

# ライオンシューティング

このラボでは、ごく初歩的なシューティングゲームを作りながら、Mind Render の基本的な使い方やブロック命令を学ぶことができます。ラボの構成は以下のようになっています。

## ラボの使い方

テキストを参考に、「1.プレイヤーをつくる」ラボを作ります。完成すると、「2.敵をつくる」のスタート時と同じ状態になります。

「2.敵をつくる」は、「1.プレイヤーをつくる」が完成した状態から始まります。

以下、同様に完成形が、次のラボのスタート時の状態となっています。

## ラボの構成

ラボ名	概要	学ぶこと	P
0. ゲームをプレイしよう！	ゲームを作る前にまずは遊んでみましょう。	-	3
1. プレイヤーをつくる	プレイヤーとなるキャラクターを追加します。 プレイヤーやカメラを操作するプログラムを作ります。	<ul style="list-style-type: none"><li>・オブジェクトの追加</li><li>・装備の追加</li><li>・モーションの確認と追加</li><li>・キャラクターやカメラを操作するプログラムの作成</li></ul>	4
2. 敵をつくる	敵を追加し、プレイヤーに向かってくるプログラムを作ります。	<ul style="list-style-type: none"><li>・オブジェクトの追加</li><li>・グループ機能</li><li>・オブジェクトの位置（座標）</li><li>・オブジェクトのコピー</li><li>・オブジェクトを動かすプログラム</li></ul>	10
3. 弾を发射する	弾を作って飛ばすプログラムや、弾が背景や敵にヒットした時のプログラムを作ります。	<ul style="list-style-type: none"><li>・分身機能</li><li>・重力の設定</li><li>・オブジェクトを飛ばすプログラム</li><li>・当たり判定</li><li>・オブジェクトの削除</li></ul>	18
4. おさらい	ここまで作った内容を確認します。自由に撃ってみましょう。	-	23

5. 制限を加える	<ul style="list-style-type: none"> <li>・弾数の上限を設定します。</li> <li>・弾数やライオンの数を画面に表示します。</li> </ul>	<ul style="list-style-type: none"> <li>・変数の設定</li> <li>・変数の初期値の設定</li> <li>・弾やライオンの数をカウントする（変数の値を減らす）</li> <li>・テキスト表示のブロックとプログラム</li> </ul>	24
6. ゲームオーバーとゲームクリア	ゲームオーバーとゲームクリアを作ります。	<ul style="list-style-type: none"> <li>・2D オブジェクトの追加</li> <li>・条件分岐のブロックとプログラム</li> <li>・変数関連のブロックとプログラム</li> </ul>	31
7. 演出をする	ゲームの演出として、BGM、効果音、エフェクトを加えます。	<ul style="list-style-type: none"> <li>・BGM、効果音の追加</li> <li>・エフェクトの追加</li> </ul>	34
8. オープニングをつくる	ゲーム開始前に、タイトル表示、ルール・操作説明を表示します。	<ul style="list-style-type: none"> <li>・テキストを表示するブロック</li> <li>・オブジェクトの表示/非表示を切り替えるプログラム</li> </ul>	38

## 0. ゲームをプレイしよう！

これから作るゲームがどんなゲームか、まずは遊んでみましょう。20発の弾で5頭のライオンを倒すシンプルなシューティングゲームです。



# 1. プレイヤーをつくる

始めに、プレイヤーを作りましょう。



## <この章の内容>

概要	学ぶこと
プレイヤーとなるキャラクターを追加します。 プレイヤーやカメラを操作するプログラムを作ります。	<ul style="list-style-type: none"><li>・オブジェクトの追加</li><li>・装備の追加</li><li>・モーションの確認と追加</li><li>・キャラクターやカメラを操作するプログラムの作成</li></ul>

## プレイヤーにするキャラクターを追加する

始めに、プレイヤーにするキャラクター（オブジェクト）を追加しましょう。



画面右下の「オブジェクト追加」ボタンをクリックします。



プレイヤーにしたいキャラクターを選びます。

ここでは、「ドンポンプ」を選びます。



「オブジェクトリスト」に「ドンポンプ」が追加されます。

クリックすると、オレンジ色の枠がつかます。

虫眼鏡アイコンをクリックすると、「オブジェクト詳細」画面が開き、

このオブジェクトの各種設定ができます。



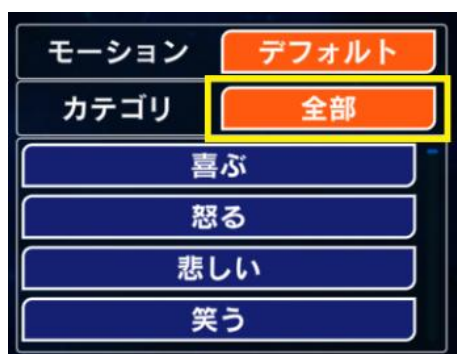
キャラクターの頭上に表示されているアイコンで、手や足に装備する部品などが選択できます。

「右手 (R)」または「左手 (L)」アイコンをクリックします。

手に装備できるものが表示されるので、画面を下方にスクロールして「サブマシンガン 1」を選択します。

装備を解除する場合は、画面の一番上までスク

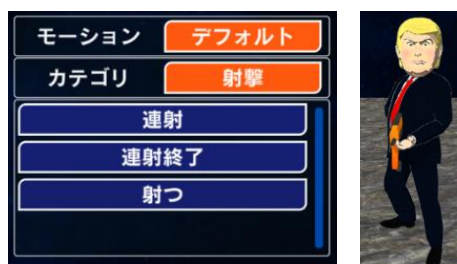
ロールして戻り、「X」をクリックします。



オブジェクト詳細画面では、モーションを確認することもできます。

サブマシンガンを撃つ動作を確認してみましょう。

カテゴリの「全部」ボタンをクリックし、表示されるカテゴリー一覧から「射撃」を選択します。



射撃に関連するモーションが表示されるので、順番に試してみましょう。

(注) この画面ではモーションを追加することはできません。動きを確認するのみです。

モーションを確認したら、画面右上の「閉じる」ボタンでプログラム編集画面に戻ります。



マシンガンを持ったドンポンプを確認してみましょう。

画面上部の「実行」ボタンをクリックします。



ドンポンプが右手に銃を持っているのがわかります。



プログラム実行画面に戻るには、「停止」ボタンをクリックします。

## プレイヤーを操作するプログラムをつくる

次に、プレイヤーを操作するプログラムを作りましょう。



画面右下の「オブジェクト追加」ボタンをクリックします。



画面上部の「ツール」アイコンをクリックします。



「左スティック」「右スティック」「Aボタン」を追加します。

三つを一度に追加することはできないので、一つずつ選択して、追加していきます。



オブジェクトリストにこれらのオブジェクトが追加されます。

まず「左スティック」に、ドンポンプが歩く/走るプログラムを作ります。オブジェクトリストで「左スティック」をクリックします。

(注) プログラムは、「ドンポンプ」にも、「左スティック」にも、それ以外のオブジェクトにも作ることができます。後からわかりやすいように作っていきましょう。ここでは、「左スティック」に作成します。



「左スティック」に、左図のようなプログラムを作ります。

「スタートボタンが押されたら」ブロック、「コントローラ(WASD)で操作」ブロックは「イベント」カテゴリにあります。

- ・ 操作の対象は「ドンポンプ」です。
- ・ コントローラの番号「1」はそのままにします。この数字は、外部ゲームコントローラを2つ以上接続した場合に、どちらのコントローラを操作するプログラムかを区別するために使用します。

- ・ 移動速度は、ドンポンプが歩く/走る速度です。数字を変えて、良いと思う値に調整してください。
- ・ 回転速度は、ドンポンプが向きを変える速度です。同じく調整してください。



プログラムを実行してみましょう。



左スティック (L) をマウスや (タッチパネルの場合は) 指で動かすか、キーボードの WASD キを押すと、ドンポンプが歩き/走ります。お使いの PC 等に外部ゲームコントローラーを接続すると、ゲームコントローラーの左スティックで動かすことができます。

## カメラを操作するプログラムをつくる

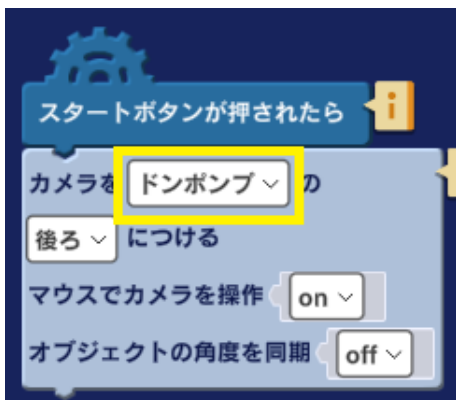
ドンポンプを動かしていると、ドンポンプが消えてしまうことがあります。Mind Render では、ゲームの世界を“カメラ”を通して見えています。今はカメラが固定されて動かないようになっているため、ドンポンプがフレーム外に移動すると見えなくなってしまう。そこで、カメラが動くようにしてドンポンプが消えないようにしましょう。



実行を停止して、プログラム編集画面に戻ります。



「右スティック」に左図のようなカメラのプログラムを作成します。



「スタートボタンが押されたら」ブロックは「イベント」カテゴリにあります。

「カメラ操作」ブロックは「カメラ」カテゴリにあります。

- ・ カメラを取り付ける対象は「ドンポンプ」です。
- ・ その他の選択項目は、初期設定のままにします。



プログラムを実行してみましょう。



左スティック（または WASD キーやコントローラ）でドンポンプを操作しながら、右スティックをマウスや（タッチパネルの場合は）指でカメラを動かします。

左スティックと同様に、外部ゲームコントローラーを接続して動かすこともできます。

## モーションのプログラムをつくる

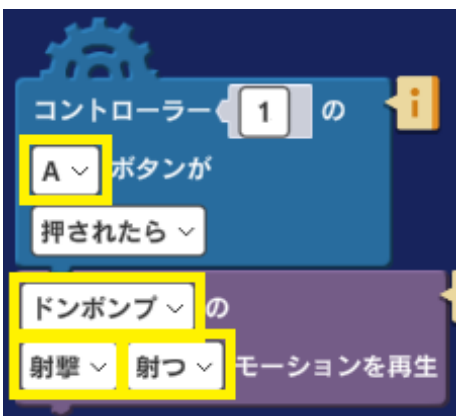
最後に、撃つモーションを追加しましょう。



実行を停止して、プログラム編集画面に戻ります。



「A ボタン」を押したら撃つようにしたいので、ここでは、「A ボタン」にプログラムを作成します。



左図のようなプログラムを作ります。

「コントローラーのボタンが押されたら」ブロックは「イベント」カテゴリにあります。

- ・ コントローラーの番号は「左スティック」のプログラムと同様に「1」のままにします。
- ・ 「A」ボタンを押したら打ちたいので、ボタンは「A」を選択します。

「モーションを再生」ブロックは「効果」カテゴリにあります。

- ・ モーションをするのは「ドンポンプ」です。
- ・ モーションのカテゴリは「射撃」を選択します。
- ・ 「射つ」モーションを選択します。



プログラムを実行してみましょう。



左スティックでドンポンを、右スティックでカメラを操作しながら A ボタンを押すと、マシンガンを撃つモーションが再生されます。

## 2. 敵をつくる

次に、敵を作りましょう。



### <この章の内容>

概要	学ぶこと
敵にするオブジェクトを追加し、プレイヤーに向かってくるプログラムを作ります。	<ul style="list-style-type: none"><li>・オブジェクトの追加</li><li>・オブジェクトの位置（座標）</li><li>・オブジェクトのコピー</li><li>・オブジェクトを動かすプログラム</li></ul>

### 敵にするオブジェクトを追加する

敵にするオブジェクトを追加しましょう。



画面右下の「オブジェクト追加」ボタンをクリックします。



画面上部の「モデル」アイコンをクリックします。



敵にするオブジェクト（ライオン）を選択します。（画面を下の方にスクロールしてください。）



「オブジェクトリスト」に「ライオン」が追加されます。



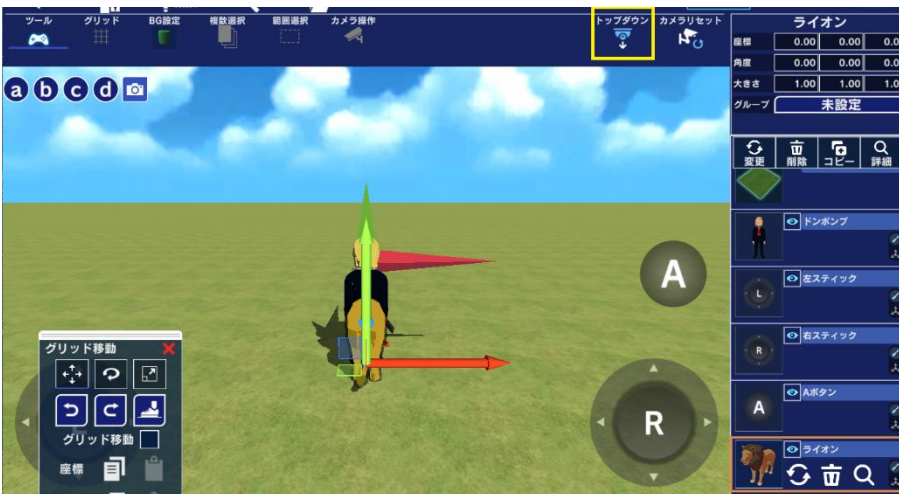
シーン編集画面で確認してみましょう。



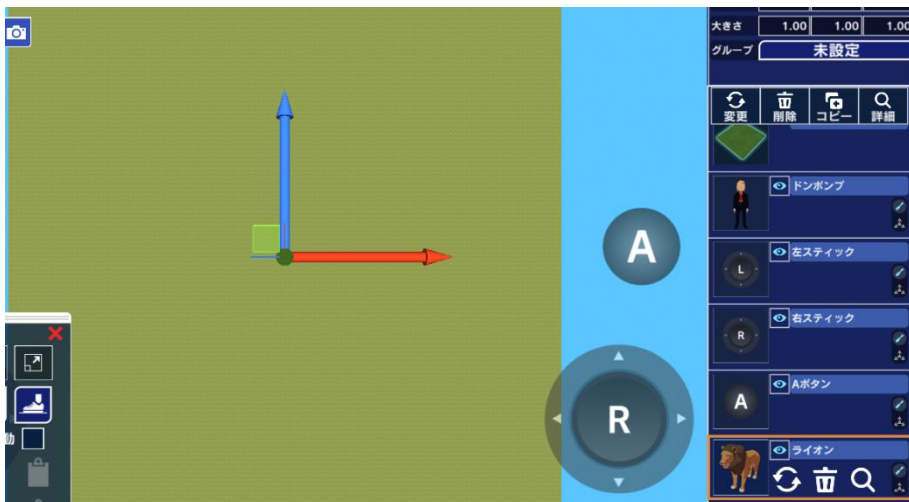
オブジェクトを追加すると、基本的に座標の“原点”に表示されます。このため、先に追加したドンポンプとライオンが重なってしまっています。

このままではおかしいので、ライオンをドンポンプから離れたところに移動します。

## オブジェクトの位置を変更する



オブジェクトリストで「ライオン」が選択されている状態にし、「トップダウン」ボタンをクリックします。カメラ位置が変わり、草原を真上から見下ろします。



矢印は「ライオン」を示しています。赤の矢印、青の矢印をドラッグして好きな位置に移動します。



黄色い円がライオンのいる場所、赤い円がドンポンプのいる場所(原点)です。「カメラリセット」ボタンで、最初のカメラ位置に戻ります。

ライオン			
座標	-20.01	0.00	59.74
角度	0.00	0.00	0.00
大きさ	1.50	1.50	1.50

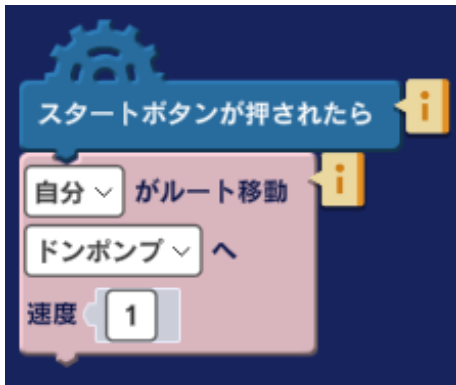
また、ライオンが小さいと弾が当たりにくいので、サイズも大きくしておきましょう。

## 敵がプレイヤー向かってくるプログラムをつくる

ライオンがドンポンプに向かってくるプログラムを作りましょう。



画面右上の「鉛筆」アイコンをクリックし、プログラム編集画面に移動します。



オブジェクトリストで「ライオン」を選択します。

左図のようなプログラムを作ります。

「スタートボタンが押されたら」ブロックは「イベント」カテゴリにあります。

「ルート移動」ブロックは「移動」カテゴリ（イベントカテゴリの下のピンク色の枠）にあります。

- ・ 移動するのはライオンです。ライオンのところにプログラムを作

っているの、「自分」（＝「ライオン」）です。

- ・ 移動先は「ドンボンブ」です。
- ・ 速度は、ライオンが走る速度です。数字を変えて、良いと思う値に調整してください。

ルート移動のブロックを使うと、「自分（ライオン）」が「ドンボンブ」の位置を自動的に認識し、速度「1」でその場所に向かって動くようになります。



プログラムを実行してライオンがドンボンブに向かって走っていくことを確認しましょう。

## ライオンをグループに入れる

ライオンを“グループ”に入れます。

### グループについて

“グループ”とは、複数のオブジェクトを一つにまとめて扱えるようにする機能です。弾が当たったらライオンが倒れるようにしたい場合、“グループ”を使わないと、各ライオンごとにプログラムを作る必要があります。

（ライオン A に当たったら倒れる、ライオン B に当たったら倒れる、・・・）。“グループ”を使うと、そのグループ（に属するライオン）に弾が当たったらライオンが倒れるというプログラムを一つ作るだけで済みます。敵のライオンは複数用意したいので、便利な“グループ”機能を使ってみましょう。

変数

拡張

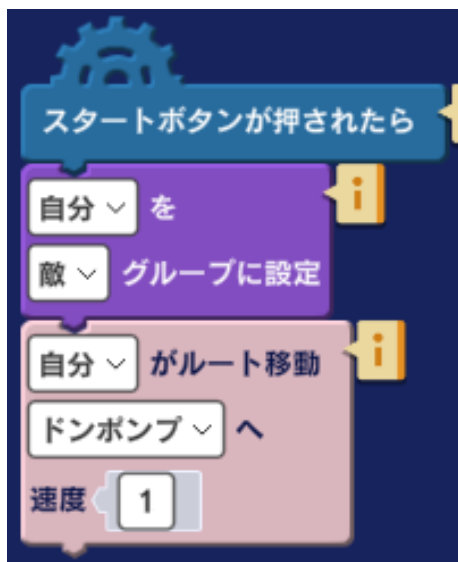
最初にグループを作成します。

プログラム編集画面に戻り、画面左下の「拡張」ボタンをクリックします。



「拡張設定」画面の「グループ設定」欄で、グループ名（ここでは、「敵」）を入力し、「追加」ボタンをクリックします。

「閉じる」ボタンで画面を閉じます。



「ライオン」のプログラムに、「テクニック」カテゴリーから「グループに設定」ブロックを追加します。

これで、このライオンが敵グループに入りました。

## オブジェクトをコピーする

ライオンをあと4頭作ります。



オブジェクトやプログラムはコピーすることができるので、コピーして手間を省きましょう。

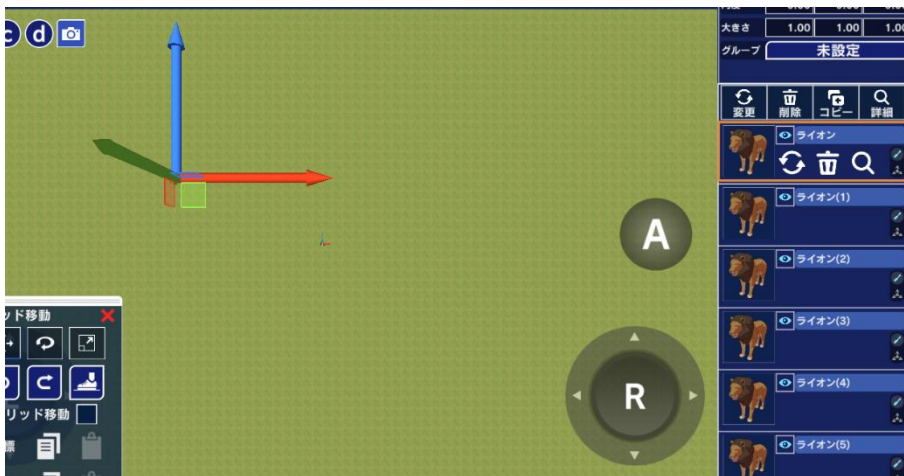
オブジェクトリストで「ライオン」を選択し、リストの上部にあるメニューで「コピー」を4回クリックします。



コピー元のライオンを含めて 5 頭になりました。

オブジェクトをコピーすると、座標などオブジェクトの設定や、オブジェクトに作ったプログラムもコピーされます。

このため、現在、5 頭のライオンは全てコピー元のライオンと同じ位置にいます。



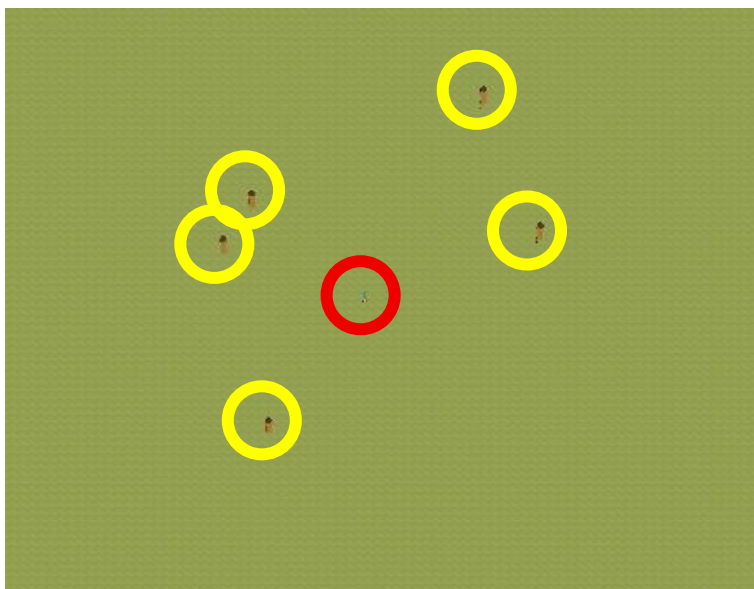
そこで、ライオンが散らばるように位置を変更します。

シーン編集画面に移動し、「トップダウン」ボタンをクリックして、シーンを真上から見下ろす形にします。

オブジェクトリストで「ライオン(1)」を選択し、赤または青の矢印をつかんで、お好みの位置に移動

します。

同様に他のライオンの位置を移動して散らします。

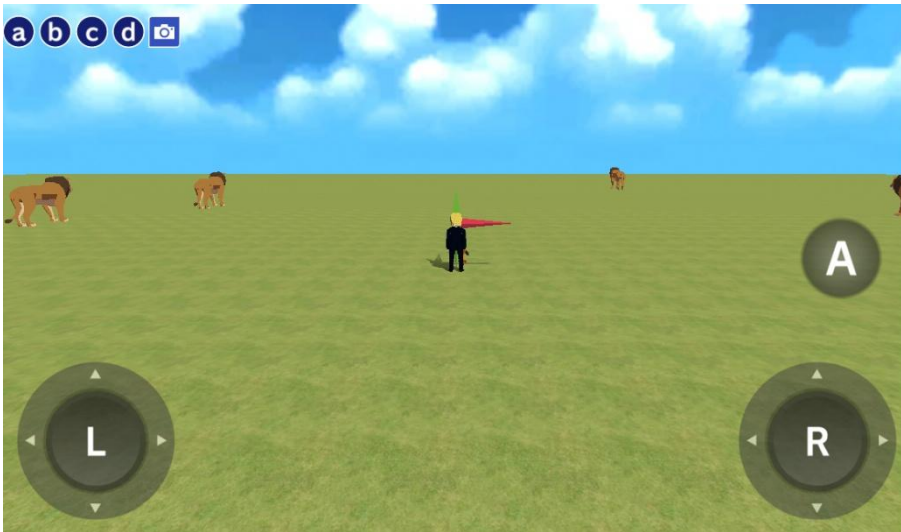


左図では、黄色い円がライオン、赤い円がドンポンプの場所です。

カメラリセット



配置が終わったら、「カメラリセット」ボタンを押して、元のカメラ位置に戻します。



プログラムを実行して、ライオンがドンポンプの方に向かってくるか確認しましょう。

LスティックやRスティックでドンポンプやカメラも操作してみましょう。

## 座標について

Mind Render では、位置や方向は (X、Y、Z) の3つの座標で表します。

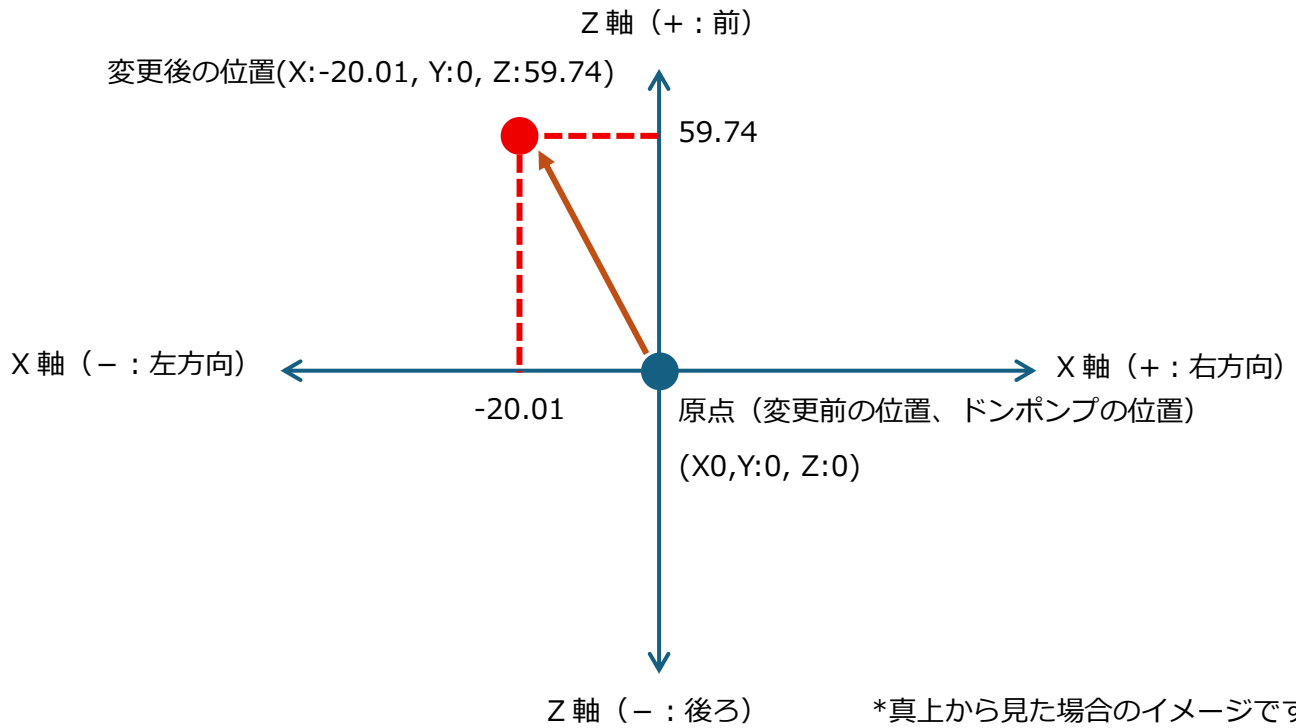
ライオン			
座標	-20.01	0.00	59.74
角度	0.00	0.00	0.00
大きさ	1.50	1.50	1.50

ライオンの位置を動かすと、シーン編集画面右上に表示されている「座標」の数字も変化します。

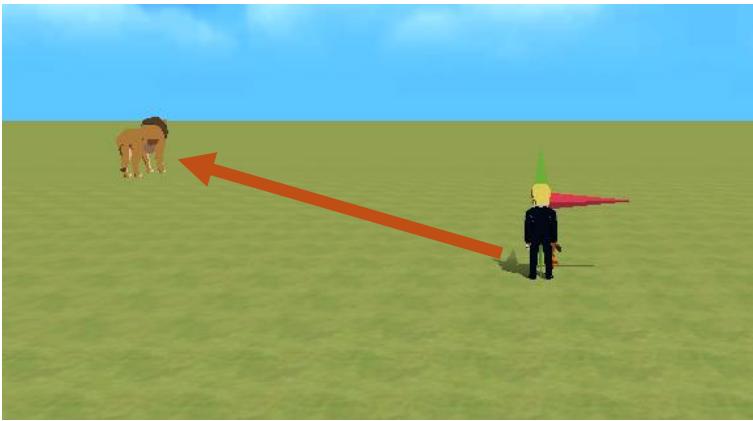
この数字はライオンの位置を表しており、左からX座標、Y座標、Z座標の値になっています。

この例では、ライオンが (X : -20.01、Z : 59.74) にいます。ライオンは地面に接して立っているためYは0になっています。

XとYの位置を真上から見ると、下図のようになります。



- X (X軸) : 左右。左方向はマイナス (-) で表す
- Y (Y軸) : 上下。下方向はマイナス (-) で表す
- Z (Z軸) : 前後。後方はマイナス (-) で表す



### 3. 弾を発射する

プレイヤーと敵ができたので、いよいよ弾を発射するプログラムを作りましょう。



#### <この章の内容>

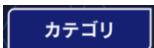
概要	学ぶこと
弾を作って飛ばすプログラムや、弾が背景や敵にヒットした時のプログラムを作ります。	<ul style="list-style-type: none"><li>・重力の設定</li><li>・分身機能</li><li>・オブジェクトを飛ばすプログラム</li><li>・当たり判定</li><li>・オブジェクトの削除</li></ul>

#### 弾にするオブジェクトを追加する

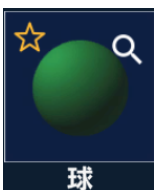
最初に弾となるオブジェクトを追加します。



画面右下の「オブジェクト追加」ボタンをクリックします。



左上の「カテゴリ」ボタンをクリックし、表示される画面で「素材」をクリックします。

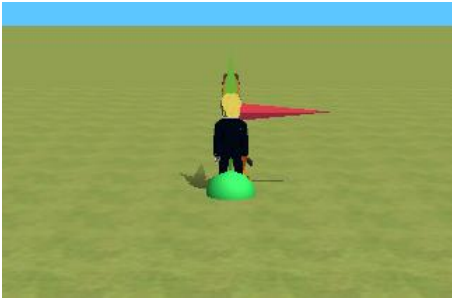


素材の一覧から「球」をクリックします。



オブジェクトリストの一番下に「球」が追加されます。

「球」の文字をクリックして、「弾」に変更します。



シーン編集画面で弾を確認すると、左図のようになっています。



弾が大きすぎるので、小さくしたいです。また、もう一つ設定を行いたいのので、「オブジェクト詳細」画面に移動します。オブジェクトリストで「虫眼鏡」アイコンをクリックします。



「オブジェクト詳細」画面が開きます。

画面右上の「大きさ」の数字を (0.50, 0.50, 0.50) に変更します。

\*座標、角度、大きさの変更は、シーン編集画面でも行えます。

「重力の影響」を「ON」にします。

「OFF」の場合、弾は重力の影響を受けないので、地面に落ちることなく直進していきます。地面に落ちるようにしたいので、重力の影響を受けるようにします。

Mind Render のオブジェクトには、重力の他にも、重さ、はね返り、摩擦、抵抗など様々な物理設定を行うことができます。

## 弾の分身をつくる

弾はたくさん必要ですが、上記のようにオブジェクトを1つずつ追加するのは大変です。このような時は“分身”機能を使うと便利です。

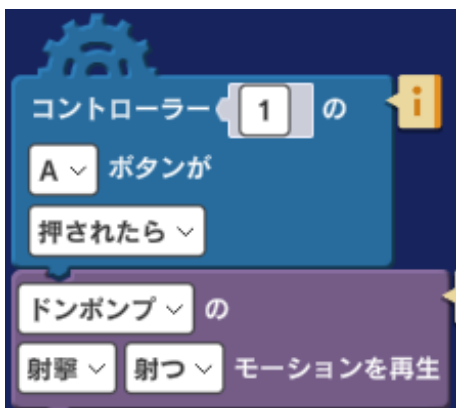
### 分身について

“分身”とは、オリジナルのオブジェクトからコピー（クローン）を作る機能です。プログラム実行中に指定のタイミングで指定の位置に作成でき、作成された分身にプログラムすることができます。

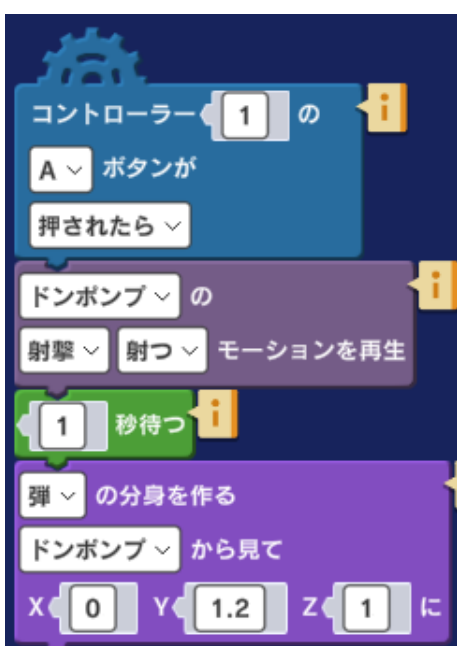


「A ボタン」を押したら弾（弾の分身）が発射されるようにしてみましよう。

「A ボタン」にプログラムを追加します。



「1.プレイヤーをつくる」で、A ボタンには左図のようなプログラムを作りました。



ここに、「分身を作る」ブロックを追加します。

「分身を作る」ブロックは「テクニク」カテゴリにあります。

- ・ 「弾」の分身を作ります。

弾の分身をどこに作りたいかを設定します。弾が銃から発射されるように見せたいので、「ドンポンプ」の構える銃の先端当たりの位置を指定しましょう。「[ドンポンプ]から見て」を選択すると、ドンポンプの体の中心を原点 (0, 0, 0) として位置を指定することができます。

左図では、

- ・ X:0.00（ドンポンプの体の左右中心）
- ・ Y:1.2（地面からの高さ。銃を構えた時の銃身の高さくらい）
- ・ Z:0.50（ドンポンプの前方。銃を構えた時の銃の先くらい）

となります。

「1秒待つ」ブロックは、タイミングを調整するために入っています。銃を撃つモーションの再生が終わるまでに1秒くらいかかるため、モーションの途中で分身が作成され、弾が発射されてしまうのを防ぐために入れます。

「●秒待つ」ブロックは「if」カテゴリにあります。



分身を作っただけでは、弾のように飛んでいきません。分身を動かすためのプログラムを作ります。



「弾」オブジェクトに、左図のようなプログラムを作ります。  
弾の分身が作成されたら、その分身に後ろから押して飛ばします。

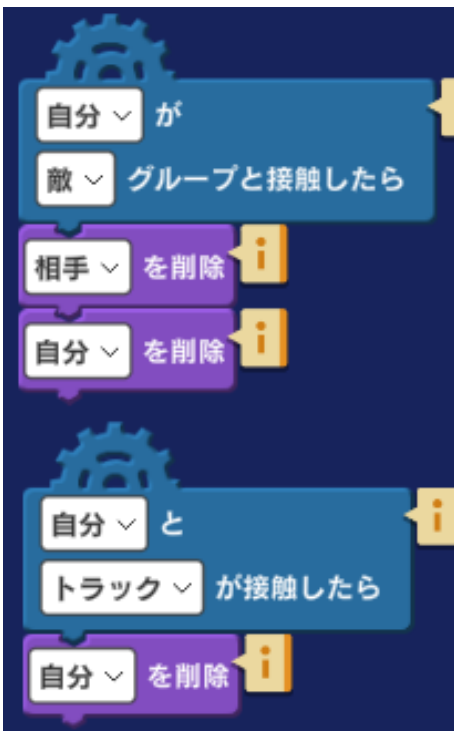
プログラムを実行して、好みの距離に飛ぶように

## 弾がライオンに当たった時、外れた時のプログラムをつくる

弾がライオンに当たった時と、外れた時のプログラムを作りましょう。



「弾」オブジェクトに、左図のようなプログラムを作ります。



上が当たった時、下が外れた時です。

「グループと接触したら」ブロック、「～と接触したら」ブロックは「イベント」カテゴリーにあります。

「～を削除」は「テクニック」カテゴリーにあります。

<当たった時>

弾がライオンに当たると、弾もライオンも画面から消えるプログラムです。

弾の分身（自分）が、敵グループ（のライオン）に当たったら、相手（のライオン）を削除し、続いて、自分自身も削除します。

「相手」とは、敵グループに属するライオンを指します。グループ内のどのライオンでも、弾があたったライオンが「相手」となります。

<外れた時>

弾が外れたら、弾を消します。

弾の分身（自分）が外れ、背景（トラック）に落ちたら、自分を削除します。

## オリジナルの弾を非表示にする



最後に、オリジナルの弾が原点に表示されているため、非表示にしておきます。

## 当たり判定

Mind Render では、オブジェクトとオブジェクトが接触したことを検知し（当たり判定）、それをきっかけにプログラムを実行することができます。オブジェクトやグループ以外にも、サーチエリア/ヒットエリアや装備などとの接触を利用することができます。「イベント」カテゴリには、当たり判定を利用するいろいろなブロックがあります。

## 4. おさらい

ここまで、プレイヤーや敵を動かすプログラムや、プレイヤーが弾を撃つプログラムを作ってきました。今まで作ったものを実行して動作を確認してみましょう。



## 5. 制限を加える

第4章で作成した弾は制限なしで撃つことができました。ゲーム性を高めるため、弾数の上限を設定します。また弾数とライオンの残りの頭数がわかるように表示しましょう。



### <この章の内容>

概要	学ぶこと
<ul style="list-style-type: none"><li>・弾数の上限を設定します。</li><li>・弾数やライオンの数を画面に表示します。</li></ul>	<ul style="list-style-type: none"><li>・変数の設定</li><li>・変数の初期値の設定</li><li>・弾やライオンの数をカウントする（変数の値を減らす）</li><li>・テキスト表示のブロックとプログラム</li></ul>

### 変数を設定する

#### 変数について

弾数をカウントするために、“変数”を使用します。

“変数”とは、データを入れておく箱のようなものです。箱には名前を付け、数値や文字を入れたり取り出したりすることができます。

## 変数

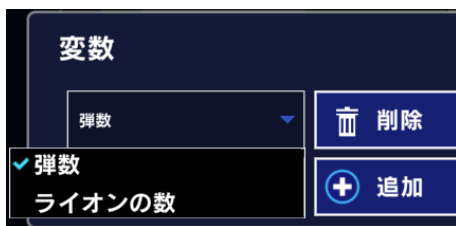
## 拡張

変数を作成するには、プログラム編集画面左下の「変数」ボタンをクリックします。



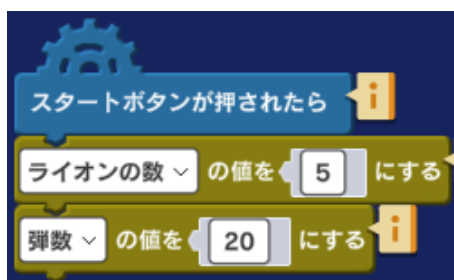
変数の名前を入力します。ここでは、「弾数」とし、「追加」ボタンをクリックします。

続けて、「ライオンの数」と入力し、「追加」ボタンをクリックします。



2つの変数が追加されたら。右上の「閉じる」ボタンで画面を閉じます。

プログラム編集画面に戻ります。



オブジェクトリストで「トラック」を選択し、変数の初期値を指定するプログラムを作ります。

プログラムが実行されるごとに、変数の値が初期化され、「ライオンの数」は5、「弾数」は20に設定されます。

「スタートボタンが押されたら」は「イベント」カテゴリにあります。

「～の値を～にする」は「X/Y」カテゴリにあります。

## 弾数をカウントする

弾が発射されたら変数「弾数」の値が減っていくようにします。



オブジェクトリストで「弾」を選択します。



すでに、左図のプログラムが作られています。



弾はライオンを外れた時も（地面に当たったとき）、ラインに当たった時も弾は1つずつ減っていくので、左図のようにブロックを追加します。

「変数-1」ブロックは「X/Y」カテゴリにあります。

変数「弾数」は初期値 20 からスタートし、19、18、17・・・と減っていきます。

## 弾数を表示する

次に、弾数を表示するプログラムを作成します。

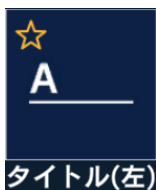


画面右下の「オブジェクト追加」ボタンをクリックします。



ツール

画面上部の「ツール」アイコンをクリックします。



「タイトル(左)」というアイコンをクリックします。

タイトル(左)



オブジェクトリストの一番下に、「タイトル(左)」が追加されます。



このままでは、何を表示するオブジェクトかがわかりにくいので、「弾数表示」に変更します。

ここに弾の数を表示するプログラムを作成します。



「～が変更されたら」ブロックは「イベント」カテゴリにあります。変数の値が変化したときに、プログラムが実行されるようにするブロックです。

「～のテキスト設定」ブロックは「せりふ」カテゴリにあります。このブロックは、テキストの字体、サイズ、色を指定するためのブロックです。一度設定すると他でも同じ設定が適用されます。

① 「弾数表示」オブジェクトで設定しているため、[自分]を選択します。

② [システム][セリフ]の2つの選択肢があります。システムは通常の字体、セリフはや

や丸みを帯びた自體です。お好みの字体を選択します。ここでは[セリフ]を選択します。

③ 文字の大きさを設定します。

④ 色の部分をクリックすると、カラーパレットが表示されます。ここでは、文字の色を白、ふちを黒にします。

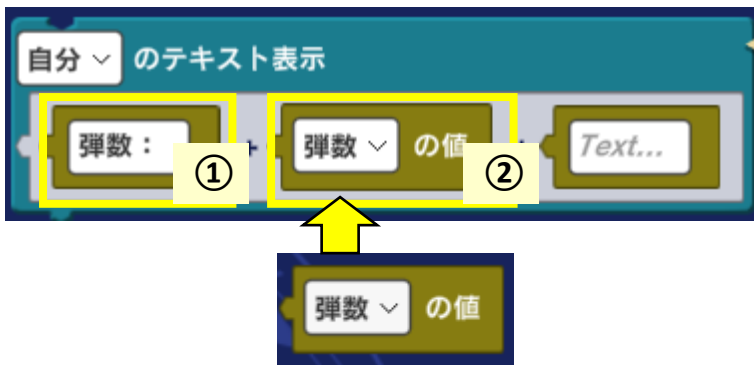


「～のテキスト表示」ブロックは「せりふ」カテゴリにあります。このブロックで実際に表示する内容（文字）を指定します。



このブロックは、元は左図のような形をしています。黄色の枠の部分に、文字を入力したり、他のブロックを埋め込んで使います。

ここでは、「演算」カテゴリの左図のブロックを埋め込みます。



「弾数：20」などと表示されるようにします。

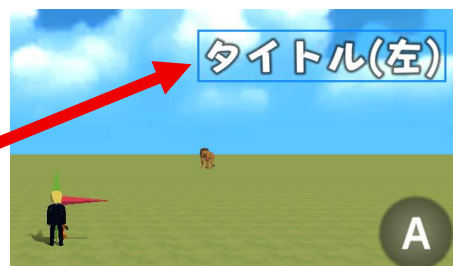
- ① 項目名です。キーボードで「弾数：」と入力します。
- ② 弾を撃つに従って数字が減っていくように、変数（「弾数」）を埋め込みます。「～の値」は「X/Y」カテゴリにあります。



プログラムを実行して確認してみましょう。弾数の表示が中央にあって邪魔なので、位置を調整します。



シーン編集画面に移動します。



「タイトル（左）」をクリックすると、青い枠が表示されます。この枠をつかんで、適当な位置に移動します。



「タイトル(左)」という表記はわかりにくいので、画面上の名前を変えておきます。

「タイトル(左)」を長押しすると、入力エリアが変更されます。「せりふ」ブロックで入力したのと同じように「弾数：」と入力し、「決定」ボタンをクリックします。

## ライオンの数をカウントする

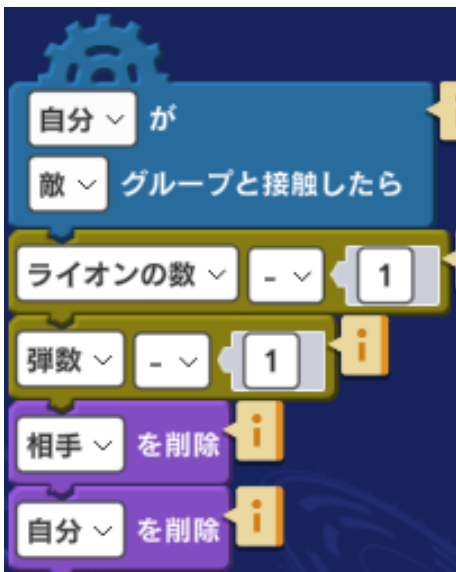
続いて、弾が当たったら変数「ライオンの数」の値が減っていくようにしましょう。

引き続き、「弾」にプログラムを作成します。



すでに、左図のようなプログラムが作られています。

ここに、弾がライオンに当たったらライオンの数を1つずつ減らしていくブロックを追加します。



「変数-1」ブロックは「X/Y」カテゴリにあります。

自分（弾）が敵グループ（ライオン）に当たったら、ライオンの数を1減らします。

変数「ライオンの数」の値は初期値5からスタートし、4、3・・・と減っていきます。

## ライオンの数を表示する

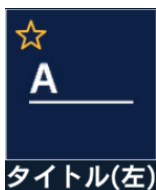
ライオンの数を表示するプログラムを作成しましょう。



画面右下の「オブジェクト追加」ボタンをクリックします。



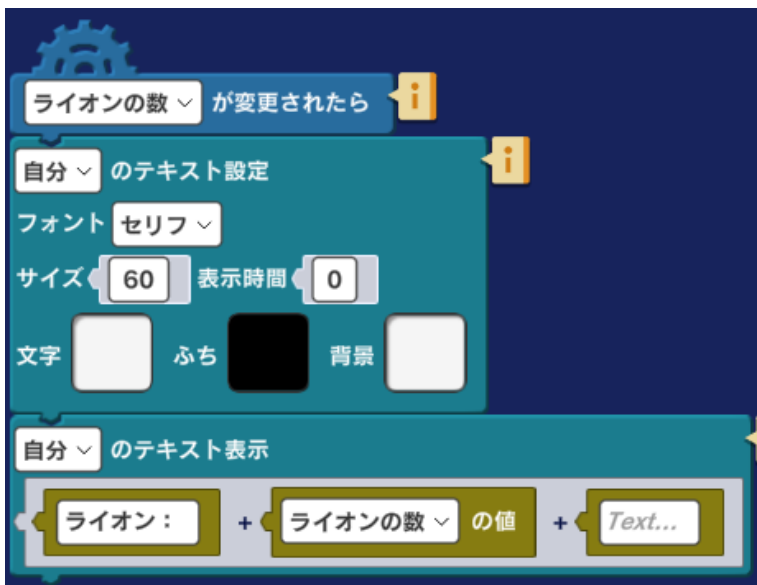
画面上部の「ツール」アイコンをクリックします。



「タイトル(左)」というアイコンをクリックします。



わかりやすくするため、オブジェクトリストで、「タイトル(左)」を「ライオンの数表示」に変更します。



ここに、ライオン弾の数を表示するプログラムを作成します。

弾数を表示するプログラムと同様なので、先に作ったプログラムを参考に作ってみましょう。



プログラムができたなら、実行して確認してみましょう。

ライオンの数の表示が中央にあって邪魔なので、位置を調整します。

弾数の場合と同様なので、弾数の手順を参考に調整してみましょう。



実行して、位置を確認しましょう。

## 6. ゲームオーバーとゲームクリア

ゲームオーバーとゲームクリアの条件を作りましょう。



### <この章の内容>

概要	学ぶこと
ゲームオーバーとゲームクリアを作ります。	<ul style="list-style-type: none"><li>・2D オブジェクトの追加</li><li>・条件分岐のブロックとプログラム</li><li>・変数関連のブロックとプログラム</li></ul>

### ゲームオーバーをつくる

始めに、ゲームオーバーとゲームクリアを作るのに必要なオブジェクトを追加します。



画面右下の「オブジェクト追加」ボタンをクリックします。



モデル

画面上部で「モデル」が選択されていることを確認します。



カテゴリ

クリックします。

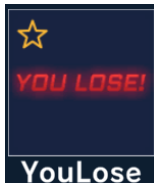


オブジェクト「プログラム」を選択します。

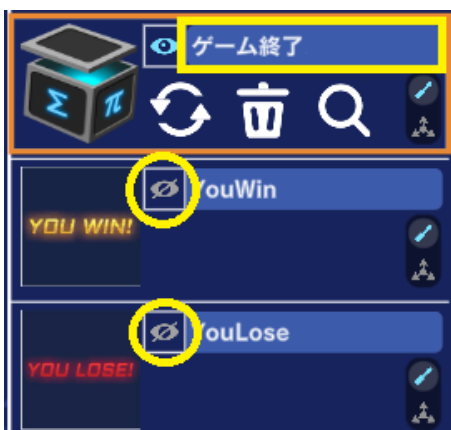
続いて、ゲームオーバーとゲームクリアの表示を追加します。



「オブジェクト追加」ボタンをクリックし、表示される画面で「ツール」アイコンをクリックします。



「YOU WIN!」を選択し、追加します。まとめて選択はできないので、もう一度手順を繰り返して、「YOU LOSE!」を追加します。



オブジェクトリストに3つのオブジェクトが追加されました。

「プログラム」は、画面には表示されないオブジェクトです。プログラムをまとめて置く時などに使用すると便利です。わかりやすくするために、「プログラム」の名称を「ゲーム終了」に変更します。

「YOU WIN!」、「YOU LOSE!」は、それぞれ「目」のアイコンをクリックして表示オフにします。「目」のアイコンに青い色が付いている場合は、プログラムスタート時から表示されます。グレーにするとプログラムスタート時には表示されず、実行中の指定したタイミングで表示する

ことができます。

「YOU WIN!」、「YOU LOSE!」の文字は、ゲームの結果が出た時に表示したいので、表示オフにします。

## ゲームオーバーとゲームクリアをつくる

ゲームクリアの条件は、ライオンが0頭になっていることです。このとき、弾数は残っていても0個でも構いません。

逆に、ゲームオーバーとなるのは、ライオンがまだいるのに、弾数が0になっている時です。

弾もライオンも残っている場合は、ゲーム続行です。

この条件を作りましょう。



「ゲーム終了」オブジェクトに左図のようなプログラムを作成します。上から順番に見ていきましょう。

変数「弾数」が変更されたら、弾数をチェックします。

弾数が0でも、ライオンが0ならゲームクリアです。「You Win」を表示します。

しかし、ライオンが0でなかったら（ブロックでは「違う場合は」）、それ以上ゲームを続けられないので、ゲームオーバーです。「You Lose」を表示します。

いずれの場合も、Aボタン、左スティック、右スティックの表示をオフにして画面から消します。

後半の「違う場合は」は、弾が残っている場合です。ライオンが0ならゲームクリアなので「You Win」を表示します。

弾が残っていて、ライオンも残っている（上記の条件に当てはまらない）場合はゲーム続行です。

「変数が変更されたら」ブロックは「イベント」カテゴリにあります。

「もし～なら実行する 違う場合」、「もし～な

ら実行する」は「if」カテゴリにあります。似たブロックがあるので、気をつけましょう。

“もし～なら”の条件となっている「変数=0」ブロックは「演算」カテゴリにあります。

「～の表示 ON/OFF」ブロックは「テクニック」カテゴリにあります。

## 7. 演出をする

今まで作ってきたプログラムに、ゲームらしい演出を加えましょう。

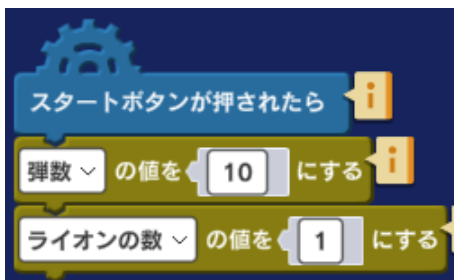
BGM、弾の発射音、弾が当たったときのエフェクトを加えます。



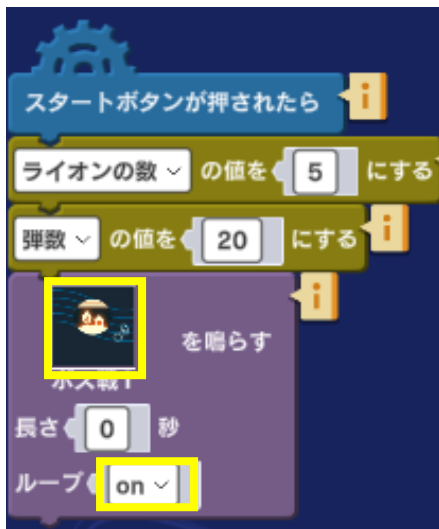
概要	学ぶこと
ゲームの演出として、BGM、効果音、エフェクトを加えます。	<ul style="list-style-type: none"><li>・サウンドの追加</li><li>・エフェクトの追加</li></ul>

### BGM を追加する

BGM を追加しましょう。



「トラック」に左図のプログラムを作りました。ここに追加します。



「～を鳴らす」ブロックは「効果」カテゴリにあります。

ブロックの中のアイコンをクリックすると、サウンドの一覧が表示されます。



「▶ (再生)」ボタンをクリックすると音が再生されます。選択するときは、画像の部分をクリックします。

「ループ」を「on」にすると、ゲームの間、繰り返し流れます。

## エフェクトと効果音を追加する



弾の発射音や、弾が当たったときのエフェクトを追加します。

「弾」に左図のようなプログラムを作りました。



これに、効果音、エフェクトをそれぞれ追加します。

「～を鳴らす」ブロック、「～を表示」ブロックは「効果」カテゴリにあります。

BGMと同様、アイコンをクリックすると、サウンドやエフェクトの一覧が表示されます。



左図は、エフェクトの一覧です。サウンドと同様、「▶ (再生)」ボタンをクリックするとエフェクトが再生されます。選択するときには、画像の部分をクリックします。

## BGM を止める

ゲームが終わったら BGM を止めましょう。



オブジェクトリストで「ゲーム終了」オブジェクトを選択します。

A ボタンや左右スティックを非表示にするにブロックの下に、「音を停止」ブロックを追加します。

2 か所あります。

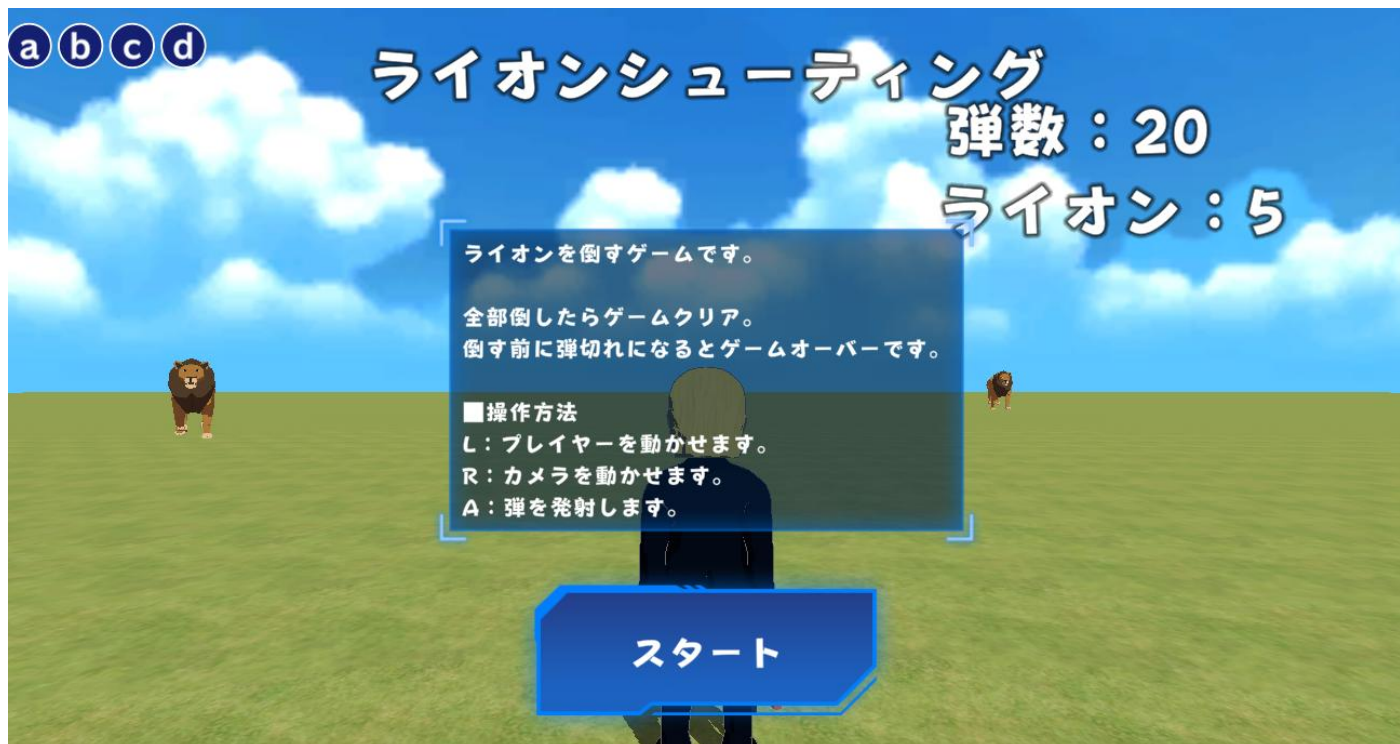
「～の音を停止」ブロックは「効果」カテゴリーにあります。

アイコンをクリックし、BGM と同じ曲 (ここでは「ボス戦 1」) を選択するのを忘れないようにしましょう。

## 8. オープニングをつくる

最後に、ゲームのオープニングを作って完成させましょう。

ゲームのタイトル、ゲームのルールや操作説明を書いておくと、ゲームをプレイする人に親切です。



### <この章の内容>

概要	学ぶこと
タイトル表示、ルール・操作説明を作ります。	・テキストを表示するオブジェクトとプログラム



シーン編集画面に移動します。

### タイトルを追加する

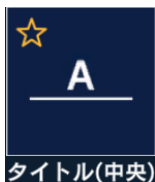
ゲームタイトルを表示します。



「オブジェクト追加」ボタンをクリックします。



「ツール」を選択します。



「タイトル(中央)」をクリックします。



シーン編集画面に移動し、「タイトル(中央)」を長押しします。

入力エリアが表示されるので、ゲームタイトル（ライオンシューティング）を入力し、「決定」ボタンをクリックします。



この変更は、オブジェクトリストのオブジェクト名には反映されません。

オブジェクトリストで「タイトル(中央)」の文字をクリックして、「ゲームタイトル」に変更してください。

## ゲームの説明を追加する

このゲームを遊ぶ人のために、ルールや操作方法を記載します。



「オブジェクト追加」ボタンをクリックします



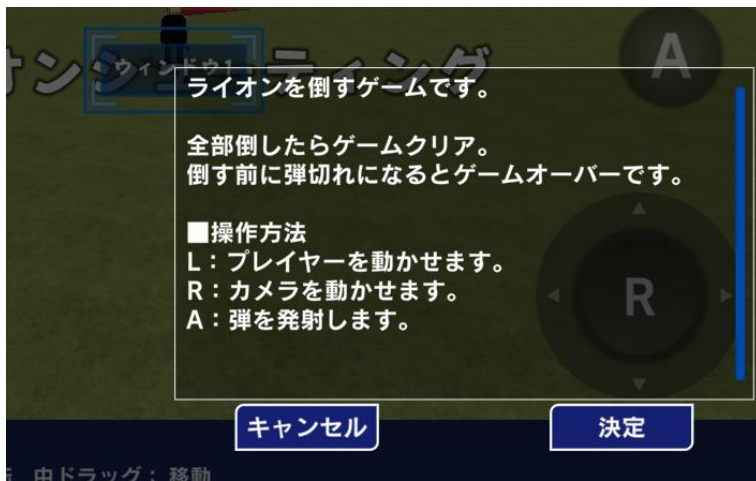
ツール

「ツール」を選択します。



ウィンドウ1

「ウィンドウ1」を選択します。



シーン編集画面で「ウィンドウ 1」を長押しし、入力エリアを表示します。

ゲームのルール、操作方法などを記入し、「決定」ボタンをクリックします。

直接入力するよりも、「メモ帳」などで書いたものをコピー&ペーストすると楽です。

入力が終わったら「決定」ボタンで確定します。



わかりやすくするために、オブジェクトリストの「ウィンドウ 1」の名前を「ゲーム説明」に変更します。

## スタートボタンを追加する

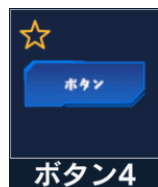
スタートボタンを押したら、タイトルと説明が消えて、ゲームが開始するようにします。



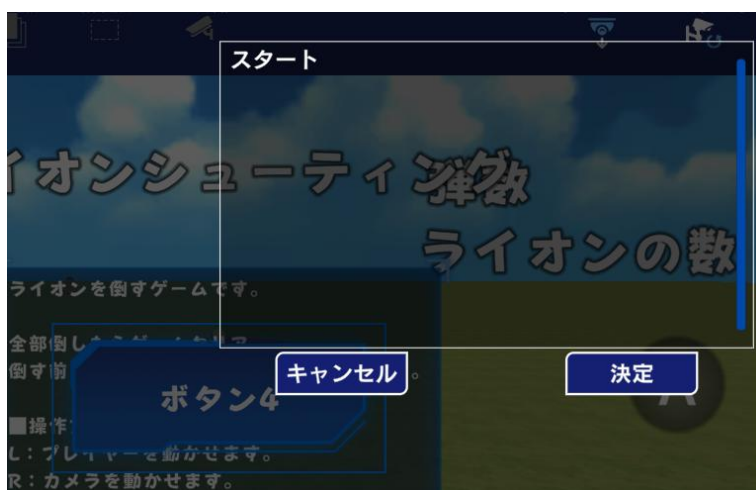
「オブジェクト追加」ボタンをクリックします。



「ツール」を選択します。



「ボタン 4」をクリックします。



シーン編集画面で「ボタン 4」を長押しし、入力エリアを表示します。

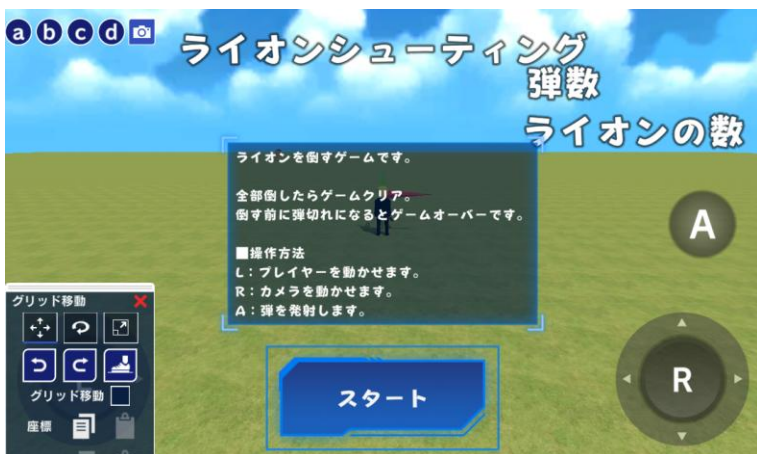
「スタート」と入力し、「決定」ボタンをクリックします。



オブジェクトリストで、ボタン名を「スタート」に変更します。



左図のように、初期状態では、画面中央にタイトル、説明、スタートボタンが重なって表示されます。

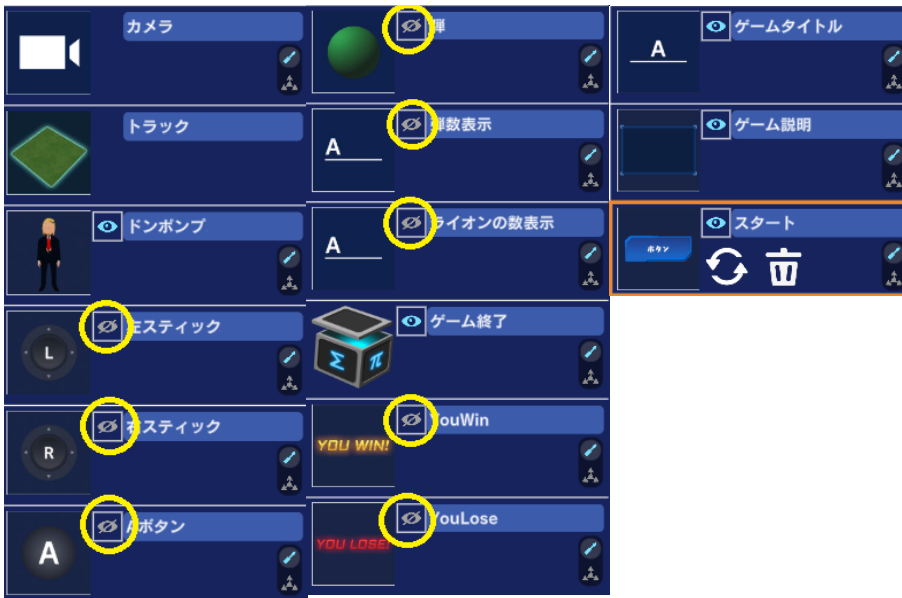


シーン編集画面で各オブジェクトをマウスでクリックすると青い枠が表示されます。クリックしたままドラッグし、適当な位置に移動します。

## オープニングで不要なオブジェクトを非表示にする

タイトルやゲーム説明を表示しているときに、左スティックなどコントロールオブジェクトや、弾数・ライオンの数などは表示されていない方がすっきりします。

これらのオブジェクトは、オブジェクトリストの「目」のアイコンをクリックして、非表示にします。



黄色い円のオブジェクトは非表示にします。

\* 「ゲーム終了」は、表示されないオブジェクトなので、表示オンでもオフでも構いません。

\* 「弾数表示」「ライオンの数表示」は表示オフにしても、プログラムを実行すると表示されます。

ゲームが始まったら表示されるようにプログラムを作ります。



プログラム編集画面に移動します。



トラックにプログラムを作ります。

「トラック」には、すでに左図のようなプログラムがあります。



これを左図のように変更・追加します。

「スタート」ボタンをクリックしたら

- ・BGM を鳴らす
- ・ゲームタイトル、ゲーム説明、スタートボタンを非表示にする
- ・弾数とライオンの数、左スティック、右スティック、A ボタンを表示する

ようにします。

「～が表示」ブロックは「テクニク」

「～がタップされたら」ブロックは「イベント」

「～が表示」ブロックは「テクニク」

これでゲームは完成です！遊んでみましょう。

## 9. 発展課題

ゲームができれば発展課題に挑戦してみましょう。

### (Q1) ライオンの数を増やす

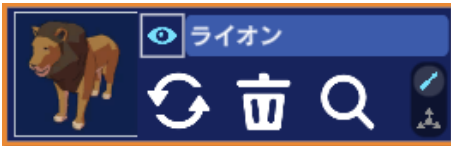
---



ライオンの数を 10 頭に変えるには、どこを変える必要がありますか？

### (Q2) ライオンの数をもっと簡単に増やす

---



ライオンをもっと増やしたい場合、何頭もコピーするのは面倒です。オブジェクトリストもライオンであふれかかってしまいます。

また、変数「ライオンの数」とも連動していません。

ライオンを“分身”機能を使って表示することもできます。「弾」の分身を参考に考えてみましょう。

(答え)

A1 オブジェクトリストでライオンを 5 回コピーして、10 頭にする

「トラック」のプログラム 「ライオンの数の値を「5」にする」→「10」に変更する