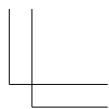
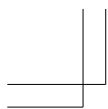


Unity で作るそれっぽいアーケード ゲーム

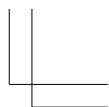
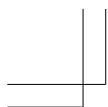
トミー (@Tomy_0331) 著

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
	たのしいコラム	2
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	シューティングのスク립トを書く	5
第 3 章	Re:View のテスト	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。

■コラム: たのしいコラム

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

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択できますが，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。

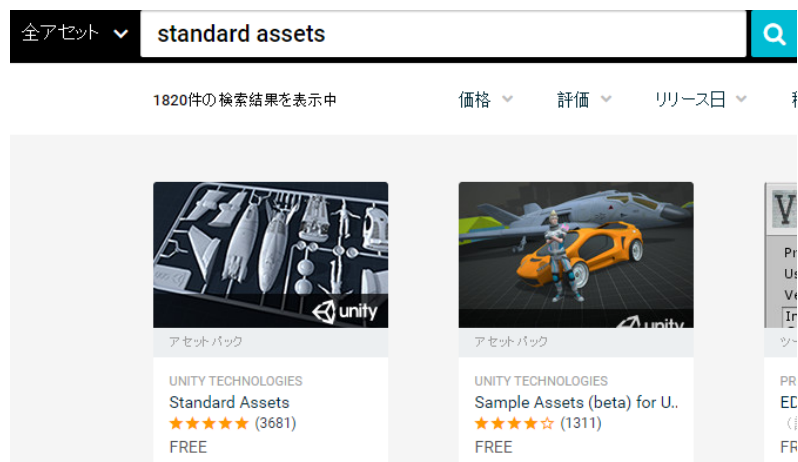


図 2.1: Standard Assets

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。

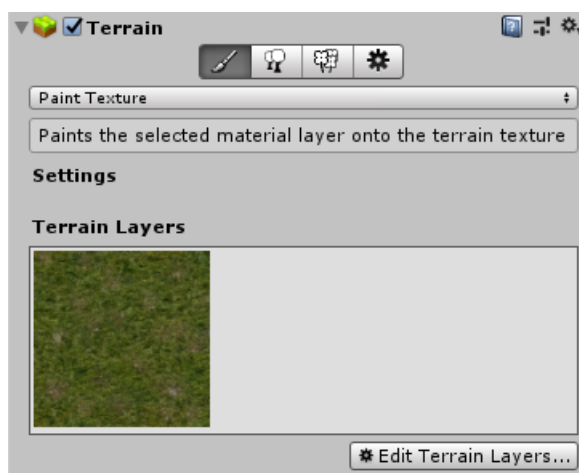


図 2.2: 草テクスチャの指定

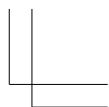
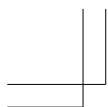
2.3 シューティングのスク립トを書く

それでは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립トを新規作成し^{*1}、以下のコードを書きます。

ShotCam.cs

```
public GameObject getClickObject() {
    GameObject clickObject = null;
    if (Input.GetMouseButtonDown(0)) {
        Ray ray = Camera.main.ScreenPointToRay(Input.mousePosition);
        RaycastHit hit = new RaycastHit();
        if (Physics.SphereCast(ray, 0.1f, out hit)) {
            clickObject = hit.collider.gameObject;
        }
        if (clickObject.gameObject.tag == "enemy") {
            Destroy(clickObject); //オブジェクトの削除
        }
    }
    return clickObject;
}
```

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



第 3 章

Re:View のテスト

ハロワ

```
1: private string str="Hello World";  
2: console.Log(str);
```

ぼくのついったー*1

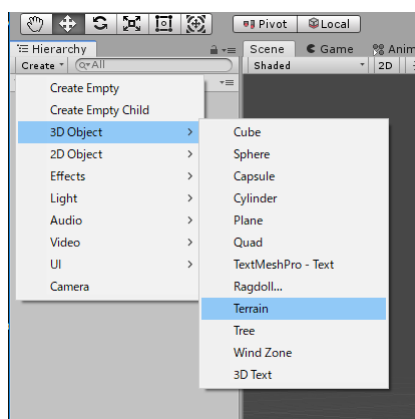
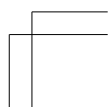
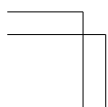
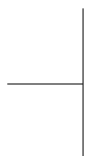
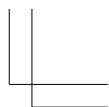
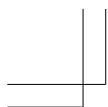


図 3.1: 図の名前

*1 https://twitter.com/Tomy_0331



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)
