1]やリミックスの能力を高めるのを助けること、生徒が協力して作業をして他人のクリエイティブな作品を土台にして構築するチャンスを与えることに重点を置きます。

ユニット2での最初の経験を足場にして、このユニットのアクティビティは、生徒がコンピュータ処理の概念（イベント処理、並列処理）とコンピュータ上の実践（実験や反復、再利用とリミックス）を自在に使いこなす能力を身に付けるのに役立つようデザインされています。

各能力開発アクティビティは、スプライトと背景の間の対話処理をプログラミングするための新しいブロックとメソッドを発見することで、生徒がストーリーテリング[訳注1]プロジェクトを作り上げるのに役立つようデザインされています。

結果として「続けて創ろう(Pass-It-On)」プロジェクトとして結実させます。

[訳注1] 物語を語ること、読み聞かせ

可能なルート



登場人物

 **推奨時間**
30-45分

目的

このアクティビティを完了すると、生徒は、
+ Scratchの“ブロックを作る”の機能を使って、登場人物の動きを定義することを体験します。
+ コンピュータ処理の概念（イベント処理、並列処理）と実践（実験と反復）をより深く理解します。

アクティビティの説明

- 必要に応じて、Charactersスタジオのサンプルプロジェクトを示したり、生徒をガイドする「登場人物」配布用プリントを用意します。
- 生徒に時間を与え、“その他”カテゴリにある“ブロックを作る”の機能を使って、独自のScratchブロックを作成してもらいます。生徒がそれぞれの行動を持った2つのスプライト（登場人物）をデザインするのを手助けします。必要に応じて、クラスで“ブロックを作る”の機能のウォークスルー[訳注1]を行います。
- 生徒が登場人物や行動を互いに共有できるようにします。デザインデモのアクティビティがお勧めです。少数の生徒に自分の作品を発表するよう頼んで、どうやって“ブロックを作成する”の機能を実装したのかをデモンストレーションしてもらいます。必要に応じて、生徒にCharactersスタジオや独立したクラス・スタジオにプロジェクトを追加してもらいます。
- 生徒に、デザイン日誌やグループ討議の中で振り返りを促す質問に回答させてデザインのプロセスを振り返ってもらいます。

[訳注1] 実際に登場人物を動かしてみて、他の人にその動きを確認してもらう。

メモ

- + 生徒が“ブロックを作る”の機能の使い方を理解するのに苦労している場合は、生徒にCharactersスタジオのプロジェクトのコードを調査して、他の人がその機能を実装した方法を調べるよう勧めます。
- + 詳しくは、ビデオチュートリアルの“ブロックを作る”機能をご覧ください。: <http://bit.ly/makeablock>

リソース

- 「登場人物」配布用プリント
- Charactersスタジオ
<http://scratch.mit.edu/studios/475545>

振り返りを促す質問

- + どうやって誰かに“ブロックを作る”を説明しますか？
- + どんな時に“ブロックを作る”を使うと良いですか？

作業の確認ポイント

- + プロジェクトには、“ブロックを作る”の機能で、各々行動するスプライトが2つ含まれているか？
- + 生徒は、“ブロックを作る”の機能の使用方法をお互いで説明し合うことができるか？
また、あなたにも説明できるか？

追加メモ記入欄

- _____
- _____
- _____
- _____

登場人物

独自のScratchブロックを作成したいですか？

Scratchの“ブロックを作る”の機能を体験します。2人の異なる登場人物用に独自のブロックを作成し、それぞれの行動を定義します。

ここからスタート

- 2人分のスプライト（登場人物）を、ライブラリから選んだり、描いたり、アップロードします。
- “作ったブロック”カテゴリの“ブロックを作る”ボタンをクリックしてブロックを作成し、名前を付けます。
- カスタムブロックの行動をコントロールするには、“定義”の下にブロックを追加します。
- 登場人物の行動をプログラムしたブロックを試みに動かしてみましょう。
- 繰り返しましょう！



作ったブロック

ブロックを作る



作ったブロック

ブロックを作る

ジャンプ

定義 ジャンプ

試してみよう

- + 行き詰まりましたか？ 大丈夫です！以下の“ブロックを作る”の機能の入門ビデオをチェックしましょう。
<http://bit.ly/makeablock>
- + Charactersスタジオの中の他のプロジェクトを検索して、他の人がどんな新しいブロックを作成しているのか調べます。
- + しばしば、同じ行動を定義するのに複数の方法があることがあります。複数のオプション[注1]と結果にトライするよう、色々なブロックの組み合わせを試してみましょう。

[注1] 定義ブロックでオプションを指定すると、オプションの値で異なる行動や処理を起こすブロックを作成できます。

完成した？

- + Charactersスタジオにプロジェクトを追加しましょう。
<http://scratch.mit.edu/studios/475545>
- + あなた自身にもっとチャレンジしましょう！“ブロックを作る”の機能を使って、さまざまな登場人物と行動を追加してみましょう。
- + 友達を手助けしましょう！

会話

 **推奨時間**
30-45分

目的

- このアクティビティを完了すると、生徒は、
- + ジョークのプロジェクトをリミックスして、スプライト間のやりとりをシンクロさせるための2つの異なる方法（時間調整とメッセージ送信）を調査します。
 - + コンピュータ処理の概念（イベント処理、並列処理）と実践（再利用とリミックス）を自在に使いこなす能力を身に付けます。

アクティビティの説明

- 必要に応じて、Penguin Joke入門プロジェクトをグループで調べたり、生徒をガイドする「会話」配布用プリントを用意します。
- 生徒に、Penguin Joke入門プロジェクトを見るよう勧め、「～秒待つ」ブロックを使って、会話をアニメーションにする方法を観察してもらいます。
リミックス機能を使い、Penguin Joke入門プロジェクトをデザインし直して、会話をシンクロさせます
使用するブロック：“～を送る”、“～を送って～秒待つ”、“～を受け取ったとき”
- 生徒に、ジョークのプロジェクトをお互いで共有してもらいます。デザインデモのアクティビティがお勧めです。少数の生徒にクラスで作品を見せながら、メッセージ送信の実現方法をデモンストレーションしてもらいます。
必要に応じて、生徒にConversationsスタジオやクラススタジオにプロジェクトを追加してもらいます。
- 生徒に、デザイン日誌やグループ討議の中で振り返りを促す質問に回答させてデザインのプロセスを振り返ってもらいます。

リソース

- 「会話」配布用プリント
- Penguin Joke入門プロジェクト
<http://scratch.mit.edu/projects/10015800>
- Conversationsスタジオ
<http://scratch.mit.edu/studios/475547>

振り返りを促す質問

- + 他の誰かに、メッセージを送る（“～を送る”）機能をどのように説明しますか？
- + プロジェクトで、いつ時間調整（“～秒待つ”）を使いますか？ いつメッセージ送信（“～を送る”、“～を受け取ったとき”）を使いますか？

作業の確認ポイント

- + プロジェクトで“～を送る”と“～を受け取ったとき”を使っているか？
- + 生徒は、“～を送る”、“～を送って待つ”、“～を受け取ったとき”の使い方を説明できるか？

メモ

- + 生徒が、“～を送る”ブロックと“～を受け取ったとき”ブロックを対で使う方法を理解できない場合、Broadcast Examplesスタジオのサンプルプロジェクトのコードを調べるよう促します。
<http://scratch.mit.edu/studios/202853>

追加メモ記入欄

- _____
- _____
- _____
- _____

会話

スプライト間のやりとりをコーディネートするための色々な方法には、何が
ありますか？

このアクティビティでは、スプライトに会話させるプログラムの作り方を探究します！
ジョークのプロジェクトをリミックスして、時間調整を試したり、メッセージ送信を調べます。



ここからスタート

- ❑ Penguin Jokes入門プロジェクトの中を見ましよう。<http://scratch.mit.edu/projects/10015800>
- ❑ コードを調査して、会話をコーディネートさせるために、“～秒待つ”ブロックと“～と～秒言う”ブロックを使う方法を確認しましょう。
- ❑ “～秒待つ”ブロックの代わりに、“～を送る”ブロックと“～を受け取ったとき”ブロックを使って、プロジェクトをリミックスしてください。



行き詰まりましたか？

大丈夫！ 次のことを試してみてください...

完成した？

- ❑ 友達とアイデアをブレインストーミングしましょう！可能な解決策のリストを作って一緒にテストします。
- ❑ “～を送る”ブロックと“～を受け取ったとき”ブロックをプロジェクトの色々なところに置いて、トライしてみます。
- ❑ スプライト同士の会話をコーディネートさせる別の方法のヒントを得るには、Conversationsスタジオでプロジェクトを探してみましょう。

- + Conversationsスタジオにプロジェクトを追加しましょう。<http://scratch.mit.edu/studios/475547>
- + あなた自身にもっとチャレンジしましょう！他の登場人物との会話を追加します。
- + 隣の友達に自分のプロジェクトを共有し、探究とデザインのプロセスを彼らにウォークスルー[注1]しましょう
- + 友達を手助けしましょう！

[注1] 実際に動かして会話する様子を見せて確認する。

場面

 **推奨時間**
30-45分

目的

- このアクティビティを完了すると、生徒は、
- + 複数場面のある物語やスライドショーのように、背景の転換[訳注1]を体験をするプロジェクトを創作できます。
 - + コンピュータ処理の概念（イベント処理、並列処理）と実践（実験と反復）に、さらに親しみを覚えるようになります。

[訳注1] 演劇の用語で場面を変えること。場面転換とも言う。

アクティビティの説明

- ❑ 必要に応じて、Scenesスタジオのサンプルプロジェクトを示し、生徒をガイドする「場面」配布用プリントを用意します。
- ❑ 生徒に時間を与え、異なる背景を使って複数場面の転換があるプロジェクト（スライドショーのような）を作成してもらいます。背景を転換を起こすために、生徒にステージのスク립トを調べ、上手に使うようチャレンジさせます。
- ❑ 生徒が、プロジェクトを互いで共有できるようにします。デザインデモのアクティビティがお勧めです。少数の生徒に、クラスに作品を見せながら、どうやって背景の転換を実現したか、デモンストレーションするよう促します。必要に応じて、自分のプロジェクトをScenesスタジオカクラス・スタジオに追加してもらいます。
- ❑ 生徒に、デザイン日誌やグループ討議の中で振り返りを促す質問に回答させてデザインのプロセスを振り返ってもらいます。

リソース

- ❑ 「場面」配布用プリント
- ❑ Scenesスタジオ
<http://scratch.mit.edu/studios/475550>

振り返りを促す質問

- + ステージとスプライトとの共通点は何ですか？
- + ステージはスプライトとどのように違いますか？
- + ある場面でどうやってスプライトのアクションを起こしますか？
- + アニメーション以外で場面転換を使っているのは、どんなタイプのプロジェクトですか？

作業の確認ポイント

- + プロジェクトは、背景の切り替えを使って、複数の場面をうまくコーディネートさせているか？

メモ

- + 生徒が背景を切り替える方法を理解できない場合は、彼らに“見た目”カテゴリのブロック（特に“背景を～にする”、“背景を～にして待つ”、“次の背景”ブロック）を使って、試行錯誤してみるよう促します。

追加メモ記入欄

- ❑ _____
- ❑ _____
- ❑ _____
- ❑ _____

場面

ステージとスプライトは何が違いますか？

このアクティビティでは、複数場面のある物語やスライドショーのように、背景を試すプロジェクトを作成します。

ここからスタート

- 複数の背景をライブラリから選んだり、描いたり、アップロードしてください。
- “見た目”カテゴリと“イベント”カテゴリのブロックを試して、背景の転換させます。
- スクリプトをステージとスプライトに追加して、プロジェクトで背景が変わる時に起こることをコーディネートしましょう。



背景を **背景1** にする

背景が **背景1** になったとき

背景の **番号**



試してみよう

- ブロックが何をしているのか調べるために、背景に関連したスプライトとステージの中のブロックを見つけて、テストしましょう。
- もっとひらめきが必要ですか？
複数の背景を使っているプロジェクトを発見するために、Scratchオンライン・コミュニティを検索してみましょう。

完成した？

- + Scenesスタジオにプロジェクトを追加しましょう。：
<http://scratch.mit.edu/studios/475550>
- + あなた自身にもっとチャレンジしましょう！
プロジェクトにもっと背景の転換を加えてみましょう。
- + 友達を手助けしましょう！
- + 以前のプロジェクトのいずれかに戻るか、ひらめきを得たプロジェクトを見つけ、背景の切り替えを加えて、リミックスしよう。

デバッグしよう！

 **推奨時間**
15-30分

目的

- このアクティビティを完了すると、生徒は、
- + 5つの課題に対する問題を調査してデバッグし、解決策を見つけ出します。
 - + テストとデバッグの実践を通して、さまざまな概念（イベント処理、並列処理）を探究します。

アクティビティの説明

- 必要に応じて、アクティビティ中で、生徒をガイドするユニット3「デバッグしよう！」配布用プリントを用意しておきます。
- ユニット3「デバッグしよう！」配布用プリントに記載されているプロジェクトへのリンクをたどってみせ、生徒が、Unit 3 Debug It!スタジオのプログラムを開くのをガイドします。生徒に、“中を見る”ボタンをクリックするよう促して、バグのあるプログラムを調べて、コードをあちこち変えてみて、可能性のある解決方法を試してもらいます。
- 生徒に時間を与え、それぞれの「デバッグしよう！」の問題をテストし、デバッグしてもらいます。必要に応じて、生徒にScratchのリミックス機能を使って、バグを解決し修正済みのプログラムを保存してもらいます。
- 生徒に、デザイン日誌やグループ討議の中で振り返りを促す質問に回答させてテストやデバッグの体験を振り返ってもらいます。
- 生徒の問題発見と問題解決のアプローチを集めて、クラス全体のデバッグ戦略リストを作成します。

リソース

- ユニット3「デバッグしよう！」配布用プリント
- Unit 3 Debug It!スタジオ
<http://scratch.mit.edu/studios/475554>

振り返りを促す質問

- + 何が問題でしたか？
- + どうやって問題を特定しましたか？
- + どうやって問題を解決しましたか？
- + 他の人は、問題を解決する別のアプローチを取りましたか？

作業の確認ポイント

- + 生徒は、バグをすべて解決できたか？ そうでない場合は、解決できなかったプログラムで示されている概念をどうやったら明確できるか？
- + 生徒は、どんなテストとデバッグの戦略を採用したのか？

メモ

- + 他人のコードを読むことは、再利用とリミックスが実践できるようになるための価値あるスキルであり、とても重要なことです。
- + このアクティビティはペア・プログラミングの素晴らしいチャンスです。生徒を二人一組のペアに分けて、デバッグの課題に取り組ませます。
- + 生徒は、Scratchのブロックを右クリックしてコードにコメントを追加して、コードの見直し内容を説明することができます。

追加メモ記入欄

- _____
- _____
- _____
- _____

デバッグしよう！

ヘルプ・ミー！

5つのScratchプログラムをデバッグできますか？

このアクティビティでは、「デバッグしよう！」の各問題について、何がうまくいっていないのかを調査し、解決策を見つけ出します。

ここからスタート

- ユニット3用のDebug It!スタジオを開きます。
<http://scratch.mit.edu/studios/475554>
- スタジオの5つのデバッグの問題をテストし、デバッグします。
- 解決策を書き出したり、あなたの解決策を使ってバグのあるプログラムをリミックスしましょう。

行き詰まりましたか？

大丈夫！ 次のことを試してみてください...

- プログラム中で起こりうるバグを一覧にします。
- あなたの作業の経過を追ってみましょう！
この一覧は、すでに試したことを思い出すのに役立ち、次に試すべきことを示してくれます。
- 役立つものを発見するまで、友達と気づいた問題点と問題解決のアプローチを共有し、比較してみましょう！

[訳注1] MITが商標登録しているScratch上のキャラクタ

[訳注2] 演劇の用語で場面を変えること。場面転換とも言う。

□ デバッグしよう！ 3.1

<http://scratch.mit.edu/projects/24269007>

このプロジェクトでは、ScratchキヤットがGoboにニャーと鳴くのを教えます。Goboの番になった時に鳴こうとするはずですが、Goboは黙っています。どうプログラムを修正しますか？

□ デバッグしよう！ 3.2

<http://scratch.mit.edu/projects/24269046>

このプロジェクトでは、Scratchキヤットは、1から指定した数まで数えるはずですが、Scratchキヤットは必ず10まで数えてしまいます。どうプログラムを修正しますか？

□ デバッグしよう！ 3.3

<http://scratch.mit.edu/projects/24269070>

このプロジェクトでは、Scratchキヤットは、Goboの友達のGiga, Nano, Pico, Tera[訳注1]の点呼をとっていますが、全員がすぐに返事してしまいます。どうプログラムを修正しますか？

□ デバッグしよう！ 3.4

<http://scratch.mit.edu/projects/24269097>

このプロジェクトでは、ScratchキヤットとGoboがジャンプの練習をしています。Scratchキヤットが“Jump！（ジャンプ）”と言うと、Goboは上下にジャンプするはずですが、Goboはジャンプしません。どうプログラムを修正しますか？

□ デバッグしよう！ 3.5

<http://scratch.mit.edu/projects/24269131>

このプロジェクトでは、右矢印キーを押すと場面が転換[訳注2]します。プロジェクトのスターである恐竜は、場面が講堂になった時を除いて、いつも隠れています。講堂では恐竜が登場してダンスをするはずですが、恐竜は常に現れており、適切なタイミングでダンスしてくれません。どうプログラムを修正しますか？

終わった？

- + スクリプト内のブロックを右クリックしてコードにコメントを追加します。コメントを追加する事は、友達がプログラムの色々な部分を理解するのに役立ちます。
- + テストとデバッグの実践に関して、パートナーと話し合いましょう。あなたの戦略との類似点と相違点を書き出してみましょう。
- + 友達を手助けしましょう！

創造物の制作

▶ 推奨時間
15-30分

目的

このアクティビティを完了すると、生徒は、
+ スケッチを共同制作することで、コンピュータ上の実践（再利用とリミックス）が紹介されます。

アクティビティの説明

- このアクティビティでは、生徒は3つのパートに分けて“創造物”をスケッチします。
- 3分の1に折り畳んだ白い用紙を配布します。生徒ごとに以下の手順で、スケッチ（各自持ち時間1分）を共同制作してもらいます。[訳注1]
 1. 創造物の“頭の部分”をスケッチする。
 2. スケッチをつなぐ箇所（ガイド）を少し残して、紙を折り畳む。
 3. 頭の部分を隠したら、2番目の生徒に描きかけの創造物を手渡す。
 4. のぞき見しないで“頭の部分”のガイドにつないで、自分の創造物の“お腹の部分”を描き加える。
 5. スケッチをつなぐ箇所（ガイド）を少し残して、紙を折り畳む。
 6. お腹の部分を隠したら、3番目の生徒に描きかけの創造物を手渡す。
 7. のぞき見しないで“お腹の部分”のガイドにつないで、自分の創造物の“足の部分”を描き加える。
 完成したら用紙を広げて共同制作された創造物の正体を明かします！
- 壁やボードにスケッチを張り出し、生徒に、自分の創造的な共同制作の成果を見てもらいます。
- 共同著者、コラボレーション、再利用とリミックスの作品について、グループ討議をファシリテートします。

リソース

- 3分の1に折り畳まれた白い用紙（約8.5インチ×11インチ）[訳注2]
- スケッチ用具（鉛筆、ペン、マーカーなど）

振り返りを促す質問

- + あなたにとってリミックスとは何ですか？
- + “頭の部分”をスケッチした創造物を考えてみましょう。他の人の貢献によって、あなたのアイデアはどのように拡張されたり、高められましたか？
- + “お腹”や“足”の部分を描いた創造物を検討します。あなたの貢献は他の人のアイデアをどのように拡張したり、高めましたか？

作業の確認ポイント

- + 生徒は、リミックスとそのメリットを説明できるか？

[訳注1] 意識。1センテンスの原文を簡条書きにした。完成した実物（次頁の絵）を参照のこと。

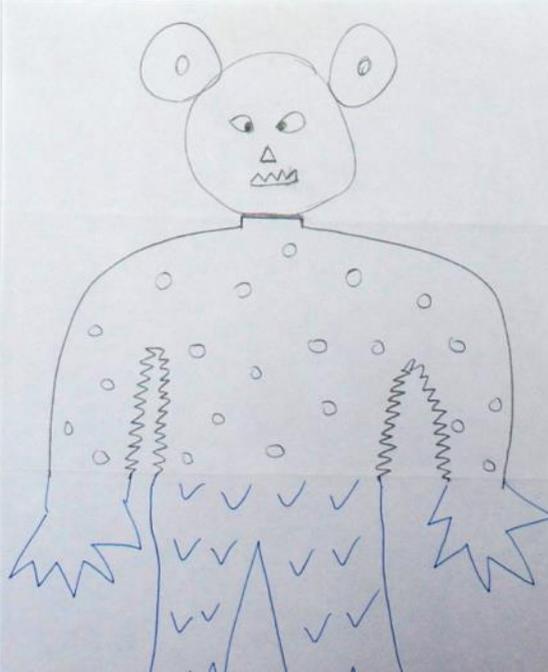
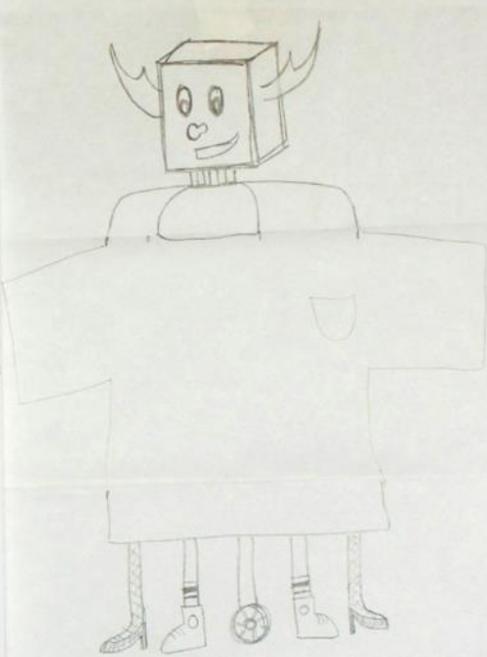
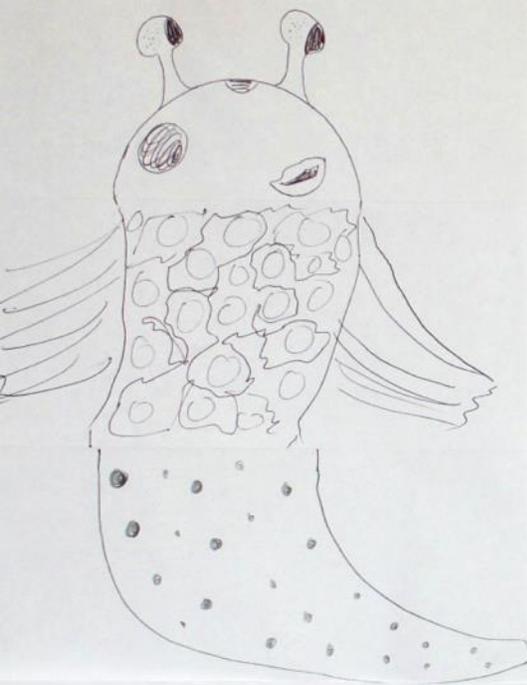
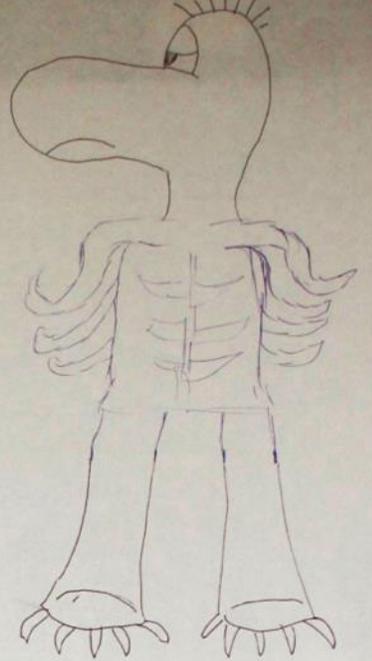
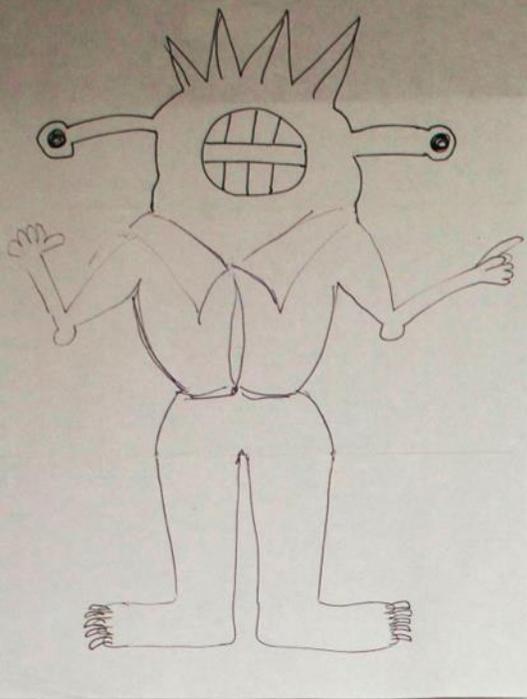
[訳注2] 原文のサイズは、レターサイズ。A4用紙で代替可能。

メモ

- + このアクティビティは、“続けて創ろう”プロジェクト用の申し分のないウォームアップ・アクティビティです。“続けて創ろう”の前に、直接、創造物の制作をフェシリテートすることを推奨します。
- + 必要に応じて、貢献したアーティストがわかるように、取り組んだ創造物の絵の下の方に自分の名前を記入してもらいます。

追加メモ記入欄

- _____
- _____
- _____
- _____



続けて創ろう! (Pass It On)

 **推奨時間**
45-60分

目的

- このアクティビティを完了すると、生徒は、
- + 他の人の作品を再利用し、リミックスすることで、物語をプロジェクトを制作できます。
 - + ストーリーテリング・プロジェクト[訳注1]を二人一組のペアで制作することで、ペア・プログラミング[訳注2]を体験します。

アクティビティの説明

- 二人一組のペアに分けます。生徒に、「続けて創ろう」の物語[訳注3]というプロジェクトのコンセプトを紹介します。それは二人一組のペアで物語を作り始めて、他の2つのペアに引き継いで、物語を展開し、イメージし直すプロジェクトのことです。必要に応じて、「続けて創ろう！」配布用プリントを印刷しておきます。
- 生徒が望むなら、どんな方法でも良いので取り掛かるよう促します。すなわち、登場人物、場面、筋書きに焦点を当てたり、エキサイトすることなら何にでも焦点を当ててプロジェクトに取り組んでもらいます。各ペアに10分与え、物語の共同制作に取り組ませます。その後、ローテーションして、プロジェクトをリミックスすることで別の物語へと展開させます。生徒にコンテンツの再利用やリミックスに対して謝辞を示すよう勧めます。
- 2回ローテーションした後、生徒に自分が貢献した物語のプロジェクトを再び訪れてもらいます。Scratchスクリーニング[訳注4]を開催するのがお勧めです。鑑賞するために周りに集めた生徒に、プロジェクターとスクリーンを使って物語プロジェクトを説明します。必要に応じて、生徒に自分のプロジェクトをPass It On スタジオかクラス・スタジオに追加するよう求めます。
- 生徒に、デザイン日誌やグループ討議の中で振り返りを促す質問に回答させて振り返ってもらいます。

メモ

- + Scratch上映会のイベントを開催することを検討して下さい！他のクラスの生徒を上映会に招待します。軽食や飲み物を提供したり、講堂やプロジェクトを表示するための大きな壁やスクリーンがある部屋でイベントを開催します。
- + プロジェクトをリミックスする別の方法としてバックパック(Scratchのプロジェクト・エディタの一番下にある)を生徒に紹介します。このツールの詳細については、Backpackのビデオチュートリアルを参照して下さい。
<http://bit.ly/scratchbackpack>

リソース

- 「続けて創ろう！」配布用プリント
- Pass It Onスタジオ
<http://scratch.mit.edu/studios/475543>
- 生徒の作品を表示するプロジェクターとスクリーン(オプション)

振り返りを促す質問

- + 他人の作品をリミックスしその上に構築してみて、何を感じましたか？リミックスされて、どう感じましたか？
- + あなたの生活のどこかで、再利用やリミックスを見たり、経験したことがありますか？例を2つ共有してください。
- + 他の誰かとの仕事は、今までのプロジェクトのデザイン作業の経験と、どのように違いましたか？

作業の確認ポイント

- + 生徒はプロジェクトのどんな部分に貢献したか？
- + 生徒は、イベント処理や並列処理の概念、再利用とリミックスの実践に対する不安が無さそうに見えるか？不安そうなら、どんな方法でこれらをさらに明確にすることができるか？

追加メモ記入欄

- _____
- _____
- _____
- _____

[訳注1] 読み聞かせを行うプロジェクト。声の代わりにフキダシが使用される。
[訳注2] 2人のプログラマーが1台のPCでコードを開発する手法。
[訳注3] 原文は、Pass-It-On Story
[訳注4] Scratchプロジェクトを生徒がオブザーブするアクティビティ

続けて創ろう!

他の人の作品を土台にして構築することによって何を創作できますか？

このプロジェクトでは、アニメ化した物語のプロジェクトの制作をスタートし、続けてその物語を友達に引き継いで、リミックスし、展開したり、想像し直します。



ここからスタート

- 登場人物、場面、筋書きに焦点を当てたり、エキサイトすることなら何にでも焦点を当てて物語プロジェクトに取り組みましょう。
- 10分ったら、プロジェクトを保存し、オンラインで共有します。
- ローテーションして、プロジェクトをリミックスすることで別の物語プロジェクトへと展開させます。
- 繰り返しましょう!



試してみること

- 物語をリミックスし、展開し、イメージし直すために、さまざまな可能性をブレインストーミングしましょう。最後に別の場面を追加します。物語が始まる前に何が起きたのか想像できますか。新たな登場人物を加えたら、どうなりますか。どんでん返しを付け加えるのはどうですか。他に何かありませんか。

- コードにコメントを追加すると、他の人がプログラムの色々な部分を理解するのに役立ちます。スクリプトにコメントを付けるには、ブロックを右クリックして説明を追加します。



使用するブロック



完成した?

- + Pass It Onスタジオにプロジェクトを追加しましょう。 : <http://scratch.mit.edu/studios/475543>
- + 友達を手助けしましょう!
- + あなたが貢献したすべてのプロジェクトに戻って、どう物語が進化したのか確認しましょう!

ユニット4 ゲーム



インデックス

内容



理想のゲームリスト	74
入門用ゲーム	76
スコア機能	80
機能拡張	82
対話処理	84
デバッグしよう!	86

ユニット4

概要

“ビッグ・アイデア”

「パーソナライゼーション」は、創造的コンピューティングの体験のデザインにおける、重要な指導原理です。我々にとって「パーソナライゼーション」とは、個人的な興味につなげることと、個人的な興味が大幅に変わる可能性があることと認めることの両方を意味しています。多くの知識と実践があり、そしてこれらの複数の方法を探究することは、若い学習者の間で興味、動機づけ、持続性を支えるのに役立ちます。

このユニットでは、学習者はゲームデザインに関連した高度な概念と難しい問題をいくつか探究します。個人的に意味のあるアクティビティに根差していれば、高度な概念や難しい問題でも理解を深められるようになります。力強いメッセージ[訳注1]のひとつの例として、私たちは、MITのScratchプロジェクトのディレクターであるMitch Resnickによって共有された話に注目しています。

「私は数年前、コンピュータ放課後クラブハウスのセンターのひとつで、13歳の男子が自身のゲーム作成に取り組んでいるのを見かけました。彼は、キャラクタ（この場合、魚）をコントロールできていました。彼はスコアを記録するゲームを欲しがりました。大きな魚に食べられた小さな魚が何匹いるのか、現実の世界では[訳注2]すぐにわかりますが、彼にはプログラムでどうすれば良いのかわかりませんでした。

私は、このことをチャンスとみて、変数のアイデアを紹介しました。私が変数のことを教えると、彼は、彼のゲームで何匹の魚が食べられたかを記録するために、変数のブロックをどう使えばいいのかわかりました。彼はそのブロックを取って大きな魚が小さな魚を食べるようにスクリプトの中に [訳注3]正しく置きました。彼は素早く試しました。案の定、大きな魚が小さな魚を食べるたびに、スコアは1ずつ加算されたのです。

彼は実際に変数を使いたかったので、本当に変数を深く理解したのだと、私は考えています。このことは、Scratchの包括的なゴールのひとつです。変数に関するだけでなく、全てのタイプの概念のためのものです。子供たちが意味があってモチベートされる方法でその概念を利用している時こそ、学んだ概念をはるかに深く理解するのだと、我々は理解しています。」

[訳注1] 原文は、the power of context

[訳注2] 補足（現実の世界）を追加

[訳注3] スクリプトタブの中のスクリプトを配置するエリアを指す。

学習の目的

生徒は、

- + コンピュータ処理の概念（条件分岐、演算子、データ（変数とリスト））を紹介されます。
- + 自らで方向性を決めた迷路、ピンポンゲーム、スクロールゲームのプロジェクトを制作し、拡張することで、コンピュータ上の実践（実験と反復、テストとデバッグ、抽象化とモジュール化）に、さらに親しみを覚えるようになります。
- + 一般的なゲームのメカニズムを特定し、理解します。



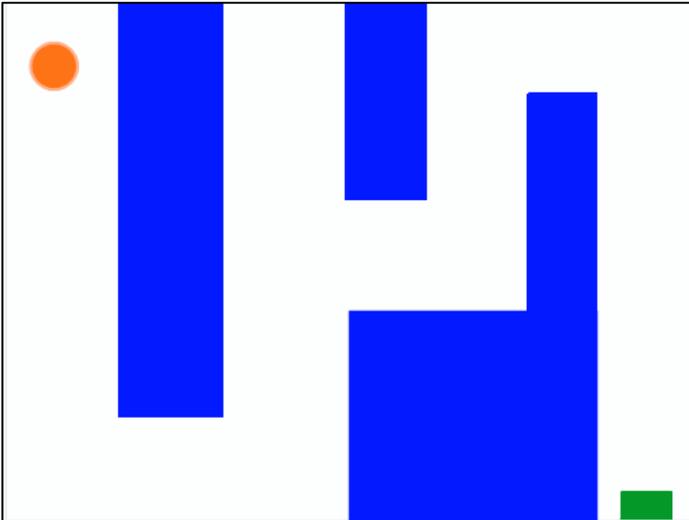
キーワード、概念、実践

- | | |
|--------------|---------------|
| + 抽象化とモジュール化 | + センサー |
| + 条件分岐 | + フィードバック・フェア |
| + 演算子 | + アーケード・ディ |
| + データ | + パズル・ジャー |
| + 変数とリスト | + プレイン・ダンブ |

メモ

- + このユニットでは、多くの新しい概念を探究していきます。そのため、サンプルプロジェクトのスタジオ、追加の実践用のプログラムされたパズル、入門用ゲームプロジェクトの形で付け加えられたサポート資料が含まれています。必要に応じて、リミックスしたり、再利用したりすることをお勧めします。

アドベンチャーを選択しよう



このユニットでは、学習者はゲームデザイナーとなり、独自のゲームプロジェクトを制作する体験ができます。生徒は、このユニットのアクティビティのガイドに沿って、ゲームの仕組みとゲーム開発の手ほどきされる、一方でコンピュータ処理の概念（条件分岐、演算子、データ）とコンピュータ上の実践（抽象化とモジュール化）の理解を深めます。

入門用ゲームのアクティビティを使って生徒にゲームプロジェクトに取り掛かせ、他のアクティビティを通してさらなる成長をサポートすることができます。ユニット4のアクティビティは、スコアや横スクロールなどの一般的なゲームの仕組みを学ぶことから、マルチプレイヤーゲーム（例えば、ピンポンゲーム）の作成に至るまで、ゲーム開発を実践するための複数の機会を生徒に与えます。

可能なルート

セッション1



理想の
ゲームリスト

すべてのゲーム
に共通するもの
は何ですか？

セッション1 - 5



入門用
ゲーム

Scratchを使っ
てインタラクティブ
なゲームを制作
するには、どうす
れば良いです
か？

セッション2



スコア

変数を使って
ゲームにスコア機
能を追加するに
は、どうすればい
いですか？

セッション3



機能拡張

ゲームを拡張し
難易度を上げる
にはどんな方法
がありますか？

セッション4



対話処理

9つのScratch
でプログラムされ
たパズルに取り
組めます。

セッション5



デバッグしよ
う！

助けて！
5つのScratch
プログラムをデ
バッグできます
か？

理想の ゲームリスト

🕒 推奨時間
15-30分

目的

このアクティビティを完了すると、生徒は、
+ 一般的なゲームのデザイン要素を確認します。

アクティビティの説明

- ❑ 生徒を、2、3人の小グループに分けます。
- ❑ 小さなグループで、生徒が楽しく遊んでいるゲームをリストアップしてもらいます。デザイン日誌や紙の用紙を使用してリストを作成することができます。ブレインダンプ[訳注1]というブレインストーミングのアクティビティをファシリテートするのがお勧めです。生徒に少し時間（1～2分）を与えて、できるだけ多くのゲームを書き出してもらい、次にそのブレインダンプしたリストから、お気に入りのゲームだけに絞り込んでもらいます。
- ❑ 数分後、絞り込んだゲームリストについてグループに質問します。
ゲームの共通点は何ですか。
それらのデザインのどの機能がゲームにしていますか。
- ❑ 「どんな特性がゲームを作っているのか」についてのクラス討論会をファシリテートし、クラス全体で共通的なゲームのメカニズムのリストを作り出します。
続けて、生徒に理想のゲームを想像してもらって、ゲームのデザイン要素をリストアップしてもらいます。
- ❑ フィードバックや提案してもらうために、生徒に自分の小グループや批評グループの中で理想のゲームリストを共有してもらいます。
(ユニット0の“批評グループ”のアクティビティを参照)

リソース

- ❑ ゲームのデザイン要素を書き出すための紙（任意）
- ❑ スケッチ用具（鉛筆、ペン、マーカー等）

振り返りを促す質問

- + お気に入りのゲームをリストアップしましょう。
- + ゲームの共通点は何ですか？
- + それらのデザインのどの機能がゲームにしていますか？
- + あなたの理想のゲームのためのデザイン要素をリストアップしてください。

作業の確認ポイント

- + 理想のゲームリストにはゲームの機能が含まれているか？
- + どんなデザイン要素が、クラスのグループリストと類似しているか、相違しているか？
- + 生徒が楽しんでいるゲームの種類とプレイタイプついて、そのリストから何がわかるか？

[訳注1] テーマについて、頭に浮かんだことを短時間ではき出す手法。

メモ

- + 生徒に、ユニット4の他のアクティビティで、ゲームをプログラミングする際に、この「理想のゲームリスト」に戻って参照するよう促します。

追加メモ記入欄

- ❑ _____
- ❑ _____
- ❑ _____
- ❑ _____

Chess

Monopoly

Mario

Clue

Football

Candyland

Pac Man

Jump Rope

Baseball

Tennis

Flappy Bird

Wheel of Fortune

Four Square

入門用 ゲーム

 **推奨時間**
45-60分

目的

このアクティビティを完了すると、生徒は
+ 自らで方向性を決めたゲームプロジェクトに取り
組むことで、コンピュータ処理の概念（条
件分岐、演算子、データ）と実践（実験と
反復、テストとデバッグ、再利用とリミックス、
抽象化とモジュール化）を使いこなす能力を
身に付けます。

アクティビティの説明

- このアクティビティで、生徒は、スコア、機能拡張、対話処理[訳注1]のアクティビティ中に再び訪れたり、拡張したりできる入門用ゲームプロジェクトを作成します。
必要に応じて、迷路、ピンポンゲーム、スクロールゲームの入門用プロジェクトのサンプルを見せます。そして、生徒をガイドする「迷路」、「ピンポンゲーム」、「スクロール」配布用プリントを用意します。
- クラスとしてファシリテートしてゲームプロジェクトを1つ選ぶか、制作したいゲームを生徒に選ばせます。：
迷路、ピンポンゲーム、スクロールゲーム
生徒に時間を与えて、ゲーム制作を始めさせるか、入門用プロジェクトの中の1つをリミックスさせます。
- 生徒が制作中のゲームについてのフィードバックをもらうよう促します。半分の生徒は、自分のプロジェクトをオープンしたままで席にとどまり、残り半分の生徒は歩き回ってプロジェクトを探索したり、質問したり、フィードバックします。それから交代します。
必要に応じて、出来上がったゲームプロジェクトをゲームスタジオまたはクラス・スタジオに追加させます。
- 生徒に、デザイン日誌やグループ討議の中で振り返りを促す質問に回答してもらいます。

リソース

- 「迷路」配布用プリント
- Maze Starter サンプルプロジェクト
<http://scratch.mit.edu/projects/11414041>
- 「ピンポンゲーム」配布用プリント
- Pong Starter サンプルプロジェクト
<http://scratch.mit.edu/projects/10128515>
- 「スクロール」配布用プリント
- Scrolling Starter サンプルプロジェクト
<http://scratch.mit.edu/projects/22162012>
- Game スタジオ
<http://scratch.mit.edu/studios/487504>

振り返りを促す質問

- + 自分ゲームのデザインについてチャレンジした事は何か？
- + 自慢できることは何ですか？

作業の確認ポイント

- + 条件分岐、演算子、データを含んだゲームになっているか？

メモ

- + 完成を祝して、最終版のゲーム作品を共有するために、アーケードデー[訳注2]を開催するのがお勧めです。
出来上がったゲームプロジェクトをプレゼンテーションモードにしておきます。生徒は、歩き回り、お互いのゲームで遊びます。
- + スクロールゲームのオプションとしてクローン機能を紹介します。ユニット5の“高度な機能”の「クローン」配布用プリントを使って、クローンのブロックの詳細を生徒に教えます。

[訳注1] 原文ではInteraction。本ガイドではプロジェクトとユーザーとの間の双方向でやりとりを対話処理と表現する。

[訳注2] プロジェクトをオープンにしたたまに見せ合うイベント。

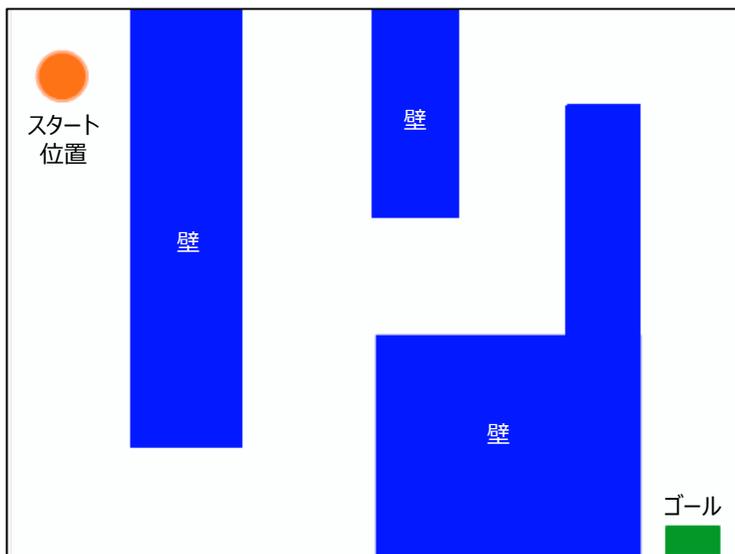
追加メモ記入欄

- _____
- _____
- _____
- _____

迷路

Scratchを使ってインタラクティブゲームを制作するにはどうすれば、いいですか？

このプロジェクトでは、ゲームを制作します。このゲームには、スプライト間の対話処理[訳注1]、スコア、レベル付け[訳注2]が含まれています。壁に触れることなく、迷路のスタート位置からゴールまで、スプライトを動かしましょう。



ここからスタート

- 迷路に適した背景を描きましょう。迷路の出口の目印[訳注3]は、壁と異なる色を使います。
- スプライトを追加します。
- あなたのゲームをインタラクティブにします！

試してみること

- ゲームに複数のレベルを追加してみよう。レベル付けは色々な背景を使って実現します。メッセージ送信（“～を送る”など）のブロックを使って次のレベルのトリガー[訳注4]とします。
- “変数を作る”の機能を使って、スコアをつけてみましょう。
- 迷路に新たなチャレンジを加えるために、“タイマー”ブロックを試してみましょう。



プレイヤーが迷路上でスプライトの動きをコントロールするスクリプト



スプライトにスタートする場所を教え、スタート位置につかせるスクリプト

このスプライト（迷路の出口）がボールに触れた時に、プレイヤーが勝った事を話すスクリプト



スプライトを迷路の青色の壁から跳ね返らせるスプライト

使用するブロック

完成した？



- + Gameスタジオにプロジェクトを追加しましょう。
<http://scratch.mit.edu/studios/487504>
- + パートナーとゲームを交換し、作品をお互いでワークスルー[訳注5]しましょう。

[訳注1] プロジェクトとユーザーが双方向でやりとりすることです。
[訳注2] 難易度をつけることです。
[訳注3] この迷路ゲームのゴールです。
[訳注4] 次のレベル用のスクリプトを開始させることです。

[訳注5] 製作者が主体となって、パートナーにゲームをプレイして見せたり、体験してもらい、良い点、改善点を指摘してもらいイベントです。

ピンポンゲーム

Scratchを使ってインタラクティブゲームを制作するには、どうすればいいですか？

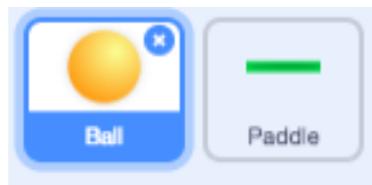
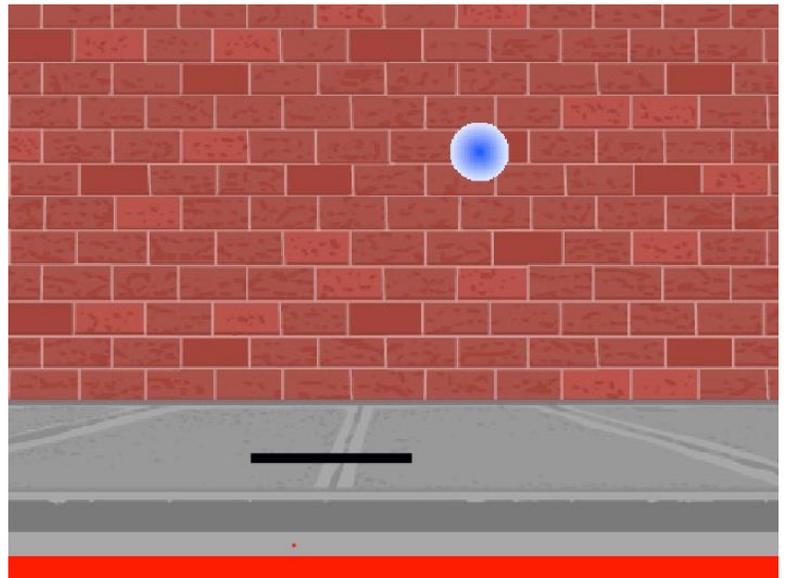
このプロジェクトでは、ゲームを制作します。このゲームには、スプライト間の対話処理[訳注1]、スコア、レベル付け[訳注2]が含まれています。このゲームは古典的なピンポンゲームに似ており、スプライトが地面に落ちないようにするのが、このゲームのゴールです。

ここからスタート

- スプライトを2つ作成します。:
コントロールするパドル、ユーザが対戦するボール
- パドルのスプライトをインタラクティブ[訳注3]にします。
- ゲームらしくします！

試してみるごと

- どうやってゲームに難易度を加えますか？
いくつかの可能な例として、色々なレベルの作成、タイマーの使用、スコアの保持があります。
- 背景を編集してゲームの外観を変えてみましょう！
- 色々なキーを使ってスプライトをコントロールしてみましょう！



パドルとの対話処理[訳注5]



壁との対話処理[訳注4]

これらはボールをコントロールします。パドルや壁に触れている限り動き続けますが、赤い線に触れたら（地面に落ちることを意味します）ゲームエンドです。

使用するブロック

完成した？

Key events: スペース キーが押されたとき, 上向き矢印 キーが押されたとき, m キーが押されたとき, メッセージ1 を受け取ったとき

Variables: 作った変数 を 0 にする, 作った変数 を 1 ずつ変える, 変数 作った変数 を表示する, 変数 作った変数 を隠す

Logic/Math: +, -, > 50, = 50, < 50, ではない, かつ, または, 色が 色に触れた, タイマー タイマーをリセット, 色に触れた, マウスのポインターに触れた, 1 から 10 までの乱数

+ Gameスタジオにプロジェクトを追加しましょう。
<http://scratch.mit.edu/studios/487504>

+ パートナーとゲームを交換し、作品をお互いでウォークスルー[訳注6]しましょう。

[訳注1] プロジェクトとユーザーが双方向でやりとりすることです。
[訳注2] 難易度をつけることです。
[訳注3] 双方向性をもたせることです。矢印キーでパドルを動かせるようにします。
[訳注4] ボールが端に着くと跳ね返ります。

[訳注5] ボールがパドルに振れると方向を変え、赤色にふれると停止します。
[訳注6] 製作者が主体となって、パートナーにゲームをプレイして見せたり、体験してもらい、良い点、改善点を指摘してもらうイベントです。

スクロール

Scratchを使ってインタラクティブゲームを制作するには、どうすればいいですか？

このプロジェクトでは、ゲームを制作します。このゲームには、スプライト間の対話処理[訳注1]、スコア、レベル付け[訳注2]が含まれています。

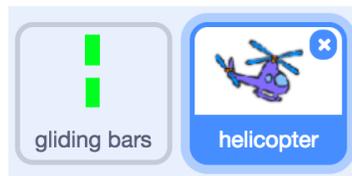
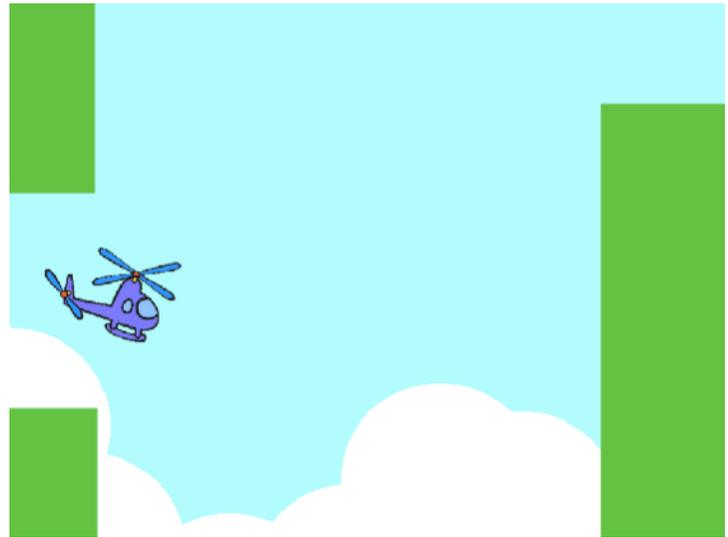
このゲームは羽ばたく鳥（Flappy Bird）に似ており、物体が地面に落ちたり、特定のオブジェクトに触れないようにすることが、このゲームのゴールです。

ここからスタート

- スプライトを2つ作成します。プレイヤーがコントロールするスプライト(helicopter)と回避するためのスプライト(gliding bars)です。
- “Helicopter”をインタラクティブ[訳注3]にします。
- ステージ上を“gliding bars”がスクロールするスクリプトを加えて、ゲームらしくします

試してみることに

- どうやったらゲームの難易度を上げられますか？いくつかの可能な例として、タイマーの使用、スコアの保持があります。
- 背景を編集してゲームの外観を変えてみましょう！
- 色々なキーを使って、スプライトをコントロールしてみましょう！



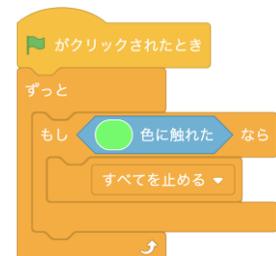
クローンを作成するスクリプト。下記のスクリプトでバーをスクリーンを上から下にスクロールさせるために使用される。



スプライトの動きをコントロールする。



スプライトを絶えず落下させる。



ゲームが終了する条件を指定する。

使用するブロック

完成した？



+ Gameスタジオにプロジェクトを追加しましょう。
<http://scratch.mit.edu/studios/487504>

+ パートナーとゲームを交換し、作品をお互いでウォークスルー[訳注4]しましょう。

[訳注1] プロジェクトとユーザーが双方向でやりとりすることです。

[訳注2] 難易度をつけることです。

[訳注3] 双方向性をもたせることです。絶えず落下するヘリコプターをスペースキーで浮上させます。

[訳注4] 製作者が主体となって、パートナーにゲームをプレイして見せたり、体験してもらい、良い点、改善点を指摘してもらうイベントです。

スコア機能

 **推奨時間**
30-45分

目的

- このアクティビティを完了すると、生徒は
- + 変数とは何か、なぜ有用なのかを説明できます。
 - + コンピュータ処理の概念（データ）を紹介されます。
 - + プロジェクトやプロジェクトの一部のリミックスと再利用を体験します。

アクティビティの説明

- 必要に応じて、グループでFish Chomp入門プロジェクトを検索したり、生徒をガイドする「スコア」配布用プリントを用意します。
- 生徒が、Fish Chomp入門プロジェクトを開くのをガイドします。生徒に変数を調べる時間を与え、Fish Chomp入門プロジェクトをリミックスして、ゲームにスコアを追加してもらいます。必要に応じて、生徒に時間を与え、着手済みの、迷路、ピンポンゲーム、スクロールゲームのプロジェクトにスコアを組み込んでもらいます。
- 生徒に、Fish Chompのリミックスやスコアのあるゲームプロジェクトを共有してもらいます。デザインデモのアクティビティがお勧めです。数人の生徒に、グループに向けてプロジェクトをプレゼンし、変数を使うスコア機能をどうやって組み込んだのかをデモンストレーションするよう促します。必要に応じて、生徒にFish Chomp Remixスタジオやクラス・スタジオに、リミックスを追加してもらいます。
- 生徒に、デザイン日誌やグループ討議の中で振り返りを促す質問に回答させてデザインのプロセスを振り返ってもらいます。

リソース

- 「スコア」配布用プリント
- Score Samples スタジオ
<http://scratch.mit.edu/studios/218313>
- Fish Chomp入門プロジェクト
<http://scratch.mit.edu/projects/10859244>
- Fish Chomp Remix スタジオ
<http://scratch.mit.edu/studios/475615>

振り返りを促す質問

- + 変数を他の人にどうやって説明しますか？
- + 変数は何のために役立ちますか？

作業の確認ポイント

- + 生徒は、変数が何か、変数が何に役立つかを説明できるか？

メモ

- + Score Samples スタジオでサンプルプロジェクトのコードを調べることで、生徒に変数について理解している事を明らかにするよう促します。
- + 変数は、数学的にもコンピュー処理上の概念にも重要です。生徒は、数学や理科の教室で変数について教えられますが、多くの生徒にとって学ぶのが難しい時間になっています。ゲームは、変数の有用性をより具体的にする方法の1つです。

自身に対する注釈

- _____
- _____
- _____
- _____

スコア機能

Scratchを使ってインタラクティブゲームを制作するにはどうすればよいですか？

Fish Chompは、プレイヤーがマウスでスプライトを誘導して、できるだけ多くの魚を捕まえるゲームです。このアクティビティでは、変数を使ってスコア機能を追加して、Fish Chompをリミックスします。



ここからスタート

- ❑ Fish Chomp入門プロジェクトのページを開きます。
<http://scratch.mit.edu/projects/10859244>
- ❑ "データ"カテゴリの"変数を作る"ボタンをクリックして、スコアの変数を作成し、名前を付けます。
- ❑ 新しい変数ブロックを試して、プロジェクトにスコア機能を組み込みましょう！



行き詰まりましたか？
大丈夫！次ことを試してみてください...

完成した？

- ❑ 変数を扱う方法がわかりませんか？
詳細は、以下プロジェクトをチェックしてみましょう。
<http://scratch.mit.edu/projects/2042755>
- ❑ あるいは、以下のビデオを参照してください。
<http://youtu.be/uXq379XkhVw>
- ❑ 変数の作成とプロジェクトへの組み込みの詳細を調べるには、スコアを使っているゲームのコードを調べて研究してみましょう。

- + Fish Chomp Remixスタジオにプロジェクトを追加しましょう。
<http://scratch.mit.edu/studios/475615>
- + あなた自身にもっとチャレンジしましょう！
- + もっと難易度の高いゲームデザインにするには、どのようにスコア機能を使用しますか？
- + ひらめきが得られたゲームを見つけて、リミックスしましょう！

機能拡張

 **推奨時間**
30-45分

目的

このアクティビティを完了すると、生徒は
+ 一般的なゲームのメカニズムを説明しているプログラムを調べることで、概念（条件分岐、演算子、データ）に、さらに親しみを覚えるようになります。

アクティビティの説明

- 必要に応じて、Extensionsスタジオのサンプルプロジェクトを示し、生徒をガイドする「機能拡張」配布用プリントを用意します。
- 生徒にExtensionsスタジオでプログラムのコードを調べる時間を与え、ゲームの難易度を上げたり拡張したりする色々なやり方を調査します。
着手済みの迷路、ピンポンゲーム、スクロールゲームのプロジェクトに追加する機能拡張[訳注1]を1つ以上、選択してもらいます。生徒に時間を与え、自分のゲームに機能拡張を組み込んでもらいます。
- 生徒に拡張されたゲームプロジェクトを互いに共有してもらいます。生徒が自分のゲームを共有したり、学んだことをデモンストレーションできるよう、ペアシェアやデザインデモのアクティビティをファシリテートすることを推奨します。
- 生徒に、デザイン日誌やグループ討議の中で振り返りを促す質問に回答させてデザインのプロセスを振り返ってもらいます。

リソース

- 「機能拡張」配布用プリント
- Extensions スタジオ
<http://scratch.mit.edu/studios/475619>

振り返りを促す質問

- + ゲームの難易度を上げるための色々な方法には、何がありますか？
- + ゲームプロジェクトにどんな機能拡張を追加しましたか？
- + 機能拡張をゲームに組み込む手順を説明してください。

作業の確認ポイント

- + 生徒は、オリジナルのゲームプロジェクト[訳注2]に機能拡張を追加することができたか？

メモ

- + 追加のサポートが必要な生徒に、より多くの足場を提供するために、「クラスで機能拡張のサンプルプログラムのひとつをウォークスルーすること（例：レベル付け）」と「彼らのゲームプロジェクトに機能拡張を追加するのを手助けすること」を提案します。
- + バックパックツールは、生徒が機能拡張のプロジェクトの一部を自分の入門用ゲームに組み込むことができる方法の1つです。バックパックについての詳細は、以下を参照して下さい。
<http://bit.ly/scratchbackpack>

追加メモ記入欄

- _____
- _____
- _____
- _____

[訳注1] ScratchのExtensions(拡張機能)と区別するため、このアクティビティのExtensionsは“機能拡張”と訳す。

[訳注2] 以前のアクティビティで作成したゲームプロジェクトを指す。

機能拡張

Scratch上のゲームを拡張し、イメージし直すにはどうすればよいですか？

自分のScratchプロジェクトの中に機能拡張を追加してゲームのデザインに取り掛かります！
右の機能拡張を少なくとも1つ（または複数）選択して、着手済みの迷路、ピンポンゲーム、スクロールゲームに追加します。

ここからスタート

- Extensions スタジオを開きます。
<http://scratch.mit.edu/studios/475619>
- 機能拡張の1つ（または複数）を選んで、調べてください。
- 選んだ機能拡張を着手済みのゲーム・プロジェクトに組み込みましょう！

- + **スコア** <http://scratch.mit.edu/projects/1940443>
スコアの設定と変更方法のデモンストレーションです。Scratchキャットがクリックされるたびに10ポイントもらいます。
- + **レベル付け** <http://scratch.mit.edu/projects/1940453>
レベルを変更するデモンストレーションです。スペースを押すたびにスコアが1ずつ増えます。10ポイントごとにレベルが1ずつ上がります。
- + **タイマー** <http://scratch.mit.edu/projects/1940445>
タイマーの使用方法のデモンストレーションです。マウスを使ってScratchキャットをGobolに移動します。
- + **敵** <http://scratch.mit.edu/projects/1940450>
敵の追加方法のデモンストレーションです。上下の矢印キーを使ってテニスボールを避けてください。
- + **ご褒美** <http://scratch.mit.edu/projects/1940456>
アイテムを集め方のデモンストレーションです。矢印キーを使用してScratchキャットを移動して探しものを集めます。
- + **マウス** <http://scratch.mit.edu/projects/25192659>
ゲームプレイをコントロールするマウスのプログラミング方法のデモンストレーションです。マウスを動かしてパドルを動かします。
- + **リスタート** <http://scratch.mit.edu/projects/25192935>
リスタートボタンの作成方法のデモンストレーションです。リスタートするには、RESTARTボタンをクリックします。
- + **メニュー** <http://scratch.mit.edu/projects/25192991>
ゲーム開始時のメニュー画面の表示方法のデモンストレーションです。メニュー画面でSTARTまたはDIRECTIONSをクリックします。
- + **マルチプレー** <http://scratch.mit.edu/projects/25192711>
他のプレイヤーの追加方法のデモンストレーションです。プレイヤー1は矢印キーを使って迷路の中のPicoをナビゲートし、プレイヤー2はW, A, S, Dキーを使って迷路の中のNanoをナビゲートします。

試してみよう

- + バックバックは、Scratchでプログラミングする際に、とても便利なツールです。コードの行から音楽ファイル、スプライトに至るまで、すべて保存できます。バックバックを使って機能拡張を自分のゲームプロジェクトに組み込みましょう。
- + あるいはデザイン日誌にアイデアやコードをスケッチすることは、機能拡張を組み込む方法をプランする上で優れた方法の1つです。

終わった？

- + あなたの迷路、ピンポンゲーム、スクロールゲームに別の機能拡張を追加しましょう。
- + あなた自身にもっとチャレンジしましょう！各機能拡張の調査を続け、あなたのゲームに追加しましょう。
- + 友達を手助けしましょう！
- + 友達と自分のプロジェクトを共有し、お互いでフォードバック合いましょう。

対話処理 (インタラクション)

 **推奨時間**
30-45分

目的

- このアクティビティを完了すると、生徒は
- + 一連の9つのパズル・プログラムを解くことで、プロジェクトをインタラクティブにするさまざまなアプローチを探究します。
 - + コンピュータ処理の概念（条件分岐、演算子、データ）と実践（テストとデバッグ）を自在に使いこなせるようになります。

アクティビティの説明

- 自分達だけで、または2～3人の少人数のグループで、Scratchプログラムを作成して、9つの対話処理[訳注1]プログラミングパズル（謎解き[訳注2]）のそれぞれを解決することによって、生徒がさらにScratchを探究するようチャレンジさせます。これらの対話処理パズルは“調べる”のブロックを探し、対話性に関連するScratchのより高度な概念のいくつかに取り組みます。必要に応じて、生徒をアクティビティの間にガイドする「対話処理」配布用プリントを用意します。
- 各パズルにはいくつかの解き方があります。生徒やグループに、色々な解き方や作戦を共有してもらいます。生徒が自分の作品を共有し、その過程を説明できるようにするため、ペアシェアやデザインデモのアクティビティがお勧めです。必要に応じて、生徒にInteractionsスタジオまたはクラス・スタジオにプロジェクトを追加してもらいます。
- 生徒に、デザイン日誌やグループ討議の中で振り返りを促す質問に回答させて、このチャレンジを振り返ってもらいます。

リソース

- 「対話処理」配布用プリント
- Interactions スタジオ
<http://scratch.mit.edu/studios/487213>

振り返りを促す質問

- + どのパズルに取り組みましたか？
- + パズルを解くためにどんな作戦を取りましたか？
- + どのパズルが、あなたゲームのプロジェクトを考えるのに役立ちましたか？

作業の確認ポイント

- + パズルが解けているか？
- + 生徒はパズルを解くための他のアプローチを試したか？
- + 未だに生徒が苦労しているブロックや概念があるか？ どうやったら手助けできるか？

メモ

- + 生徒に探究して欲しいと思っている新しいブロックや概念にハイライトを当てている特定の課題を選択します。あるいは、独自の対話処理のパズルの台本 [訳注3]を考案してもらいます。
- + 他のアクティビティを早く終わらせている生徒のために非構造的なアクティビティ[訳注4]として、またはウォーミングアップのチャレンジとして、これらのパズルを再利用します。
パズル・ジャーの作成方法：
各パズルの説明のコピーを印刷してカットし、重ねて瓶の中に入れておきます。生徒に瓶からパズルを選ばせて解かせます。

[訳注1] プロジェクトとユーザーが双方向でやりとりすること

[訳注2] 一般的なパズルゲームではなく、謎解きする課題を指している

[訳注3] ユーザー入力を促す（ボタン、キー入力、スプライトなど）台本

追加メモ記入欄

- _____
- _____
- _____
- _____

[訳注4] あらかじめ組込まれているユニットや順序を決めていないアクティビティ。ここでは、ランダムにパズルを選んで、パズルを解きます。

対話処理 (インタラクション)

Scratchプロジェクトは、静止画像やビデオと何が違いますか？ [訳注1]

対話性に関連するScratchのさらに高度な概念のいくつかに参与するパズル（謎解き）に取りかかります。これらのパズルには、それぞれいくつかの解決策があります。

ここからスタート

- 9つの各対話型パズル用のScratchプログラムを作成します。

[訳注1] 対話のない静止画やビデオとの対比

行き詰まりましたか？
大丈夫！ 次のことを試してみてください...

□ **パズル1**： Bキーを押すたびにスプライトが大きくなり、Sキーを押すたびに小さくなります。

□ **パズル2**： スプライトが大きな音を聞いたときに、色が変わります。

□ **パズル3**： スプライトがスクリーンの上（25%）にいくたびに、「私、上にいるのが好き(I like it up here)」と言います。

□ **パズル4**： スプライトが青色のものに触れると高い音が鳴り、赤い部分に触れると低い音が鳴ります。

□ **パズル5**： 2つのスプライトがぶつかるたびに、1人が「失礼！（Excuse me）」と言います。

□ **パズル6**： 猫のスプライトが犬のスプライトの近くに来ると、犬はターンして猫から走って逃げます。

□ **パズル7**： 背景をクリックするたびに、その場所に花が出現します。

□ **パズル8**： スプライトをクリックするたびに、他のすべてのスプライトがダンスします。

□ **パズル9**： マウスポインタを動かすたびに、スプライトが追いかけます。しかし、マウスポインタには触れません。

完成した？

- Scratchで作業を始める前に、それぞれの対話型パズルの考えられるプログラミング方法を求めて、デザイン日記にアイデアを書き出してみましょう。
- 友達と一緒に作業しましょう。パートナーとのコラボレーションすることは、問題を解決したり、Scratchでプログラミングする方法についての新しい視点を獲得するための最適な方法です！

- + Interactionsスタジオに制作したプロジェクトを追加しましょう。 <http://scratch.mit.edu/studios/487213>
- + 友達を手助けしましょう！
- + 各パズルにアプローチするための戦略についてパートナーと話し合います。自分の手順との類似点と相違点についてメモを残しておきましょう。

デバッグしよう！

▶ **推奨時間**
15-30分

目的

- このアクティビティを完了すると、生徒は
- + 5つの課題に対する問題を調査してデバッグし、解決策を見つけ出します。
 - + テストとデバッグの実践を通して、さまざまな概念（条件分岐、演算子、データ）を探究します。

アクティビティの説明

- 必要に応じて、アクティビティ中で、生徒をガイドするユニット4「デバッグしよう！」配布用プリントを用意しておきます。
- ユニット4「デバッグしよう！」配布用プリントに記述されているプロジェクトへのリンクをたどってみせ、生徒が、Unit 4 Debug It!スタジオのプログラムを開くのをガイドします。生徒に、“中を見る”ボタンをクリックするように促して、バグのあるプログラムを調べて、コードをあちこち変えてみて、可能性のある解決方法を試してもらいます。
- 生徒に時間を与え、それぞれの「デバッグしよう！」の問題をテストし、デバッグしてもらいます。必要に応じて、生徒にScratchのリミックス機能を使って、バグを解決し修正済みのプログラムを保存してもらいます。
- 生徒に、デザイン日誌やグループ討議の中で振り返りを促す質問に回答させてテストやデバッグの体験を振り返ってもらいます。
- 生徒の問題発見と問題解決のアプローチを集めて、クラス全体のデバッグ戦略リストを作成します。

リソース

- ユニット4「デバッグしよう！」配布用プリント
- ユニット4 Debug It! スタジオ
<http://scratch.mit.edu/studios/475634>

振り返りを促す質問

- + 何が問題でしたか？
- + どうやって問題を特定しましたか？
- + どうやって問題を解決しましたか？
- + 他の人は、問題を解決する別のアプローチを取りましたか？

作業の確認ポイント

- + 生徒は、バグをすべて解決できたか？
そうでない場合は、解決できなかったプログラムで示されている概念をどうやったら明確できるか？
- + 生徒は、どんなテストとデバッグの戦略を採用したのか？

メモ

- + このアクティビティでは、特に条件分岐（例：if）、演算子（例：算術、論理）、データ（例：変数、リスト）の概念を中心に、追加的な配慮とサポートを必要としている生徒と一緒にチェックイン[訳注1]する機会を提供します。

追加メモ記入欄

- _____
- _____
- _____
- _____

[訳注1] 参加者がお互いの率直な気持ち、気がかりな事、期待などを共有する時間を取ることに

デバッグしよう！

ヘルプ・ミー！

5つのScratchプログラムをデバッグできますか？

このアクティビティでは、「デバッグしよう！」の各問題について、何がうまくいっていないのかを調査し、解決策を見つけ出します。

ここからスタート

- ユニット4用のDebug It!スタジオを開きます。
<http://scratch.mit.edu/studios/475634/>
- スタジオの5つのデバッグの問題をテストし、デバッグします。
- 解決策を書き出したり、あなたの解決策を使ってバグのあるプログラムをリミックスしましょう。

行き詰まりましたか？

大丈夫！ 次のことを試してみてください...

- プログラム中で起こりうるバグを一覧にします。
- あなたの作業の経過を追ってみましょう！
この一覧は、すでに試したことを思い出すのに役立ち、次に試すべきことを示してくれます。
- 役立つものを発見するまで、友達と気づいた問題点と問題解決のアプローチを共有し、比較してみましょう！

□ デバッグしよう！ 4.1

<http://scratch.mit.edu/projects/24271192>

このプロジェクトでは、Scratchキヤットが新しいアイテムをピックアップするたびに、「インベントリ」リスト[訳注1]を更新する必要がありますが、ラップトップだけしか持ち上げることができません。どうプログラムを修正しますか？

□ デバッグしよう！ 4.2

<http://scratch.mit.edu/projects/24271303>

このプロジェクトでは、Scratchキヤットは黄色のGobosを集めると10ポイントを獲得し、ピンクのGobosにぶつくと10ポイントを失いますが、うまく動きません。どうプログラムを修正しますか？

□ デバッグしよう！ 4.3

<http://scratch.mit.edu/projects/24271446>

このプロジェクトでは、Scratchキヤットは1から10の数字を考えていますが、予測の値のチェックに何か問題があります。いつも、うまく動きません。どうプログラムを修正しますか？

□ デバッグしよう！ 4.4

<http://scratch.mit.edu/projects/24271475>

このプロジェクトでは、Scratchキヤットにテニスボールが当たるたびに、「# of hits」の表示[訳注2]が1ずつ加算されるはずですが、当たったとき「# of hits」は1より多く加算されてしまいます。どうプログラムを修正しますか？

□ デバッグしよう！ 4.5

<http://scratch.mit.edu/projects/24271560>

このプロジェクトでは、Scratchキヤットが黄色の長方形に到達するために迷路を進みます。しかし、Scratchキヤットは壁を通り抜けて歩くことができず、どうプログラムを修正しますか？

終わった？

- + スクリプト内のブロックを右クリックしてコードにコメントを追加します。コメントを追加する事は、友達がプログラムの色々な部分を理解するのに役立ちます。
- + あなたのテストとデバッグの実践をパートナーと話し合ってみましょう。あなたの作戦との類似点と相違点を書き出しましょう！
- + 友達を手助けしましょう！

[訳注1] 在庫リスト。持ち物リストを指します。

[訳注2] ヒット回数。ボールにあたった回数を指します。

ユニット5 もっと探究しよう



インデックス

内容



学びたいことを知ろう	92
第2ラウンド	94
高度な概念	96
ハードウェアと拡張機能	100
アクティビティのデザイン	102
マイデバッグ！	106

ユニット5

概要

“ビッグ・アイデア”

以前のバージョンのガイドのリリース後に、我々が教師の方からもらったフィードバックには、彼ら（と彼らがサポートしている学習者達）が、もっと“キャッチアップ”する時間や、居残って、過去のユニットで作成済みのアイデアとプロジェクトを再び訪れて拡張する時間を欲しがっているという共通点がありました。回答として、我々はこのユニット「もっと探究しよう」を追加しました。

高度な概念と実践を推進しているのか、過去の経験を再び訪れているのかに関係なく、これは、学習者がじっくりと考え、振り返る瞬間に取り組むチャンスです。

クリアになりきっていないものは何でしょうか。

学習者は、Scratchの何を更に知りたがっているのでしょうか。

他の人は、どうやって学習者を助けるのでしょうか。

学習者は、どうやって他の人を助けるのでしょうか。

このことは、また教育者としてのあなたにとっても、じっくりと考え、振り返る同様の行為に取り組むチャンスです。

何があなたを驚かせましたか。

何があなたを不安にしましたか。

次回そうではなく何をやりたいですか、それは何故ですか。



学習の目的

生徒は、

- + 現在の学習のゴールとニーズを自己評価するために過去の体験を振り返ります。
- + 過去に着手したプロジェクトを拡張してセルフ・リミックス [訳注1]を制作します。
- + Scratchを物理的な世界とつながっているさまざまなハードウェアの拡張機能が紹介されます。
- + 最新のScratchの機能（ビデオセンサー、クローン）を探究することで、コンピュータ処理の概念や実践を使いこなせるようになります。
- + 他の人のための学習体験をデザインしてみましょう。

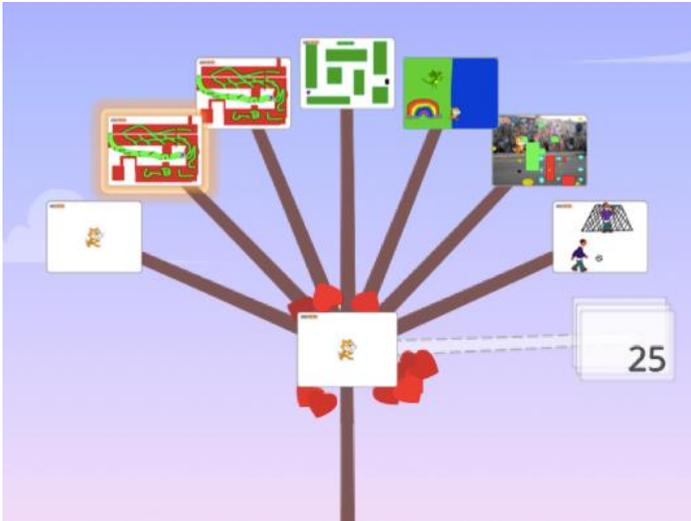
キーワード、概念、実践

- + ビデオセンサー + ピアインタビュ + 拡張機能
- + クローン + ハードウェア

メモ

- + 探し物が見つかりませんか？あなたや学習者にとってベストなものにするため、このガイドのいずれかのアクティビティを自由に、リミックスし、再利用し、イメージし直してください。
- + ScratchEdウェブサイトで、具体的なカリキュラム用にデザインされたLESSONプラン、アクティビティ、リソースを探し出して下さい。 <http://scratched.gse.harvard.edu>

アドベンチャーを選択しよう



このユニットは、直前の3つのユニットのように特定のテーマやジャンルに焦点を当てるのではなく、これまでの作業を見直し、振り返るための空間を作り出すことを目的としています。このユニットのアクティビティは、特に柔軟性があり、今までの課題を復習し、スキルを伸ばし、実践を磨くことで、創造的コンピューティングをさらに深く探究します。

生徒に、過去の作品を確認してもらい、“学びたいことを知ろう”のアクティビティで自分の学習のゴールの自己評価に取り組みってもらうことから始めます。

続けて、どのアクティビティをフォローアップして追い求めるかを選ぶことによって、生徒にScratchをさらに深く探究するよう促します。

可能なルート

セッション1

学びの意欲
を知ろう

何を知っていますか？
何を知りたいですか？
何を学びましたか？

セッション2

第2ラウンド

過去のプロジェクトを
リミックスします。
欠席したアクティビ
ティに戻ります。新し
いことを学ぶよう、あ
なた自身チャレンジ
します。

セッション3

ハードウェア
と拡張機能

Scratchの創作
は、スクリーン上
で起こる事以上
の事ができます。

セッション4

アクティビティ
のデザイン

他の人がトライ
するための学び
の体験をデザ
インします。

セッション5

私の
デバッグ！

独自の"Debug It!"
プログラムを制作し、他
の人がそれを解決でき
るかどうかを確認します。

学びたいことを知ろう

 **推奨時間**
30-45分

目的

- このアクティビティを完了すると、生徒は
- + 過去のプロジェクトと体験を振り返ります。
 - + 現在の知識と学習のゴールを自己評価します。
 - + 自らで方向性を決める研究アクティビティで、個人の学びに対する関心^[訳注2]を追求します。

アクティビティの説明

- 自らで方向性を決めるこの学習アクティビティでは、生徒が現在理解していることを振り返り、自分の関心に基づいた新しい知識を構築します。
必要に応じて、生徒をガイドする「学びたいことを知ろう^[訳注1]」ワークシートを用意します。
- 生徒に、Scratchと創造的コンピューティングについて既に知っていること、次に知りたいことを振り返ってもらいます。生徒に「振り返りを促す質問」の最初の2つの質問に、デザイン日誌か、自己評価用の「学びたいことを知ろう」ワークシートに回答するようガイドします。
次に、生徒に時間を与え、「何を知りたいですか？」の回答からの学びに対する関心^[訳注2]を追求してもらいます。
最後に、生徒に「振り返りを促す質問」の3番目と4番目の質問に、デザイン日誌か、「学びたいことを知ろう」ワークシートに回答してもらいます。
- 生徒が、振り返りや学びに対する関心をお互いで共有するよう手助けします。ピアインタビューがお勧めです。生徒をペアに分け、振り返り、自己評価、研究のプロセスについて、交互にインタビューしてもらいます。

リソース

- 「学びたいことを知ろう」ワークシート
- Scratch Wiki <http://wiki.scratch.mit.edu>
- Scratchディスカッションフォーラム <http://scratch.mit.edu/discuss>
- Scratch FAQ <https://scratch.mit.edu/info/faq>

振り返りを促す質問

- + 何を知っていますか？
- + 何を知りたいですか？
- + 何を学びましたか？
- + あなたが知りたかったものを調査するための戦略は何でしたか？

作業の確認ポイント

- + 生徒は知りたいことを学ぶことができたか？
- + 生徒はどのような作戦とリソースを採用したか？

メモ

- + 知識がある仲間を活用したり、家族や友人に質問を持ちかけたり、Scratchディスカッションフォーラムに質問を投稿するなど、生徒が研究中に他のリソースを見つけて利用するのを手助けしましょう。

[訳注1] 原文は、KNOW WANT LEARN
[訳注2] 原文は、learning interests

追加メモ記入欄

- _____
- _____
- _____
- _____

学びことを知ろう

名前: _____

創造的コンピューティングとScratchに関して、あなたは何を知っていますか、次に何を知りたいですか。
このアクティビティは、

自分自身がScratchのどの領域を快適に進められるのか（何を知っていますか？）、
どの領域をさらに探究したいのか（何を知りたいですか？）
を考えるチャンスです。
周りの色々なリソースを使って知りたいことを調べて、気付いたこと（何を学びましたか？）を共有しましょう。

何を知っていますか？

これまでのデザインの体験を振り返って、Scratchと創造的コンピューティングについて知っている事を書き出してください。

何を知りたいですか？

個人的な関心に基づいて、もっと知りたい事や次に発見したい事をリストアップしてください。

何を学びましたか？

リソースを集めて上記で作成したリストの項目を調査し、続いて調査から学んだことを共有してください。

第2ラウンド



推奨時間
45-60分

目的

このアクティビティを完了すると、生徒は
+ 過去の作品のセルフ・リミックスを制作する機会、または過去にスキップしたり、完了できなかったユニットのアクティビティに時間をかける機会を持ちます。

アクティビティの説明

- 必要に応じて、生徒をガイドするユニット0-5のアクティビティの配布用プリントを用意します。
- 生徒に自習する時間を与えます。
 1. セルフ・リミックス（自分自身のプロジェクトのリミックス）を制作して、過去のプロジェクトをイメージし直したり、拡張したりします。
 2. スキップしたり、完了できなかった過去のユニットのアクティビティを再び訪れて取り組みます。
- 生徒にセルフ・リミックスやアクティビティの成果をお互いで共有するよう促します。ペアシェアやデザインデモを使うのがお勧めです。
- 生徒に、デザイン日誌やグループ討議の中で振り返りを促す質問に回答することによって、デザインのプロセスを振り返ってもらいます。

リソース

- ユニット0-5の配布用プリント

振り返りを促す質問

- + なぜそのプロジェクトやアクティビティを選んで取り組んだのですか？
- + もっと時間があつたら、何をしますか？

作業のチェックポイント

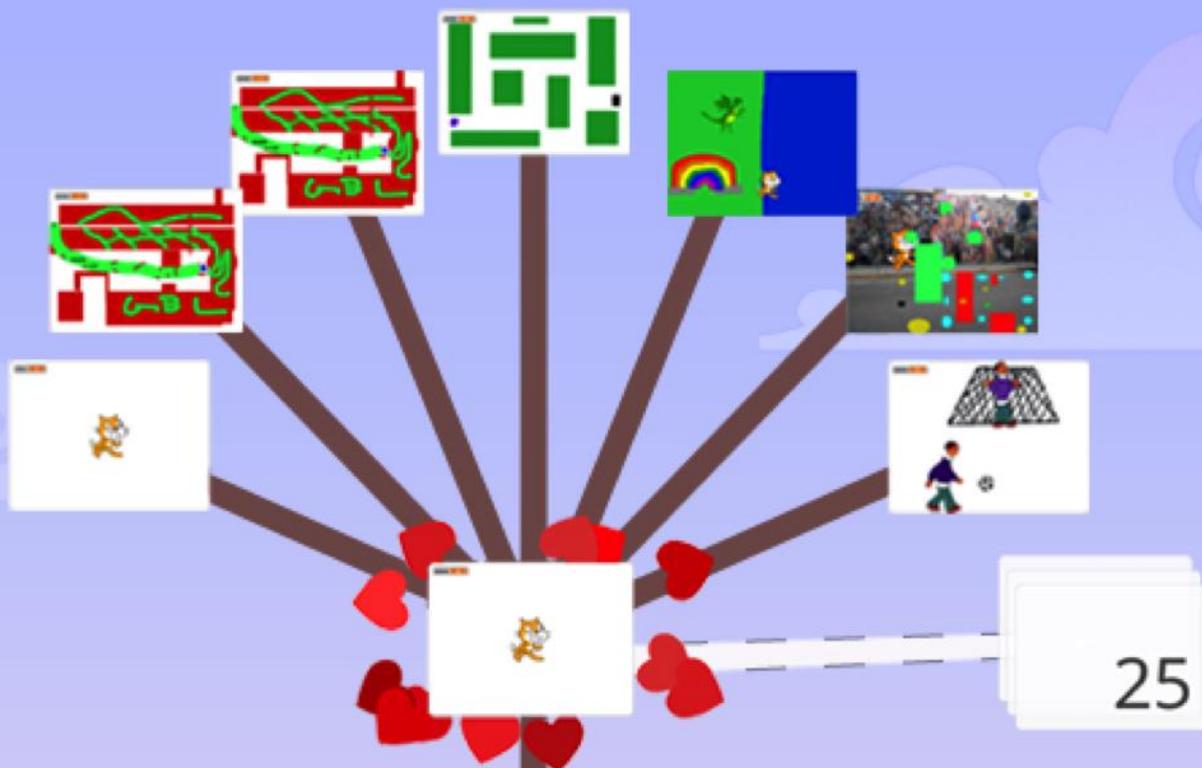
- + 生徒はセルフ・リミックスを作成したり、アクティビティに取り組んだか？
- + 生徒の関心について、あなたは何を学んだか？
- + 生徒はさらにどんなサポートが必要としているか？

メモ

- + 過去の作品やアクティビティを振り返るために、生徒に、デザイン日誌とScratchのプロフィールを確認するよう求めます。
- + 生徒に、アイデアを求めて、ユニット1の“マイスタジオ”の感動的なプロジェクトのアイデアを確認するよう勧めます。

追加メモ記入欄

- _____
- _____
- _____
- _____



25

高度な概念

 **推奨時間**
30-45分

目的

このアクティビティを完了すると、生徒は
+ ビデオセンサーやクローンを探究するプロジェクトを制作することで、コンピュータ処理の概念（イベント処理、並列処理、データ）と実践（実験と反復、テストとデバッグ、再利用とリミックス、抽象化とモジュール化）を使いこなせるようになります。

アクティビティの説明

- ❑ Advanced Conceptsスタジオ、Video SensingサンプルスタジオおよびCloning examplesスタジオを使って、サンプルを示します。そして、生徒がビデオセンサーとクローンを制御するブロックを理解できるよう助けます。必要に応じて、生徒をガイドする高度な概念の配布用プリント[訳注1]を用意します。
- ❑ 生徒に時間を与え、サンプルプログラムのコードを調べて、1つ以上の高度な概念（ビデオセンサー、クローン）を体験するプロジェクトを制作してもらいます。
- ❑ 生徒に自分の探究を他の人と共有するよう促します。デザインデモのプレゼンテーションを開催するのがお勧めです。必要に応じて、生徒にAdvanced Conceptsスタジオがクラス・スタジオにプロジェクトを追加してもらいます。
- ❑ 生徒に、デザイン日誌やグループ討議の中で振り返りを促す質問に回答させてデザインのプロセスを振り返ってもらいます。

リソース

- ❑ Advanced Conceptsスタジオ：
<http://scratch.mit.edu/studios/221311>
- ❑ 「ビデオセンサー」配布用プリント
- ❑ Video Sensingサンプルスタジオ：
<http://scratch.mit.edu/studios/201435>
- ❑ 「クローン」配布用プリント
- ❑ Clone Examplesスタジオ：
<http://scratch.mit.edu/studios/201437>

振り返りを促す質問

- + あなたが選択して探究したのは、どの高度な概念ですか？
- + あなたが選んだコンセプトについて、もっと学ぶための戦略は何でしたか？

作業の確認ポイント

- + プロジェクトは1つ以上の高度な概念を探究しているか？

メモ

- + ビデオセンサー機能を探究したい生徒には、ウェブカメラ付きのコンピュータが必要です。
- + バックパックツールが、サンプルプロジェクトからコードを借りてリミックスするのに使えることを生徒に思い出させて下さい。

追加メモ記入欄

- ❑ _____
- ❑ _____
- ❑ _____
- ❑ _____

[訳注1] リソース欄の「ビデオセンサー」と「クローン」を指す。



ビデオセンサー

どうしたらScratchプロジェクトでビデオセンサーを使えますか？

ウェブカメラでScratchプロジェクトをインタラクティブにすることができるのを知ってましたか？ビデオセンサー機能を組み込んだプロジェクトを制作して、高度なScratchの概念を探究しましょう。

ここからスタート

- ❑ 既存のScratchプロジェクトを開くか、新しいプロジェクトを開始してビデオセンサー機能を追加します。
- ❑ “拡張機能を追加”をクリックします。
- ❑ “ビデオモーションセンサー”をクリックします。
- ❑ “ビデオモーションセンサー”のカテゴリにあるビデオセンサー用ブロックをチェックします。
- ❑ ビデオの動きを検出するようにプロジェクトをプログラムするため、“ビデオを～（入・切・左右回転）にする”ブロック、“ビデオの透明度を～にする”ブロックを試してみます。



試してみよう

- ❑ ウェブカメラが接続されていることを確かめます！ “ビデオを[入]にする”のブロックを使って、ウェブカメラをテストしましょう。
- ❑ 行き詰まっても大丈夫です！ Video Sensingスタジオの他のいくつかのプロジェクトを検索して、ビデオ機能をもったブロックの使用方法を調べたり、Tipsウィンドウを使ってビデオセンサーの詳細を確認してください。

終わった？

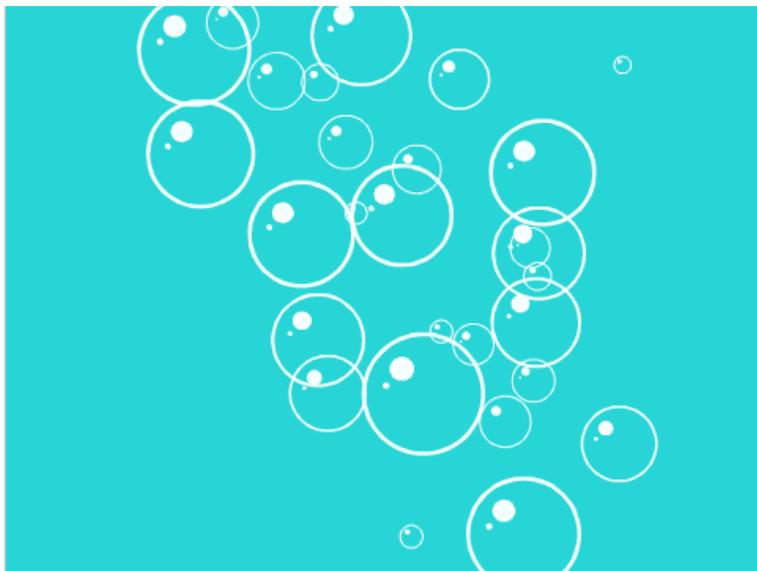
- + Advanced Conceptsスタジオにプロジェクトを追加しましょう。
<http://scratch.mit.edu/studios/221311>
- + 過去のプロジェクトの1つにビデオセンサーを追加してみましょう！
- + 友達を手助けしましょう！
- + Video Sensingスタジオのプロジェクトをリミックスしましょう。

クローン

どうしたらScratchプロジェクトでクローンを使えますか？

クローンは、複数の同じスプライトを作成する簡単な方法です。クローンを使うと、多くのオブジェクトを作成したり、プロジェクトでクールな効果を演出できます。

クローン機能を組み込んだプロジェクトを制作して、高度なScratchの概念を探究しましょう。



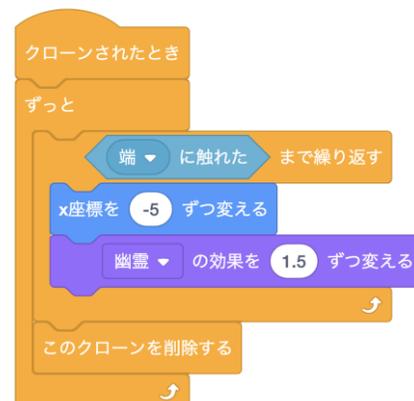
ここからスタート

- 既存のScratchプロジェクトを開くか、新しいプロジェクトを開始してクローン機能を試してください。
- “制御”のカテゴリにあるクローン用ブロックをチェックします。
- スプライトのクローンを作成するブロックを実験してみましょう。また、クローンされたスプライトの動作を定義します。

クローンされたとき

自分自身 のクローンを作る

このクローンを削除する



試してみよう

- 最初にクローンが見えない場合は、オリジナルのスプライトが同じ場所にいるかどうかチェックしてください。オリジナルのスプライトがクローンに重なっているかもしれません！オリジナルのスプライトまたはクローンをプログラムで移動させるか、手動で別の場所へ移動して、見えるようにします。
- 行き詰まっても大丈夫です！Clone Examplesスタジオの他のいくつかのプロジェクトを検索して、クローン機能の使用方法を調べたり、Tipsウィンドウを使って“～のクローンを作る”ブロックと“クローンされたとき”ブロックに詳細を確認してください。

終わった？

- + Clone Examplesスタジオにプロジェクトを追加しましょう。
<http://scratch.mit.edu/studios/201437>
過去のプロジェクトの1つにクローン機能を追加してみましょう！
- + 友達を手助けしましょう！
- + Clone Examplesスタジオのプロジェクトをリミックスしましょう。

ハードウェアと拡張機能

 推奨時間
45-60分

目的

このアクティビティを完了すると、生徒は
+ Scratchのデジタルな世界と物理的な世界をつなぐ、さまざまなハードウェア拡張機能を紹介されます。

アクティビティの説明

- ❑ **(重要：このアクティビティでは、以下のハードウェア製品のへのアクセスが必要です。)**
生徒に、Scratchがテクノロジー製品やハードウェア拡張機能に接続する方法を紹介します。
(LEGO WeDo、MaKey MaKeyおよびPicoBoard)
必要に応じて、「Scratchと他のテクノロジー製品との接続方法のビデオ」の再生リストを示します。
- ❑ 生徒を2~4人の小グループに分けます。グループに、時間を与え、利用可能なハードウェア拡張機能を使って物理的な世界のコンポーネントを組み込んだScratchプロジェクトを作制してもらいます。
- ❑ グループごとに自分の作品を他の人と共有させます。ギャラリーウォークやフィードバック・フェアをファシリテートするのがお勧めです。
- ❑ 生徒に、デザイン日誌やグループ討議の中で振り返りを促す質問に回答させてデザインのプロセスを振り返ってもらいます。

リソース

- ❑ LEGO WeDo construction set
<http://bit.ly/LEGOWeDo>
https://ja.scratch-wiki.info/wiki/LEGO%C2%AE_WeDo
- ❑ MaKey MaKey
<http://makeymakey.com>
- ❑ PicoBoard
<https://www.sparkfun.com/products/10311>
- ❑ Scratchと他のテクノロジーとの接続方法のビデオ
<http://bit.ly/hardwareandextensions>

振り返りを促す質問

- + どのハードウェアや拡張機能を探究しましたか？
- + どうやってデジタルと物理を合体しましたか？
- + 何が難しかったですか？
- + 何が驚きでしたか？

作業の確認ポイント

- + 作品にはデジタルと物理的なコンポーネントがあるか？

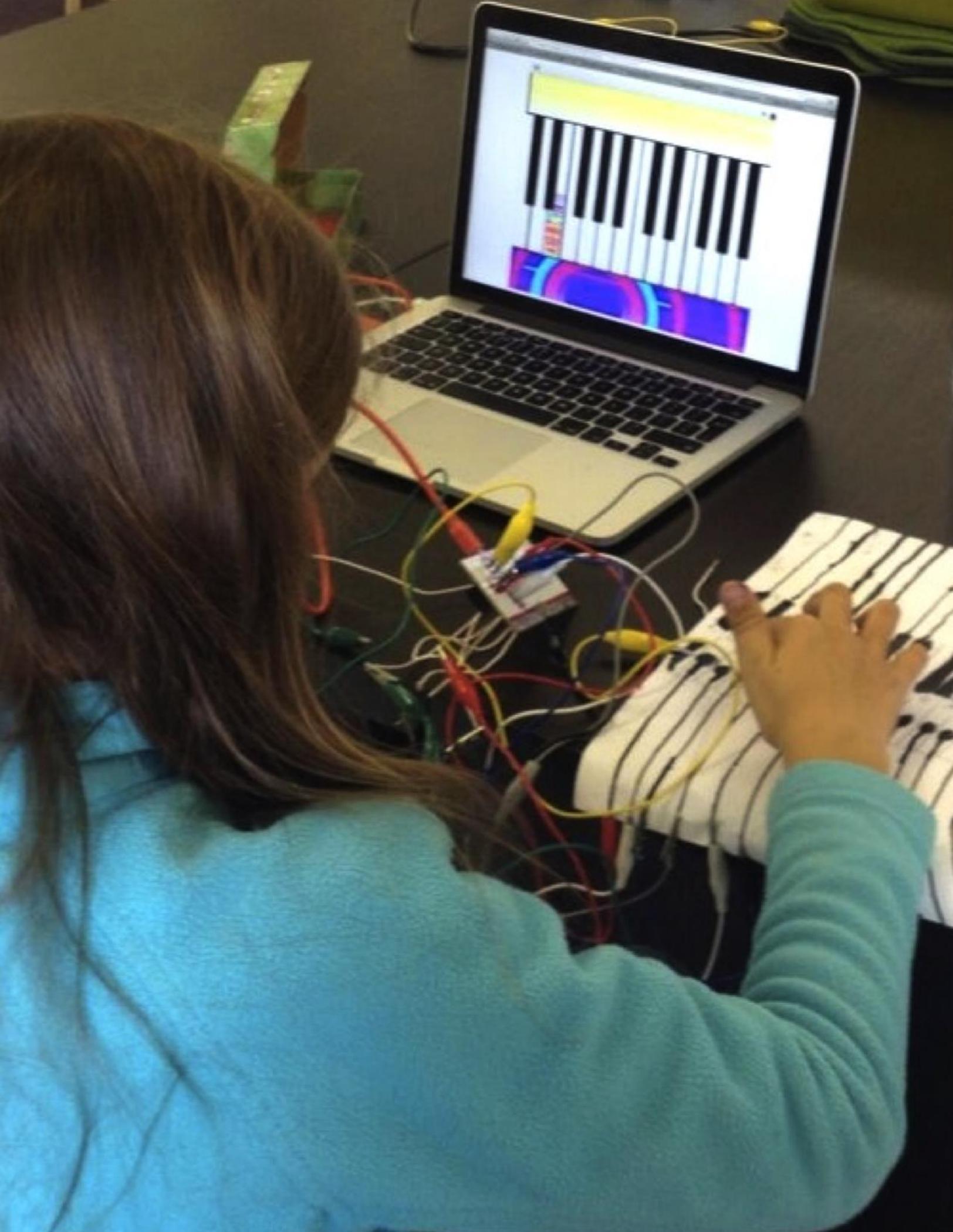
メモ

- + これをグループ全体のアクティビティにしましょう！
LEGO WeDoとScratchを使って、生徒に自分のプロジェクトに接続して、ルーベ・ゴールドバーグ・マシン[訳注1]スタイルでリアクションの連鎖を作るよう、チャレンジさせます。
サンプルのためのビデオを確認して下さい。
<http://bit.ly/ScratchChainReaction>
- + Scratchプロジェクト・エディタの“その他”カテゴリーにある“拡張機能を追加”ボタンをクリックし、ハードウェア拡張機能を制御するScratchブロックを有効にします。

追加メモ記入欄

- ❑ _____
- ❑ _____
- ❑ _____
- ❑ _____

[訳注1]からくりを多数、連鎖させる表現手法。ドミノ倒しやピタゴラススイッチに類似したアクティビティ。



アクティビティのデザイン



推奨時間
30-45分

目的

このアクティビティを完了すると、生徒は
+ 他の人がScratchとコンピュータの創造性をもっと学ぶ際のサポート用のアクティビティやリソースをデザインします。

アクティビティの説明

- 生徒に「靴の履き方を教える」[訳注1]のような体験をさせて下さい！
生徒に、創造的コンピューティングの探究をサポートするアクティビティやリソースのデザイン（制作、リミックス、再イメージ）にチャレンジしてもらいます。必要に応じて、追加サポート用の「アクティビティ・デザイン」配布用プリントを用意します。
- 生徒がブレインストーミングして、色々な種類の創造的な学習体験をイメージするのを助けます。
必要に応じて、インスピレーションを求めて、このガイドのプロジェクトのアイデアやアクティビティのサンプルを確認するか、あるいは生徒にScratchカードのリソースとScratch Design Studioのリストを調べるよう促します。
続けて生徒に時間を与え、彼ら自身の学習のアクティビティやリソースをデザインしてもらいます。
- 生徒に、学習者と一緒にアクティビティやリソースをテストする機会を与えます。生徒に、自分のアクティビティやリソースを家族や友人と共有するよう勧めたり、他のクラス、クラブやイベントのピアメンター[訳注2]になるよう求めます。
- 生徒に、デザイン日誌やグループ討議の中で振り返りを促す質問に回答させてデザインのプロセスを振り返ってもらいます。

リソース

- 「アクティビティ・デザイン」配布用プリント
- Scratchカード：
<http://scratch.mit.edu/info/cards/>
- Scratch Design Studio：
<http://scratch.mit.edu/users/ScratchDesignStudio/>

振り返りを促す質問

- + 誰が、あなたのアクティビティやリソースを使うことを想定していますか？
- + あなたのアクティビティやリソースで学習者に何を学んで欲しいですか？
- + あなたのアクティビティを催したり、リソースを使う際に、学習者はどんなチャレンジを体験できますか？
そのチャレンジに対処する際、どうやって学習者をサポートしますか？

作業の確認ポイント

- + アクティビティやリソースは、創造的コンピューティングの紹介や探究を促進しているか？
あなたはどんなフィードバックを生徒に提供できるか？

メモ

- + とりわけ他の人の学習をサポートすることに興味がある生徒は、授業中や、放課後、ランチタイムにScratchクラブでピアメンターになれる素敵な候補者です。

[訳注1] 原文は、it's like be in your teaching shoes!

[訳注2] 友達のような相談相手を指す。

追加メモ記入欄

- _____
- _____
- _____
- _____

アクティビティ デザイン

名前： _____

Scratchと創造的コンピューティングをもっと詳しく他の人に教えるにはどうすればいいですか？
他の人がScratchを学ぶのを助けるアクティビティをデザインしましょう。それは、
コンピュータを使用しないアクティビティ（“創作物の制作”のような）、プロジェクトのアイデア
（“バンドを作ろう”のような）、またはチャレンジ・アクティビティ（“デバックしよう！”のような）
から成ります。

新しいタイプのアクティビティや配布用プリントを開発するのでも構いません！

以下の質問を使ってブレーストーミングして、アクティビティと配布用プリントのプランナー[訳注1]
を使用して詳細化してください。

[訳注1] アクティビティ作成用のフォーム（次の2頁）

誰を対象にしていますか？

アクティビティの対象者は誰ですか？誰にScratchと創造的コンピューティングをもっと詳しく教えたいですか？

学習者は何を学ぶのでしょうか？

学習のゴールは何ですか？あなたのアクティビティで学習者にどんな新しい事を学んで欲しいですか？

学習者は何を必要としていますか？

学習者は、どんな物資を必要としていますか？

あなたのアクティビティを学習者が成功するには、どんなタイプのサポートが役立ちますか？

(タイトル)



推奨時間

__-__分

目的

(学習のゴール)

このアクティビティを完了すると、生徒は

+

+

アクティビティの説明

(プロジェクトのインストラクション)

□ 学習者は何を作成しますか？どうやって作成しますか？

□ 学習者はどうやって自分の作品を他の人と共有しますか？

□ 学習者はどうやって自分のデザインを振り返りますか？

リソース

(プロジェクトのリソース (スタジオ、配布用プリントなど) を2つ記述)

□

□

振り返りを促す質問

(振り返りを促す質問を3つ記述)

+

+

+

作業の確認ポイント

(アクティビティを完了したかどうかの確認ポイントを2つ記述)

+

+

メモ

(ヒントと秘訣を記述)

+

+

+

追加メモ記入欄

□

□

□

□

(タイトル)

(プロジェクトの概要)

(プロジェクトの説明)

(プロジェクトの図や写真)

ここからスタート

(プロジェクトの手順の説明)

-
-
-
-

(イラストや写真入りのプロジェクトの説明)

試してみよう

(行き詰まった場合に行うことを3つ記述)

-
-
-

完成した？

時間が余った生徒に、やってもらうことを3つ記述)

- +
- +
- +

マイデバッグ！

▶ 推奨時間
15-30分

目的

このアクティビティを完了すると、生徒は
+ デバッグの問題をデザインすることで、コンピュータ処理の実践（実験と反復、テストとデバッグ、再利用とリミックス、抽象化とモジュール化）を使いこなせるようになります。

アクティビティの説明

- 必要に応じて、アクティビティの間に生徒をガイドする「マイデバッグ！」配布用プリントを用意します。
- 生徒に、他の人が解決すべき独自のデバッグの問題を作成する機会を与えます。バグは、特定のコンピュータ処理の概念、Scratchブロック、対話処理、その他プログラミングのチャレンジに集中させることができます。Scratchプロジェクトの開発中に、自分で行き詰まったり、乗り切ったりした経験からアイデアを見つけるよう生徒を促します。
- 生徒に、交代でお互いのバグのあるプロジェクトのテストとデバッグにトライさせます。必要に応じて、生徒にMy Debug It! スタジオまたはクラス・スタジオに自分のDebug-Itプログラムを追加してもらいます。
- 生徒に、デザイン日誌やグループ討議の中で振り返りを促す質問に回答させて問題制作のアプローチを振り返ってもらいます。

リソース

- 「マイデバッグ！」配布用プリント
- My Debug It!スタジオ：
<http://scratch.mit.edu/studios/475637>

振り返りを促す質問

- + 何が問題でしたか？
- + アイディアはどこから来ましたか？
- + どうやって他の人が問題を調査し解決するのか、想像できましたか？
- + 他の人は、あなたが期待していた事以上に、問題を発見して解決するための他のアプローチを取りましたか？彼らの戦略は何でしたか？

作業の確認ポイント

- + プロジェクトには、解決すべきデバッグの問題が含まれているか？
- + 生徒は、どんなテストとデバッグの戦略を採用したか？

メモ

- + Scratchウェブサイトのプロジェクトのページのメモ欄に、課題の説明を含めることを生徒に思い出させます。
- + 追加の時間を持つか、ウィームアップ・アクティビティが必要ありませんか？
My Debug It!スタジオに投稿された他の“Debug-it”のプログラムで、生徒の問題発見スキルと問題解決スキルに磨きをかけさせましょう。

追加メモ記入欄

- _____
- _____
- _____
- _____

マイデバッグ！

うーん...

さあ、独自のDebug Itプログラムをデザインする時間です。
何を作成しますか？

このアクティビティでは、他の人が調査して解決し、リミックスするあなた独自のDebug It!の問題を作成します。



ここからスタート

- プロジェクトの作成やデバッグの際、あなたが遭遇したさまざまな種類のバグを振り返ってみましょう。
- 作成できそうなデバッグの問題をリストアップします。“マイデバック！”は、特定の概念、ブロック、対話処理、その他プログラミングチャレンジに集中させることができます。
- あなた独自のDebug It!プログラムを制作しましょう。

追加メモ
記入欄

マイデバックのためのプラン

終わった？

-
-
-
-
-

- + My Debug It!スタジオにマイデバックを追加しましょう。：
<http://scratch.mit.edu/studios/475637>
- + Debug It!プログラムを友達とを交換し、お互いでバグのあるプログラムを解決してみましょう。
- + 友達を手助けしましょう！
- + My Debug It!スタジオで他のプログラムもデバッグしてみましょう。

ユニット6 ハッカソン

インデックス

内容

0

1

2

3

4

5

6

プロジェクト・ピッチ	114
プロジェクト・プランニング	116
デザインスプリント	120
プロジェクト・フィードバック	122
プロジェクト・チェックイン	124
アンフォーカスグループ	126
ショーケースの準備	128
ショーケース	130

ユニット6

概要

“ビッグ・アイデア”

最後のユニットでは、生徒は、彼らが選ぶオープンエンドのプロジェクトのデザインに取り組む事によって、創造的コンピューティングの経験を積みます。

あなたとあなたの生徒が、このオープンエンドなデザインの体験に取り組むのに役立てるために、我々はこのユニットをハッカソンとして形作る着想を得ました。

ハッカソンは、ジャスト・イン・タイムの学習と問題解決に取り組み、反復的な計画・作成・共有（プラン・メイク・シェア）を促し、つながりのあるコラボレイティブな環境を歓迎するといった特性を持つ、理想的な創造的コンピューティングの最高の経験です。

授業は終わったけど、
何人かの生徒は気
付いていないみたい。
彼らの #Scratch
ゲームをデバッグする
のに夢中なんです。
チームの効果ですね。
@Sheena1010



学習の目的

- 受講生は、
- + ハッカソンのイベント形式が紹介されます。
 - + 個人的に意味があり、自らで方向性を決めたプロジェクトを定義し、開発し、発表することにより、コンピュータの概念（順次処理、繰り返し処理、イベント処理、並列処理、条件分岐、演算子、データ）と実践（実験と反復、テストとデバッグ、再利用とリミックス、抽象化とモジュール化）の知識を実際に実証してみせます。
 - + ピアチーム[訳注1]で作業を行い、スキルを共有し、複数回のフィードバックを与えたり受けたりして、複数のコラボレーションの機会を持ちます。

キーワード、概念、実践

- + ハッカソン
- + アンフォーカスグループ
- + デザインスプリント
- + ショーケース
- + プロジェクト・ピッチ

メモ

- + このユニットは、独立したグループか共同のグループの両方のプロジェクトに対応可能です。片方を選択するか、あるいは、生徒に選択させます。

[訳注1] 仲間やクラスの友達からなるチーム

ハッカソンとは？

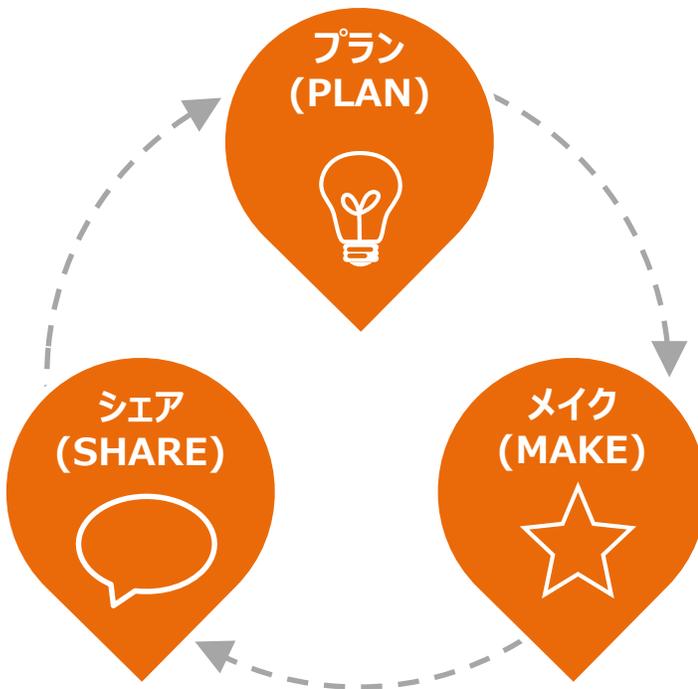
「ハック」という言葉は、いくらかの点でネガティブな印象がありますが、遊び心、好奇心、持続性、創造性の代名詞としての長い歴史があります。我々が気に入っている「ハック」の定義のひとつは、「独創性を適切に適用すること」です。この定義によれば、若い学習者にとって、ハックする方法を学ぶこと以上の知的能力があるでしょうか？

ハッカソンは、ハッキングの遊び心のある独創性をもたらします。そして真剣に集中した時間制限のある環境の中にその独創性を置きます。このユニットでは、反復的なプラン・メイク・シェアのサイクルを使って、学習者がアイデアをブレインストーミングし、プロジェクトを作成し、そして最終版のプロトタイプを紹介します。

ハッカソンは、学習者自身が個人的に意味のある重要なプロジェクトを考案して取り組む素晴らしい機会を学習者に提供します。それは独立した最終的なプロジェクトとして開発されるか、コラボレイティブなチームの中で開発されます。コラボレイティブで、創造的で、柔軟で、遊び心のある学習環境の中で、生徒にとって、彼らのScratchの知識を実証し、現在のスキルを高め、そしてアイデアを発展させて試すチャンスなのです。

作業の進め方は？

ハッカソンの期間の全体を通じて、生徒は、プラン・メイク・シェアの反復サイクルに取り組みます。この反復サイクルは、生徒に意味のある思考過程、創作、および振り返りの行為に取り組むことを奨励します。



**プラン
(PLAN)**

何をしたいですか？
アイデアをブレインストーミングし、アクションプランを準備しましょう！



**メイク
(MAKE)**

リソースを使ったり、友達から助けをもらって、プロジェクトの創作品をデザインし制作します。



**シェア
(SHARE)**

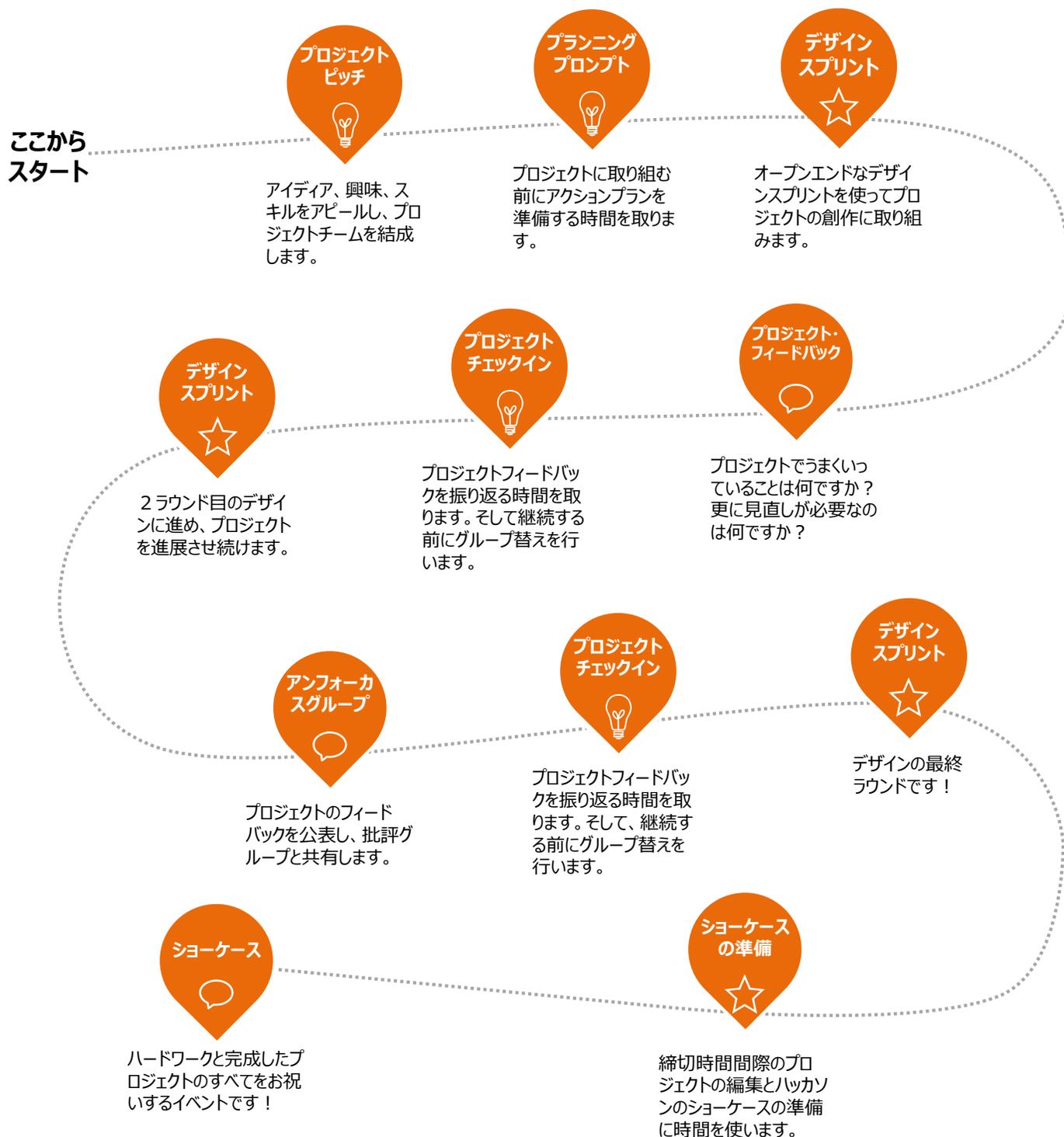
あなたの次のステップをガイドするために、プロジェクトを友達と共有し、フィードバックを集めます。

可能なルート

ハッカソンに影響を受けたこのユニットのアクティビティは、生徒に、オープンエンドのコラボレティブな学習環境の中で、より複雑なプロジェクトを制作することにチャレンジさせるようデザインされています。我々が行ってきた重要な構築のカルチャー[訳注1]のすべて（リスクテイクと持続性の奨励すること、失敗を学びの機会と容認すること、プロダクトよりもプロセスを重視すること、そして協調性と楽しさのカルチャーを育てること）が、このユニットでクライマックスに達します。

[訳注1]原書はCulture-Building

初めての皆さんを助けるために、プラン・メイク・シェアのデザインサイクルに沿った提案のアクティビティの流れを示します。





プロジェクト・ピッチ

 **推奨時間**
30-45分

目的

- このアクティビティを完了すると、生徒は、
- + 個人的な興味に基づいて、プロジェクトのアイデアをブレインストーミングします。
 - + プロジェクト・チームを結成するために、アイデア、興味、スキルをピッチ[訳注1]します。

アクティビティの説明

- 生徒にピッチの概念を紹介します。ピッチを使うと、生徒は、チームメンバーを募集するためにプロジェクトのアイデアを発表したり、チームに採用されるために自分の興味、スキル、才能を売り込むことができます。
- 「ピッチ」配布用プリントを配ります。生徒に時間を与え、ブレインストーミングを行って、配布用プリントに回答してもらいます。何人かの生徒は、既にプロジェクトのアイデアを持っているか、共有してもっと探究したいと思っている具体的な関心やスキルを持っているかもしれません。具体的なプロジェクトのアイデアや興味が無い場合は、別のチームに加わるチャンスがあることを生徒に知らせます。必要に応じて、生徒に、ユニット1の“マイスタジオ”のアクティビティの時に特定した感動的なプロジェクトを確認してもらいます。
- 各生徒にグループの残りの人にピッチするチャンスを与えます。生徒がピッチを希望している場合は、整列してもらい、各々に30秒の時間を与え、自分のプロジェクト、興味、スキルを説明してもらいます。
- 生徒に時間を与え、3～4人のグループを結成してもらいます。必要に応じて、チームビルディングを容易にするために、生徒に自分の名前とプロジェクトの興味を、壁に整理して並べられる付箋[訳注2]に書いてもらいます。

リソース

- 「ピッチ」配布用プリント
- 付箋（必要に応じて）

振り返りを促す質問

- + これまでに取り組んできたお気に入りのプロジェクトは何ですか？
- + 今回、あなたが創作に興味があるのは、どんなプロジェクトですか？
- + プロジェクトに貢献できるのはどんな知識、スキル、才能ですか？

作業の確認ポイント

- + 各々の生徒には、アイデアと興味をピッチするチャンスがあったか？
- + 各々の生徒は、参加するプロジェクトチームを見つけたか？

[訳注1] 意気込みをアピールすること

[訳注2] ポストイットのように文字の書ける大きめの付箋

[訳注3] 子供たちのアクティビティをリードする教師のことを指す

メモ

- + すべてのScratchセッションを通して、特にハッカソン・セッション中に、お互いにサポートとガイダンスを提供しあう中で、生徒は途方もなく貴重なことができます。若者に知識やスキルを他の人と共有するよう促すことは、ファシリテーター[訳注3]にとって容易なことですが、その上クリエイターの学びと理解を著しく深めてくれます。

追加メモ記入欄

- _____
- _____
- _____
- _____

プロジェクト ピッチ

プロジェクト・ピッチ作成者 _____

ハッカソンで取り組むことに興味があるプロジェクトのためのアイデアをブレインストーミングするために、以下の項目を記入してください。持ち時間30秒で、あなたのアイデア、興味、スキルをグループの中で、アピールしましょう。

お気に入りのプロジェクト

これまでに取り組んできたお気に入りのプロジェクトは何ですか？気に入っている理由は何ですか？

ハッカソン・プロジェクトの私のアイデア

今度の創作に興味があるのは、どんなプロジェクトですか？

私のスキルと興味

+ どんな知識、スキル、才能をプロジェクトに貢献できますか？

プロジェクト・プランニング



推奨時間
30-45分

目的

- このアクティビティを完了すると、生徒は、
- + 適切にスコープ決めしたプロジェクト[訳注1]を特定します。
 - + プロジェクトを完了するために必要なアクティビティやタスクのアウトラインを作成します。
 - + プロジェクト完了に必要なリソースの準備リストを作成します。

アクティビティの説明

- 仕上げプロジェクトの開始時に、しばらく時間をとって、アイデアを模索したり、プロジェクトを完了させるための必要なタスクを特定して既知（及び未知）のタスクをリストアップすることは、プロジェクトの成功にきわめて有益です。
- グループをプロジェクトチームに分けます。必要に応じて、「プロジェクト・プランニング」、「プロジェクトスケッチ」配布用プリントを各チームか個人に配布します。
- プロジェクト・プランニングのさまざまな要素（プロジェクトスケッチ、タスクのアウトライン、リソース一覧、ストーリーボード[訳注2]/ワイヤーフレーム[訳注3]）をレビューします。チームに15分間与え、プロジェクトのアイデア、プラン、そしてリソースをブレインストーミング[訳注4]してもらいます。生徒が既に明確なコンセプトとプランを持っている場合、すぐにプロジェクト・デザインに取り掛かることを歓迎します。
- 必要に応じて、このアクティビティの最後に、記入済み「プロジェクト・プランニング」、「プロジェクトスケッチ」配布用プリントを集めておきます。次のデザインプリント・セッションの始めに生徒に返却するためです。

リソース

- 「プロジェクト・プランニング」配布用プリント
- 「プロジェクトスケッチ」配布用プリント

振り返りを促す質問

- + どんなプロジェクトを制作したいですか？
- + どんなステップを踏んでプロジェクトを制作しますか？
- + 私のプロジェクトを制作するために、既にどんなリソース（人やサンプルプロジェクト等）を持っていますか？
- + 私のプロジェクトを制作するためにほとんどなリソース（人やサンプルプロジェクト等）が必要になりますか？

作業の確認ポイント

- + このハッカソンで利用可能な時間とリソースのために、プロジェクトは適切にスコープ決めされているか？
- + リソースを必要としている生徒が、どうやってリソース[訳注5]にアクセスできるようにすることが可能か？

メモ

- + プランニングは役立ちますが、それはすべての時間を消費したり物事を行う唯一の方法であってはなりません。さまざまな生徒は、異なる大きさで計画し、試行錯誤することを望んでおり、必要としています。さらにプロジェクトのさまざまなフェーズでは異なるアプローチが求められます。多様なデザインスタイルと開発スタイルを奨励し、受け入れるべきです。

追加メモ記入欄

- _____
- _____
- _____
- _____

[訳注1] プロジェクトの対象範囲。実現したい機能、対象ユーザー、利用できるリソース、作業期間・繰り返し回数などを事前に決めておく。

[訳注2] ユーザストーリー（「誰が何のために何をしたいのか」をカードに書いたもの）を見やすく張り出したカンパンのようなボード。

[訳注3] スクリーンのイメージと遷移を具体化するためのスケッチ

[訳注4] ここでは、プロジェクトスケッチとプロジェクト・プランニングの作成も含まれる。

[訳注5] 例えば、リソースの連絡先やURLの一覧を配布する



Music

Play
Air
Sound

Garden

Magic

Two player
games

Sports

Art

Reduce the amount of time spent on the game

Reduce the amount of time spent on the game

Two player magic battle

Reduce the amount of time spent on the game

Reduce the amount of time spent on the game

Reduce the amount of time spent on the game

プロジェクト プランニング

プロジェクト・プランニング作成者： _____

プロジェクトを作成するのに必要な要素を検討し始めるために、以下の質問に教えてください。

私のプロジェクト

制作したいプロジェクトを説明してください。

プロジェクトを制作するために必要なステップをリストアップしてください。

私のリソース

既にどんなリソース（人やサンプルプロジェクト等）を持っていますか？

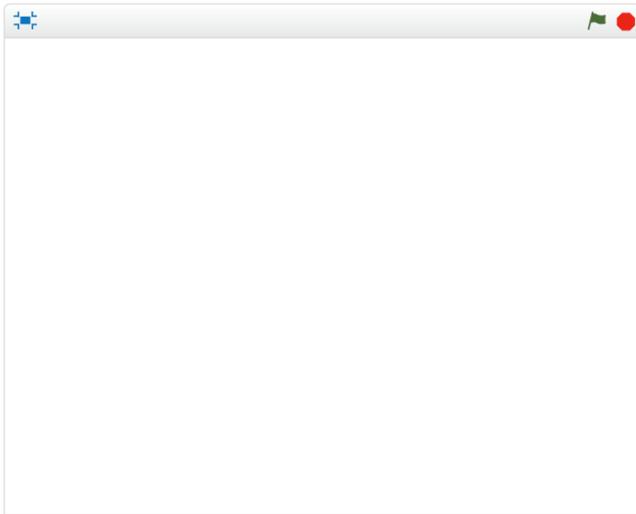
プロジェクトを制作するためにはどんなリソース（人やサンプルプロジェクト等）が必要になる可能性がありますか？

プロジェクト スケッチ

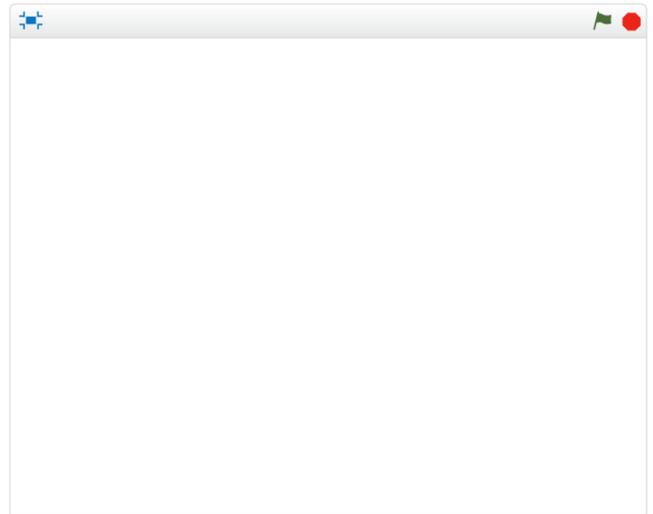
プロジェクト・スケッチ作成者： _____

以下のスペースを使って、プロジェクトがどのようなものか絵コンテを描きましょう。

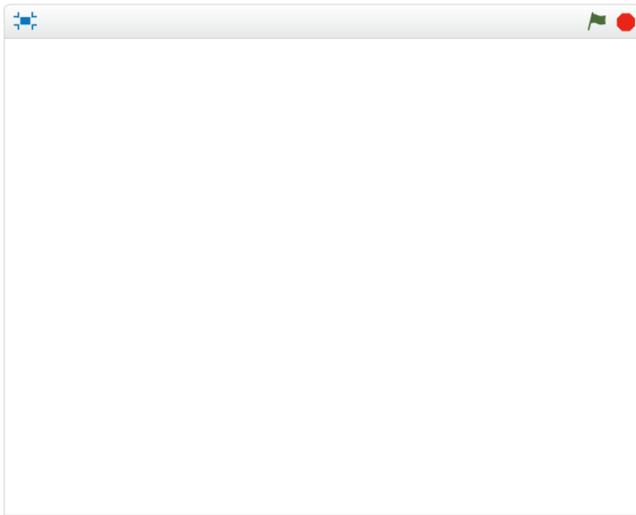
私のプロジェクト・スケッチ



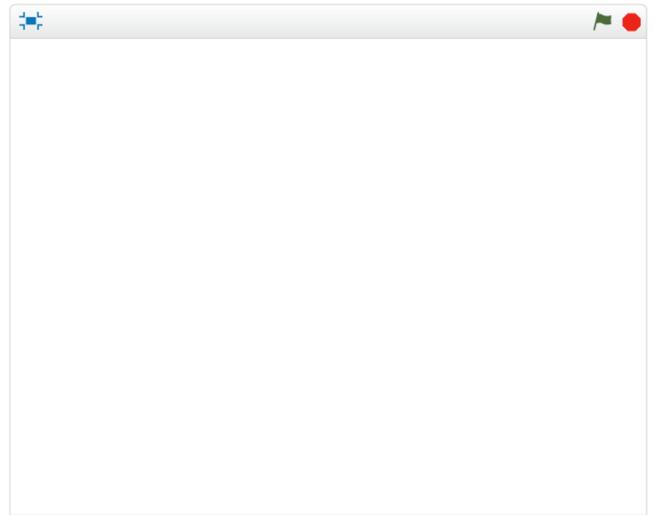
何が起こっていますか？重要な要素は何ですか？



何が起こっていますか？重要な要素は何ですか？



何が起こっていますか？重要な要素は何ですか？



何が起こっていますか？重要な要素は何ですか？

※要素例：登場人物、ストーリー、背景、心情等

デザイン スプリント



推奨時間
45-60分

目的

このアクティビティを完了すると、生徒は、
+ コンピュータ処理の概念と実践を使って、彼らの選択のScratchプロジェクトをさらに発展させます。

アクティビティの説明

- 生徒に、デザインスプリントの概念を紹介します。デザインスプリントは、情熱的にプロジェクトの開発に取り組むため専用の指定された時間です。
- 生徒に、“プロジェクト・チェックイン”のアクティビティを使うか、デザインチームやデザイン日誌の振り返りを促す質問に回答して、このセッション[訳注1]のゴールを書き出してもらいます。記入済みの「プロジェクト・プランニング」、「プロジェクト・フィードバック」、および「アンフォーカスグループ」配布用プリントを生徒に渡して、オリジナルのプロジェクトのゴールを振り返るようガイドし、そしてフィードバックを元にした改善プラン[訳注1]を作成するよう促します。
- 生徒にプロジェクトに取り組むための自習時間を与えます。必要に応じてサポート用の追加リソースを紹介したり、配布します。仲間のサポートに加え、すぐに利用できるサポートリソースを集めておくことは、生徒を進展させ続けるのを助けます。Scratchウェブサイト (<http://scratch.mit.edu>) のサンプルプロジェクトは、アイデアを提供してくれます。そして追加のリソースは、ScratchEdウェブサイトで見つけられます。 (<http://scratched.gse.harvard.edu>)
- 必要に応じて、生徒にプロジェクトのドラフトをクラス・スタジオに追加してもらいます。

リソース

- 追加リソース（例えば、サンプルプロジェクト、配布用プリント、Scratchカード、クラフト材料）

振り返りを促す質問

- + 今日、プロジェクトのどの部分に取り組むつもりですか？
- + 進展させるためには、どんな助けが必要ですか？

作業の確認ポイント

- + 個人やグループは、適切に進行しているか？
- + プロジェクトに対するどんなフィードバックや提案があるか？

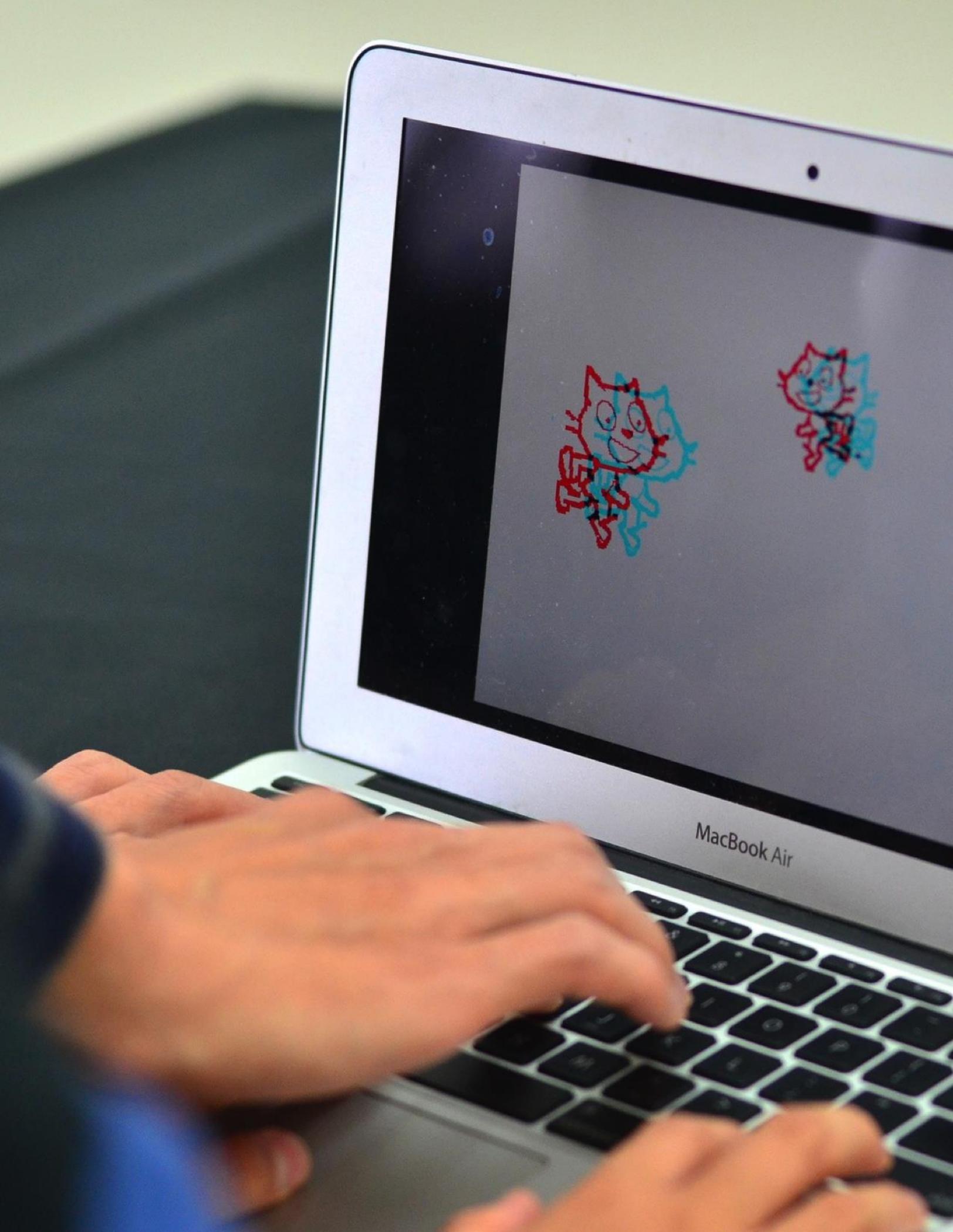
メモ

- + すべてのデザイン・アクティビティは、時間、リソースによって、また与えられた時点の自身の能力によって、制限されます。おまけに妥協が必要となる場合があります。オープンエンドのデザインセッション[訳注1]は、プロジェクトの重要な要素について、生徒と会話する絶好の機会です。
プロジェクトの最も重要な面は何ですか。
残りの時間で無理なくやり遂げられるものは何ですか。

追加メモ記入欄

- _____
- _____
- _____
- _____

[訳注1] 後続のデザインスプリント等に反映するための改善策。
[訳注2] デザインスプリント・セッションのことを指す



MacBook Air

プロジェクト・フィードバック

 **推奨時間**
30-45分

目的

- このアクティビティを完了すると、生徒は、
- + 小規模な批評グループで一緒に作業して、お互いで自分のプロジェクトについて事前にフィードバックし合います。[訳注1]
 - + プロジェクトの進展状況を調べます。
 - + 他の人のためのフィードバックを発表したり、共有します。

アクティビティの説明

- グループを3~4人のフィードバックチームに分けます。フィードバックチームのメンバーは、プロジェクトチームのメンバー以外の人にします。
必要に応じて、生徒に批評グループ（ユニット0 “批評グループ”のアクティビティを参照）を集めさせます。
- 「プロジェクト・フィードバック」配布用プリントを各人に配布し、配布用プリントのそれぞれの項目を確認します。
生徒に、配布用プリントの上部に名前とプロジェクト名を記入してもらいます。
- 生徒に時間を10分与え、フィードバックチームで各々のプロジェクトをレビューし、「プロジェクト・フィードバック」配布用プリントのレッド、イエロー、グリーンの問題を使ってプロジェクトのドラフトを批評してもらいます。
レビューが完了したら、各生徒はフィードバック・グループの他のメンバーから自分のプロジェクトに関するフィードバックを受けます。[訳注2]
- フィードバックのラウンドがすべて終了した後、生徒に時間を与え、フィードバックをレビューして次の“デザインスプリント”セッションのプロジェクトに取り入れたい意見を反映させるために、プロジェクトチームメンバーと打ち合わせさせます。
必要に応じて、このアクティビティの最後に、記入済み「プロジェクト・フィードバック」配布用プリントを集めておきます。次の“プロジェクト・チェックイン”のアクティビティがデザインスプリント・セッションの始めに生徒に返却するためです。

リソース

- 「プロジェクト・フィードバック」配布用プリント

振り返りを促す質問

- + あなたのプロジェクトに対してどんなフィードバックが得られましたか？
- + 次に、どのようなフィードバックをプロジェクトに組み込む予定ですか？（もしあれば）

作業の確認ポイント

- + 各生徒は、さまざまなソースからのフィードバックを得る機会や受ける機会を持ったか？
- + 各生徒は、「プロジェクト・フィードバック」配布用プリントを記入済みか？

メモ

- + さまざま人々は、進行中のプロジェクトについて異なる視点を提供するでしょう。学習者が自分自身を含むさまざまなソースからフィードバックを得る機会を作りましょう！

追加メモ記入欄

- _____
- _____
- _____
- _____

[訳注1] アンフォーカス・グループやショーケースのアクティビティの事前に行う。

[訳注2] 交代して、フィードバックのラウンドを重ねます。

[訳注3] フィードバックチーム、批評グループ、自分自身

プロジェクト フィードバック

フィードバック対象: _____

プロジェクト名: _____

レッド、イエロー、グリーン

フィードバックをくれた人	[レッド] うまくいかないことは何ですか？ 改善可能ですか？	[イエロー] 混乱していることは何ですか？ 別の方法でできそうですか？	[グリーン] うまくいっていることは何ですか？ プロジェクトについて本当に好きなことは何ですか？

プロジェクトに関するフィードバックを考えるのに役に立つヒント：

- + 明快さ：プロジェクトがやるはずだったことが、わかりましたか？
- + 機能性：プロジェクトにはどんな機能がありますか？期待どおりにプロジェクトは動作しますか？
- + アピール力：プロジェクトは、どんな風に魅力的ですか？
インタラクティブですか？独創的ですか？洗練されていますか？面白いですか？興味深いですか？
対話したとき、どう感じましたか？

プロジェクト・チェックイン

 **推奨時間**
15-30分

目的

- このアクティビティを完了すると、生徒は、
- + プロジェクトの進行状況とフィードバックを確認します。
 - + プロジェクト完了に必要なアクティビティやタスクのアウトラインをつかみます。
 - + プロジェクト完了に必要なリソースのリストを作成します。

アクティビティの説明

- このアクティビティでは、生徒はプロジェクト・チェックインを行います。プロジェクト・チェックインでは、仲間のチームメンバーに今までのデザインの進行状況をアップデートし、受け取ったフィードバックを元に、今後のデザインプリントの計画を立てます。
必要に応じて、このアクティビティ中に、生徒やグループをガイドする「プロジェクト・チェックイン」配布用プリントを渡します。
- グループをプロジェクトチームに分けます。
必要に応じて、記入済みの「プロジェクトプラン」、「プロジェクト・フィードバック」、および「アンフォーカスグループ」配布用プリントを生徒に再び配布します。
- チームに時間を与え、オリジナルのプロジェクトのゴールと入手済みのフィードバックを振り返ってもらいます。
生徒に、今後のデザインプリントに向けたプロジェクトの改善のための次のステップと計画の概要を説明してもらいます。

リソース

- 「プロジェクト・チェックイン」配布用プリント

振り返りを促す質問

- + これまでのところ、プロセスの中でうまくいっている部分は何ですか？
- + プロジェクトのどの部分にまだ取り組む必要がありますか？
- + 次にプロジェクトのどの部分に取り組みますか？
- + 進展させるためには、どんな助けを必要としますか？

作業の確認ポイント

- + チームは適切な進行と計画を作成しているか？
- + ディスカッションしたり、プロジェクトを役割分担しながら、グループメンバーは協調して取り組んでいるか？

メモ

- + プロジェクト・チェックインは、短いプラン作成のアクティビティです。各デザインプリントセッションの開始時に、ウォームアップ・アクティビティとして、プロジェクト・チェックインを使うことを推奨します。

追加メモ記入欄

- _____
- _____
- _____
- _____

[訳注1] 参加者がお互いの率直な気持ち、気がかりな事、期待などを共有する時間を取ることに

プロジェクト チェックイン

チェックイン作成者： _____

チームでデザインの進行状況についてディスカッションし、フィードバックを元に次のステップの計画のアウトラインを説明してください。

プロジェクト進行状況

これまでのところ、プロセスの中でうまくいっている部分は何ですか？

プロジェクトのどの部分にまだ取り組む必要がありますか？

次のステップ

各プロジェクトメンバーは、次にプロジェクトのどの部分に取り組みますか？

進展させるためには、どんな助けを必要としますか？

アンフォーカス・グループ

 **推奨時間**
30-45分

目的

このアクティビティを完了すると、生徒は、
+ プロジェクトの進行状況とフィードバックをインタビューし、オブザーブ（観察）し、他の人に質問します。

アクティビティの説明

- アンフォーカスグループを主催するのは、IDEO[訳注1]を参考にしたアイデアです。アンフォーカスグループの概念を紹介します。アンフォーカスグループでは、生徒がプロジェクトの進行状況を共有し、多様な人々の集まりからのフィードバックを求めます。
- 必要に応じて、各自に「アンフォーカスグループ」配布用プリントを配ります。
- 生徒が、アンフォーカス・グループの対象者をブレインストーミングする手助けをします。生徒に、焦点をあてている聴衆と、通常は想定しないようなユーザや想定外のケースを考慮するよう勧めます。（例えば、親、教師、兄弟、他の生徒、コミュニティメンバー）その人達は、ユニークな視点や興味深いフィードバックをしてくれます。
- 生徒に時間を与えて、2つのアンフォーカスグループのメンバーからのフィードバックを確認してインタビューし、観察して記録してもらいます。
- 色々なアンフォーカスグループのソースから集めたフィードバックを共有するため、生徒に時間を与え、プロジェクトチームのメンバーと打ち合せします。
必要に応じて、このアクティビティの最後に、記入済み「アンフォーカスグループ」配布用プリントを集めておきます。次のプロジェクト・チェックインまたはデザインスプリントのセッションの開始時に生徒に返却するためです。

リソース

- 「アンフォーカスグループ」配布用プリント

振り返りを促す質問

- + アンフォーカスグループの参加者と彼らを選んだ理由を記述して下さい。
- + 彼らのアイデアはあなたのプロジェクトにどう影響を与えますか？

作業の確認ポイント

- + 生徒は、2つのアンフォーカスグループの参加者を特定し、インタビューしたか？

メモ

- + フィードバックのソース[訳注2]を調査したり、発見する中で、生徒の創造性を高めるよう手助けします。手助けに興味を持ちそうな地元のゲームデザイン会社はありますか？プロジェクトを別の学校の生徒と共有することが可能ですか？
- + セッション期間中に、インタビューされるアンフォーカスグループのメンバー（教師、親、兄弟、コミュニティメンバーなど）を用意できない場合、授業外のアクティビティを準備したり、宿題にすることができます。

追加メモ記入欄

- _____
- _____
- _____
- _____

[訳注1] アメリカ合衆国カリフォルニア州パロアルトに本社を置くデザインコンサルティング会社。

[訳注2] フィードバックチーム、批評グループ、自分自身だけでなく学校関係者以外も対象。

アンフォーカス グループ

プロジェクト名： _____
インタビュー担当： _____

このアクティビティでは、進行中のプロジェクトについてのフィードバックをもらうため、他の人にインタビューして観察（オブザーブ）します。

特定する

- + どんな人が、プロジェクトにユニークな視点を提供してくれますか？
- + プロジェクトのドラフトの共有を計画している2つのアンフォーカスグループメンバーは誰ですか？

観察（オブザーブ）する

アンフォーカスグループとプロジェクトを共有し、リアクションを観察します。

- + 彼らは何に立ち往生していますか？
- + 彼らは、想像したようにプロジェクトとやり取りしていますか？
- + 彼らは何か驚くべきことをしていますか？

インタビューする

観察後、経験したことをグループにインタビューします。

- + インタビューからどんなフィードバックが受けましたか？
- + 次のプロジェクトにどのような提案を組み込む計画ですか？（もしあれば）

ショーケース の準備

 **推奨時間**
30-45分

目的

このアクティビティを完了すると、生徒は、
+ 最終版のプロジェクトの草案に取り組み、プロジェクト・ショーケースを準備します。

アクティビティの説明

- 頑張り認め、生徒が経験したことを振り返る方法として、自分達のプロジェクトをお互い（おそらくゲストも）で共有することを、生徒に思い出させます。
このセッションは、仕掛り中の作品を完成させて、プロジェクトを他の人に共有するための作戦を考え出すチャンスだと説明します。
- 生徒にプロジェクトに取り組む時間を与え、プロジェクト・ショーケースで、最終版の草案をプレゼンする準備してもらいます。
必要に応じて、プレゼンテーションを容易にするために、最終版の作品をクラス・スタジオに集めます。必要に応じて、生徒にHackathonスタジオにプロジェクトを追加してもらいます。
- 生徒に、「プロジェクト・リフレクション」配布用プリントを配布し、彼らが自身の経験を他の人にプレゼントする方法として、「What?, So What?, Now what?」フレームワーク[訳注1]についてディスカッションします。

リソース

- 「プロジェクト・リフレクション」配布用プリント
- Hackathonスタジオ
<http://scratch.mit.edu/studios/488267>

振り返りを促す質問

- + どんなプロジェクトですか？
- + プロジェクトの開発プロセスはどうでしたか？
- + 次に何を創作したいですか？

作業の確認ポイント

- + 各グループか個人は、「プロジェクト・リフレクション」配布用プリントを記入済みか？

メモ

- + 生徒は自分のプロジェクトを完成させなければならないと、不安やストレスを感じるかもしれません。
この機会に、以下のことをリマインドします。
(1) 今回の経験は、コンピュータ・クリエイターとして、彼らの道のりの、通過点にすぎません。
(2) ストレスのいくつかは、自分のためになる可能性があります。ゴールに集中し、物事を終わらせるのに役立ちます！

[訳注1] 振り返りを効率よく行うためのフレームワーク。

What? (何が起ったのか)、So what? (どんな意味があったのか、何が良く/何が悪かったか)、Now what? (これからどうするか) の3つの質問をして振り返りを促す。

追加メモ記入欄

- _____
- _____
- _____
- _____

プロジェクト リフレクション

作成者： _____

デザインのプロセスを振り返るために、以下の項目を記入してください。

What? (何が起こったのか)

どんなプロジェクトですか？

プロジェクトは、どのように動作しますか？ そのアイデアをどうやって思い付きましたか？

So what? (どんな意味があったか、何が良く/何が悪かったか)

プロジェクトの開発プロセスはどうでしたか？

どんな興味を持ち、どんなチャレンジをし、どんなサプライズがありましたか？それは何故ですか？

何を学びましたか？

Now what? (これからどうするか)

自分のプロジェクトについて、何を最も誇りに思いますか？

あなたは何かを変えるつもりです？

次に何を
創作したい
ですか？



ショーケース

 **推奨時間**
45-60分

目的

このアクティビティを完了すると、生徒は、
+ 自分の最終版のプロジェクトを他の人と共有し、デザインのプロセス全体とコンピュータ上の創作の体験を振り返ります。

アクティビティの説明

- 広いスペースにゲストを招待し、音楽を演奏したり、飾り物を吊るしたり、軽食を提供することで、祝賀ムードを醸し出します。
- 必要に応じて、プロジェクトを表示するプロジェクタとスクリーンを使います。
- 生徒に、最終版のプロジェクトを共有してもらい、デザインのプロセスについて、他の人とディスカッションさせます。必要に応じて、ノートや以前のプロジェクトを利用可能にしておく、生徒の進行状況を可視化できます。
- 生徒に時間を与え、デザイン日誌をレビューし、デザイン日誌やグループ討議での振り返りを促す質問に回答することで、彼らの「創造的コンピューティング」のすべての体験を振り返ってもらいます。

リソース

- プレゼンテーション用プロジェクタとスクリーン(任意)

振り返りを促す質問

- + デザインメモに目を通してください。どんなメモを取りましたか？
- + どのメモが、最も役に立ちましたか？
- + これまでのところ、取り組んだお気に入りのプロジェクトは何でしたか？なぜお気に入りなんですか？
- + 次に何を創作したいですか？

作業の確認ポイント

- + チームや生徒個人は、それぞれ自分の作品を共有する機会を持ち、褒められたか？

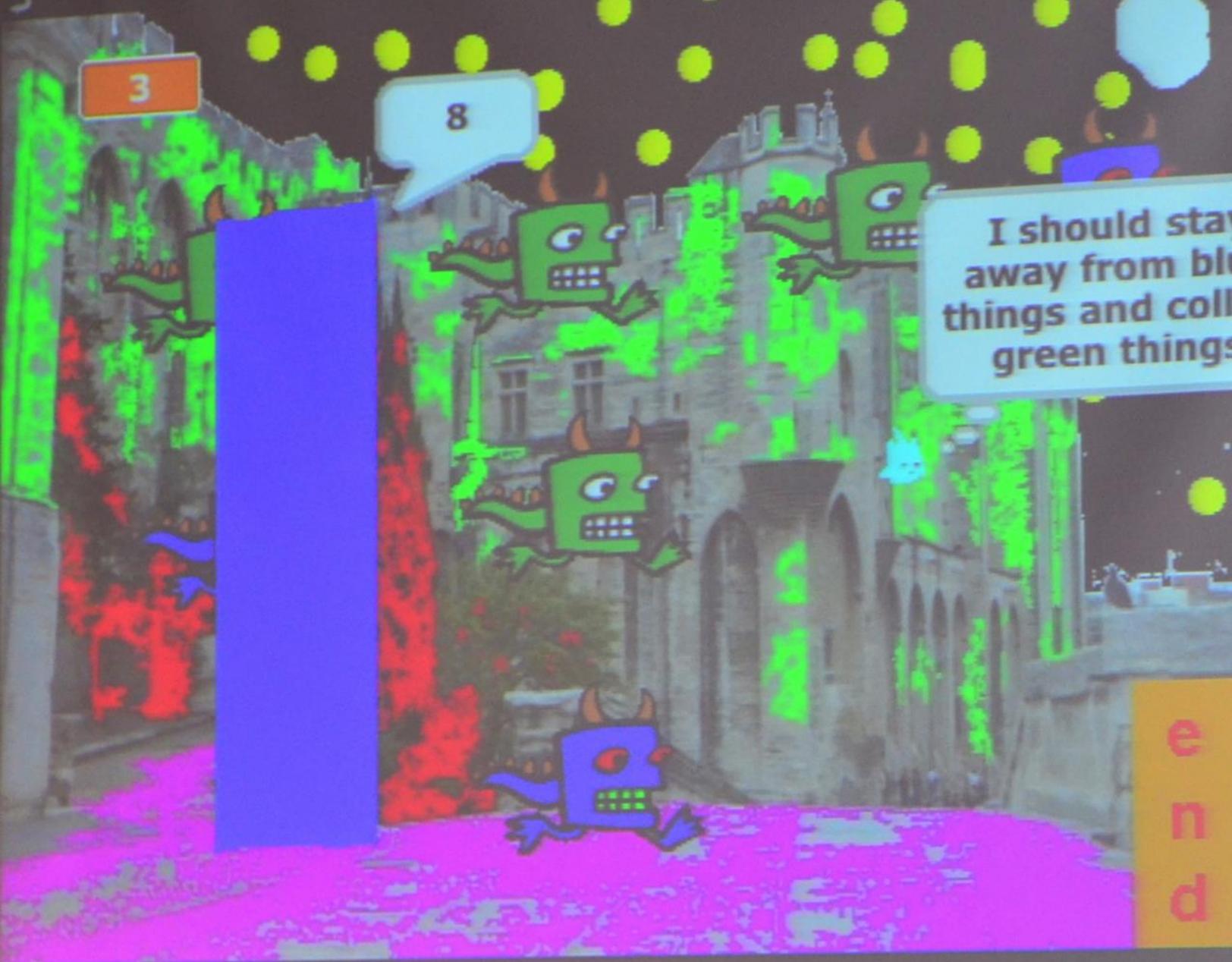
メモ

- + 共有方法には、さまざまな方法があります。
個人からグループ全体に向けたプレゼンテーション、同時並行で行う生徒同士のプレゼンテーション[訳注1]、ライブデモ、ウェブからのプロジェクトへのアクセス、等
- + 評価目的で集められそうなアーティファクト（成果物）には、プロジェクトポートフォリオ、デザイン日誌、最終版「プロジェクト・フィードバック」配布用プリント、最終版「プロジェクト・リフレクション」配布用プリントがあります。（付録を参照）

追加メモ記入欄

- _____
- _____
- _____
- _____

[訳注1] 原文は、Concurrent subset of students presenting



付録



用語集

カリキュラムガイドのキーワード、概念、および実践の手引き：

追加のScratch固有の用語は、Scratchヘルプページ(<http://scratch.mit.edu/help>)、またはコミュニティが作成したScratch Wiki(<http://wiki.scratch.mit.edu>はScratch)を参照して下さい。

Scratchスクリーニング：Scratchプロジェクトを観察するために生徒が集まり共有するためのアクティビティ。

Tipsウィンドウ：プロジェクト・エディタに直接組み込まれている。TipsウィンドウはScratchのヘルプがあるフォームです。

アーケードデー：生徒の作品とグループ全体のアクティビティを共有するためのイベント[訳注1]。

生徒は、完成したプロジェクトをプレゼンテーションモードにし、歩き回り、お互いの作品に係わります。

アニメーション：一連の静止画像をすばやく表示することによって連続的な動きが生じるように見える。一種の眼の錯覚。

アンフォーカス・グループ：生徒が進行中のプロジェクトを共有し、多様な人々の集まりからのフィードバックを要求するアクティビティ。

イベント処理：1つの事が別の事を引き起こすコンピュータ処理の概念。

インタラクティブなコラージュ：さまざまなクリック可能なスプライトを組み込んだ切り絵のScratchプロジェクト。

ギャラリーウォーク：生徒がプロジェクトをプレゼンテーションモードにして、歩き回り、お互いのプロジェクトの探究を共有するためのアクティビティ。

クローン：スプライトがプロジェクトの実行中に自分の複製を作成することを可能にするScratchの機能。

コスチューム：多くの「フレーム」の中の1つ、またはスプライトの代わりに姿（外観）の中の1つ。スプライトはコスチュームのどれかに姿（外観）を変えられる。

コンピュータ処理の概念：デザイナーがプログラムする（順次処理、繰り返し処理、条件分岐、イベント処理、並列処理、演算子、データなど）ときに係わる概念のこと。

コンピュータ処理の視点：デザイナーはコンピュータ処理を通して周囲の世界を形成する可能性があるという幅広い視点のこと。自分自身の表現、他の人とながら、その世界のテクノロジーの役割に関する疑問の提起など。

コンピュータ上の実践：プログラマーが仕事をするにつれて習得する思考の習慣。実験や反復、テストとデバッグ、作品のリミックスと再利用、抽象化とモジュール化など。

ショーケース：生徒が最終版のプロジェクトを他の人にプレゼンして、デザインのプロセスとコンピュータ上の創作の経験を振り返ることを共有するためのイベント[訳注1]。

スクリプト：順序を形成する一連につながった1つ以上のScratchブロック。スクリプトは、入力（例えば、マウスクリック、ブロードキャスト）に反応するイベントのブロックで始まる。イベントが発生すると、イベントのブロックの後ろにつながっているブロックが1回だけ同時に実行される。

スタジオ：Scratchオンライン・コミュニティ上のユーザー作成のギャラリー。1人または複数のユーザーが投稿したプロジェクトにハイライトを当てて表示するために使用可能。

ステージ：Scratchプロジェクトの背景。ステージにはスプライトと類似のスクリプト、背景（コスチューム）、および音がある。

スプライト：メディア・オブジェクト。Scratchプロジェクトのステージ上でアクションを演じる。

[訳注1] 原文は、strategy

データ：値の保存、取得、および更新に関するコンピュータ処理の概念。

デザインスプリント：プロジェクトを開発する際に集中して作業することに費やされる指定された時間。

デザインデモ：生徒がクラスに作品を発表したり、プロジェクト内で特定のブロック、スキル、またはデザイン戦略をどうやって実装するかをデモンストレートするアクティビティ。

テストとデバッグ：物事が確実に動作することを確認するコンピュータ上の実践。問題が発生した場合、問題を見つけて解決する。

ハードウェアと拡張機能：Scratchのデジタル世界と物理的世界を結びつける補足的な素材。
ハードウェア機能拡張の例として、LEGO WeDo、PicoBoard、およびMaKey MaKeyがある。

バックバック：プロジェクト間でメディアやスクリプトを引き継ぐための使用可能なScratchの便利な機能。

ピアインタビュー：振り返り、自己評価、研究のプロセスについて互いに生徒が交代でインタビューする共有のためのアクティビティ。

ピッチ：他のチームメンバーを採用するためにプロジェクトのアイデアを発表する、または他のチームから採用されるために興味、スキル、才能をプロモーションするアクティビティ。

ビットマップ：離散的なカラーの値（別名：“ピクセル”）の2次元配列（グリッド）によって定義される画像のこと。ベクターグラフィックスと対比して使用される。

ビデオセンシング：ウェブカメラのビデオを使用して動きを検出したり、ステージ上にビデオ入力を表示するScratchの機能。

フィードバックフェア：参加者の半分のプロジェクトを開いた状態で席に留まり、残りの半分はプロジェクトを探索して質問し、フィードバックをする共有のアクティビティ。完了すると、生徒は入れ替わって、もう一度、同じ手順を開始する。

プレゼンテーションモード：Scratchの表示モードで、プロジェクトを拡大して表示できる。Scratchプログラムの左上にあるボタンを押すとアクセス可能。このモードはフルスクリーンモードまたは拡大スクリーンとも呼ばれる。

ブロードキャスト：Scratchプログラムを通して送信されるメッセージ。受信スクリプトを起動する。

プロジェクト・エディタ：プロジェクトを変更できるScratchオンライン・コミュニティの機能。スクリプト領域（スクリプトが組み立てられる場所）、スプライト領域（スプライトを操作できる場所）、ステージ領域（スプライトが配置され、背景にアクセスできる場所）がある。

ブロックを作成：“その他”カテゴリにある機能で、生徒は独自のカスタムブロックや手続きを作成して定義できる。

プロフィール・ページ：Scratchオンライン・コミュニティの専用ページで、自分が作成したりブックマークしたプロジェクト（別名「お気に入り」）など、Scratchユーザーに関する情報を表示する。

ペアプログラミング：開発者がプロジェクトで二人一組になって作業を行うプログラミングのためのメソッド。

ペイントエディタ：Scratchのビルトインされたイメージ・エディタ。多くのScratcherが独自のスプライト、コスチューム、背景を作成するためにペイントエディタを使用する。

ベクターグラフィック：幾何学的シェープ（例えば、円、長方形）と色の集まりによって定義されるイメージ。ビットマップと対比して使用される。

リミックス：オリジナル作品（または別のリミックス）から派生した創造的な作品。リミックスはオリジナルの作品とある程度の類似性を保ちながら、新しいコンテンツやスタイリッシュな要素を導入するのが一般的。

レッド、イエロー、グリーン：個人が自分のプロジェクトの側面を識別する際の振り返りと共有のためのアクティビティ。
“レッド”うまくいかないか、まだ作業を必要とする、“イエロー”混乱や異論がある、“グリーン”うまく動作する。

動き：Scratchブロックの10種類のカテゴリの1つ。青色に色分けされ、スプライトの動きを制御するために使用される。

演劇のメタファー：俳優（スプライト）、コスチューム、背景、スクリプト、ステージなど、演劇の意図的な類似性を強調するScratchのデザインを記述（比喻）する方法。

演算子：数学的および論理的表現をサポートするコンピュータ処理上の概念。

音：Scratchプロジェクトで再生できるオーディオファイル。Scratchのビルトインされた音ライブラリからインポートするか、新しい録音を作成して利用できる。音量、テンポなどをコントロールするサウンド用のブロックを使って音を再生する。

繰り返し処理：同じ順次処理を複数回実行するコンピュータ処理上の概念。

再利用とリミックス：既存のプロジェクトやアイデアを基にして何かを作るコンピュータ上の実践。

実験と反復：少しずつ開発して試してから、さらに開発するコンピュータ上の実践。

順次処理：あるタスクのための連続したステップを識別するコンピュータ処理上の概念。

条件分岐：条件（例えば、現在の変数値）に基づいて判断するコンピュータ処理の概念。

調べる：Scratchブロックの10種類のカテゴリの1つ。淡青色で色分けされ、さまざまな入力形態（例えば、マウス位置）またはプログラム状態（例えば、スプライト位置）を検出するために使用される。

制御：Scratchブロックの10種類のカテゴリの1つ。ゴールドに色分けされ、スプライトやステージ[訳注1]を制御するために使用される。

続けて創ろう(Pass-It-On Story)：ペアで始めて、他の2つのペアに引き継いで、物語を展開し、イメージし直す。

抽象化とモジュール化：全体と部品との間の係わりを探るコンピュータ上の実践

背景(backdrop)：ステージの多くのフレームか、背景(background)のうちの1つ。[訳注2]

批評グループ：プロジェクトをさらに発展させる方法についてのフィードバックを得るために、アイデアを共有し、進行中のプロジェクトをテストするためのデザイナーのグループ。

並列処理：物事を同時に起こすコンピュータ処理上の概念。

変数とリスト：変更可能な値または値の集まり。

Scratchの記憶エリアに記録される。変数は一度に1つの値を保存できるが、リストには複数の値が保存できる。

見た目：Scratchブロックの10種類のカテゴリの1つ。紫色で色分けされ、スプライトの外観を制御するために使用される。

[訳注1] 本文は、Scriptであるが、文意からSpriteとStage判断した。

[訳注2] 本ガイドではbackgroundとbackdropとも訳語は同じ「背景」を使用している。

スタンダード

このガイドのアクティビティは、Common Core State Standards, CSTA K-12 Computer Science Standards, ISTE NETSなど、いくつかの異なるK-12カリキュラム・スタンダードに係わりを持っています。例として、Common Core Standardsに係わりがあります。

詳細については、ガイドのサイト<http://scratched.gse.harvard.edu/guide>を参照下さい。

Common Core State Standards for Mathematic 2010

http://www.corestandards.org/wp-content/uploads/Math_Standards.pdf

- + 問題を理解し、問題を解決するために忍耐強く続ける - 多くのガイドアクティビティでは、生徒が問題を見つけて解決するさまざまな方法を発見するように生徒に勧めます。アクティビティの例：ユニット1 - 4 デバックしよう！
- + 抽象的および定量的な理由 - 抽象的な概念を表現し、変数などの量的関係の理解をScratchで設計された視覚的表現を通じて示すことができます。アクティビティの例：ユニット4 スコア
- + 数学を使ったモデル - このガイドの特定のアクティビティでは、生徒にScratchプログラムとして過去に学習した方程式、データ比較、またはその他の数学的関係を表現するようチャレンジさせます。アクティビティの例：ユニット4 対話処理
- + 正確さへの気配り - オンとオフのスクリーンのアクティビティは、特定の結果を引き出すためのインストラクションや一連のコードを指定する際に、細部に気を配ることの重要性を認識するのに役立ちます。アクティビティの例：ユニット1 ダンスをプログラム
- + 構造を探して利用する複雑なプログラムを構築するために、デバッグのチャレンジ中にスクリプトを通して見たり、プロジェクトをリミックスしたり、作業を見直したりする間に他の人のプロジェクトのコードを読むことは、自分または他のScratchプログラムの中にある繰り返しのパターンと構造を識別するために、生徒をしっかりと目を向けることに取り組ませることができます。アクティビティの例：ユニット3 会話

Common State Standards for English Language Arts / Literacy 2010

http://www.corestandards.org/wp-content/uploads/ELA_Standards.pdf

- + 独立性を示す。 - ガイドのほとんどのアクティビティやプロジェクトは、自発的に促すように設計されるか、独立した作業に合わせて簡単に調整可能です。但し共同作業やグループ作業が奨励されています。アクティビティの例：ユニット1 自己紹介
- + 観客、作品、目的、原則のさまざまな要求に応える。 - 世界中のScratchオンライン・コミュニティにプロジェクトを共有したり、他の人たちのプロジェクトやアクティビティをデザインする際に、さまざまなタイプの聴衆、タスク、目的、原則を生徒に気付かせます。アクティビティの例：ユニット5 アクティビティのデザイン
- + 批評だけでなく理解もする。 - さまざまなフィードバックの演習と共同プロジェクトは、生徒に、進行中の作品を共有して質問して、建設的な批評を交換し合うよう、取り組ませます。アクティビティの例：ユニット0 批評グループ
- + テクノロジーとデジタルメディアを戦略的かつ可能な限り使用します。 - 自発的なアクティビティの中で、Scratchウェブサイトのさまざまな部分にナビゲートし、プロジェクトを開発し、インスピレーションを求め、他者とながら、学習目標を達成することを学びます。アクティビティの例：ユニット5 学びたいことを知ろう
- + 他の観点や文化を理解するようになります。 - 他人のプロジェクトをリミックスする際には、生徒は、自分のではないコードや意図を読み、理解し、解釈する必要があります。共同プロジェクトを構築する時、生徒は協力し、妥協し、他の人と仕事を分かち合うことを学びます。アクティビティの例：ユニット3 続けて創ろう

コンピュータ処理の概念

過去数年間、我々はScratchで行われる学習と開発を記述する方法として「計算論的思考」に魅了されてきました。このセクションで以下を共有します。

- (1) 我々の計算論的思考の定義としての、コンピュータ処理の概念、実践、視点。
- (2) コンピュータ上の実践の熟練度の評価ツール
- (3) 教師がクラスルームでのコンピュータ上の実践をどうサポートしているかを、自ら振り返るためのツール

これらの定義とツールは、教育開発センターの児童&技術センターのWendy Martin, Francisco Cervantes, Bill Tally及びMITメディアラボのMitch Resnickとで共同開発されました。

追加の計算論的思考のリソースは、<http://scratched.gse.harvard.edu/ct> から入手可能です。

コンピュータ処理の概念

概念	説明
順次処理	ある作業のための連続したステップを識別する。
繰り返し処理	同じ順次処理を複数回実行する。
並列処理	物事を同時に起こす。
イベント処理	1つの事が別の事を引き起こす。
条件分岐	条件に基づいて決定を行う。
演算子	数学的および論理的表現をサポートする。
データ	値を保存し、取得し、および更新する。

コンピュータ上の実践

実践	説明
実験と反復	少しずつ開発して試してから、さらに開発する。
テストとデバッグ	物事が確実に動作することを確認する。問題が発生した場合、問題を見つけて解決する。
再利用とリミックス	既存のプロジェクトやアイデアを基にして何かを作る。
抽象化とモジュール化	全体と部品との間の係わりを探る。

コンピュータ処理の視点

観点	説明
表現(expressing)	コンピュータ処理が創作品の媒体であることを認識する。 「私は創作できます」
つながり(connecting)	他者との創造力と他者のために創造力を認識する。 「他の人にアクセスできる時、私はいろいろなことをすることができます」
質問(questioning)	世界についての質問する能力を与えられたと感じること。 「私は（コンピュータを使用して）世界（中のコンピュータ的なもの）を理解するための質問ができます」

コンピュータ上の実践の発達状況評価

以下のツールは、生徒が計算論的思考の実践（実験と反復、テストとデバッグ、再利用とリミックス、抽象化とモジュール化）を使いこなす能力の発育を評価するために使用可能です。

最初の列は、生徒への質問（たとえば、デザイン日誌の振り返りを促す質問やインタビューの一部）を示しています。第2, 3, 4列は、熟練度（低、中、高のレベル）を示しています。

実験と反復	低	中	高
プロジェクトをステップ・バイ・ステップに構築する方法を説明してください。	生徒は、プロジェクトを構築するための基本的な説明を提供しますが、特定のプロジェクトに関する詳細は提供しません。	生徒は、特定のプロジェクトを特定の順序で構築する一般的な例を示します。	生徒は、特定のプロジェクトのさまざまな構成要素と、特定の順序でどのように開発したかに関する詳細を提供します。
あなたがプロジェクトに携わった時に試したことは何ですか？	生徒は、試したものの具体例を提供しません。	生徒は、プロジェクトで何かを試す一般的な例を示します。	生徒は、プロジェクトで試しているさまざまな事の具体例を提供します。
どのような改訂を行いましたか、なぜ、それらの改定版を作成したのですか？	生徒は、改訂をしていない、または改定したが、例は示していないだけだと述べています。	生徒はプロジェクトに行った特定の改訂版を1つ説明します。	生徒は、プロジェクトに追加した具体的な事とその理由を説明します。
プロジェクトで何かをやるうとしたか、新しいことをしようとした時のさまざまな方法を説明してください。	生徒は、新しい何かの試みの例を提供しません。	生徒は、プロジェクトでの新しい何かの試みの例を提供します。	生徒は、プロジェクトで試した特定の新しい事を説明します。
テストとデバッグ	低	中	高
あなたが望んだプロジェクトとは異なるプロジェクトを実行した時、起こった事を説明してください。	生徒は、自分が望んだ事とプロジェクトを実行した時と何が違っていたのか、説明しません。	生徒は、プロジェクトで間違っやった事を説明しますが、何をしたかったのか、説明しません。	生徒は、起こった事とプロジェクトを実行した時に起こしたかった事の具体例を示します。
問題の原因を調べるために、どうやってスクリプトを通して読んだのか説明してください。	生徒は、問題を説明しません。	生徒は、スクリプトを通して読むことについて説明しますが、コード内の問題の発見の具体例は示しません。	生徒は、スクリプトを通して読むことを説明し、コード内の問題の発見の具体例を示します。
起こった事を確認するために、どのように変更を加えてテストしたかの説明してください。	生徒は、あった問題やその解決方法について説明しません。	生徒は、動作するかどうか確認するために、変更を加えてテストする一般的な例を提供します。	生徒は、動作するかどうか確認するために、変更を加えてテストする具体例を示します。
問題を解決するための他の方法を、どのように考慮したか説明してください。	生徒は、問題の解決策の例を提示しません。	生徒は、問題の解決策の一般的な例を提供します。	生徒は、問題の解決策の具体例を示します。

再利用とリミックス	低	中	高
他のプロジェクトを試すか、自分のスクリプトを読むことによって、インスピレーションを見つけたかどうか、どのように見つけたのか、説明してください。	生徒は、他のプロジェクトのアイデアやインスピレーションをどのように見つけたか、説明しません。	生徒は、自分にインスピレーションを与えたプロジェクトの一般的な説明を提供します。	生徒は、自分にインスピレーションを与えたプロジェクトの具体例とその方法を提供します。
どうやって別のプロジェクトの一部を選んで、自分のプロジェクトに適用させましたか？	生徒は、他のプロジェクトのスクリプト、アイデア、またはリソースをどのように適合させたか、説明しません。	生徒は、他のプロジェクトに適用したスクリプト、アイデア、またはリソースを特定しています。	生徒は、他のプロジェクトに適用したスクリプト、アイデア、またはリソースの具体例とその方法を提供します。
改善や強化のために、どうやって既存のプロジェクトを変更しましたか？	生徒は、別のプロジェクトの変更について説明しません。	生徒は、別のプロジェクトに行った変更の一般的な説明を提供します。	生徒は、他のプロジェクトに行った変更の具体例とその理由を提供します。
構築した作品を提供してくれた人やインスピレーションを与えてくれた人に、どうやって謝辞を示しましたか？	生徒は、他者に謝辞を示していません。	生徒は、作品にインスピレーションを与えてくれた人の名前を挙げます。	生徒は、Scratchウェブサイト上で、作品にインスピレーションを与えてくれた人をプロジェクトで明らかにしています。
抽象化とモジュール化	低	中	高
自分のプロジェクトに必要なスプライトは何で、どこに進むべきか、どうやって決定しましたか？	生徒は、スプライトをどのように選択したかについての説明を提供しません。	生徒は、特定のスプライトを選択する決定方法についての一般的な説明を提供します。	生徒は、プロジェクトのゴールに基づいてスプライトを決定した方法の具体的な説明を提供します。
自分のプロジェクトに必要なスクリプトは何で、何を実行すべきか、どうやって決定しましたか？	生徒は、どうやってスクリプトを作成したかの説明を提供しません。	生徒は、特定のスクリプトを作成する決定方法の一般的な説明を提供します。	生徒は、プロジェクトのゴールに基づいてスクリプトを決定した方法の具体的な説明を提供します。
あなたや他の人にとって意味のある方法でスクリプトをどのように構成しましたか？	生徒は、どうやってスクリプトを構成したかの説明しません。	生徒は、どうやってスクリプトを構成したかの説明を提供します。	生徒は、どうやってスクリプトを構成したかの具体例とその理由を提供します。

クラスルームでのコンピュータ上の実践のサポート

以下のツールは、学習環境（教室、図書館、または別の学習環境など）でのコンピュータ上の実践をサポートする方法を振り返るのに役立ちます。このツールの目的は、あなたがデザインしてサポートしていることを知る機会の程度（ない、ある程度、多数）に気付くのを助けることです。

実験と反復：少しずつ開発して試してから、さらに開発する。

アクティビティは、生徒に対して機会をどの程度、与えたか	ない	ある程度	多数
プロジェクトをステップ・バイ・ステップで構築する機会			
あなたがするように物事を試す機会			
起こることに基づいて改訂版を作る機会			
さまざまな方法で試してみる機会、または新しいことを試す機会			
次回に向けてのメモ： 「ない」の場合、どうやってもっと余裕を持つか／時間を作るか？ 「ある程度」の場合、どうすればアクティビティを深め、強化することができるか？ 「多数」の場合、何を気づいたか、学んだのか？			

テストとデバッグ：物事が確実に動作することを確認し、問題が発生した場合、問題を見つけて解決する。

アクティビティは、生徒に対して機会をどの程度、与えたか	ない	ある程度	多数
プロジェクトを実行するときに、何が起こるかを観察する機会			
あなたが期待したことと異なっている事を書き出す機会			
問題の原因を調査するためにスクリプトを通して読む機会			
変更を加えてテストして何が起こるかをテストする機会			
問題を解決する他の方法を検討する機会			
次回に向けてのメモ： 「ない」の場合、どうやってもっと余裕を持つか／時間を作るか？ 「ある程度」の場合、どうすればアクティビティを深め、強化することができるか？ 「多数」の場合、何を気づいたか、学んだのか？			

再利用とリミックス:既存のプロジェクトやアイデアを基にして何かを作る。

アクティビティは、生徒に対して機会をどの程度、与えたか	ない	ある程度	多数
他のプロジェクトを試してスクリプトを読んで、アイデアやインスピレーションを見つける機会			
別のプロジェクトの一部を選んで、あなたのプロジェクトに適用する機会			
既存のプロジェクトを修正して改善したり、変更して強化する機会			
作品の構築に貢献したり、インスピレーションを与えてくれた人に謝辞を示す機会			
次回に向けてのメモ： 「ない」の場合、どうやってもっと余裕を持つか／時間を作るか？ 「ある程度」の場合、どうすればアクティビティを深め、強化することができるか？ 「多数」の場合、何を気づいたか、学んだのか？			

抽象化とモジュール化：全体と部品との間の係わりを探る。

アクティビティは、生徒に対して機会をどの程度、与えたか	ない	ある程度	多数
あなたのプロジェクトに必要なスプライトは何か、そしてどこに進むか、決定する機会			
あなたのプロジェクトに必要なスクリプトは何か、そして何を実行するのか、決定する機会			
あなたや他の人にとって意味のある方法でスクリプトを構成する機会			
次回に向けてのメモ： 「ない」の場合、どうやってもっと余裕を持つか／時間を作るか？ 「ある程度」の場合、どうすればアクティビティを深め、強化することができるか？ 「多数」の場合、何を気づいたか、学んだのか？			

参考書籍、論文、レポート

創造的コンピューティングの探究をさらにサポートする記事の抄録：

書籍

- + Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. New York, NY: Basic Books.
- + Papert, S. (1993). *The children's machine: Rethinking school in the age of the computer*. New York, NY: Basic Books.
- + Kafai, Y. B. (1995). *Minds in play: Computer game design as a context for children's learning*. Mahwah, NJ: Lawrence Erlbaum. Available at <http://www.yasminkafai.com/minds-in-play/>
- + Margolis, J., & Fisher, A. (2002). *Unlocking the clubhouse: Women in computing*. Cambridge, MA: MIT Press.
- + Margolis, J., Estrella, R., Goode, J., Holme, J.J., & Nao, K. (2008). *Stuck in the shallow end: Education, race, and computing*. Cambridge, MA: MIT Press.
- + Kafai, Y. B., Peppler, K. A., & Chapman, R. N. (2009). *The computer clubhouse: Constructionism and creativity in youth communities*. New York: Teachers College Press.
- + Rushkoff, D. (2010). *Program or be programmed: Ten commands for a digital age*. New York, NY: OR Books.
- + Kafai, Y. B., & Burke, Q. (2014). *Connected code: Why children need to learn programming*. Cambridge, MA: MIT Press.

論文

- + Monroy-Hernandez, A. (2012). *Designing for remixing: Supporting an online community of amateur creators*. Doctoral dissertation, Massachusetts Institute of Technology.
- + Brennan, K. (2013). *Best of both worlds: Issues of structure and agency in computational creation, in and out of schools*. Doctoral dissertation, Massachusetts Institute of Technology.

レポート

- + Brennan, K., & Resnick, M. (2012). *New frameworks for studying and assessing the development of computational thinking*. American Educational Research Association meeting, Vancouver, BC, Canada.
- + Brennan, K. (2013). *Learning computing through creating and connecting*. *IEEE Computer, Special Issue: Computing in Education*. doi:10.1109/MC.2013.229

リンク

創造的コンピューティングに役立つリンク:

タイプ	サイト名	リンク先
ウェブサイト	Scratch	http://scratch.mit.edu
ウェブサイト	ScratchEd	http://scratched.gse.harvard.edu
リソース	Offline Version of Scratch	http://scratch.mit.edu/scratch2download
リソース	Scratch Cards	http://scratch.mit.edu/info/cards
リソース	Scratch Community Guidelines	http://scratch.mit.edu/community_guidelines
リソース	Scratch Remix FAQ	https://scratch.mit.edu/info/faq#remix
リソース	Scratch Wiki	http://wiki.scratch.mit.edu
リソース	Scratch Discussion Forums	http://scratch.mit.edu/discuss
リソース	Scratch FAQ	http://scratch.mit.edu/info/faq
リソース	LEGO WeDo Construction Set	http://bit.ly/LEGOWeDo
リソース	MaKey MaKey	http://makeymakey.com
リソース	PicoBoard	https://www.sparkfun.com/products/10311
リソース	Scratch Design Studio List	http://scratch.mit.edu/users/ScratchDesignStudio
ビデオ	Scratch Overview Video	http://vimeo.com/65583694 http://youtu.be/-SjuiaWRMU4
ビデオ	Unit 1 Programmed to Dance Videos	http://vimeo.com/28612347 http://vimeo.com/28612585 http://vimeo.com/28612800 http://vimeo.com/28612970
ビデオ	Backpack Video Tutorial	http://bit.ly/scratchbackpack
ビデオ	Make a Block Video Tutorial	http://bit.ly/makeablock
ビデオ	Variables Video Tutorial	http://bit.ly/scratchvariables
ビデオ	How can I connect Scratch with other technologies? Video Playlist	http://bit.ly/hardwareandextensions
ビデオ	Scratch Chain Reaction Video	http://bit.ly/ScratchChainReaction

原書のCreative Computingは、
ハーバード大学 教育学大学院のScratchEdチームによって作成され、
クリエイティブ・コモンズ・ライセンスの下でリリースされました。
また、下記の訳者により日本語に翻訳され、
クリエイティブ・コモンズライセンスの下でリリースされました。

原書名 : Creative Computing

著者 : Karen Brennan, Christan Balch, Michelle Chung
Harvard Graduate School of Education

日本語版タイトル : 創造的コンピューティング

訳者 : 大山 哲志郎 (Tetsushiro Oyama), Scratcher, PMP,
Co-Funder, Mentor at CoderDojo Minami-Kashiwa
Scratch User ID: DobbsBigbear
E-mail: drdobbsbigbear@gmail.com

