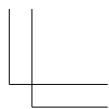
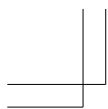


Unity で作るそれっぽいアーケード ゲーム

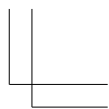
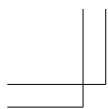
トミー (@Tomy_0331) 著

2019-01-26 版 発行



目次

第 1 章	はじめに	1
1.1	この本について	1
1.2	開発環境	1
第 2 章	Unity でゲーム部分を作ろう	3
2.1	プロジェクトを作成する	3
2.2	地形を作る	3
	Standard Assets のインポート	3
	Terrain の設置	4
2.3	敵 (ゾンビ) の用意	4
	ゾンビのモデルをインポート	4
	ゾンビの死体を作る	5
2.4	ゾンビを撃つスクリプトを書く	6
	プログラム解説	7



第 1 章

はじめに

1.1 この本について

この本では「Unity でサンプルゲームは作ってみたけどそれからどうしたらいいかわからない」「入門レベルより少しクオリティの高いゲームを作りたい」というような Unity 初心者～中級者手前の人をターゲットにしています。

なお、この本では画面の見方や操作など基本的な部分の説明は省略することが多いため、分からない場合適宜調べるか一度入門サイトでミニゲームを作ってみることをおすすめします。

1.2 開発環境

以下の環境・ソフトで開発しました。

Windows10

一般的な OS. Unity は Mac 版でも操作は基本的に同じですが、今回のテーマのアーケード制作において Wii リモコンやペダルの入力を Mac では方法を載せていませんのでご了承ください。Linux も同様です。

Unity2018.3.2f1

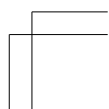
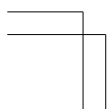
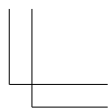
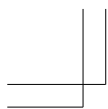
今回のメインで使うゲームエンジン。3D ゲームを作るのに最適で個人からゲーム会社まで広い場所で使われています。

blender

フリーの 3D モデリングソフト。造形からテクスチャのマッピング、アニメーション作成までできます。3D ビューの操作が独特で混乱しがち。

Visual Studio 2017

Unity でスクリプトを書くときに使います。自動で Unity 環境の補完をしてくれます。



第 2 章

Unity でゲーム部分を作ろう

2.1 プロジェクトを作成する

Unity を起動し「New」から新規に作るプロジェクトの設定をします。Project name は「shoot」など任意の名前，Template は「3D」，他パスの設定などして Create project でプロジェクトを作成します。

2.2 地形を作る

ゲームの舞台となる地形を作ります。Unity には山や植物をペイントツールのように作る Terrain という機能があります。

Standard Assets のインポート

Terrain を使う前に，地面に適用するテクスチャや草木のデータを使うために公式の「Standard Assets」をインポートします。

Unity の Asset Store から「Standard Assets」で検索するとパッケージが出てくるのでダウンロードし，インポートします。インポート時に色々選べますが，必要なテクスチャや草木データが入っている「Environment」のみでいいでしょう。

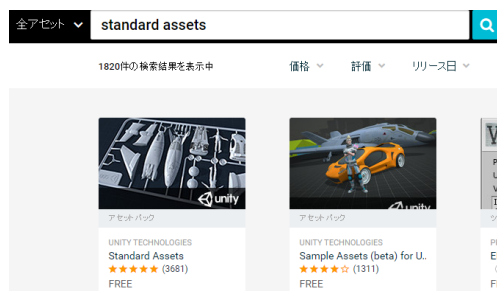


図 2.1: Standard Assets

Terrain の設置

「Hierarchy」の「Create」から「3D Object>Terrain」で作ります。「Inspector」の「Terrain」で設定ができ、「Paint Texture」を選択し「Edit Terrain Layers」から草のテクスチャ画像を指定しましょう。

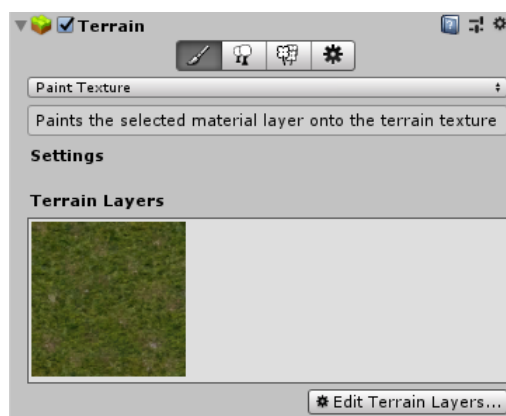


図 2.2: 草テクスチャの指定

2.3 敵 (ゾンビ) の用意

恐らく一番難しい敵キャラの動作を作ります。

ゾンビが今回唯一登場するキャラクターです。アニメーションやラグドール、移動 AI など初心者には難しいところがあるかもしれませんが、ここを作ることができると”ゲーム感”がとてもよく出てそれっぽくなります。

ゾンビのモデルをインポート

Asset Store で「zombie」と検索すると出てくる一番人気のアセットをインポートします。



図 2.3: ゾンビの素材

筆者はこの時なぜかインポートしたファイル名の一つが大文字小文字の区別に関するエラーが出てきたので手動で修正しました。

ゾンビの死体を作る

恐ろしい小見出し名ですが、ゾンビを撃った後に物理法則に従って倒れるようにラグドールを適用します。インポートしたアセットの Prefabs というフォルダにゾンビのプレハブがあるのでシーンに追加し、メニューの「GameObject>3D Object>Ragdoll...」から Create Ragdoll ウィンドウを開きます。

頭、腕、足など指定するためシーンのゾンビの階層を展開してドラッグアンドドロップで一つずつ体のパーツを当てはめていきます。

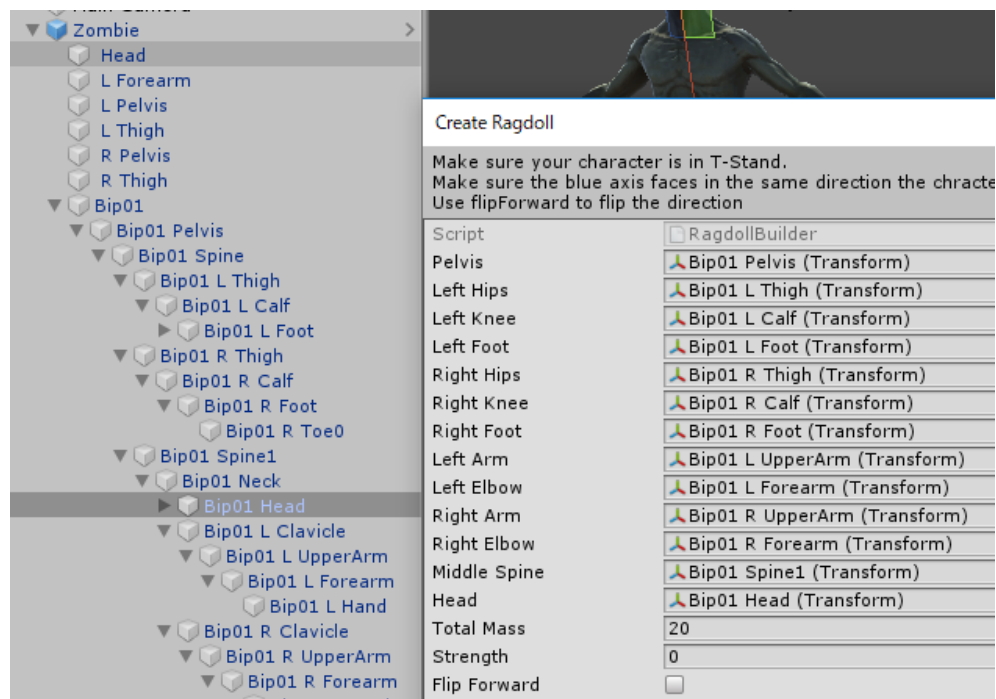


図 2.4: ラグドールの設定

enemy タグ付ける

2.4 ゾンビを撃つスクリプトを書く

ここでゲームの動作部分になるプログラムを作ります。「ShotCam」という名前でスクリプトを新規作成し^{*1}、以下のコードを書きます。

スクリプトは Main Camera にアタッチします。

ShotCam.cs

```
public class ShotCam : MonoBehaviour {
    void Start() {
    }
    void Update() {
        GameObject clickObject=getClickObject();
        //クリックしたのが敵なら
```

^{*1} Unity では C# の他 JavaScript, Boo といったプログラミング言語を使えましたが、廃止となっています。

```
if (clickObject!=null && clickObject.gameObject.tag == "enemy") {
    //アニメーション無効
    clickObject.transform.root.GetComponent<Animator>().enabled = false;
    Vector3 vec = clickObject.transform.position - this.transform.position;
    //射撃した部位に力を加える
    clickObject.GetComponent<Rigidbody>().velocity = vec.normalized*15;
    //ゾンビ側のスクリプトの death() 呼び出し
    clickObject.transform.root.GetComponent<Zombie>().death();
}
}
// 左クリックしたオブジェクトを取得する関数
public GameObject getClickObject() {
    GameObject clickObject = null;
    if (Input.GetMouseButtonDown(0)) { //左クリック
        Ray ray = Camera.main.ScreenPointToRay(Input.mousePosition);
        RaycastHit hit = new RaycastHit();
        if (Physics.SphereCast(ray, 0.1f, out hit)) {
            clickObject = hit.collider.gameObject;
        }
    }
    return clickObject;
}
}
```

続いてゾンビのオブジェクト (一番上の階層) に `Zombie` というスクリプトを作成し以下のクラスを追加しアタッチします。

`Zombie.cs`

```
public void death() {
    Invoke("destroyObject", 5f); //5 秒後に実行
}
void destroyObject() {
    Destroy(gameObject); //オブジェクトを消す
}
```

プログラム解説

まずは `ShotCam` の説明をします。

`getClickObject` 関数は左クリックされると画面上に `Ray` と呼ばれるレーザーのような線を出し、衝突したオブジェクトを取得し `return` で返しています。毎フレーム呼び出される `Update` 関数では `getClickObject` 関数でクリックしたオブジェクトを検知し続け、`if` 文にて `enemy` タグが付いているかを判別します。`enemy` タグはゾンビの胴体、手足、

変数 `vec` は撃った部位に力を加えて吹き飛ばす演出をするためのベクトルを保持しています。撃った部位の座標からゲーム画面を映すカメラの座標を減算することでベクトルが取得できます。そして `normalized` で単位ベクトル化し `Rigidbody` コンポーネントで力を加えています。「*15」は力の大きさです。試しに大きくして遊んだり自分好みの値にしてみるのもいいでしょう。

最後にゾンビ側のスクリプト `Zombie.cs` の関数 `death` を呼び出して一定時間後オブジェクトを消すようにしています.

Unity ってね・・・ポケモン GO にも使われるんだって!!!!!!!!!!!!!!

Unityで作るそれっぽいアーケードゲーム

2019 年 1 月 26 日 初版第 1 刷 発行

著 者 トミー (@Tomy_0331)
