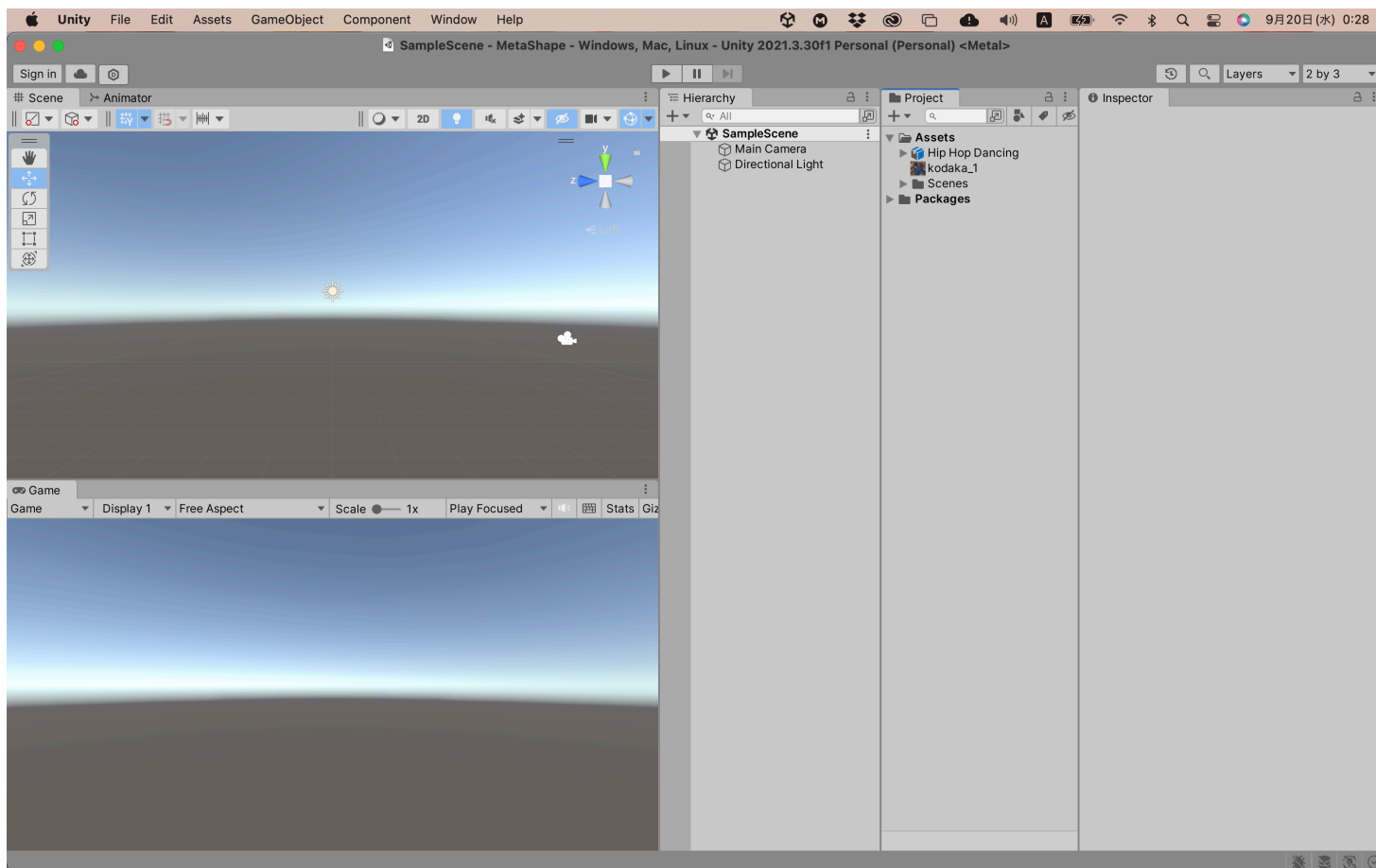


## 4. Unity(実際に動かしてみる)編

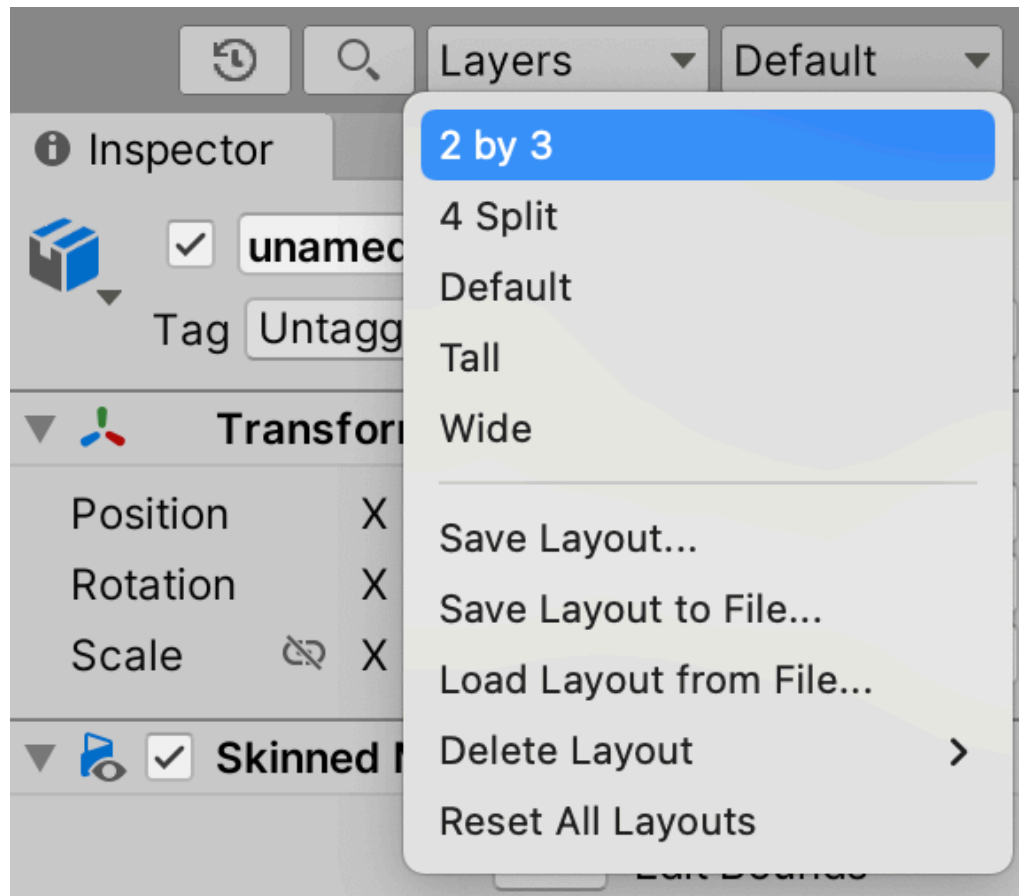
# 4. Unity編

アニメーション付きのモデルをUnityで動かします



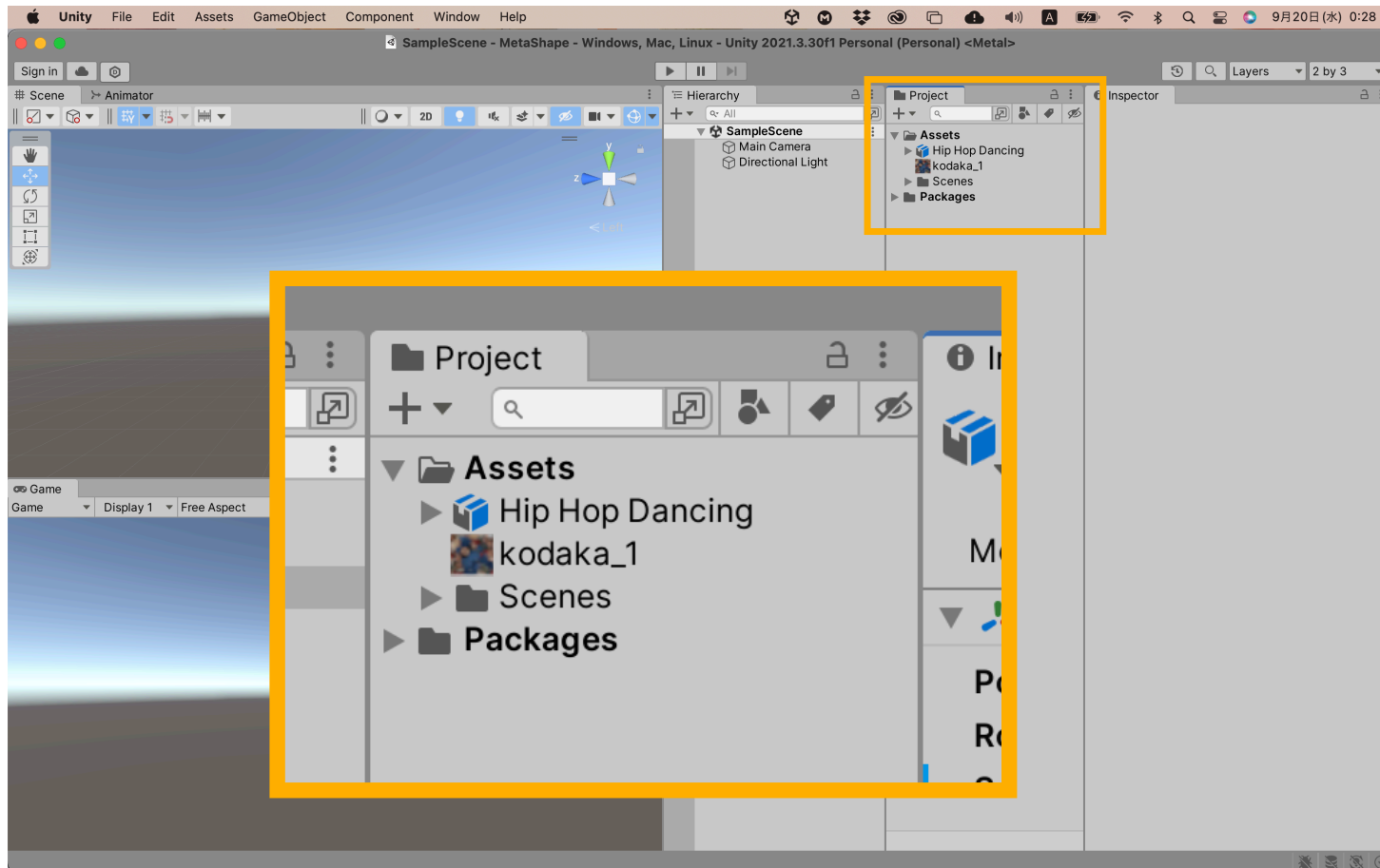
## 4. Unity編

今回は右上の[Default▼]でレイアウトを2by3にしています



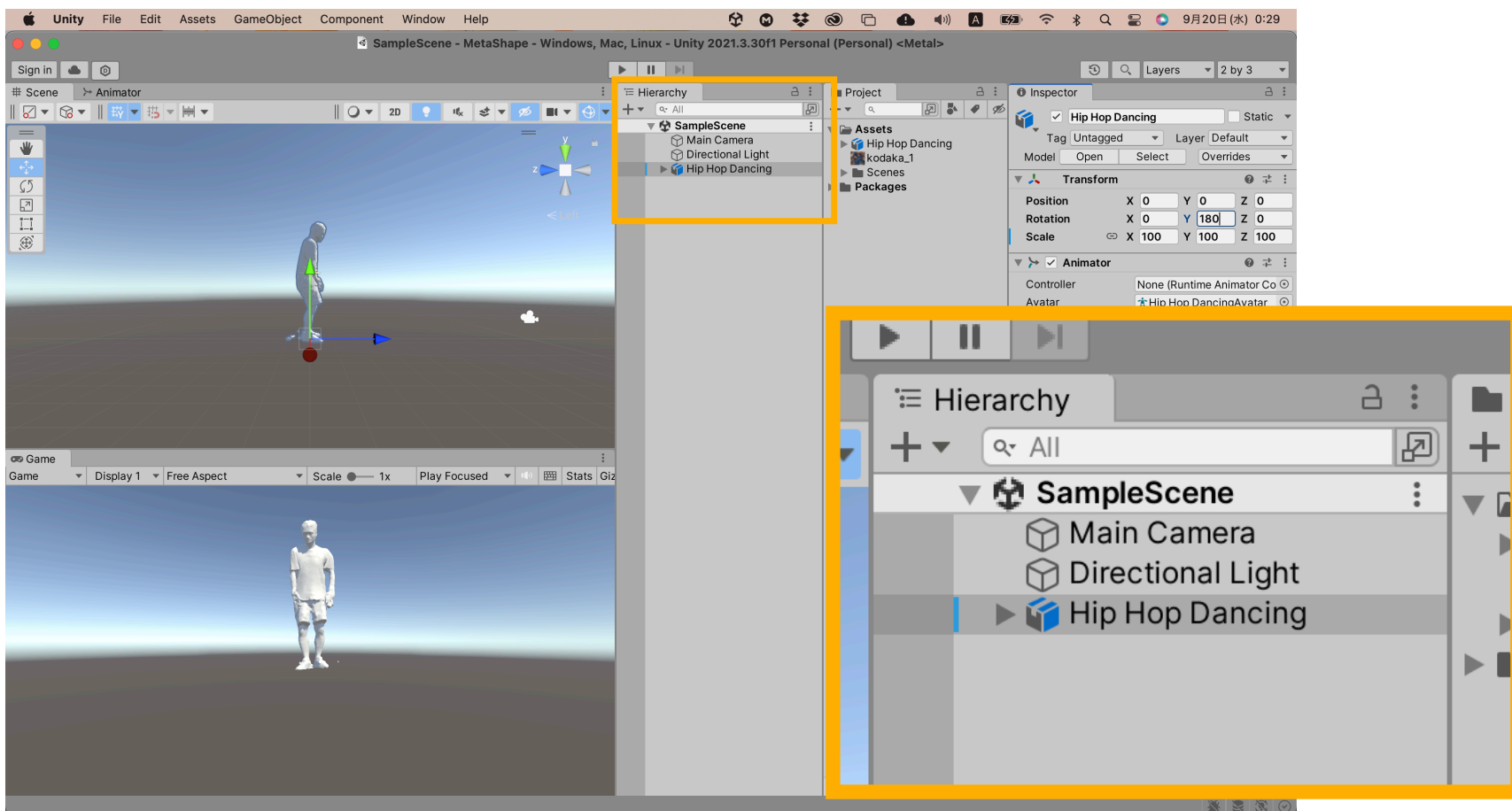
## 4. Unity編

まずMixamoからDLしたFBXとMetaShapeで書き出したテクスチャをProjectウィンドウのAssets配下にドラッグ&ドロップしてください



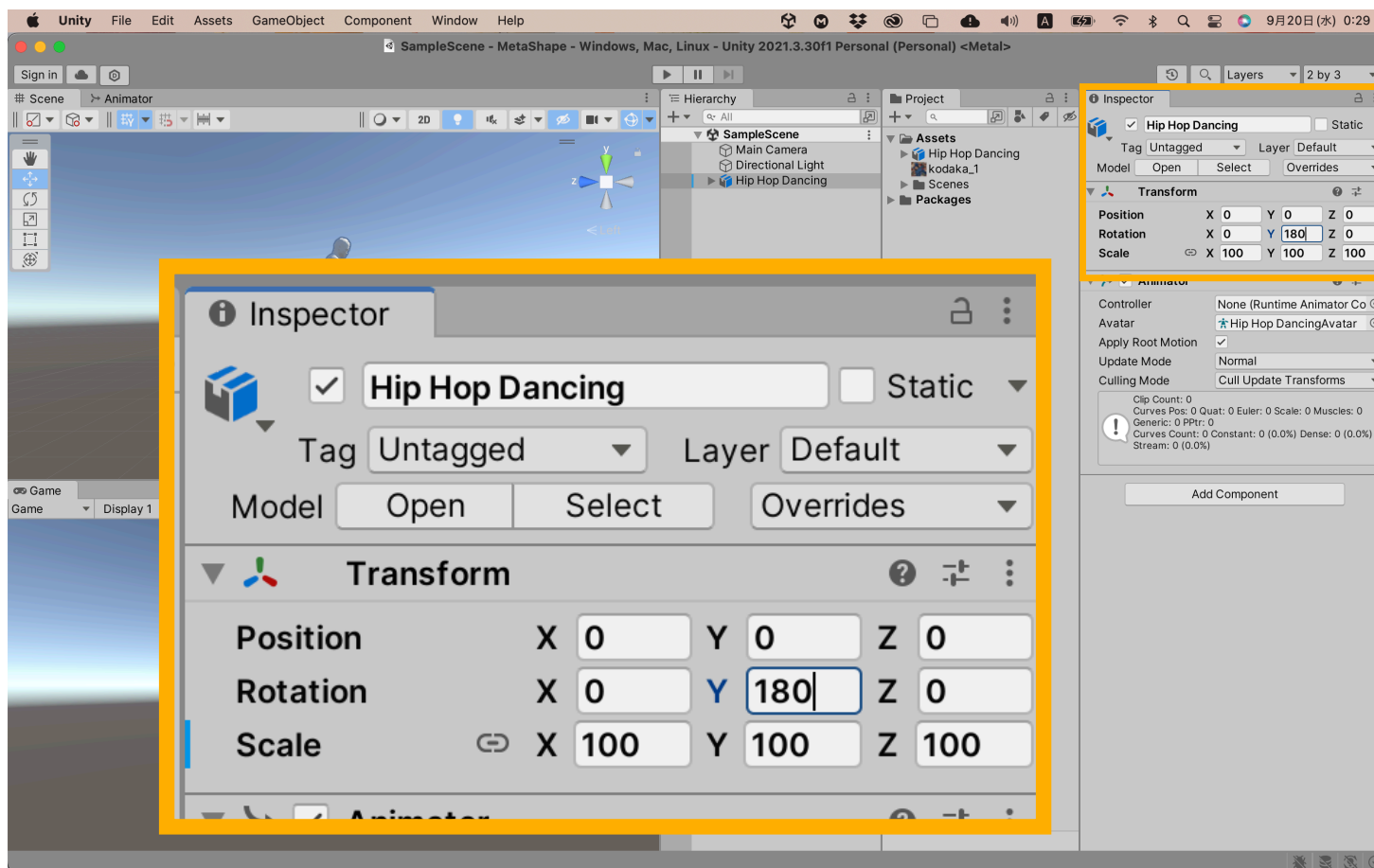
## 4. Unity編

次に今入れたFBXを横のHierarchyウィンドウのSampleScene配下にドラッグ&ドロップしてください



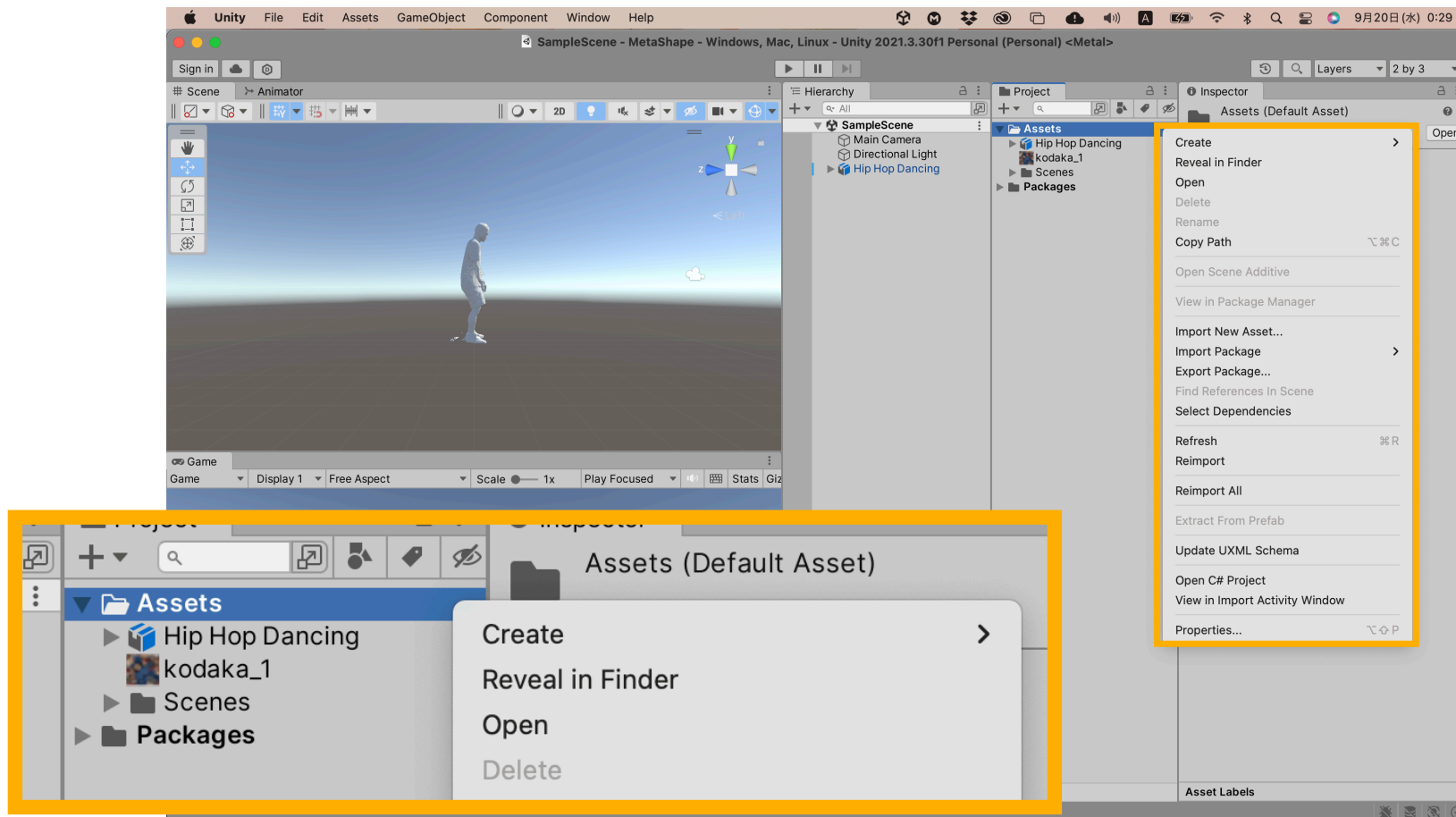
## 4. Unity編

Hierarchyに置いたFBXを選択し、右のInspectorウィンドウのTransformで位置、回転、サイズを調整できます



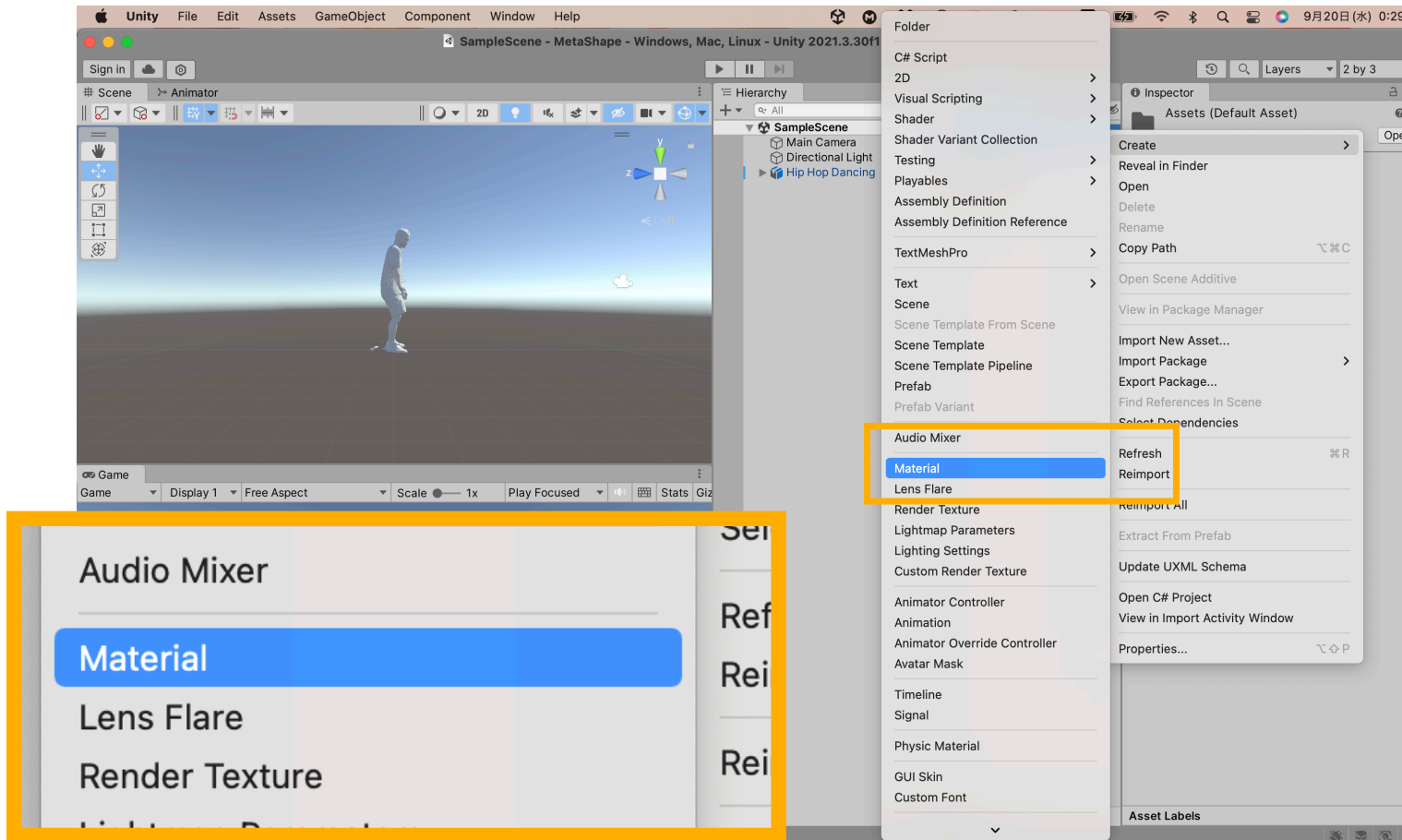
## 4. Unity編

次にProjectウィンドウでAssetsを選択し右クリックでメニューを出します



# 4. Unity編

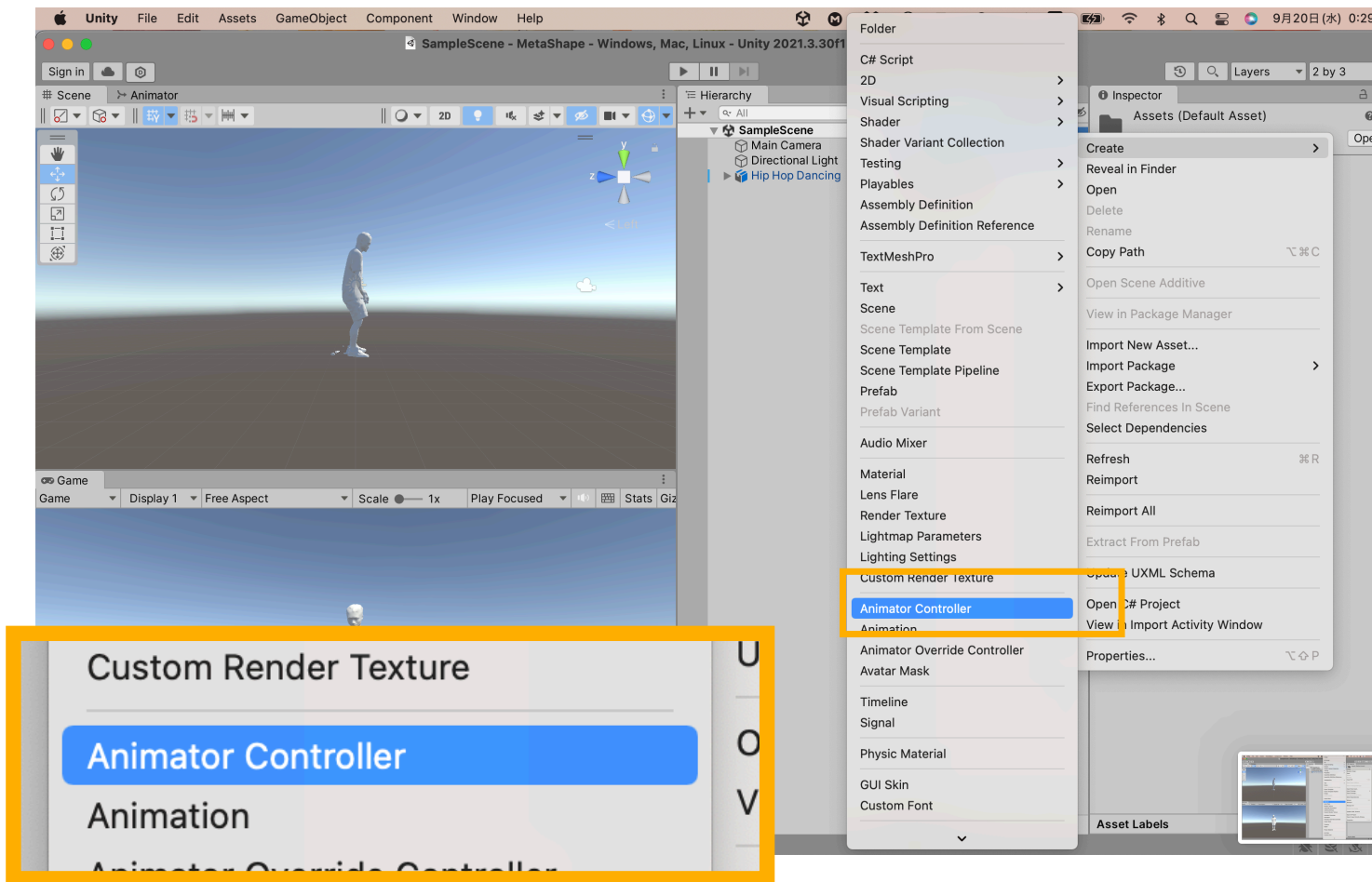
メニューのCreateからMaterialと





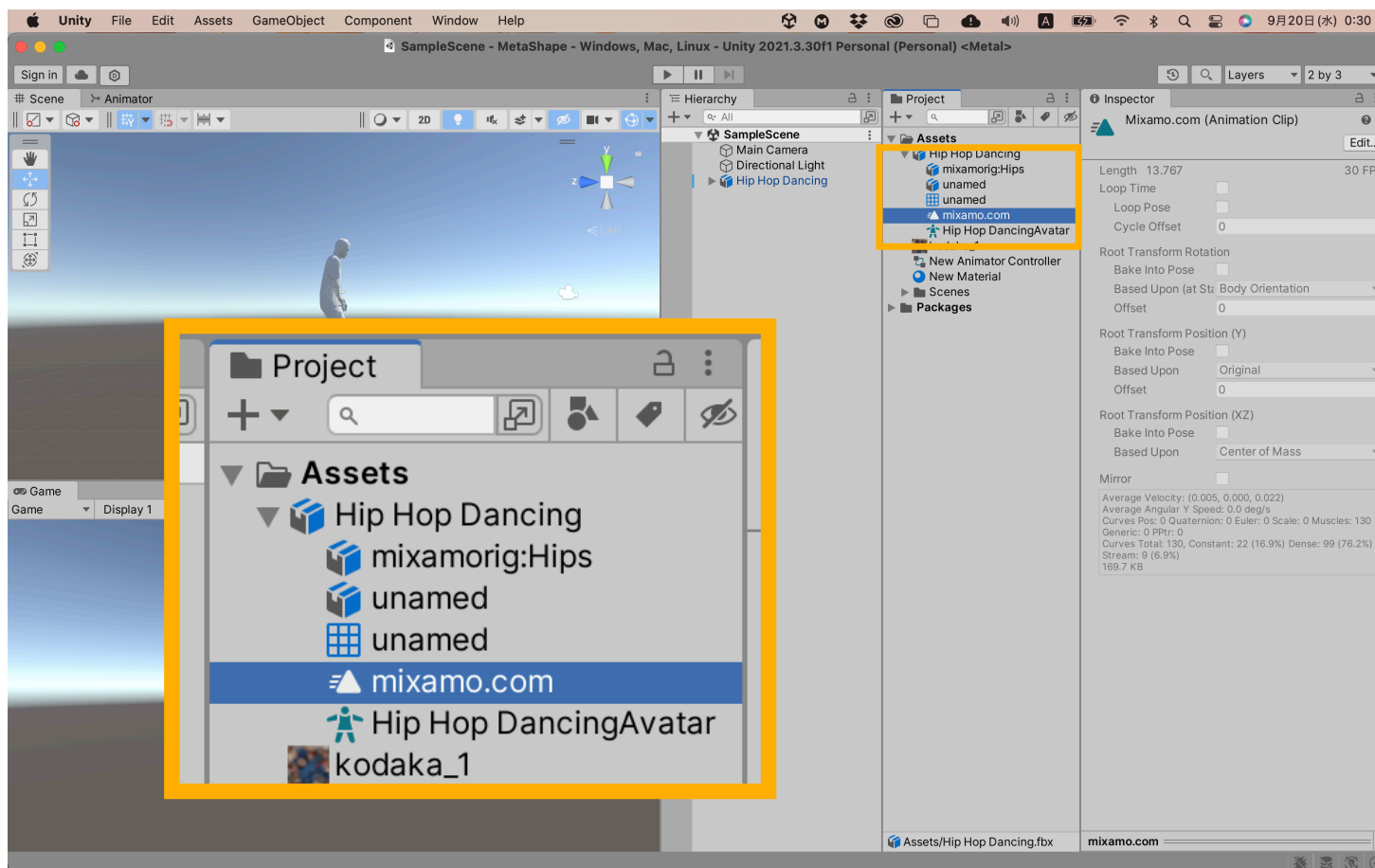
# 4. Unity編

AnimatorControllerを選択して生成します



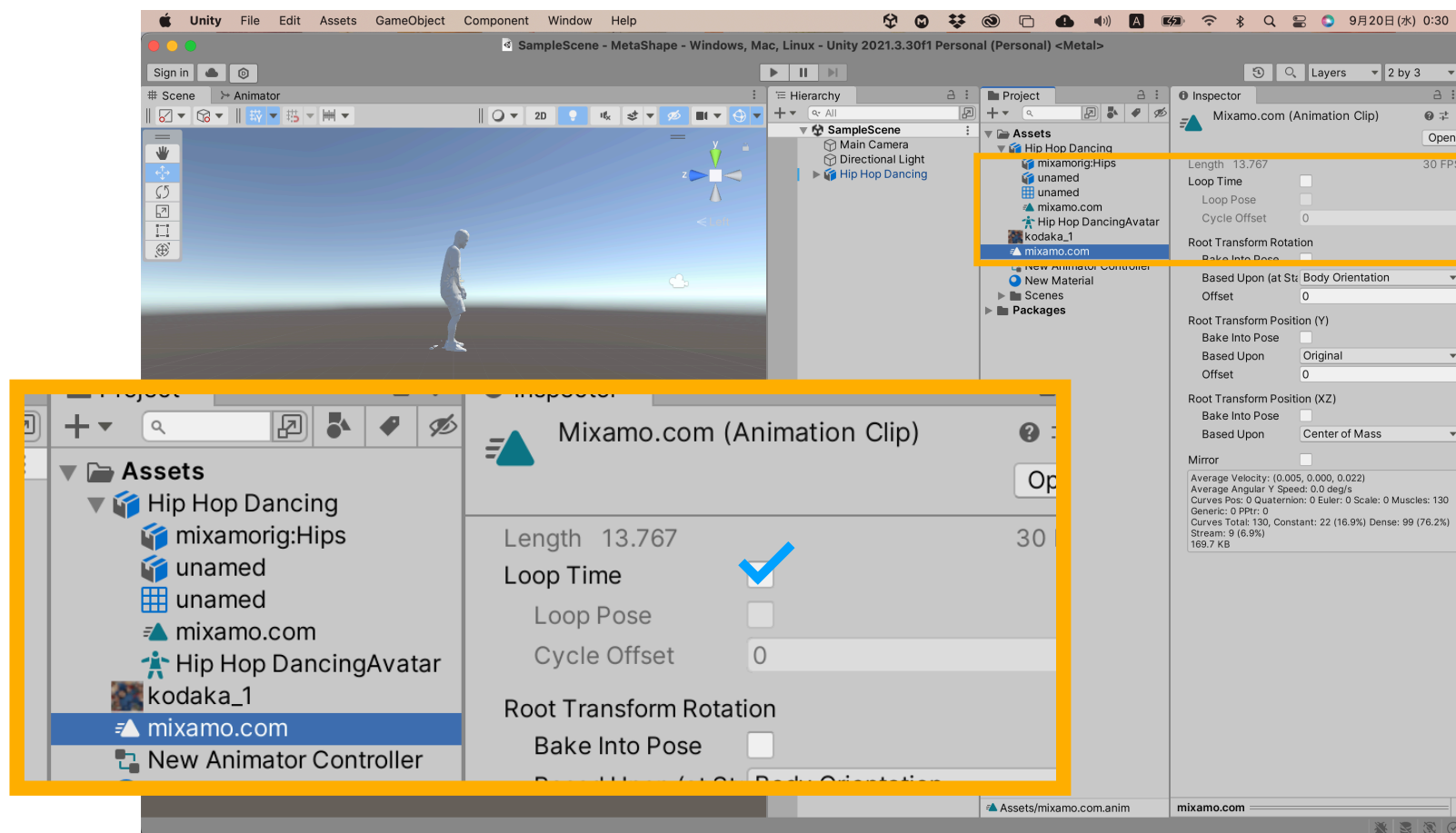
## 4. Unity編

次にProject/Assets/\*mixamoでDLしたfbxに格納されている  
mixamo.comというアニメーションファイルを選択して⌘+Dで複製します



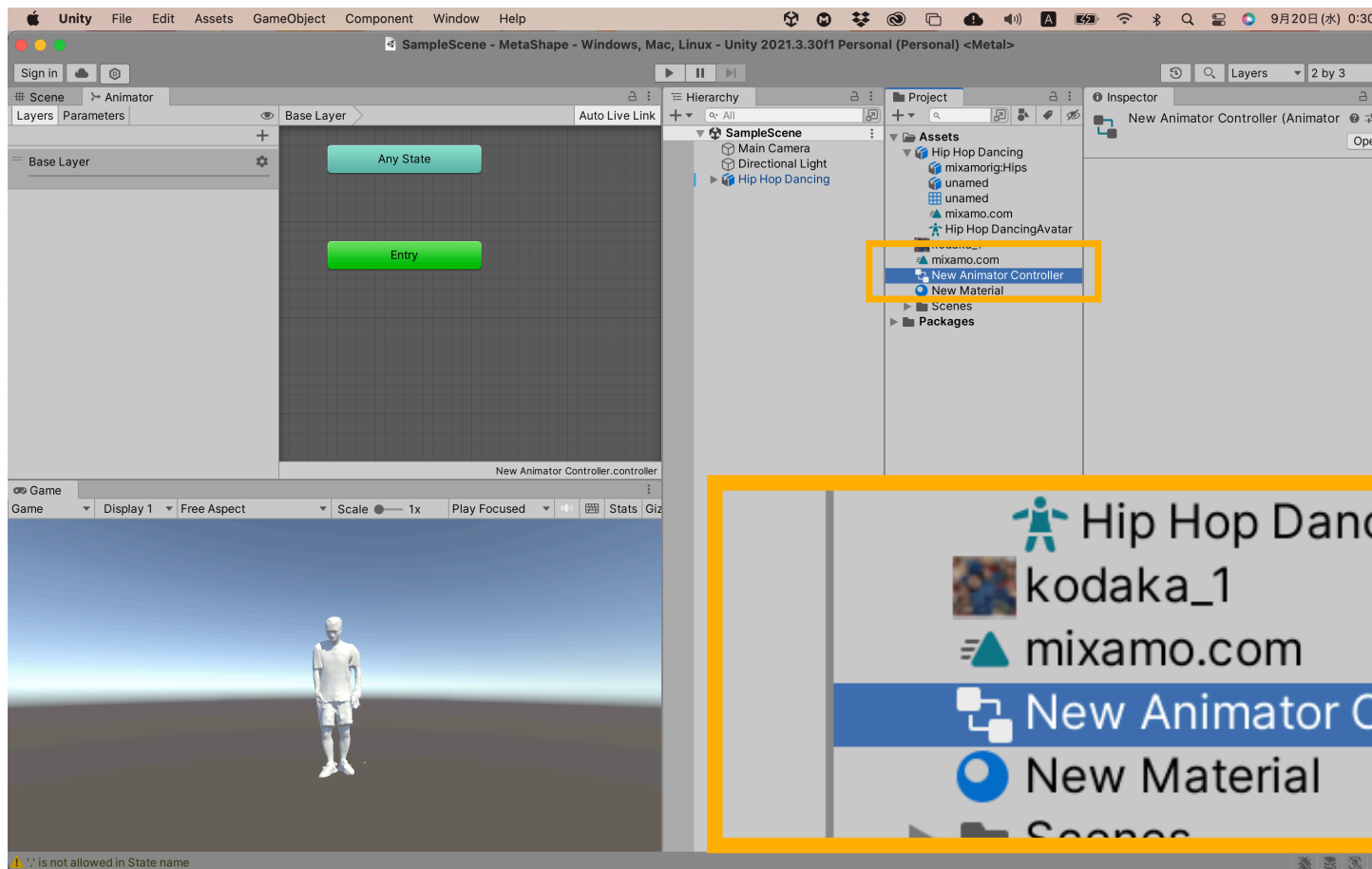
## 4. Unity編

FBXに格納されているアニメーションはそのままだと編集できないのですが、複製したアニメーションは編集できるのでループをONにします



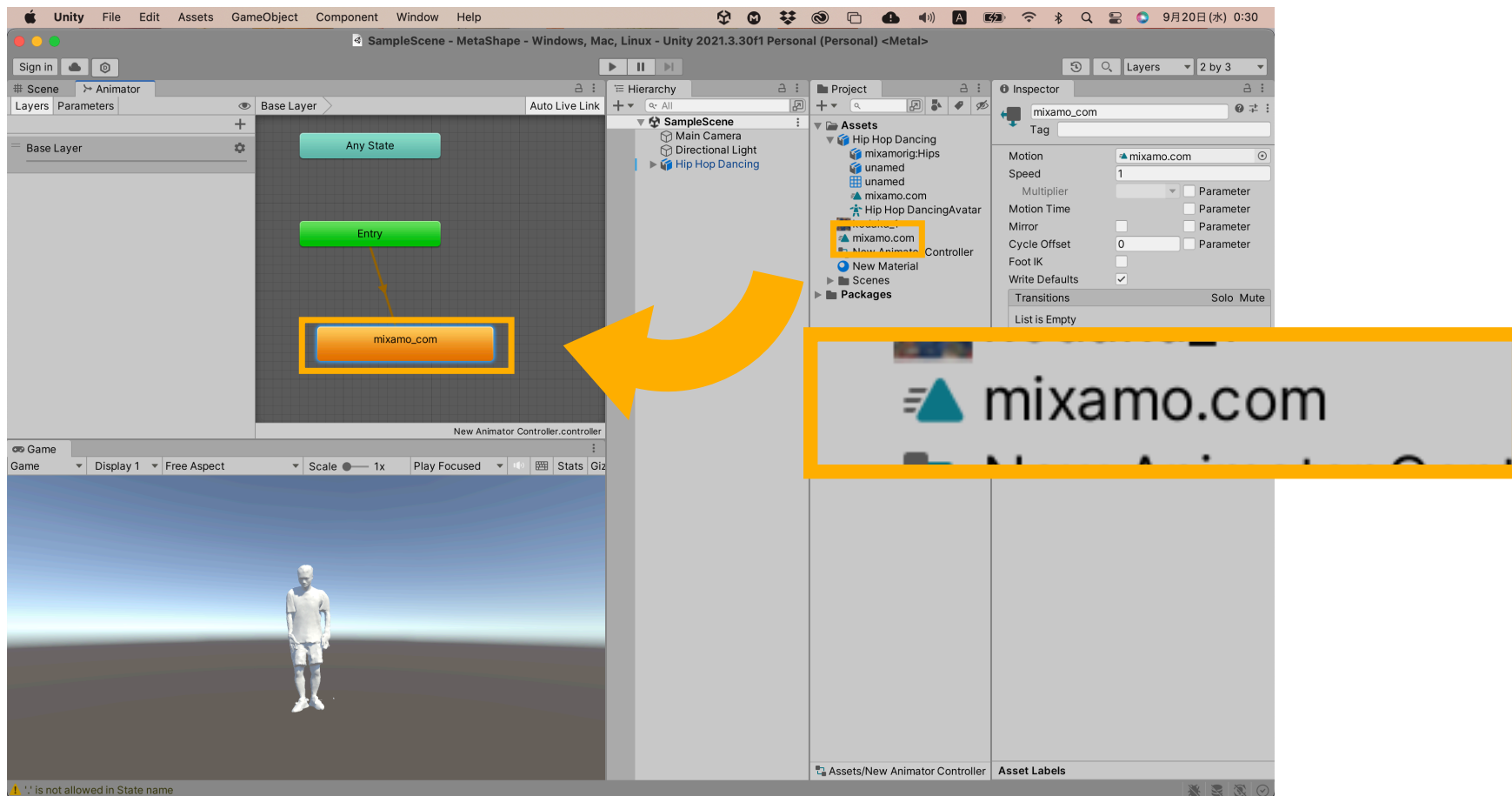
## 4. Unity編

先ほど生成したProject/Assets/New Animator ControllerをダブルクリックしてAnimatorウィンドウを開きます



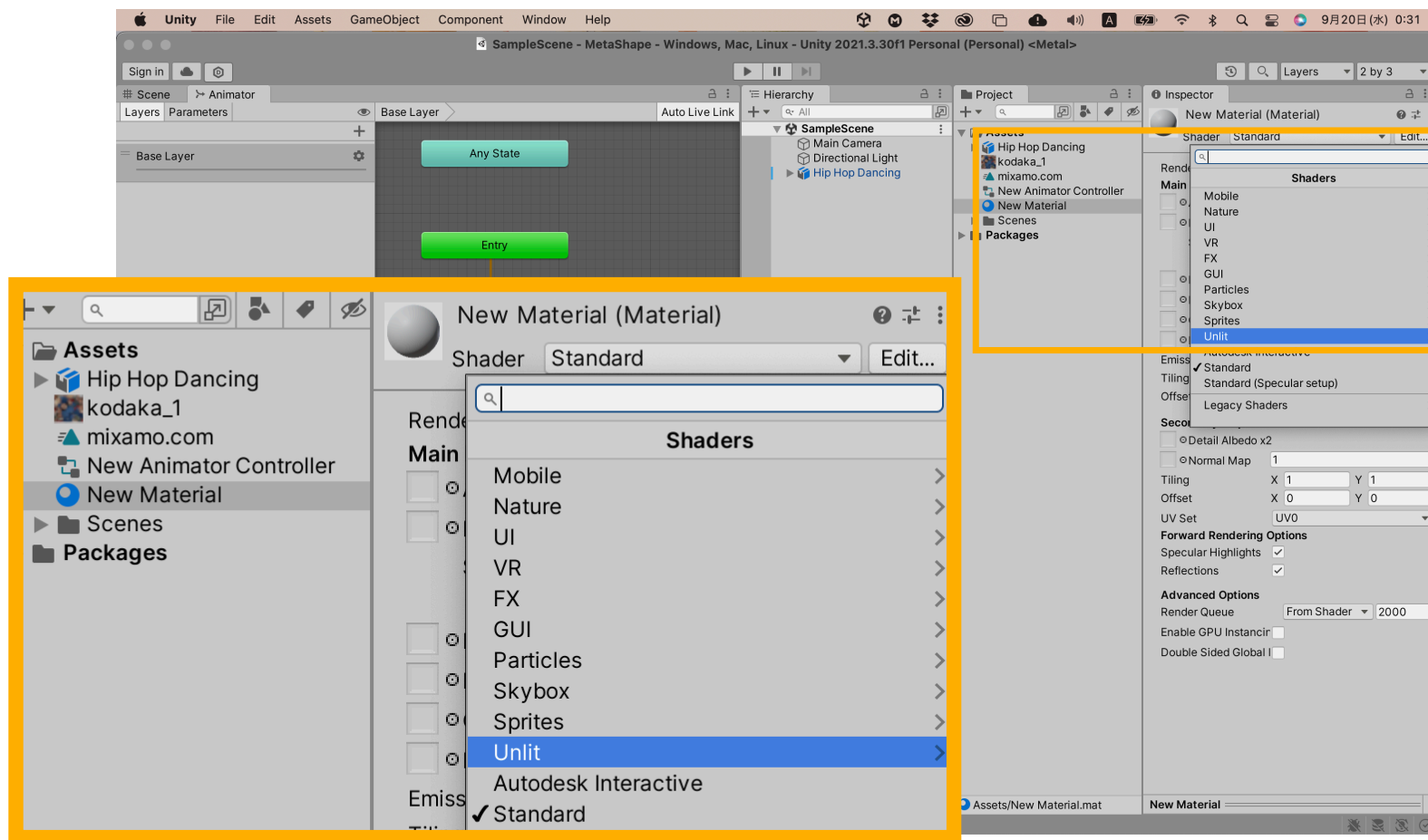
## 4. Unity編

先ほど生成したProject/Assets/mixamo.comを  
Animatorウィンドウにドラッグアンドドロップします



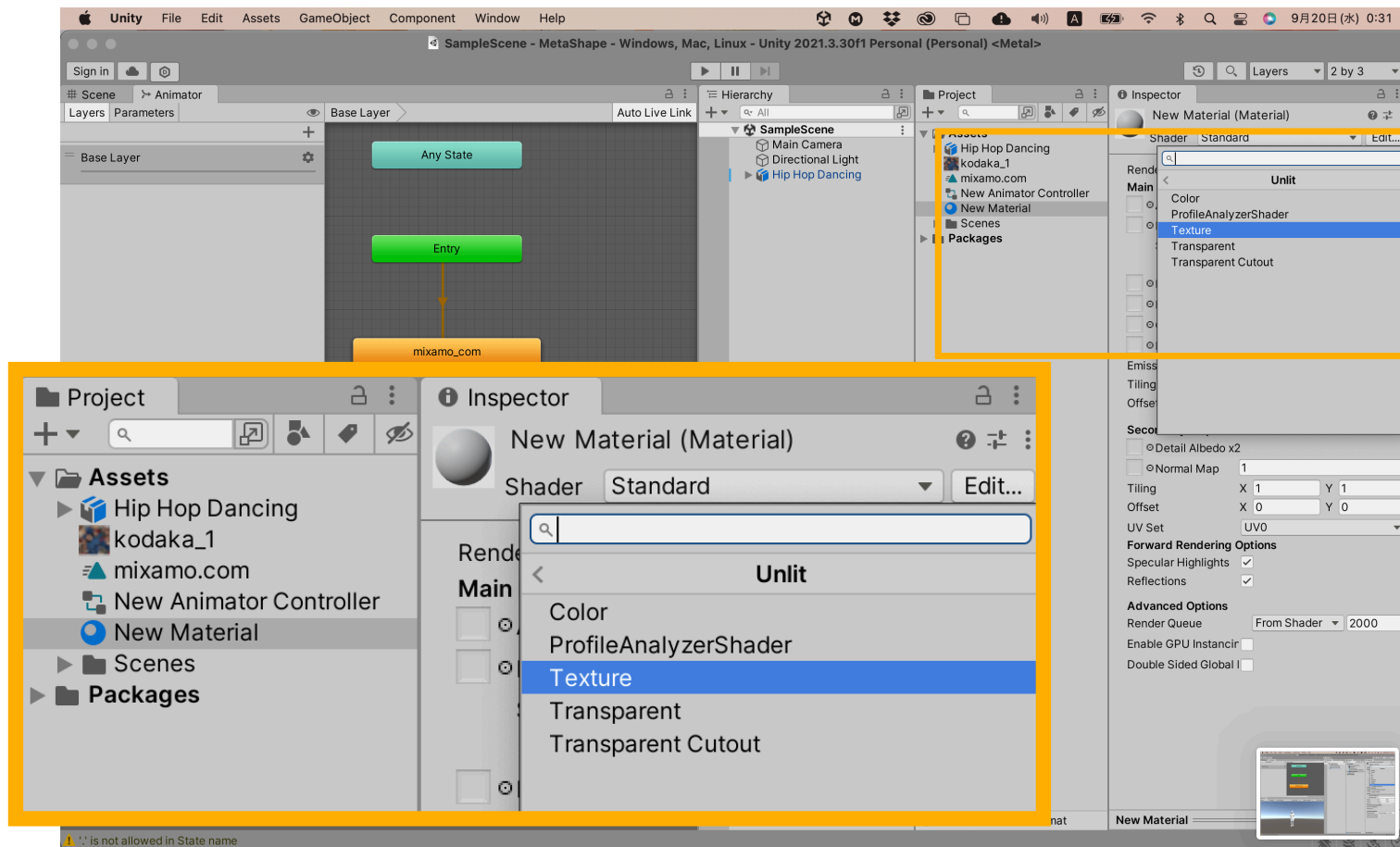
## 4. Unity編

先ほど生成したProject/Assets/New Materialを選択して  
ShaderからUnlit→



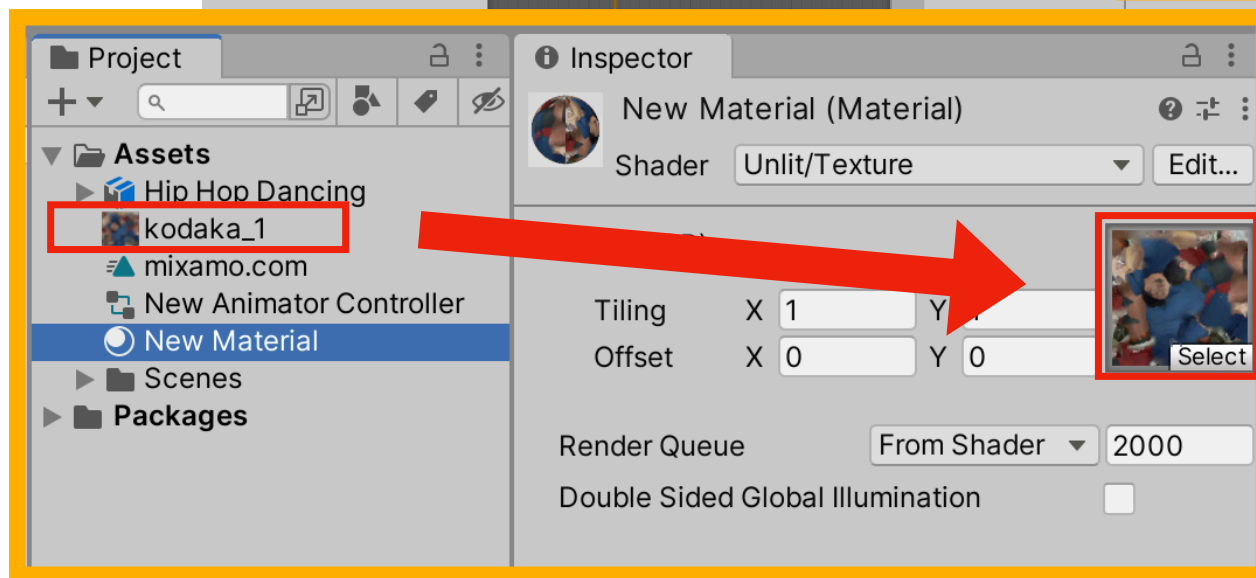
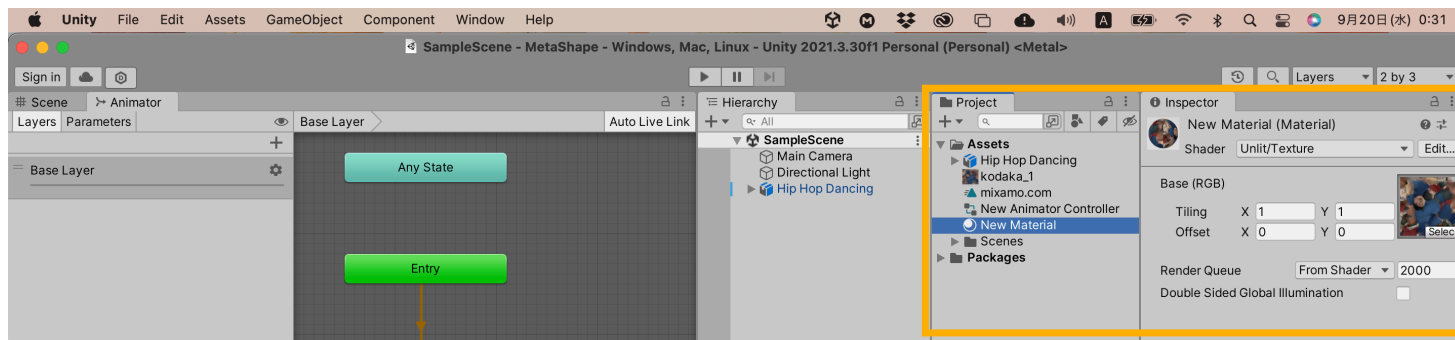
# 4. Unity編

ShaderからUnlit→Textureを選択  
(ライティングに影響されないマテリアル)



# 4. Unity編

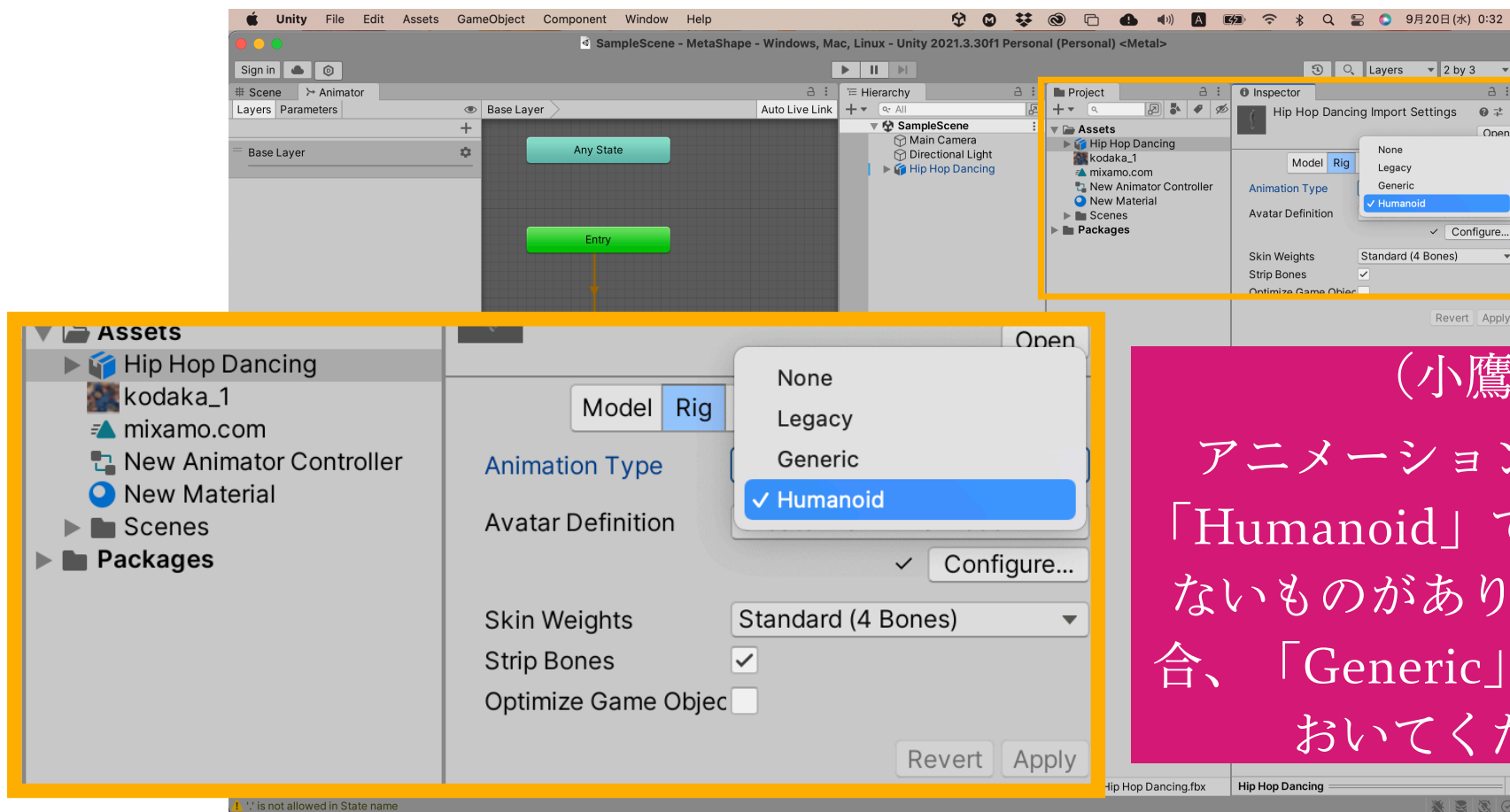
MetaShapeで生成したテクスチャをマテリアルに設定します





## 4. Unity編

Project/Assets/\*mixamoでDLしたfbxを選択し、  
右のInspectorのRigからAnimation TypeをHumanoidに設定します

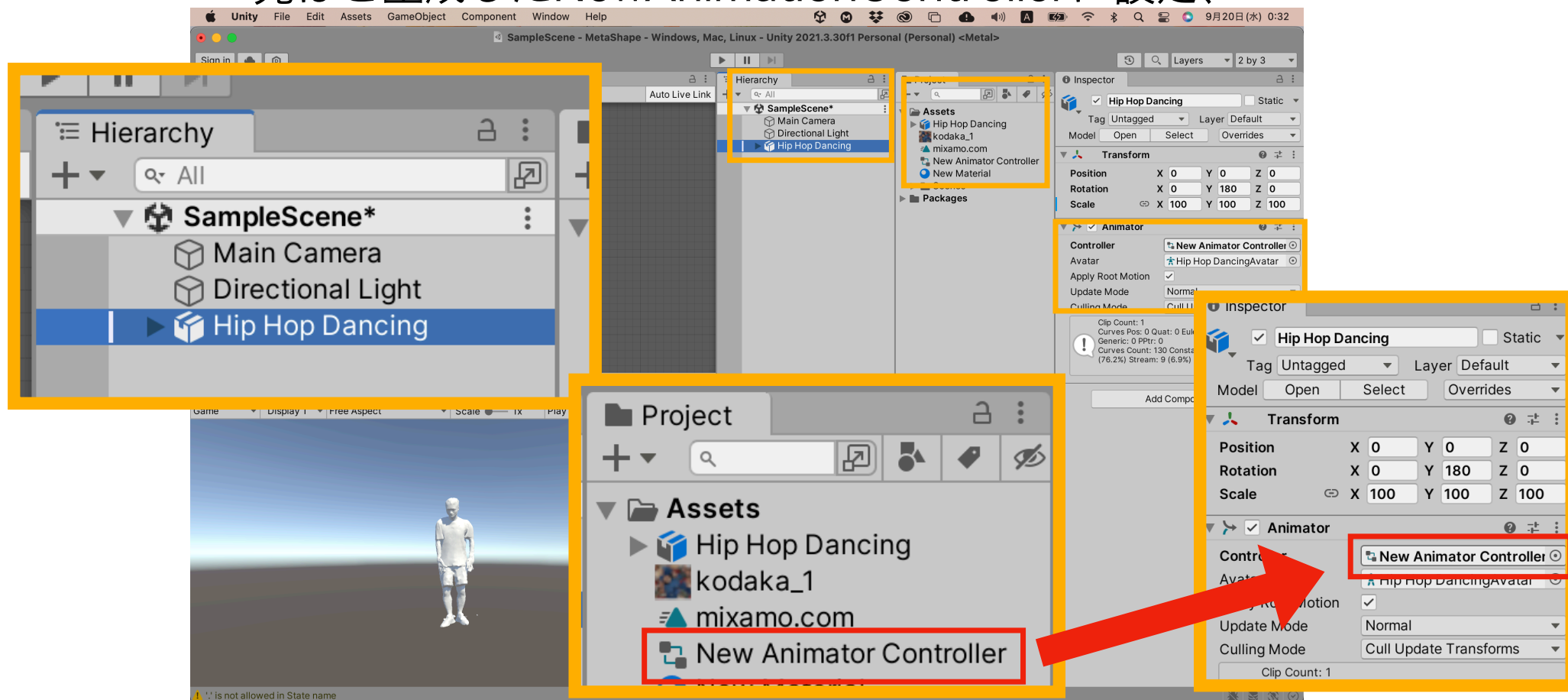


(小鷹注)

アニメーションによっては「Humanoid」ではうまく動かないものがあります。その場合、「Generic」のままにしておいてください。

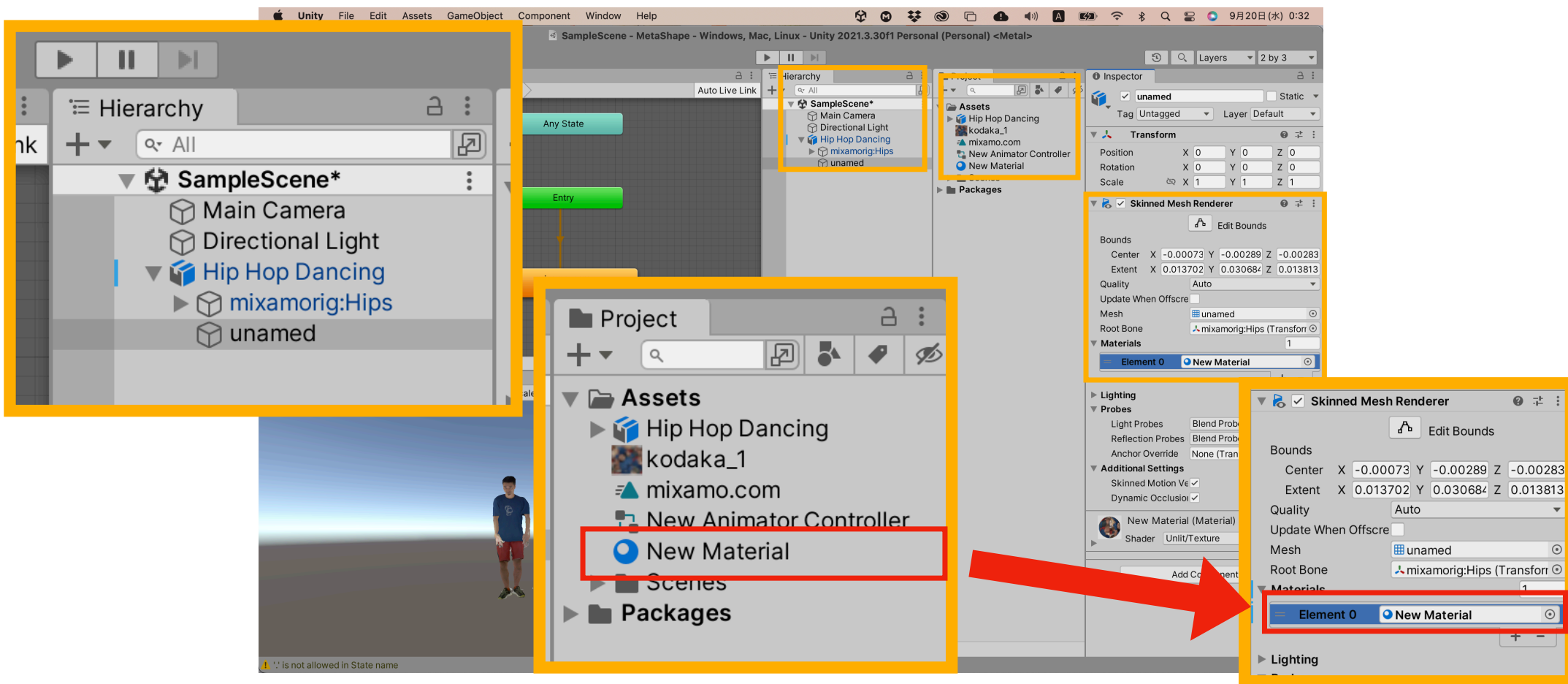
## 4. Unity編

最後にHierarchy/SampleScene/\*mixamoでDLしたfbxを選択し、  
右のInspectorのAnimatorからControllerを  
先ほど生成したNewAnimationControllerに設定、



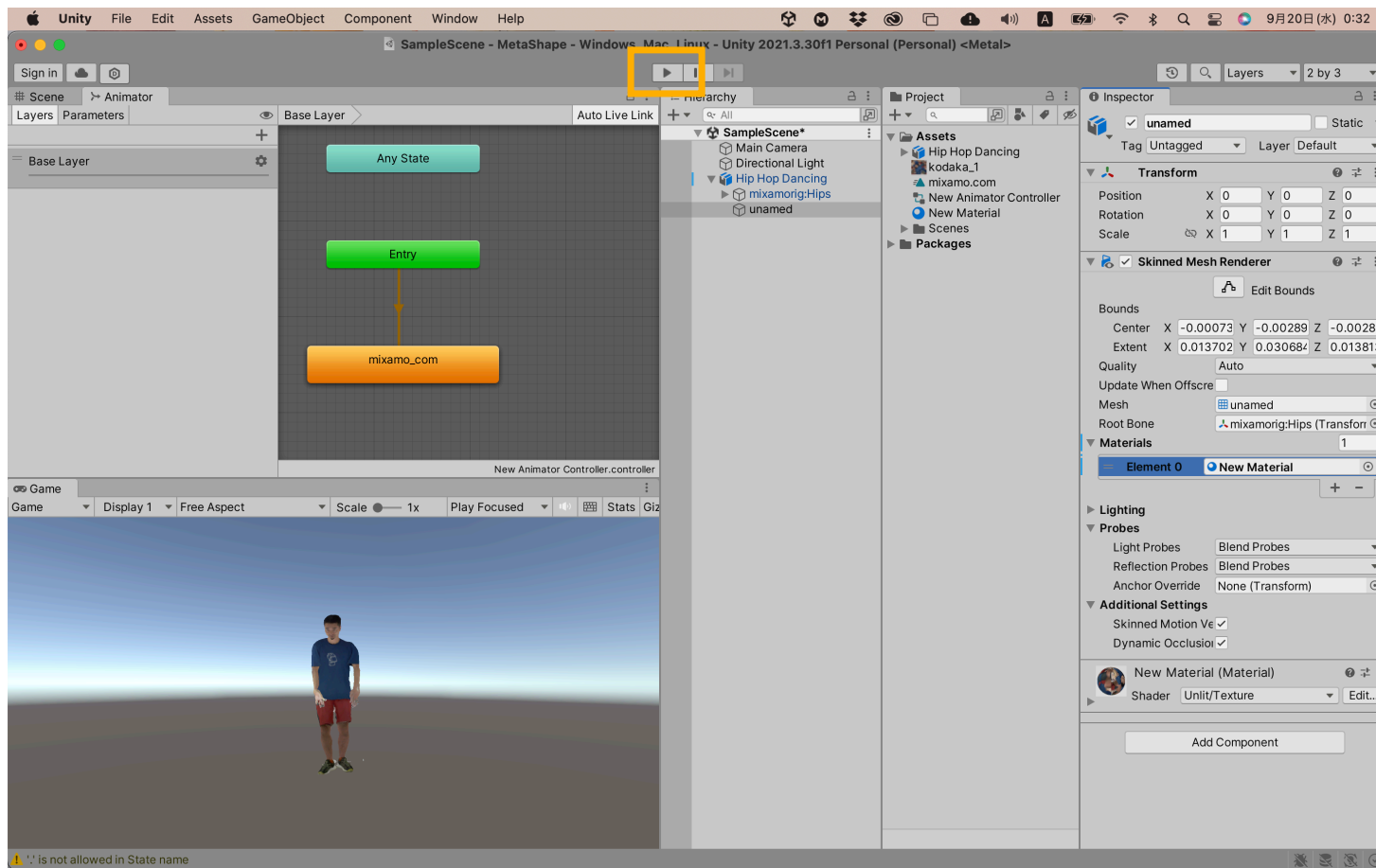
## 4. Unity編

Hierarchy/SampleScene/\*mixamoでDLしたfbx/unnamedを選択し  
右のInspectorのSkinnedMeshRendererのMaterialを生成したMaterialに設定して



# 4. Unity編

真ん中上にある再生ボタンを押すと…



## 4. Unity編

動きます

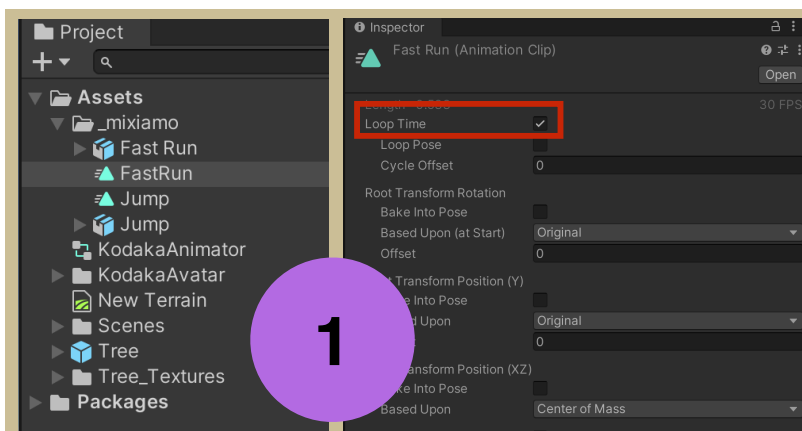


## 0. 補足

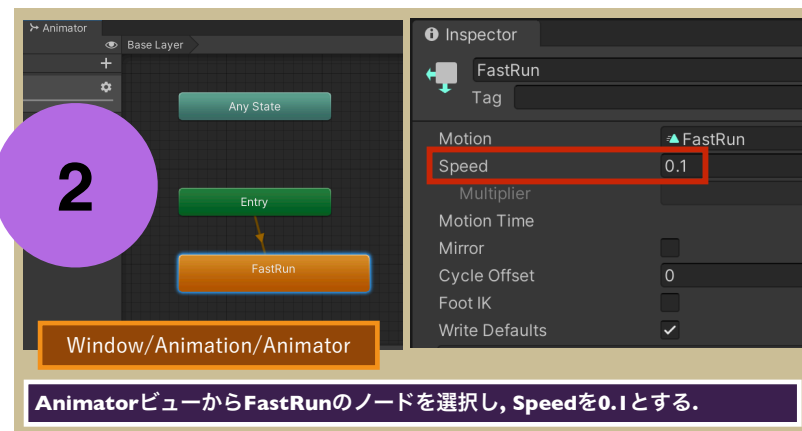
# MIXAMO→UNITY (補足1)

● アニメーションの実行

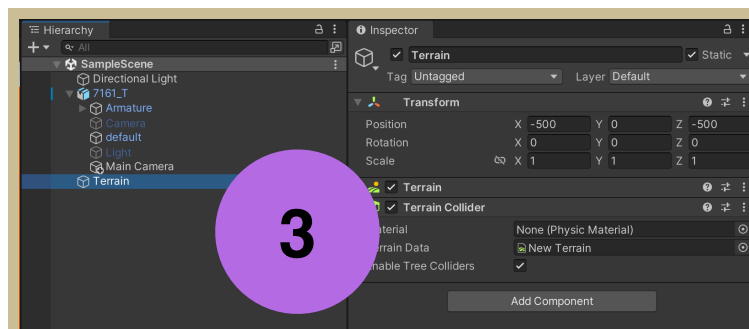
アバターが地面を走り続けるアニメーションを作成します。



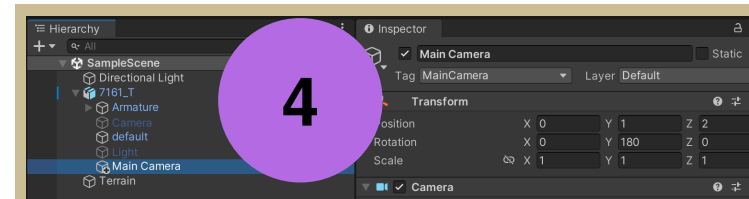
ProjectビューからFastRunを選択し、Loop Timeにチェックを入れる。



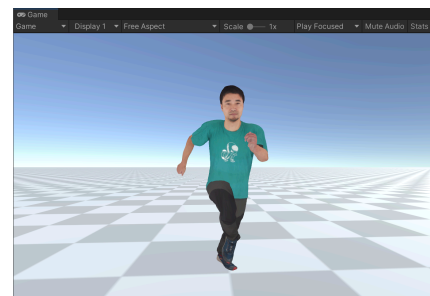
AnimatorビューからFastRunのノードを選択し、Speedを0.1とする。



シーンにTerrainを配置. 位置を (500,0,-500) とする。



カメラをアバターの子オブジェクトとし、位置を (0,1,2) ・方向を (0,180,0) とする。

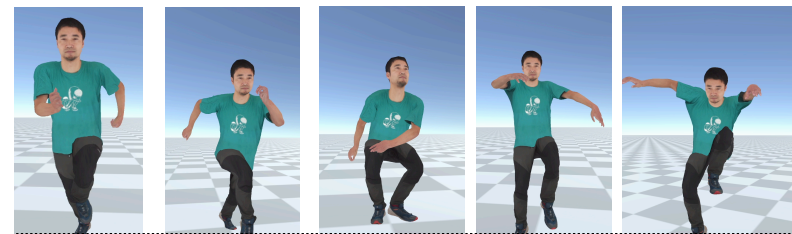
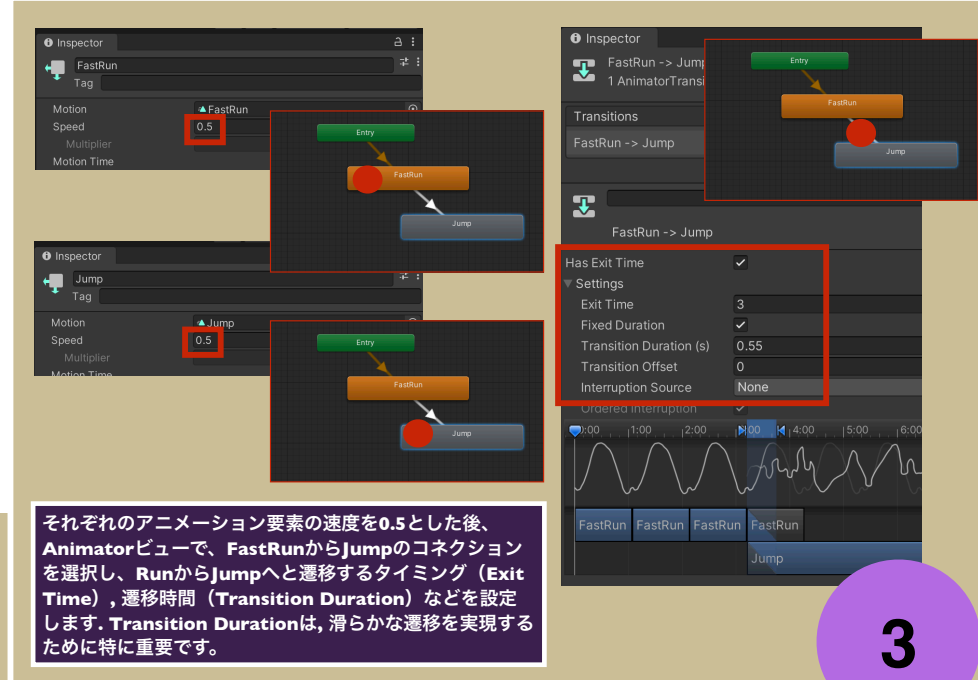
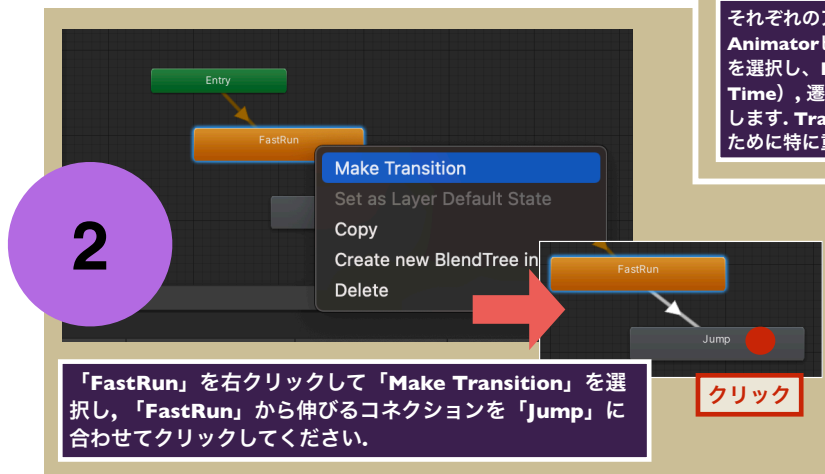
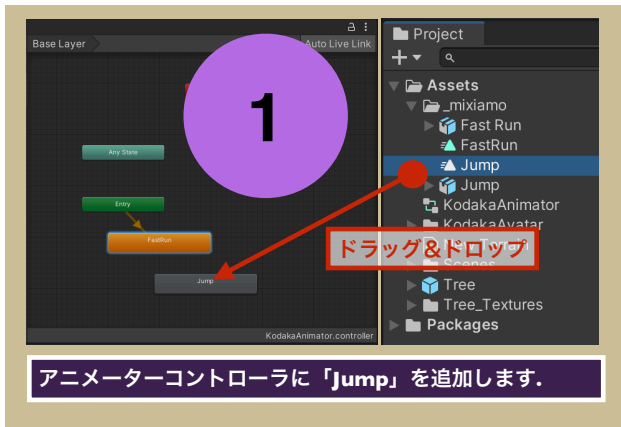


スローモーション気味に走り続けるアバターを正面から捉えるシーンが完成します。

# MIXAMO→UNITY (補足2)

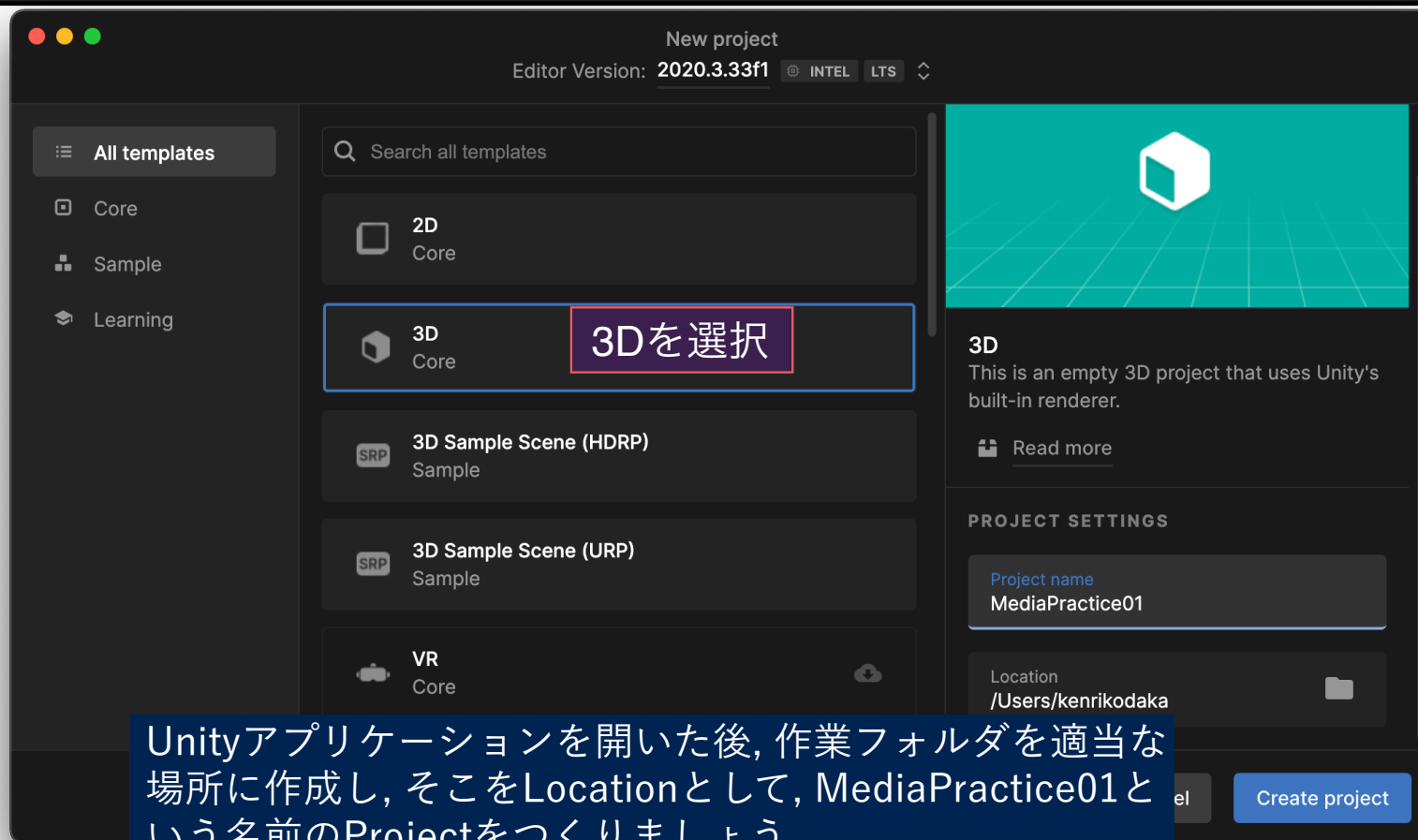
## ● 複数のアニメーションの接続

2つのアニメーション要素 (RUNとJUMP) を接続させます。

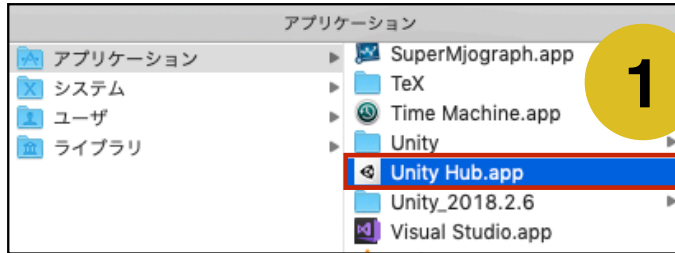




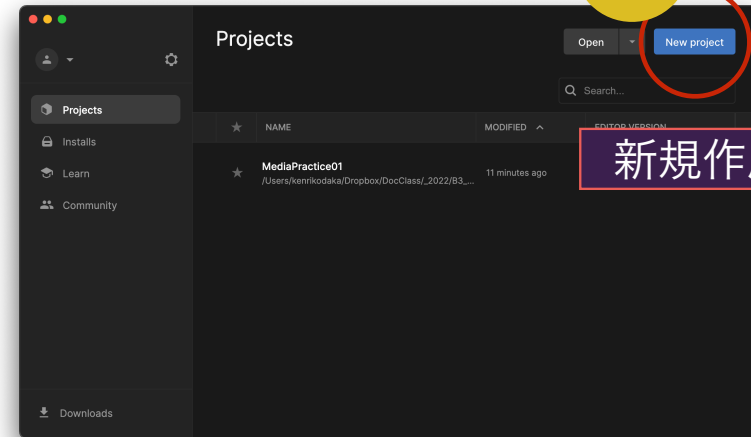
MediaPractice01

Unityとエディタの連携

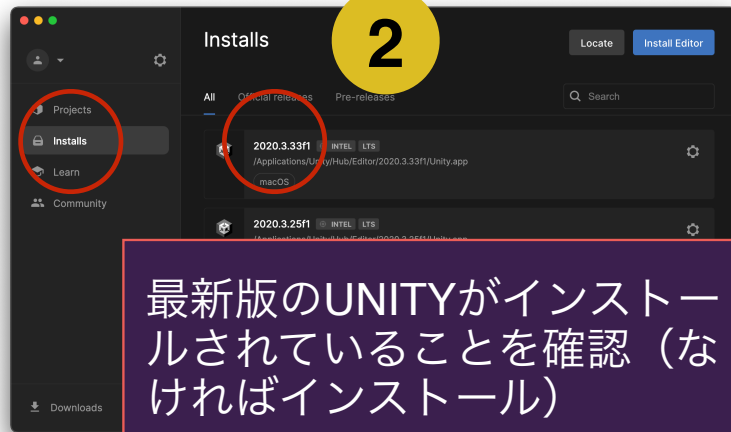
# 新規プロジェクトの作成 (MediaPractice01)



Unity Hubを起動

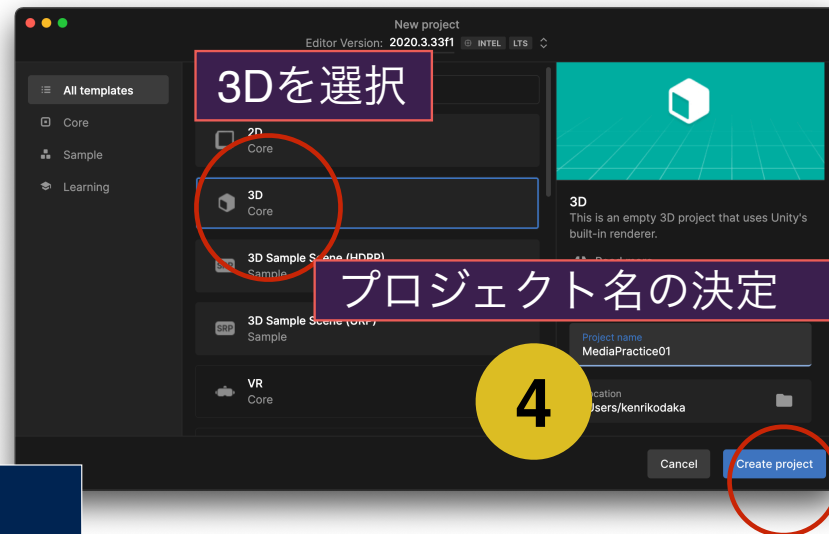


新規作成



最新版のUNITYがインストールされていることを確認 (なければインストール)

2022年4月時点の、最新の安定版は「2022.3.33f1」です。

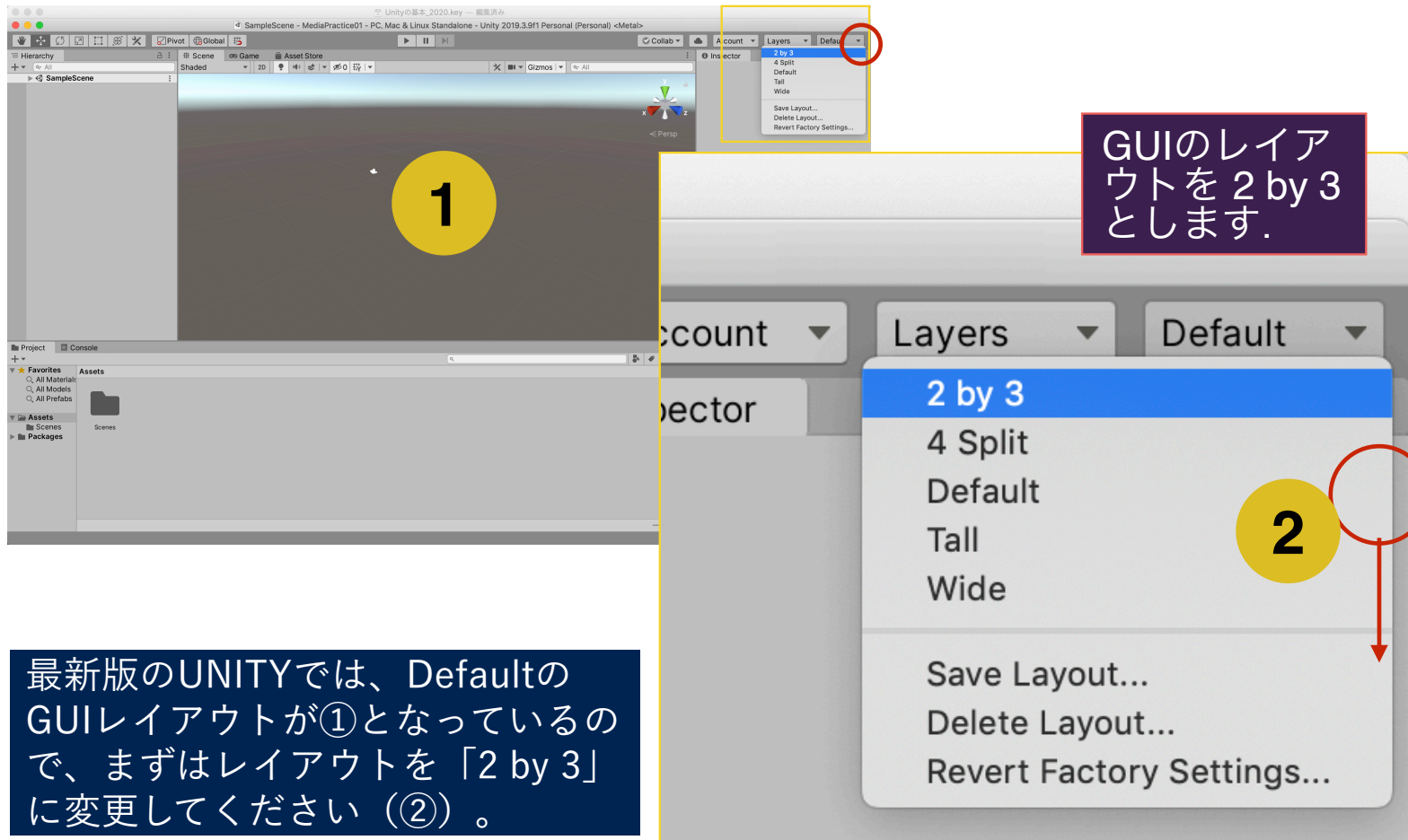


3Dを選択

プロジェクト名の決定

4

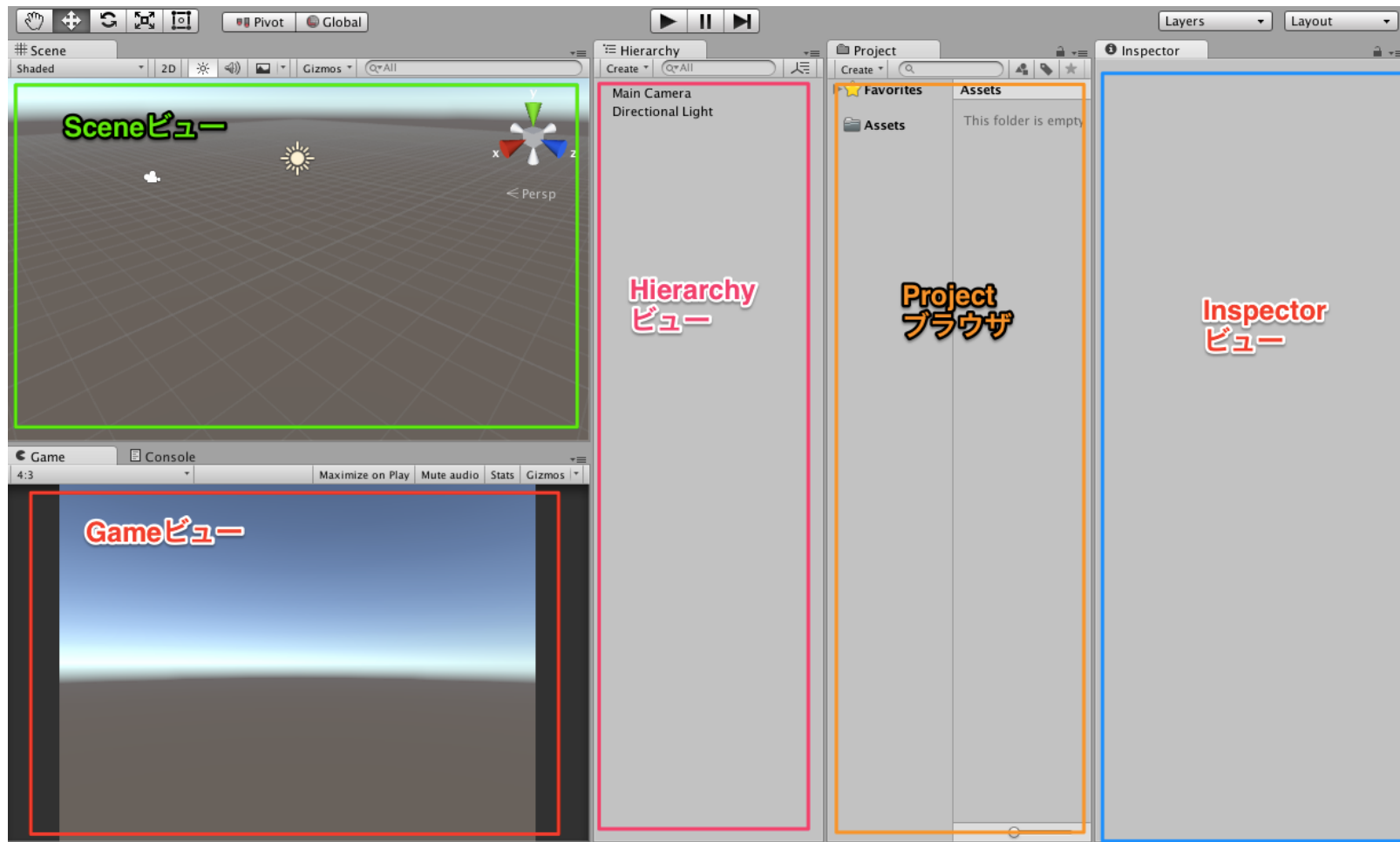
# GUIレイアウトの変更



最新版のUNITYでは、DefaultのGUIレイアウトが①となっているので、まずはレイアウトを「2 by 3」に変更してください(②)。

授業資料は、「2 by 3」のレイアウトを前提に進めていきます。

## GUIの構造・各パネルの呼び名 (2 by 3 レイアウト)



<https://github.com/unity3d-jp/FirstTutorial/wiki/0.プロジェクトの作成>

# スクリプト作成の準備

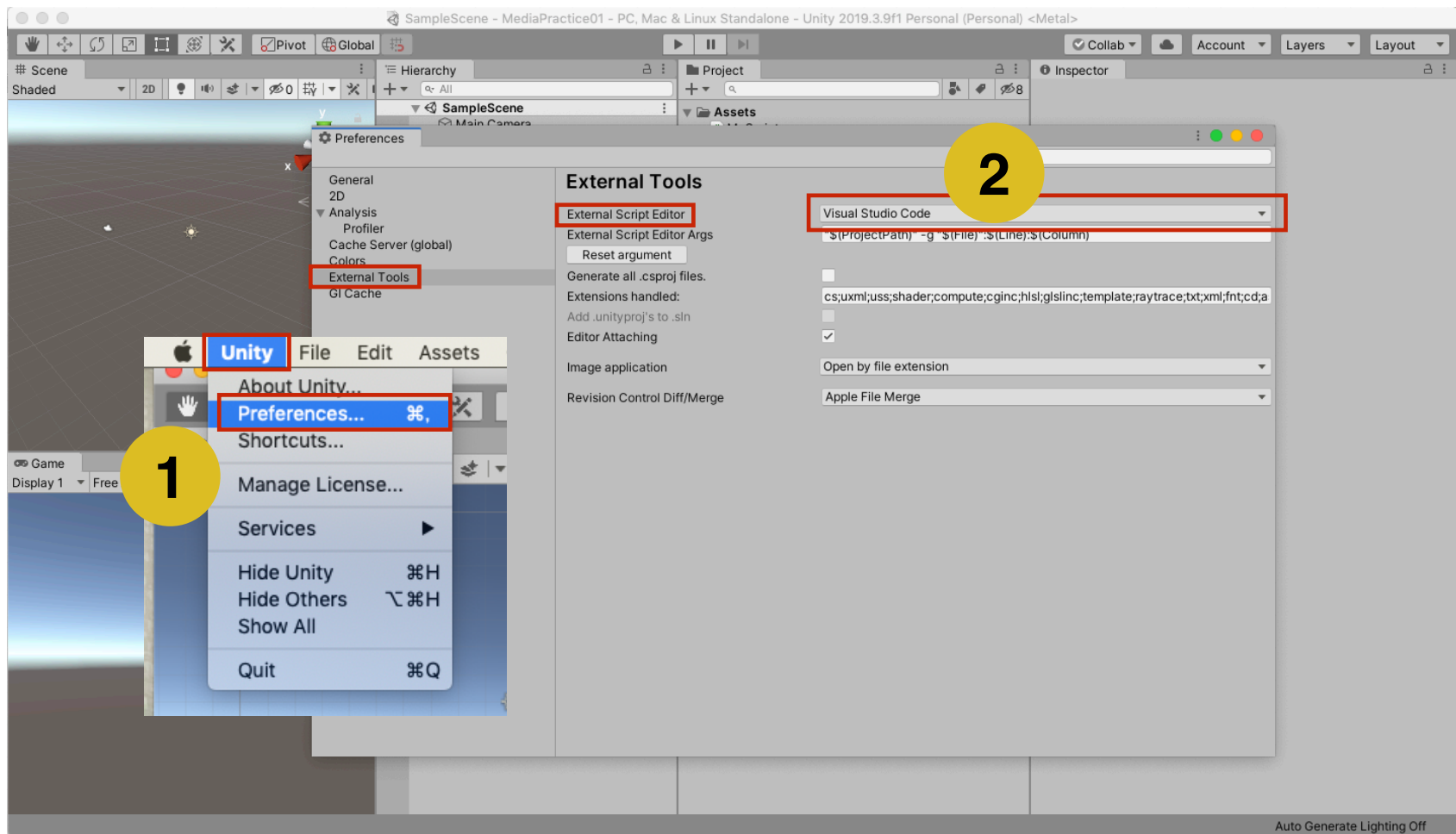
1 Projectブラウザのレイアウトが一行となっていることを確認してください。

2 C# Scriptのファイルを作成します。

3 Projectブラウザに、適当な名前のC#ファイルが表示されます。ここでは、ファイルの名前を「MyScript」に変更しましょう。

The image shows the Unity 2019.3.9f1 interface. The Project browser is set to 'One Column Layout'. A context menu is open over the Project browser, showing 'C# Script' as the selected option. The Inspector panel is also visible, showing the 'One Column Layout' option is checked. A new C# script named 'MyScript' is being created in the Project browser.

# 外部スクリプトエディタの設定



はじめに、外部スクリプトのエディタとして「Visual Studio Code」を設定しておきます。これをしないと、プログラム環境として別のテキストエディタが開きます。