

## Unity ちゃんを歩かせる

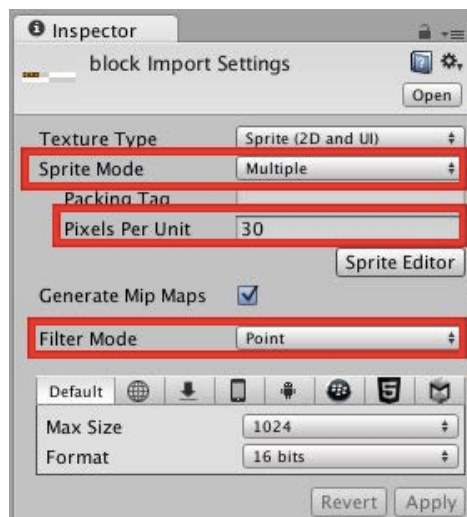
### 1. 地面を作る

最初に、¥¥mmnas01¥student¥GC2015¥02\_授業¥2017¥Unity¥20170515\_Unity2Dにある、「haikei.png」を Sprites フォルダへ Import します。

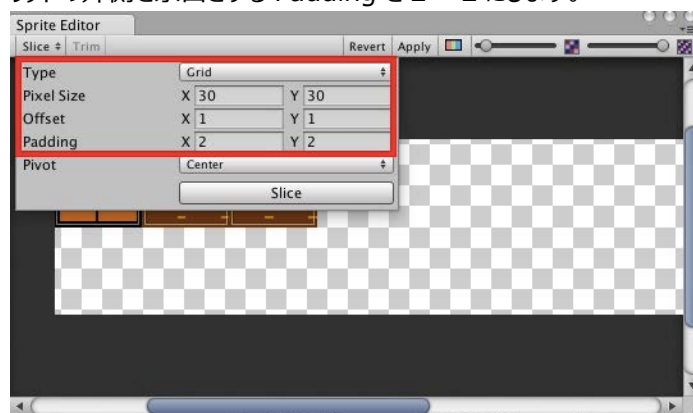


このようなマップチップ画像を Sprite Editor にてカットして使う時、Sprite と Sprite とが隣り合うと画像がちらついてしまうことが多々あります。そのために周囲をトリミングする場合があります。

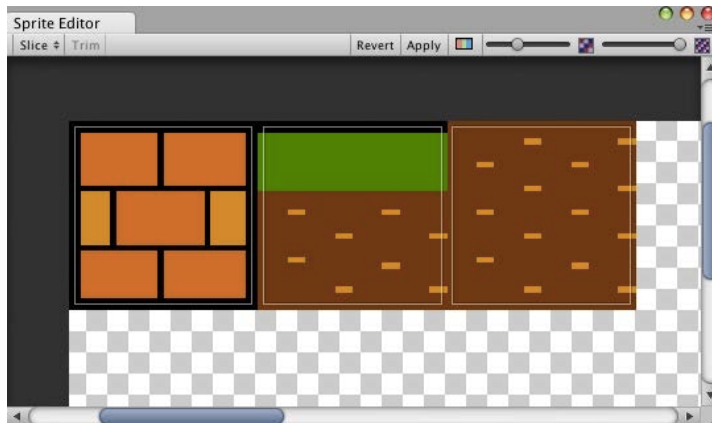
Inspector ビューにて Sprite Mode を Multiple、Pixels Per Unit を 30、Filter Mode を Point に変更し、Sprite Editor ボタンを押して下さい～Sprite 画像は 32 \* 32 の大きさですが、2px 小さい 30 を指定します。



Sprite Editor を開いたら Slice ボタンを押し、自分でカットする範囲を指定するため Type を Grid にし、画像サイズである Pixel Size を 30 \* 30、画像を左上から指定した値の分だけずらす Offset を 1 \* 1、グリッドの外側を余白とする Padding を 2 \* 2 にします。

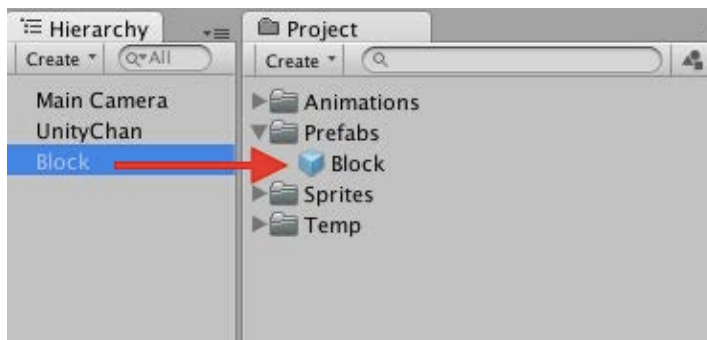


そうすると、下記の通りになります。

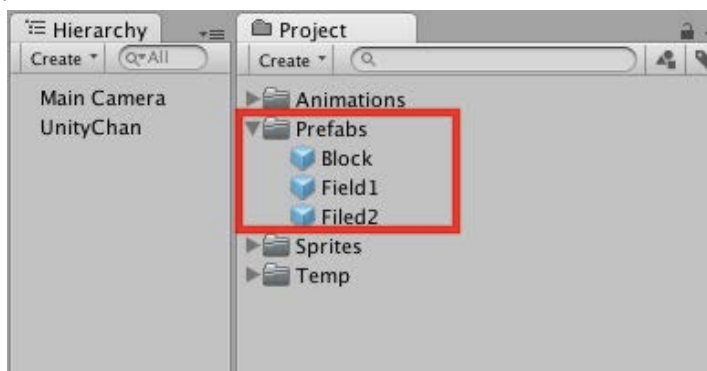


カットした Sprite 画像を Prefab 化しておきます。block\_0 を Hierarchy ビューヘッドラッグ&ドロップし、名前を「Block」に変更します。

その後、Box Collider2D コンポーネントを取り付け、Prefabs フォルダを新規作成しそこに保存してください。



これらの処理を Block\_1、Block\_2 に対しても行います～名前はそれぞれ「Field1」、「Field2」として下さい



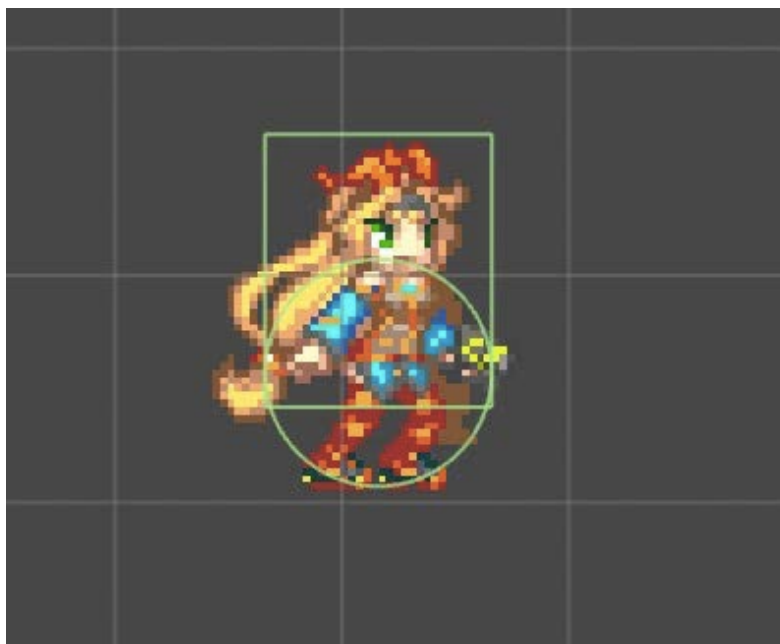
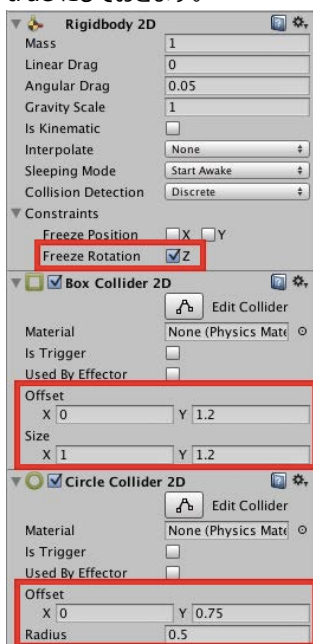
今度は、策ほど作成した Field1 と Field2 を並べて、地面を作成します。その際、下記の図のように Field1 なら Field1 という名前の、Field2 なら Field2 という名前の空のゲームオブジェクトを作成し、その下に Prefab を置き整頓しながら作ってゆくと整理ができます。



### 1.1. ユニティちゃんを地面に立たせる

続いてユニティちゃんを地面に立たせます。地面との接地面を最小にするためユニティちゃんに Rigidbody2D コンポーネント、Box Collider2D コンポーネント、Circle Collider2D コンポーネントを付け、下記の図のようにすう知恵尾調整します。

また、ユニティちゃんは常に垂直に立った状態にしたいので、Rigidbody2D の Constraints>Freeze Rotation.z を true にしておきます。

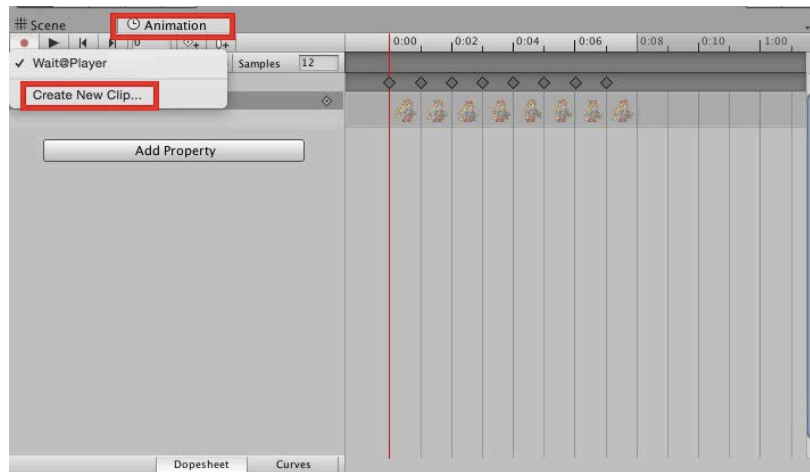


これでゲームを再生すると、Unity ちゃんが地面の上に立つはずです。

## 2. 歩きのアニメーション設定

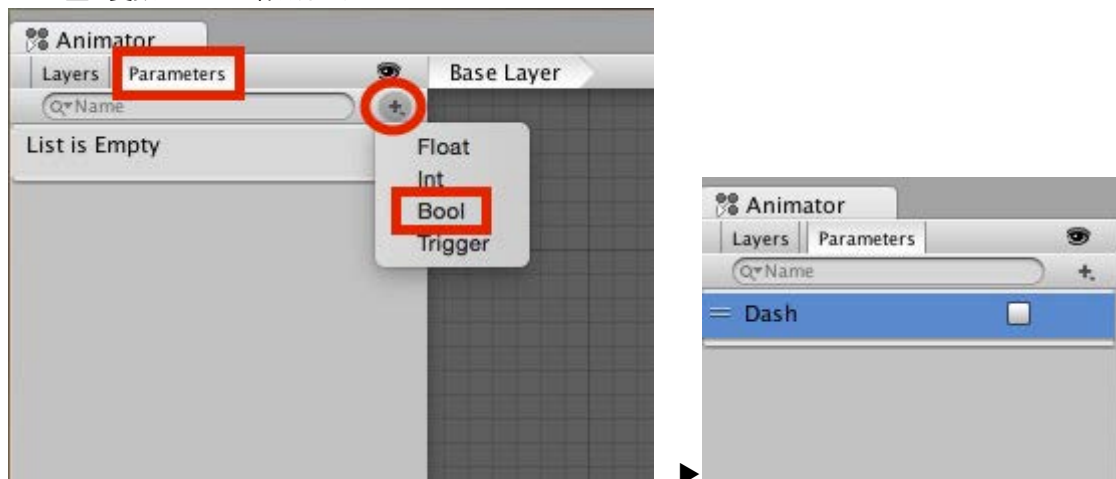
Hierarchy ビューに配置した Unity ちゃんの名前を、“UnityChan”に変更しておきます。その UnityChan を選択したまま、メニュー ▶ Window ▶ Animation から Animation ビューを開きます。現在待機状態のアニメーションが設定されています。そこに、“走る”アニメーションの動作を追加します。

Animation ビューから下記のように “Create New Clip” を選択し、“Dash@Player”という名前でアニメーション・セットを追加します。



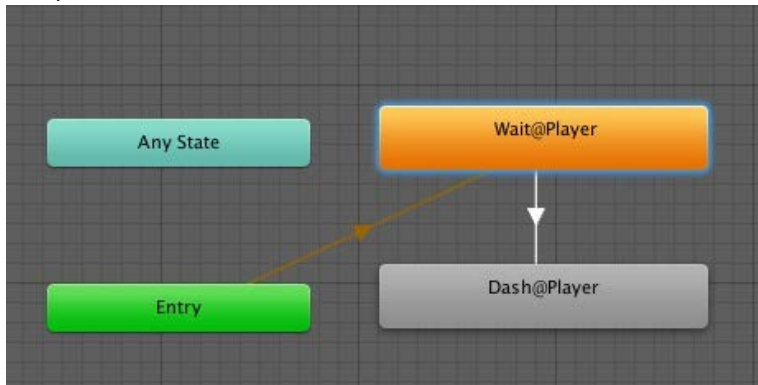
そこに、Sprite にある走るアニメーションの画像をドラッグアンドドラッグし画像を配置します。

続いて、アニメーションコントロール用に変数を作成するために Parameters をクリックし、プラスボタン>Bool を押して、Bool 型の変数“Dash”を作成します。

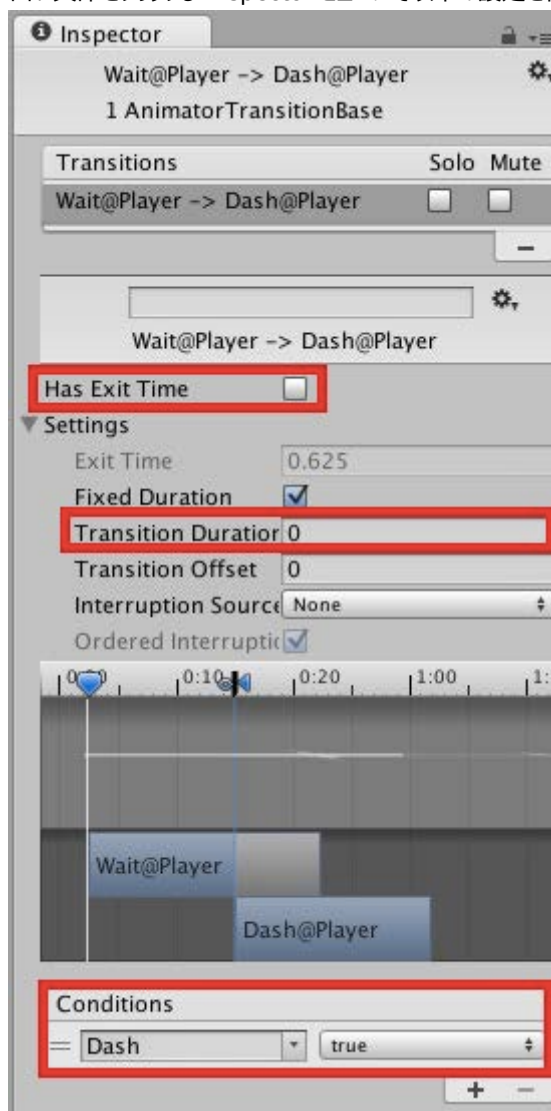


### 3. アニメーションの状態遷移を設定する。

UnityChan を選択したまま、Animator タブを押します。そうすると下記の用に画面が変わります。



白い矢印をクリックし Inspector ビューにて以下の設定を行います。

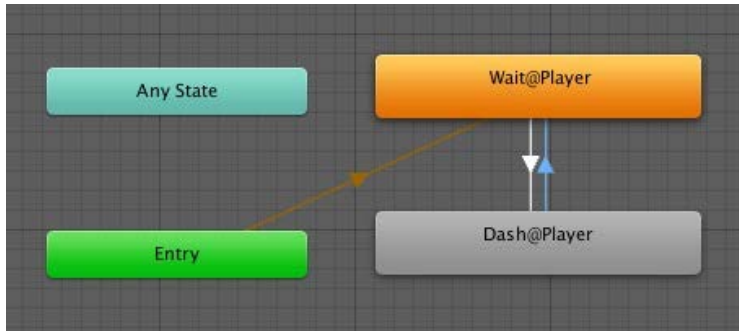


← アニメーションの継続時間、チェックを外すとループする

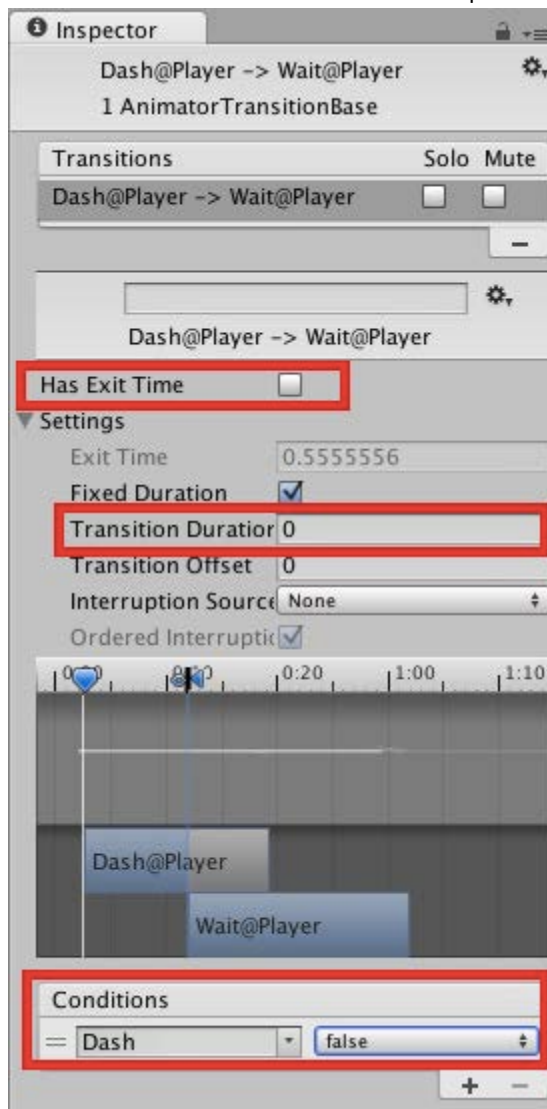
← アニメーションの切り替え時デレイ時間  
今回は、すぐに切り返したいので 0 とする。

← アニメーションが遷移する条件

同様に、""Dash@Player""から待機に向けて矢印を引きます。



作成した矢印をクリックし白い矢印をクリックし Inspector ビューにて以下の設定を行います。



← ここだけ異なります。

#### 4. Unity ちゃんをスクリプトで歩かせる。

Player.cs スクリプトを作成し、Unity ちゃんを動かしてみましょう。

Player.cs

```
1  using UnityEngine;
2  using System.Collections;
3
4  public class Player : MonoBehaviour {
5
6      public float speed = 4f; //歩くスピード
7      private Rigidbody2D rigidbody2D;
8      private Animator anim;
9
10     void Start () {
11         //各コンポーネントをキャッシュしておく
12         anim = GetComponent<Animator>();
13         rigidbody2D = GetComponent<Rigidbody2D>();
14     }
15
16     void FixedUpdate ()
17     {
18         //左キー: -1、右キー: 1
19         float x = Input.GetAxisRaw ("Horizontal");
20         //左か右を入力したら
21         if (x != 0) {
22             //入力方向へ移動
23             rigidbody2D.velocity =
24                 new Vector2 (x * speed, rigidbody2D.velocity.y);
25             //localScale.x を-1 にすると画像が反転する
26             Vector2 temp = transform.localScale;
27             temp.x = x;
28             transform.localScale = temp;
29             //Wait→Dash
30             anim.SetBool ("Dash", true);
31             //左右も入力していなかったら
32         } else {
33             //横移動の速度を 0 にしてピタッと止まるようにする
34             rigidbody2D.velocity =
35                 new Vector2 (0, rigidbody2D.velocity.y);
36             //Dash→Wait
37             anim.SetBool ("Dash", false);
38         }
39     }
40 }
```