

Blade Muskietier



■アセット名

Blade Musketier

■初めに

当アセットをダウンロードしていただき、まことにありがとうございます。
このキャラクターがあなた方のゲーム開発において、有意義なサンプルとして、
もしくは実際ゲームに使われるキャラクターとして活躍されることを心よりお祈りし
ております。

今後ともよろしく願いいたします！！:)

■ 3Dモデルの詳細。

about character :

High : 7376 Tris , 7327 verts , 69 Joint

Low 0 : 4954 Tris , 5447 verts , 67 Joint

Low 1 : 3564 Tris , 4428 verts , 53 Joint

SD Model : 3488 Tris , 4413 verts , 53 Joint

1のマテリアルと、1のテクスチャセットを使用します。 Texture size : 2048 x 2048.

about weapon :

High Model : 1497 Tris , 1850 verts , 4 Joint

Low 0 : 964 Tris , 1308 verts , 4 Joint

SD Model : 964 Tris , 1323 verts , no skinned

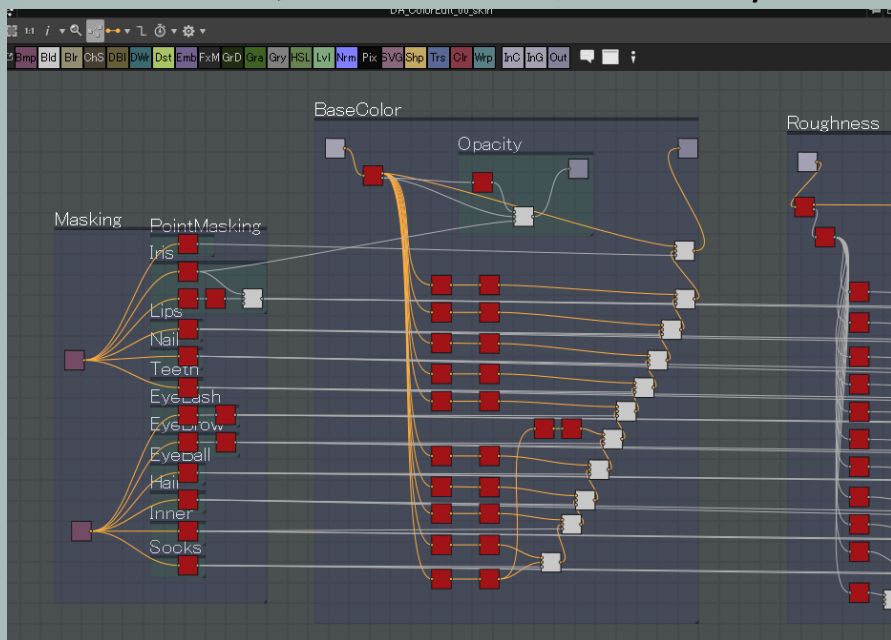
1のマテリアルと、1のテクスチャセットを使用します。 Texture size : 1024 x 256.

■ プロシージャリアルマテリアル

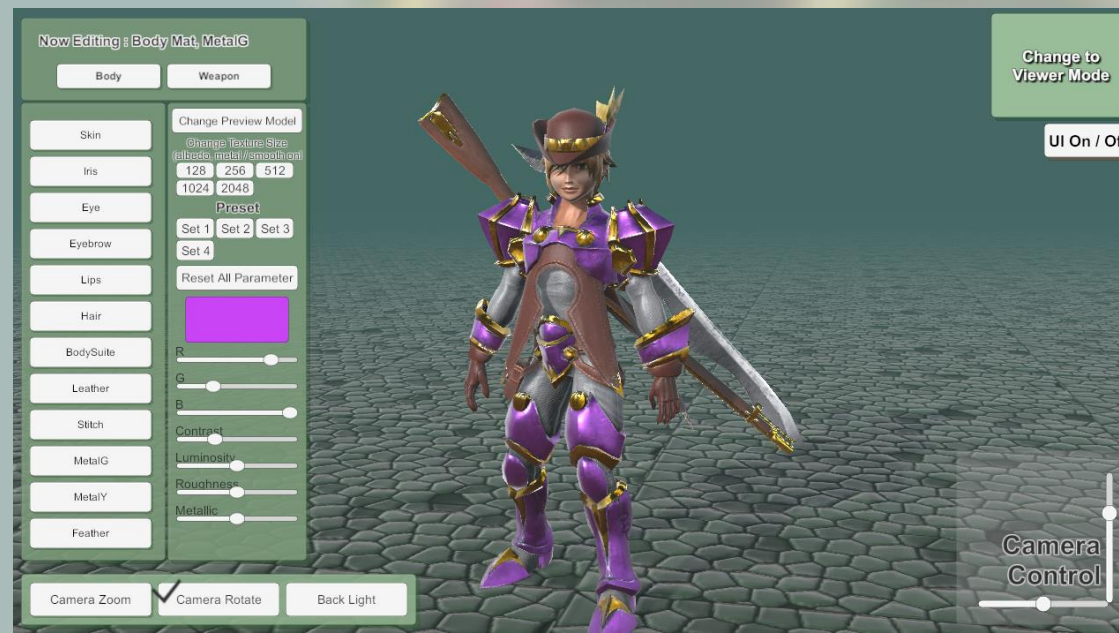
当アセットにはSubstance Designerによって作成された、プロシージャリアルマテリアルが含まれています。

プロシージャリアルマテリアルにはカラーエディットのためのパラメータが設置されており、各パラメータを調整することで、自分だけのカラーに変更することができます。

(当アセットの場合はAlbeo、Metallic/Smoothnessマップに影響します。)



In Substance Designer.

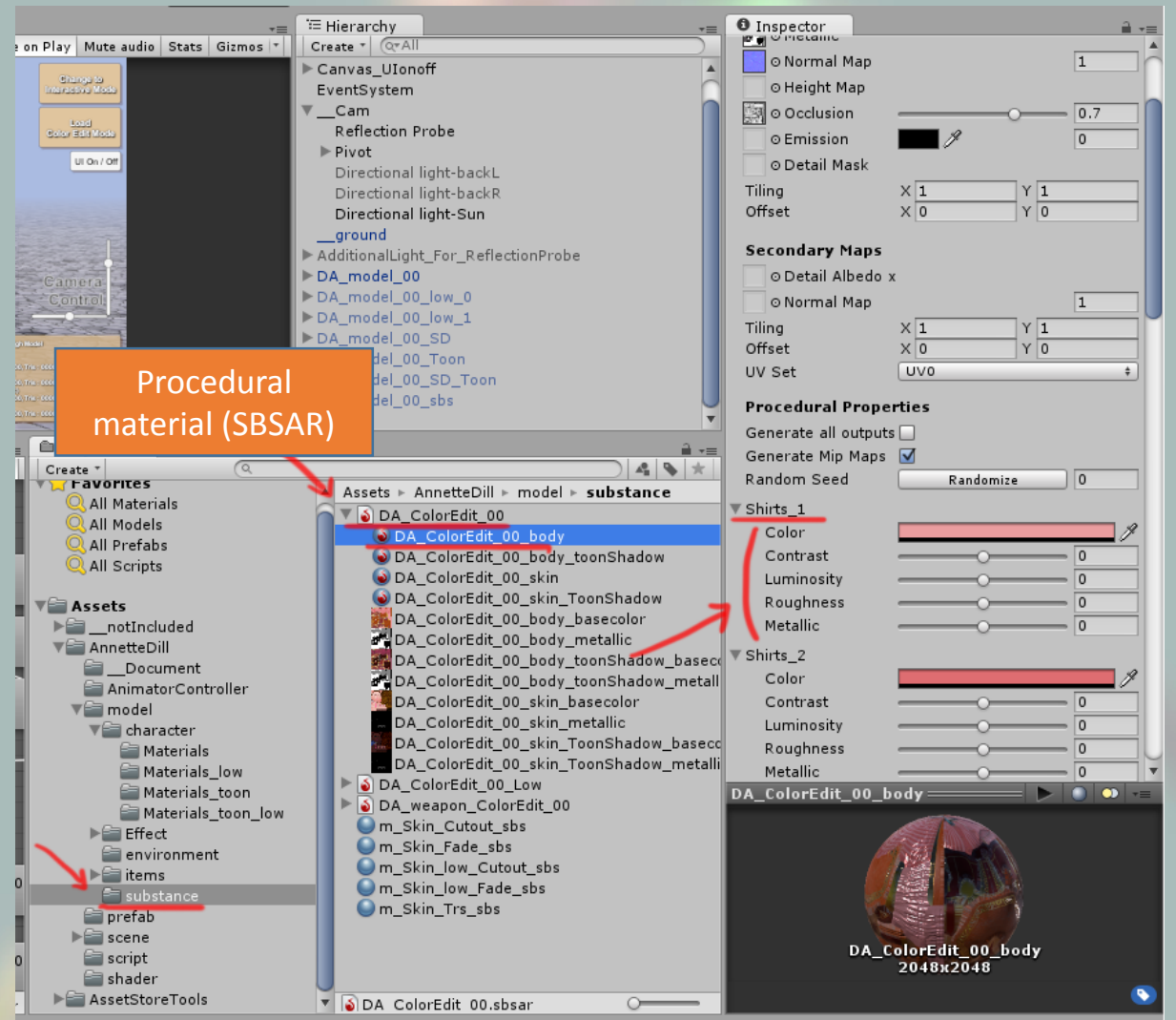
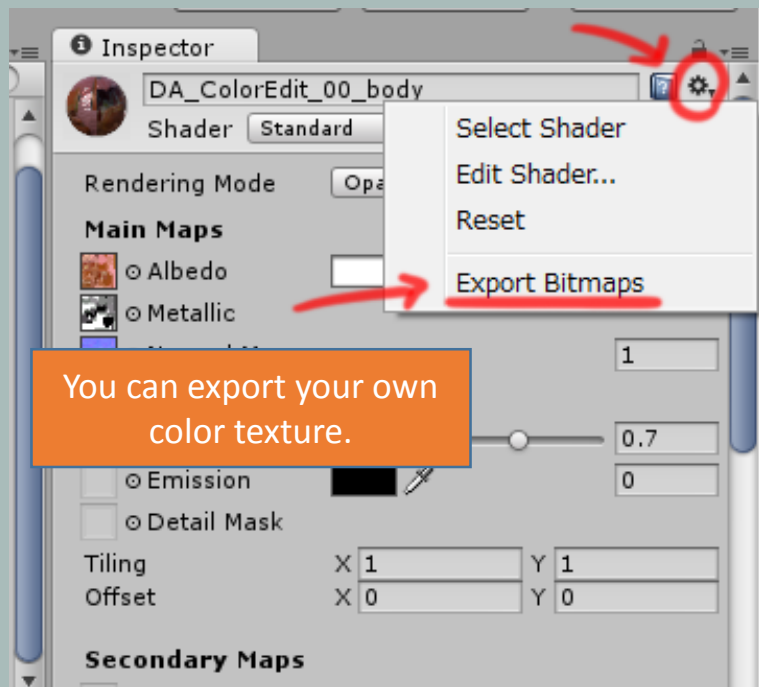


Real time Color Edit

■ プロシージャルマテリアル

プロシージャルマテリアルはリアルタイムでも動作しますが、エディター上で予めパラメータを変更し、テクスチャを用意しておくこともできます。

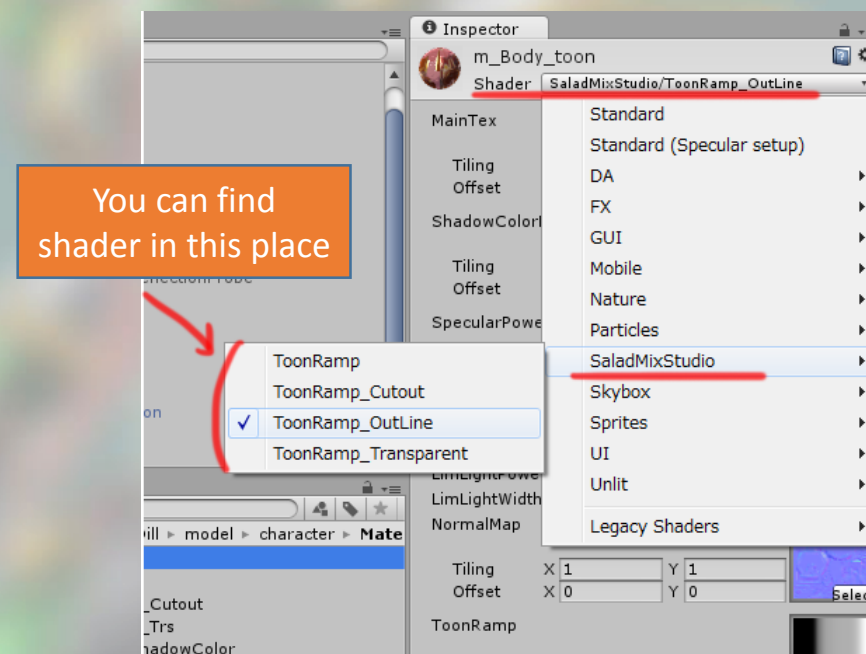
生成されるテクスチャは、Export Bitmapsメニューで画像ファイルとして保存し、通常のマテリアルで使用することもできます。



■カスタムトゥーンシェーダー

当アセットにはカスタムトゥーンシェーダーが含まれていて、マテリアルのshader選択メニューの [SaladMixStudio/ToonRamp_????]のところに追加されます。

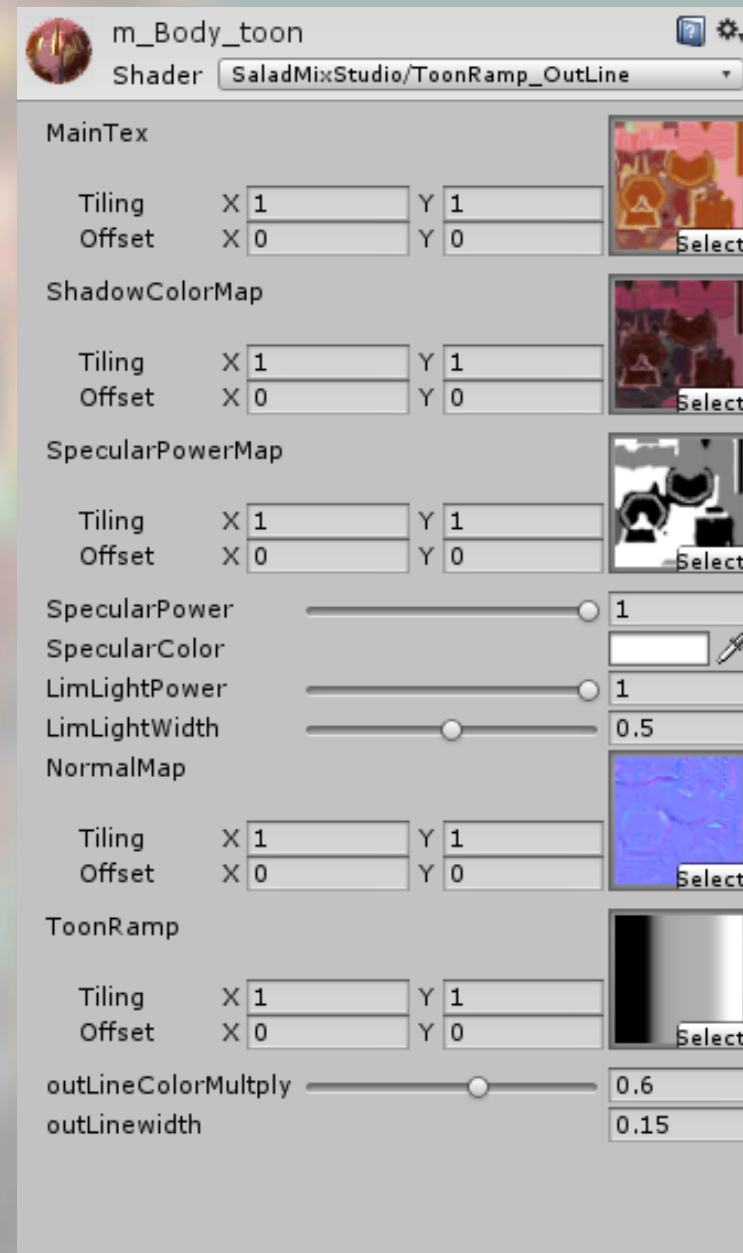
Shader Forgeで作成されたシェーダーですので、もしShader Forgeをお持ちでしたらコードを書かずとも、ご自身でカスタマイズすることもできます。



■カスタムトゥーンシェーダー

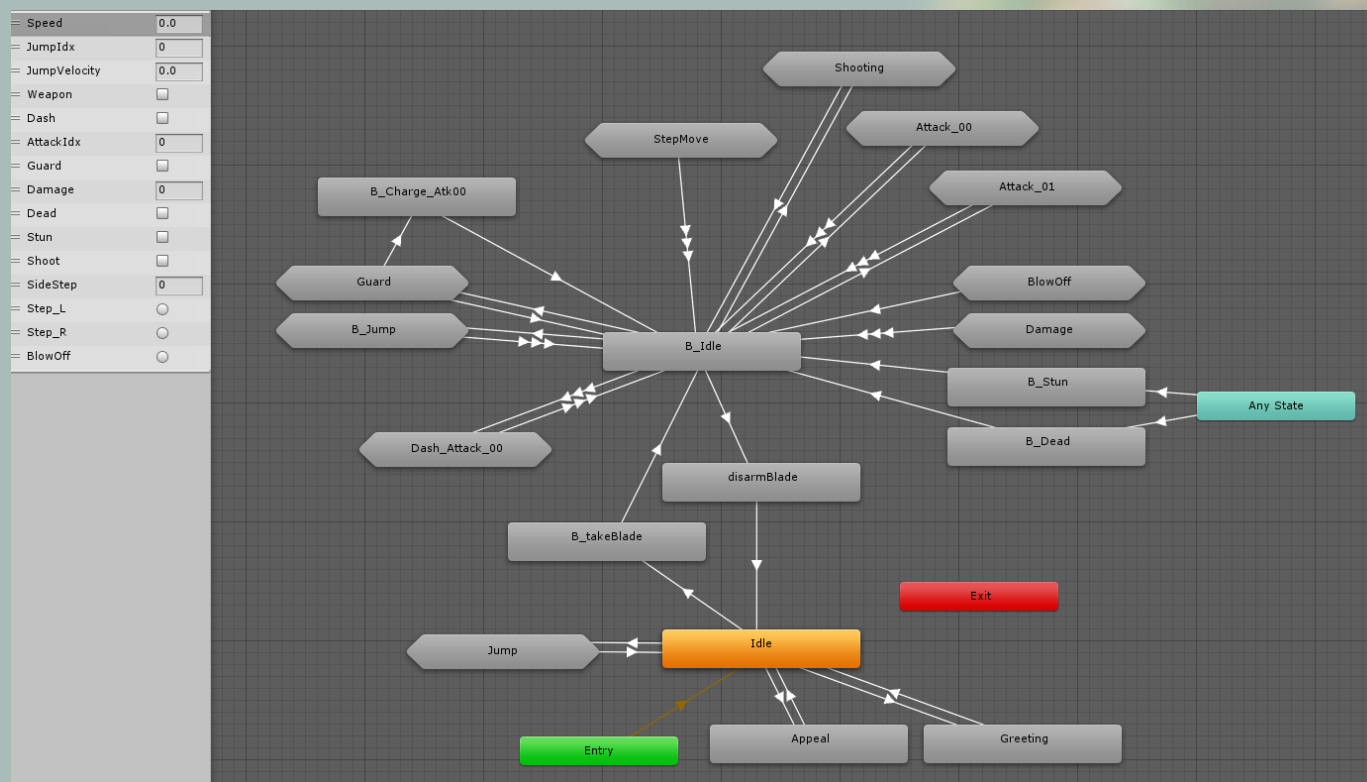
Toon Shaderの設定。

- MainTexはいつものカラーテクスチャです。
- Shadow Color Mapは影色を設定するマップとなります。
影になる領域にShadow Color MapとMainTexがブレンドされます。
- Specular Power MapはSpecularMaskやGlossinessMapとして機能します。
- Toon ramp mapは影の掛り具合を調整します。
完全黒のところはShadowColorMapの色になり、黒からグレイへのグラデーションでは、Shadow Color MapとMainTexがブレンドされます。
グレイ(50%)ではMainTex、グレイから白へいくとSpecularの計算によりMainTexを明るくしていきます。
このグラデーションの具合を調整することでシェーディングの具合を調整できます。
- OutLineColorMultiplyはアウトラインの色を調整します。
OutLineColorMultiplyが0なら黒、1ならShadowColorMapの色になります。



■ モーション

MECANIM(ヒューマノイド)対応のモーションが40種類ほど、
武器用のアニメーション(トリガーを引く)、
制御サンプルのanimator controllerが付属しています。



■ デモシーンに関して

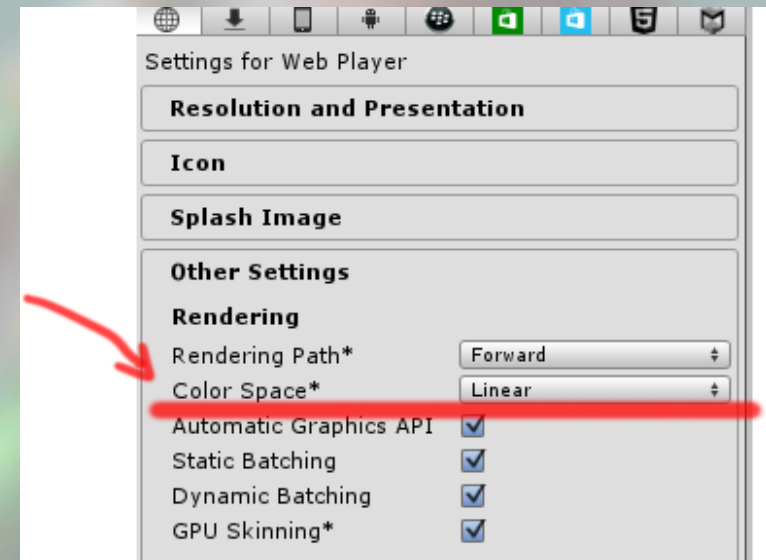
Viewer mode - アニメーションを一つずつ再生しながら確認

Interactive mode - ゲームのようにキーボードでキャラクターを操作

ColorEdit mode - プロシージャルマテリアルを利用し、テクスチャカラーの変更を行う。ここで変更したカラーはビューアモードに戻ってプレイに使う事も可能です。

アニメーターコントローラはViewerMode, InteractiveModeの2種類があります。

すべてのスクリーンショットやデモシーンなどは Linear Space上で撮影、稼働されたものです。Unityの基本設定であるGamma Spaceでは若干色合いが変わることに注意してください。



■デモシーンに関して

デモシーンの制御用に、いくつかのモーションにはイベントが設定されています。

ジャンプ時 : (MbladeAnimatorControl.cs > SetJump())
jump_00_up, jump_01, B_jump_00_up, B_jump_01

吹き飛ばされる時 : (MbladeAnimatorControl.cs > SetBlowOff())
B_BlowStart

武器を持ち変える時 : (MbladeAnimatorControl.cs > AttachWeapon(int idx))
B_takeBlade , DisarmBlade

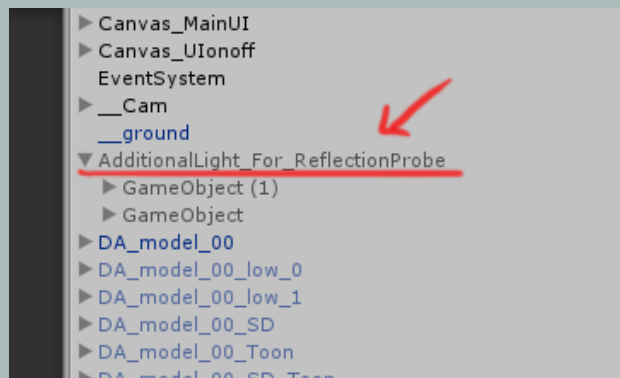
トリガーを引く、銃口から火を噴くタイミング : (MbladeAnimatorControl.cs > PullTrigger(), SetFire())
B_Atk00_03 , B_Shoot_00 , B_Charge_Atk_00 , B_Dash_Atk00_02

■ デモシーンに関して

ライティングに関して

デモシーンを開くと、ヒエラルキーから[AdditionalLight_For_ReflectionProbe]という名前のオブジェクトを発見できます。このオブジェクトグループは、standardマテリアルのEmissionを高くしてライト扱いにしたシンプルなものです。このアセットには環境光用のIBLが用意されていなく、ライティング環境が乏しいので、このライトを利用しました。普通の使い方ではないかもしれませんが、ライティングをビルドする際に、このグループをONにしてビルドしてください。Reflection Probeに光がベイクされますので、キャラクターの反射に影響するようになります。プレイ中は邪魔なのでOFFにします。

turn on [AdditionalLight_For_ReflectionProbe] > lighting build > turn off [AdditionalLight_For_ReflectionProbe] > playing



キャラクターの移動に関して

基本ルートモーションにて動いていますが、ジャンプ制御の時はスクリプト制御で移動させています。

■ サポート

質問やコメントなどがありましたら以下のところまで。
日本語でのサポートいたしますので遠慮なくお願いいたします😊

E-mail : trhunter@naver.com

twitter : https://twitter.com/0z_TM