

株式会社アウスタ さいたまゲーむす 勉強会資料

勉強会情報

<https://atnd.org/users/184556>

アプリ作ってます！

Android

https://play.google.com/store/apps/developer?id=SAITAMA_GAMES

IOS

<https://itunes.apple.com/jp/app/pichannobaningu!/id880087045>

●勉強会で学べること

Unityのアニメーションなど

●作る内容

音ゲーのような感じ

右から左へスライドするキーが枠に入ったタイミングで打ち込んでいく
キーに対応して画面にいるU n i t yちゃんが動く

●〇準備

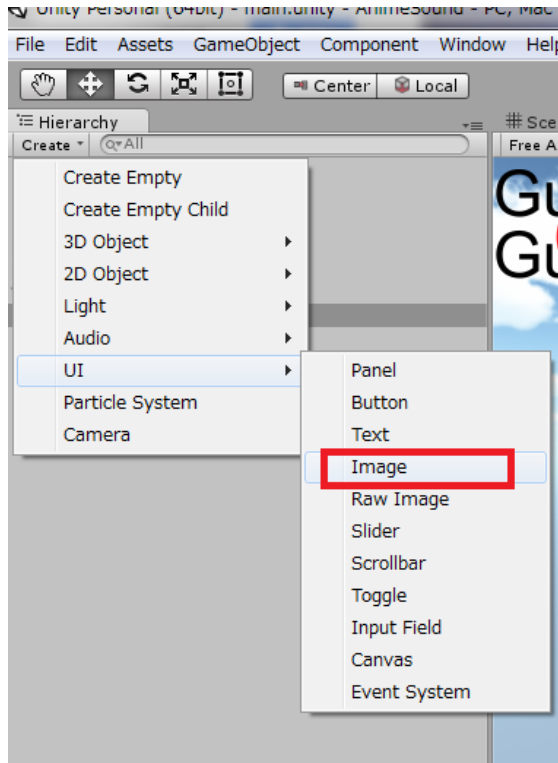
0-1) AnimeSound.zipをダブルクリック 解凍する

0-2) Assetsフォルダ内のmain.unityをダブルクリック unityを起動する。

● 1 音ゲームのUI(ユーザーインターフェース)の作成

1-1) 2次元表示の作成

Hierarchy > Create > UI > Image で2d画像表示の元を作成

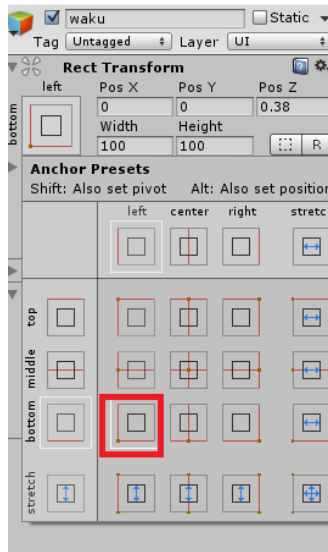


ImageがHierarchyウィンドウに作成される。
Imageをクリックしたときに右側に出てくる
Inspectorの内容を変更
Name back
以下backと呼びふ。

1-2) 下寄せ 中央に変更

RectTransformの項目の
左のほうに四角があるのでクリック
その後 bottom Left を選択

次ページ画像参照



1-3)中心点 位置 大きさ の変更

まず中心点を変更する

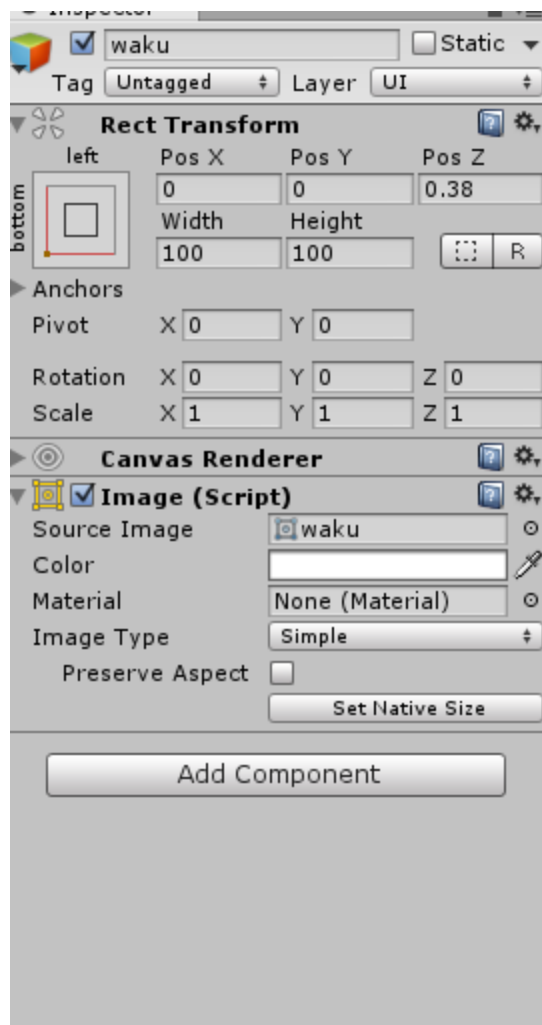
Pivot x 0 y 0

位置を設定

pos X 0 pos Y 0

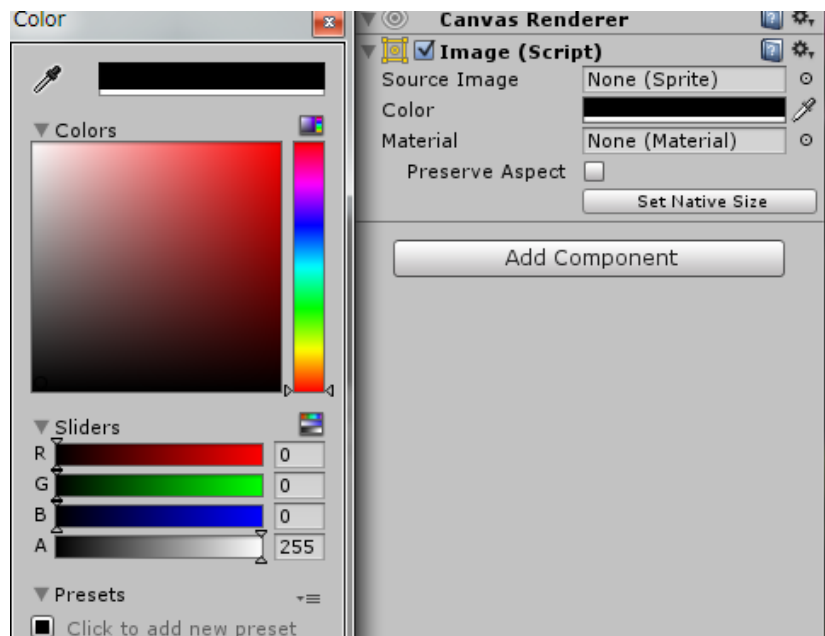
大きさを設定

Scale X 25 Y 1



1-4)Colorを変更

Imageの Color を R O G O B O に設定



1-5) 枠の作成準備

枠内に文字が入ったらキーを押すゲームなので
その枠を作成。

Hierarchy > Create > UI > Image で2d画像表示の元を作成

ImageがHierarchyウィンドウに作成される。

1-6) 枠の位置の設定

Imageをクリックしたときに右側に出てくる

Inspectorの内容を変更

Name waku

RectTransform

左のほうに四角があるのでクリック

その後 bottom left を選択

Pivot x 0 y 0

pos X 0 pos Y 0

以下wakuと呼ぶ。

1-7)wakuに画像をセット

projectフォルダ内の
Assets>textuer>waku を

ImageないのSouce Imageに設定する

●2キーアイコン生成システム

2-1) キーアイコン2次元表示の作成

Hierarchy > Create > UI > Image で2d画像表示の元を作成

ImageがHierarchyウィンドウに作成される。

ImageをクリックしてCanvasにドラッグ
U n i t yで親子関係と呼ばれる状態にする。

2-2)位置調整など
位置の設定を変更

Name KeyObjC

RectTransform

左のほうに四角があるのでクリック

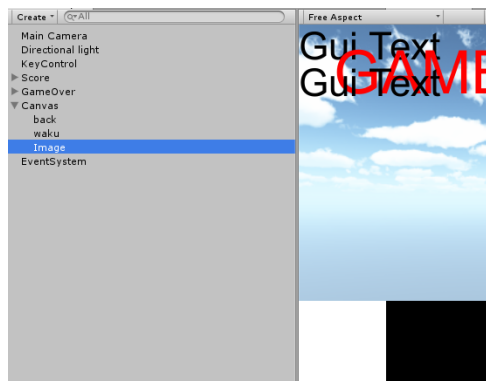
その後 bottom left を選択

Pivot x 0 y 0

pos X 0 pos Y 0

2-3)描画順調製

描画順が上から順になるのでこのような配置に変更する



2-4)KeyObjCに画像をセット

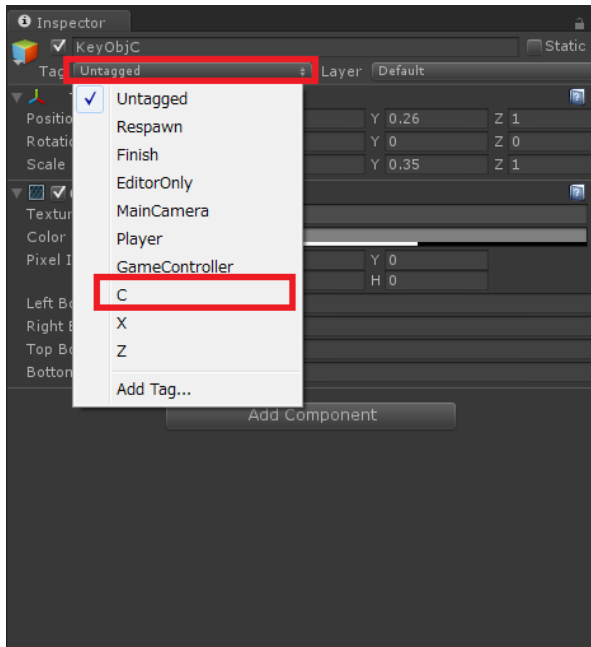
projectフォルダ内の
Assets>textuer>C を

ImageないのSouce Imageに設定する

2-5)タグの設定

そのキーアイコンに対応したキーを入力することで成功失敗を判定するので、その判定に必要なタグを設定します。

KeyObjC の Inspector のTag項目をクリック
表示された項目のCをクリック



これでCが設定される

2-4)KeyObjCに動きを設定

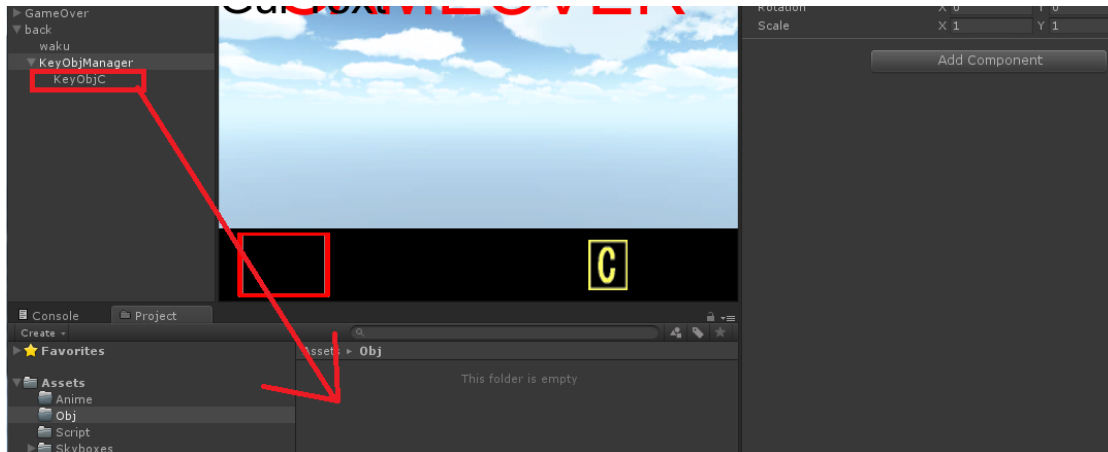
KeyObjCのInspector 下にある
AddComponent をおして
Script->KeyObj をクリック
キーの動きをさせるスクリプトが設定される。

再生すると右から左へ移動する。

2-5)KeyObjCをプレファブ化

KeyObjCを
Projectウィンドウの

Assets->Objにドラッグ



これでプレファブ化が完了。

2-6)KeyObjを量産

今作ったのはCキーのオブジェクト

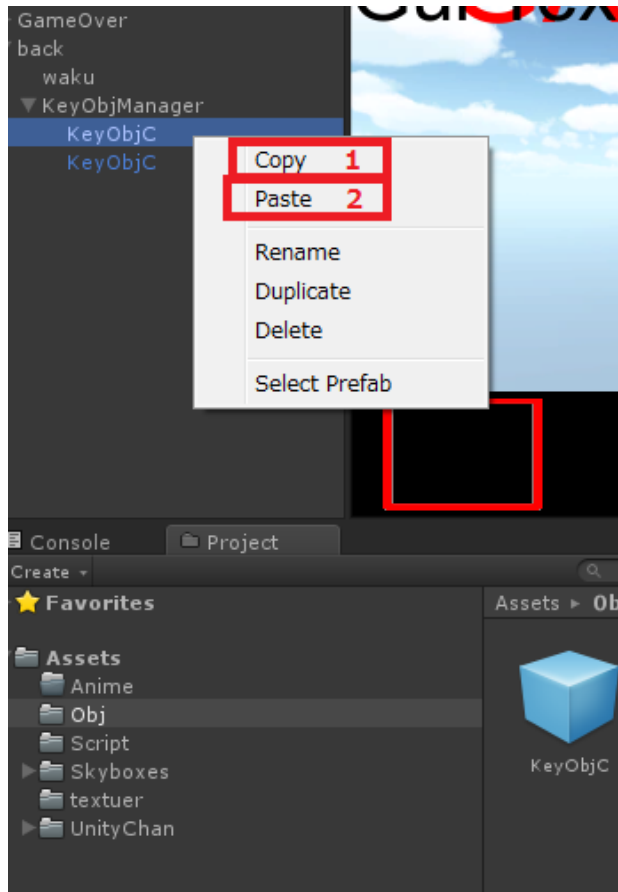
Z X も作らないといけないので同じような手順で作っていく。

KeyObjCを右クリック

Copyを押してコピー

また右クリックでPaste

貼り付ける



2回繰り返すことでKeyObjCが3つになる

2個目のKeyObjCの Inspector 編集

Name KeyObjZ

TagをZにへんこう

Assets>textuer>Z を

ImageないのSouce Imageに設定する

できあがったらProjectウィンドウの

Assets->Objにドラッグしてプレファブ化

3個目のKeyObjCの Inspector 編集

Name KeyObjX

TagをXにへんこう

Assets>textuer>X を

ImageないのSouce Imageに設定する

Assets->Objにドラッグしてプレファブ化

これで3つのキーができた

HierarchyのKeyObjC KeyObjZ KeyObjX

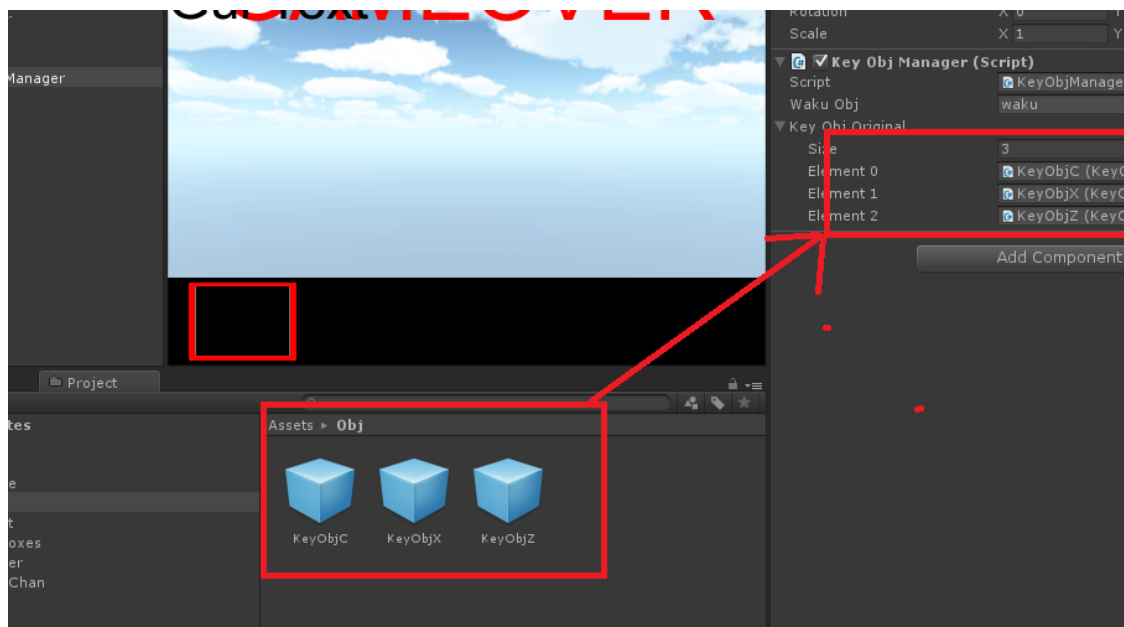
は削除。

2-6)キー生成システムマネージャーの作成！

Hierarchy内の
CanvasのInspector -> AddComponent
Scripts->KeyObjManagerを選択
スクリプトが付与される

KeyObjManagerの
WakuObjに
Hierarchyにあるさっきつくったwakuをドラッグして追加

KeyObjManager内の
KeyObjOriginalの項目の左にある三角を押すと
下にSizeという項目が出る そこを3に設定
Element 0~2がでてくるので
そこにProjectウィンドウ内の
ObjフォルダのKeyObjC KeyObjX KeyObjZ
を設定する。



これでキーが自動で生成される。

●3キャラクターの配置 アニメーションなど

3-1)unitychan(女の子のキャラクター)の配置
Projectウィンドウ
UnityChan->Prefabs->unitychan
をHierarchyにドラッグ

Inspectorで位置を調整
transform
position x 0 y 0 z 0
rotation x 0 y 180 z 0

以下この女の子のキャラクターをunitychanと呼びます。

3-2)unitychanにアニメーション用のスクリプトを設定

unitychanのInspector 下にある
AddComponent をおして
Script -> U n i t y C h a n C o i n をクリック
同様に
Script -> ActionManager をクリック

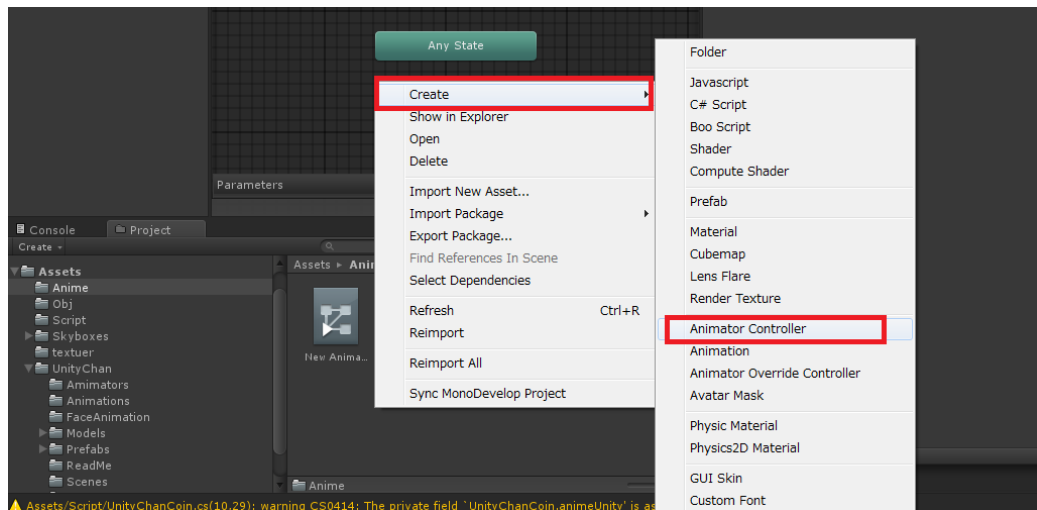
キャラクターのアニメーションをつかさどるスクリプトが追加される。

3-3)アニメーションの作成
3Dキャラクターのモーションの設定をするAnimator Controller
を作成。

projectウィンドウの

Assets->Animeフォルダ内で右クリック
create->Animator Controller

New Animator Controller ができます。



New Animator Controllerをダブルクリックすることで
Animatorウィンドウが開きます。

ここにアニメーションの遷移を作成していきます。

3-3)パラメーター作成

アニメーションに使うパラメーターの作成をしておきます。

Animator左上のParamatersを押す
出てきた項目の+ボタンを押し、
さらに出てきた項目の中の[Bool]をおします。

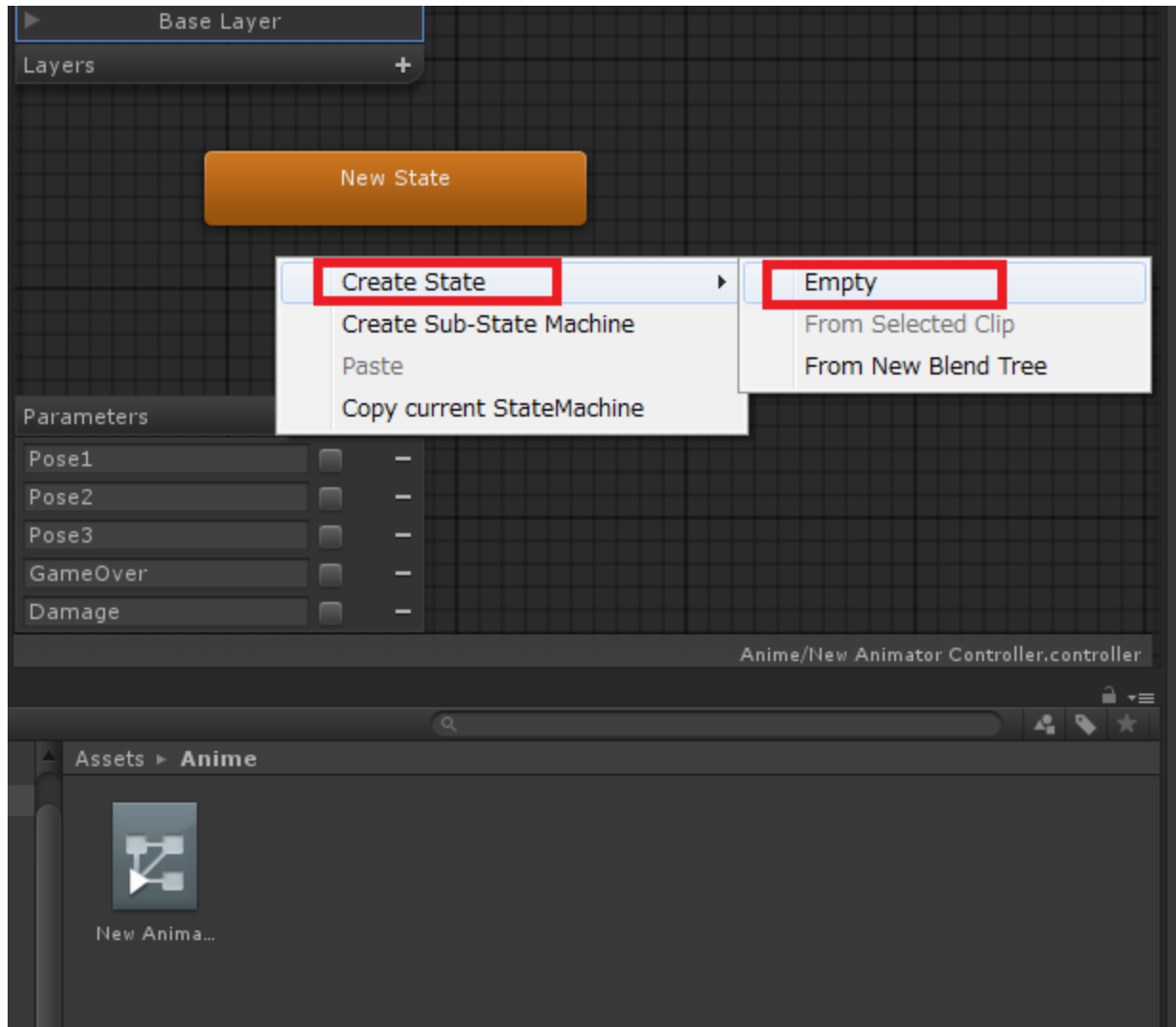
作成されたNewBool
の名前をPose1にへんこう

同じ感じで
Pose2
Pose3
GameOver
Damage
を作成。

※ 1 字一句間違えなく作成

3-4)待機モーション作成

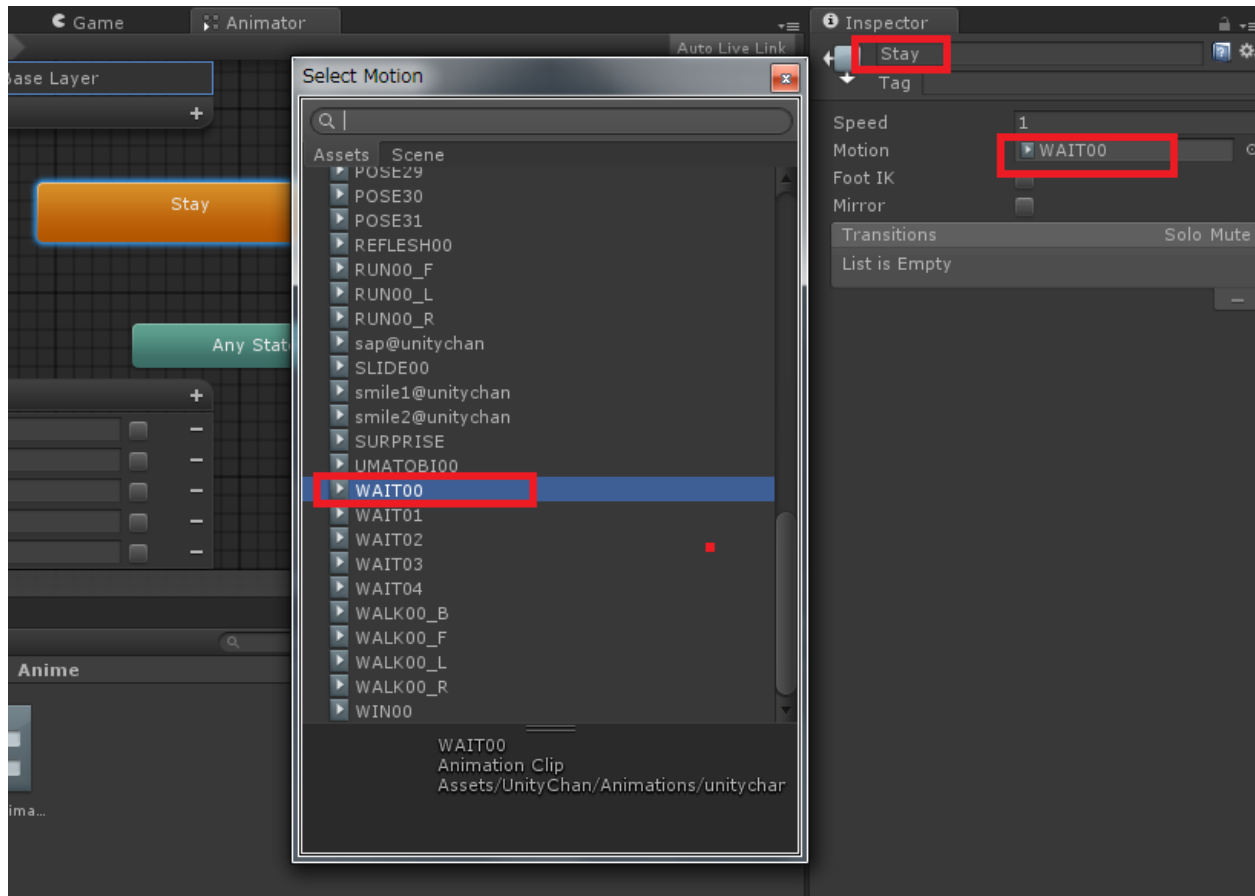
右クリックを押して
Create Status -> Empty を入力
モーションの元を作成。



作成されたNew Stateをクリックすると
Inspectorが右に出る。
New Stateという名前はStayに変えておく。

Motion 項目の右側にある◎マークをクリックすることで
モーション（キャラの動き）選択用のウィンドウが出る。
左上の項目からAssetsをクリック

その中からWA I T 00を選んで選択。



3-5)各種ポーズを作成

キー入力成功ポーズ3種
 ミスったときのモーション
 ゲームオーバーのモーション

を作成していきます。

・ 1 個目

まずキー成功のポーズ1です
 右クリックを押して
 Create Status -> Empty を入力

Inspectorで名前をPOSE1
 motionをPOSE04に設定

・ 2 個目

このPOSE1を右クリックしCopy
 何もないところでもう一回右クリックで Paste
 出来上がったPOSE1 0の
 名前をPOSE2に設定
 motionをPOSE05に設定

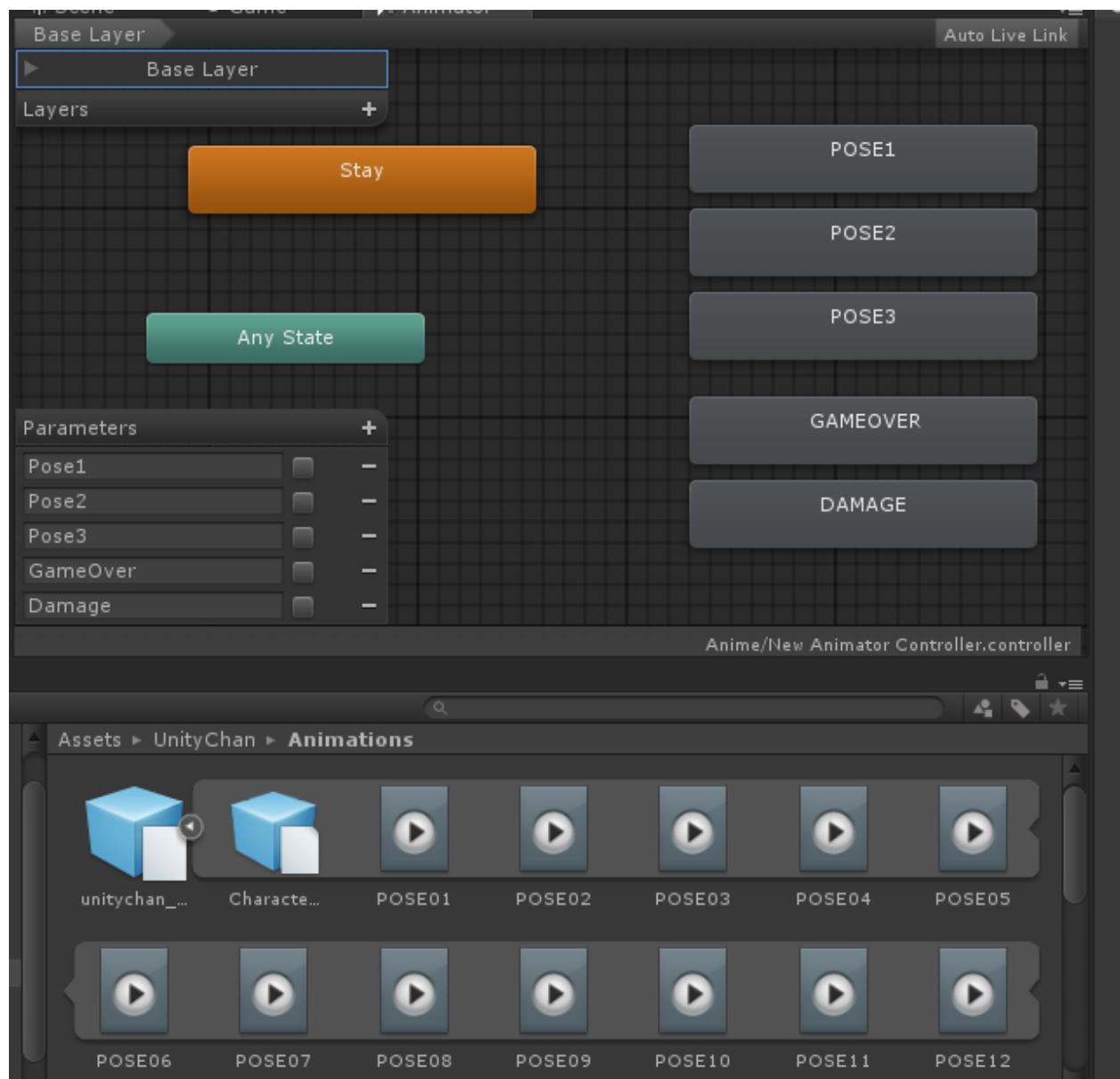
・ 3 個目

さらにコピーペーストで
3個目を作成
名前をPOSE3
Motion を POSE30に設定

・ 4 個目
さらにコピーペーストで
4個目を作成
名前をGAMEOVER
Motion を LOSE00に設定

・ 5個目
さらにコピーペーストで
5個目を作成
名前をDAMAGE
Motion を DAMAGE00に設定

位置を調整したりすると大体こんな感じになります↓



3-6)各種アニメーションの関連付け

まずさっき作った **S t a y** を右クリック
 一番上の **MakeTransition** を選択
 その状態で **POSE1** をクリックすると
 二つをつなぐ矢印が出てくる

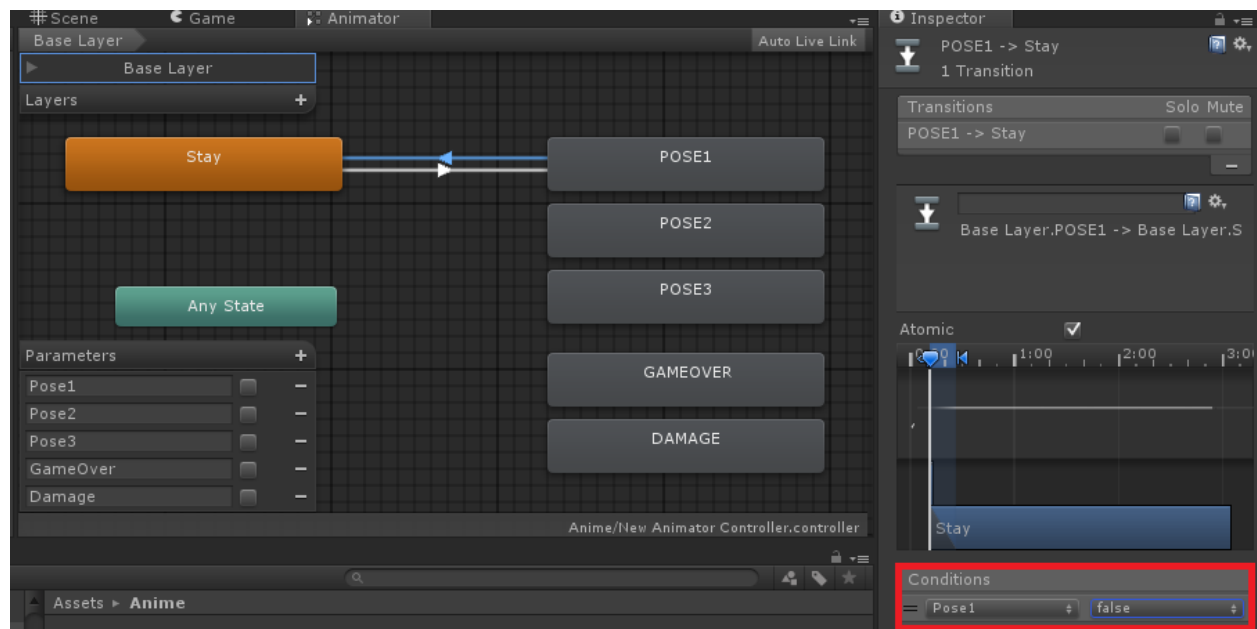
さらに **POSE1** を右クリック
MakeTransition をクリック
 その状態で **STAY** をクリック

これで二つのモーションがつながる。

この二つの矢印のうちの
STAY から **POSE1** に向かう矢印をクリック

Insoectorに矢印の情報が表示
Cndtionsの+ボタンをクリック
Pose1にへんこう 右側の項目はtrue
Has Exit Time を off

POSE1からSTAYに向かう矢印をクリック
Insoectorに矢印の情報が表示
Cndtionsの
ExitTimeをクリック
Pose1にへんこう 右側の項目はfalse
Has Exit Time を off



フラグのオンオフで
モーションが遷移するようになる。

これをPOSE2 POSE3も設定

・ P O S E 2

S t a y を右クリック
一番上のMakeTransitionを選択
その状態でPOSE2をクリックすると
二つをつなぐ矢印が出てくる

さらにPOSE2を右クリック
MakeTransitionをクリック
その状態でSTAYをクリック

これで二つのモーションがつながる。

この二つの矢印のうちの
STAYからPOSE2に向かう矢印をクリック
Insoectorに矢印の情報が表示

Condtionsの+ボタンをクリック
Pose2にへんこう 右側の項目はtrue
Has Exit Time を off

POSE2からSTAYに向かう矢印をクリック
Insoectorに矢印の情報が表示
Condtionsの+ボタンをクリック
Pose2にへんこう 右側の項目はfalse
Has Exit Time を off

・ P O S E 3

S t a y を右クリック
一番上のMakeTransitionを選択
その状態でPOSE3をクリック

さらにPOSE3を右クリック
MakeTransitionをクリック
その状態でSTAYをクリック

これで二つのモーションがつながる。

この二つの矢印のうちの
STAYからPOSE3に向かう矢印をクリック
Insoectorに矢印の情報が表示
Condtionsの+ボタンをクリック
Pose3にへんこう 右側の項目はtrue
Has Exit Time を off

POSE3からSTAYに向かう矢印をクリック
Insoectorに矢印の情報が表示
Condtionsの+ボタンをクリック
Pose3にへんこう 右側の項目はfalse
Has Exit Time を off

・ G A M E O V E R

G A M E O V E Rは一方通行です。

S t a y を右クリック
一番上のMakeTransitionを選択
その状態でGAMEOVERをクリック

STAYからGAMEOVERに向かう矢印をクリック
Insoectorに矢印の情報が表示
Condtionsの+ボタンをクリック
GameOverにへんこう 右側の項目はtrue
Has Exit Time を off

・ D A M A G E

S t a y を右クリック
一番上のMakeTransitionを選択
その状態でDAMAGEをクリック

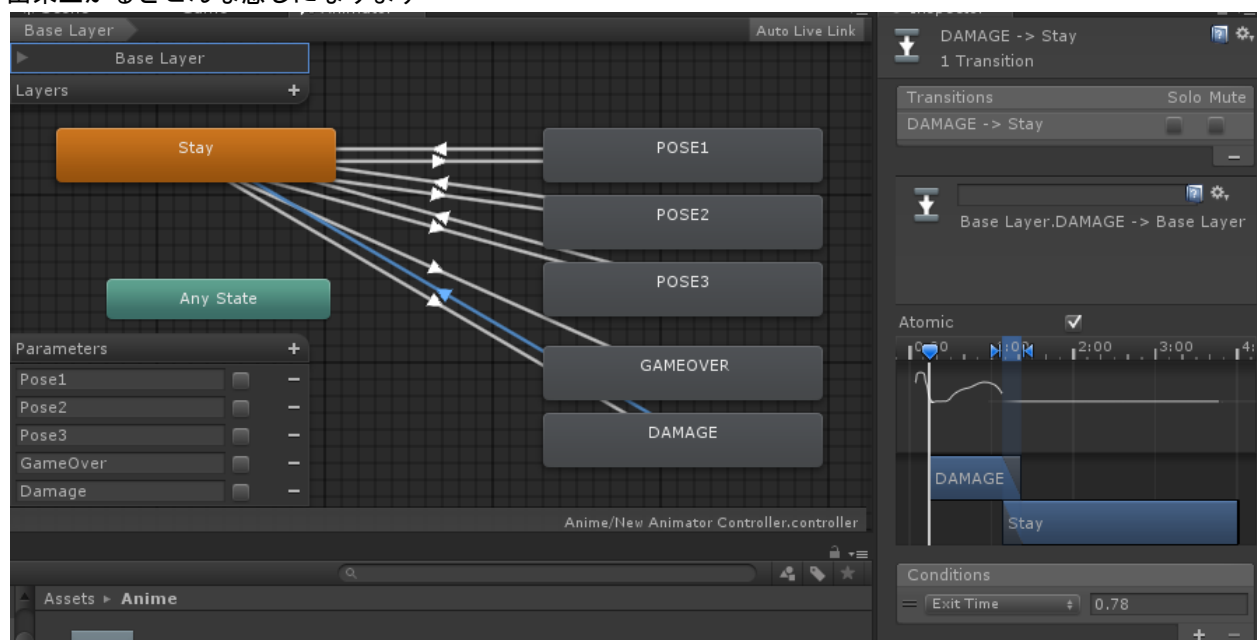
Insoectorに矢印の情報が表示
Cndtionsの+ボタンをクリック
Damageにへんこう 右側の項目はtrue
Has Exit Time を off

さらにDAMAGEを右クリック
MakeTransitionをクリック
その状態でSTAYをクリック
Cndtionsの+ボタンをクリック
Damageにへんこう 右側の項目はfalse
Has Exit Time を off

これで二つのモーションがつながる。

この二つの矢印のうちの
STAYからDAMAGEに向かう矢印をクリック
Insoectorに矢印の情報が表示
Cndtionsの
ExitTimeをクリック
Pose3にへんこう 右側の項目はtrue
Has Exit Time を off

出来上がるとこんな感じになります



3-7)unitychanにNew Animator Controllerを設定

unitychanをクリック
Inspectorの
Animatorの
Controllerに
projectファイルのNew Animator Controller
をドラッグして設定
アニメーションが適応されます。

時間が余ったら
動きに合わせて声を出すとか