

## 目次

|  |    |
|--|----|
| 1. 内容物紹介 .....   | 2  |
| 2. K2Engine セットアップ手順 .....                               | 3  |
| 2.1 「右クリックで管理者として実行してね.bat」を実行 .....                     | 3  |
| 2.2 GameTemplate/tools の「右クリックで管理者として実行してね.bat」を実行 ..... | 4  |
| 2.3 GameTemplate をコピーして制作を開始 .....                       | 7  |
| 3. 3D モデル出力 .....  | 8  |
| 3.1 tkExporter を起動 .....                                 | 9  |
| 3.2 「tkm ファイル」を出力 .....                                  | 12 |
| 4. 画像表示 .....  | 13 |
| 4.1 dds ファイル .....                                       | 13 |
| 5. K2Engine 用 3D モデルの max マテリアルの設定 .....                 | 14 |
| 5.1 マテリアルの種類を「DirectX Shader」に設定 .....                   | 14 |
| 5.2 マテリアルの DirectX シェーダに「k2EngineShader.fx」を設定 .....     | 15 |
| 5.3 マテリアルにテクスチャを設定 .....                                 | 16 |
| 5.3.1 User Normal Map .....                              | 17 |
| 5.3.2 Albedo .....                                       | 17 |
| 5.3.3 Normal .....                                       | 17 |
| 5.3.4 Metallic+Smooth .....                              | 17 |
| 5.3.5 Metallic(金属度) .....                                | 18 |
| 5.3.6 Smooth(滑らかさ) .....                                 | 19 |
| 6. 3D モデルのアニメーションの出力 .....                               | 20 |
| 6.1 3D モデルにボーンとアニメーションを設定 .....                          | 20 |
| 6.2 tkExporter を起動し、出力するアニメーションのフレームを設定 .....            | 20 |
| 7. Mixamo の紹介 .....                                      | 22 |
| 7.1 モデルのダウンロード .....                                     | 22 |
| 7.2 アニメーションの適用とダウンロード .....                              | 25 |
| 8. ナビゲーションメッシュと A* アルゴリズム .....                          | 29 |
| 8.1 ナビゲーションメッシュ、A* アルゴリズムとは？ .....                       | 29 |
| 8.2 tkn ファイル出力 .....                                     | 30 |
| 8.3 出力した tkn ファイルを使用して、移動処理を実装する .....                   | 32 |

# 1. 内容物紹介

K2Engine のファイル、フォルダ構成について説明する

## GameTemplate

→これをコピーしてゲーム制作を行う

## Sample

→k2Engine の様々な機能を紹介している  
詳しくは Sample の readme.txt を参照

## GameTemplate/tools

→ゲーム制作の際に必要な、様々なツール類が入っている

### Effekseer160Win

エフェクト制作ツール

### K2SLEditor

画像の配置情報の出力、画像を dds ファイルに変換できる  
k2SLEditor.exe で起動

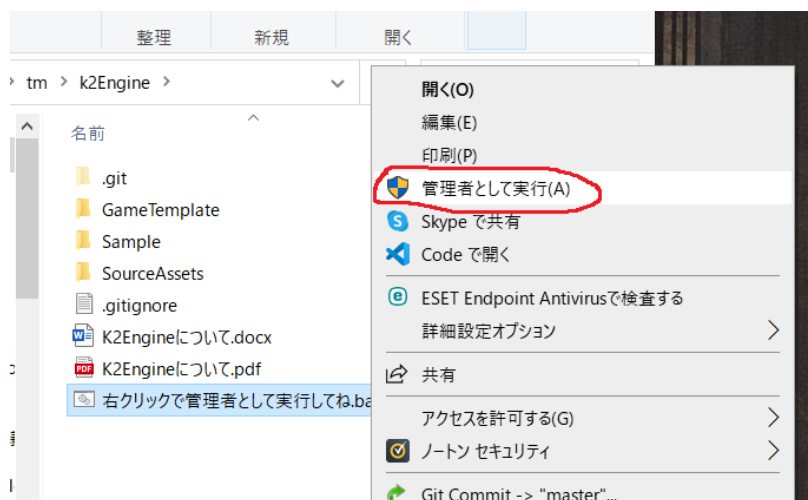
## 2. K2Engine セットアップ手順

K2Engine での、ゲーム制作開始までの手順を説明する

### 2.1 「右クリックで管理者として実行してね.bat」を実行

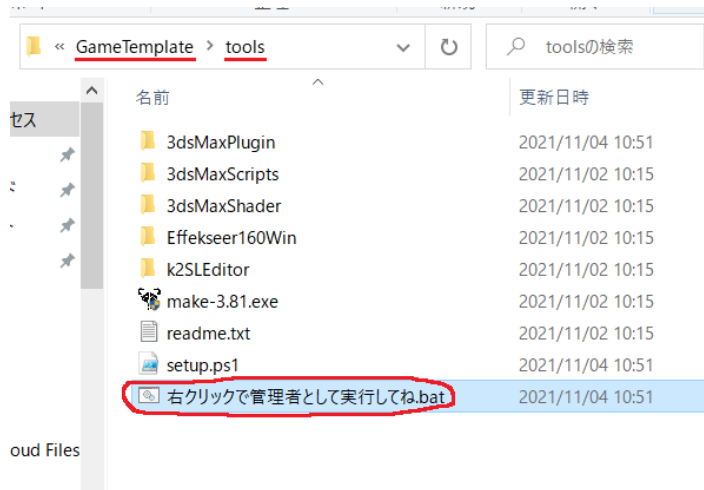
| 名前                    | 更新日時             |
|-----------------------|------------------|
| .git                  | 2021/10/06 10:42 |
| GameTemplate          | 2021/10/06 9:37  |
| Sample                | 2021/10/06 9:38  |
| SourceAssets          | 2021/10/06 9:38  |
| .gitignore            | 2021/10/06 9:36  |
| K2Engineについて.docx     | 2021/10/06 9:37  |
| K2Engineについて.pdf      | 2021/10/06 9:37  |
| 右クリックで管理者として実行してね.bat | 2021/10/06 9:38  |

「右クリックで管理者として実行してね.bat」を選択

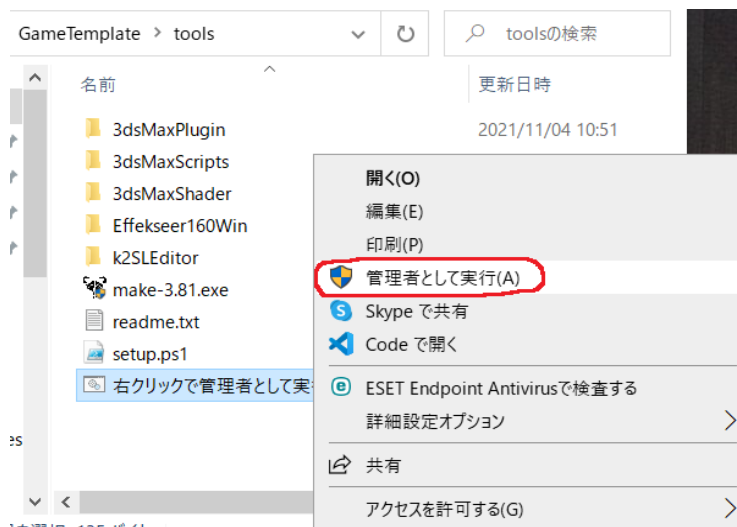


右クリック→「管理者として実行」を選択する

## 2.2 GameTemplate/tools の「右クリックで管理者として実行してね.bat」を実行



GameTemplate/tools の  
右クリックで管理者と  
して実行してね.bat を選  
択



右クリック→「管理者  
として実行」を選択す  
る

Setup - Make



## Welcome to the Make Setup Wizard

This will install Make-3.81 (GNU make utility to maintain groups of programs; Complete package, except sources) on your computer.

It is recommended that you close all other applications before continuing.

Click Next to continue, or Cancel to exit Setup.

Next >

Cancel

Setup - Make

## License Agreement

Please read the following important information before continuing.

Please read the following License Agreement. You must accept the terms of this agreement before continuing with the installation.

### GNU GENERAL PUBLIC LICENSE Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.  
59 Temple Place - Suite 330, Boston, MA  
02111-1307, USA.

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

☒ I accept the agreement

☐ I do not accept the agreement

Print

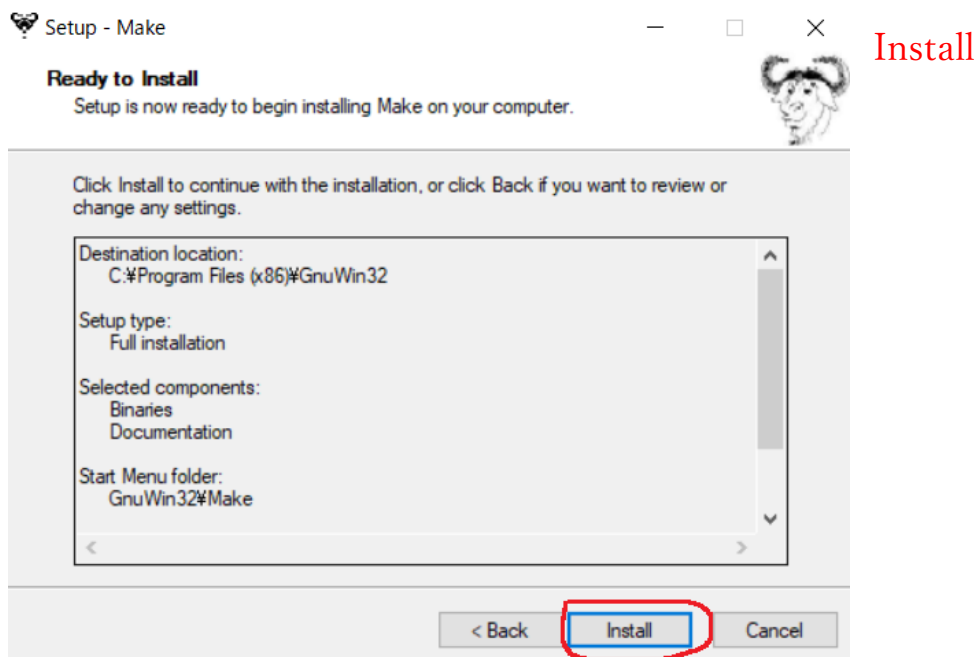
< Back

Next >

Cancel

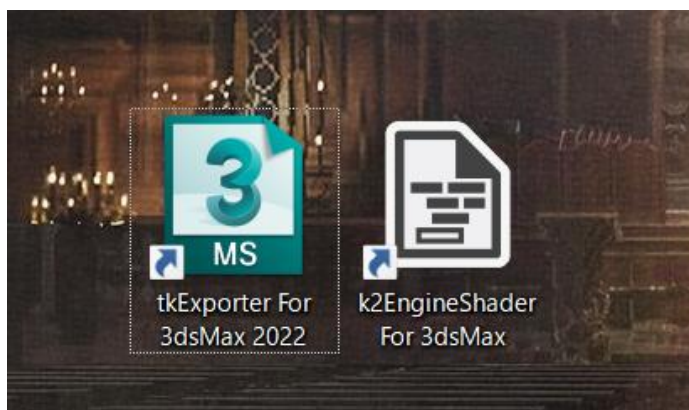
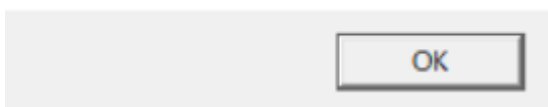
Make の Setup 画面が表示される

I accept the... を選択



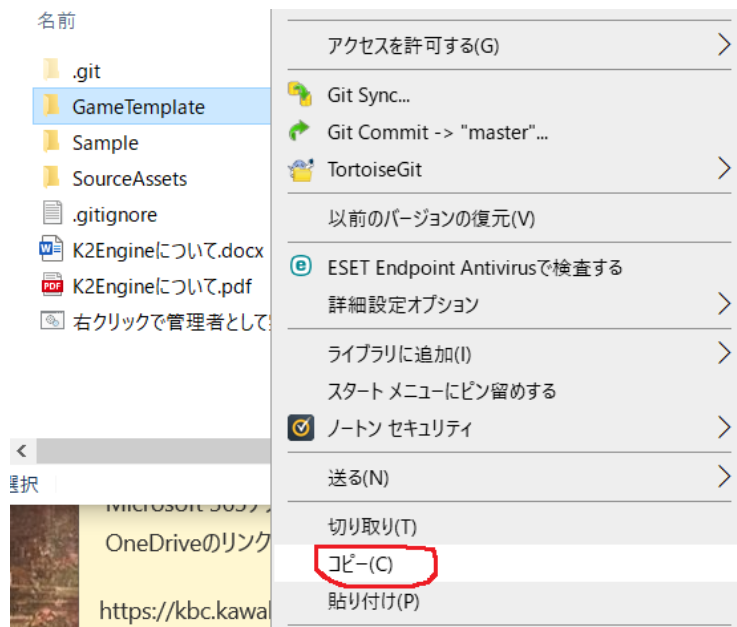
× make のインストールが完了してしばらくすると、このようなメッセージが表示されるので、再起動する

セットアップ完了。PCを再起動してください。

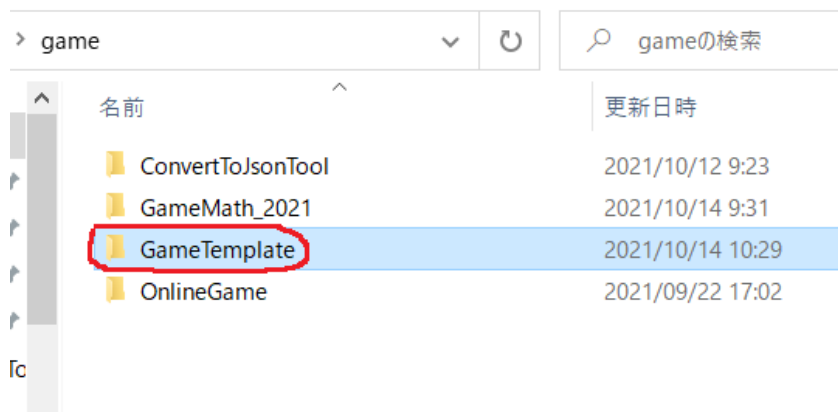


デスクトップに「tkExporter For 3dsMax」と「k2EngineShader For 3dsMax」のショートカットが作成されていることを確認する

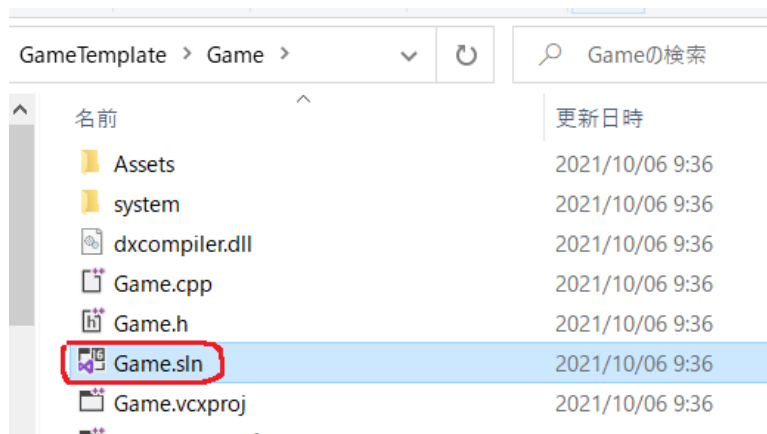
## 2.3 GameTemplate をコピーして制作を開始



GameTemplate を丸ごとコピー



コピーした  
GameTemplate



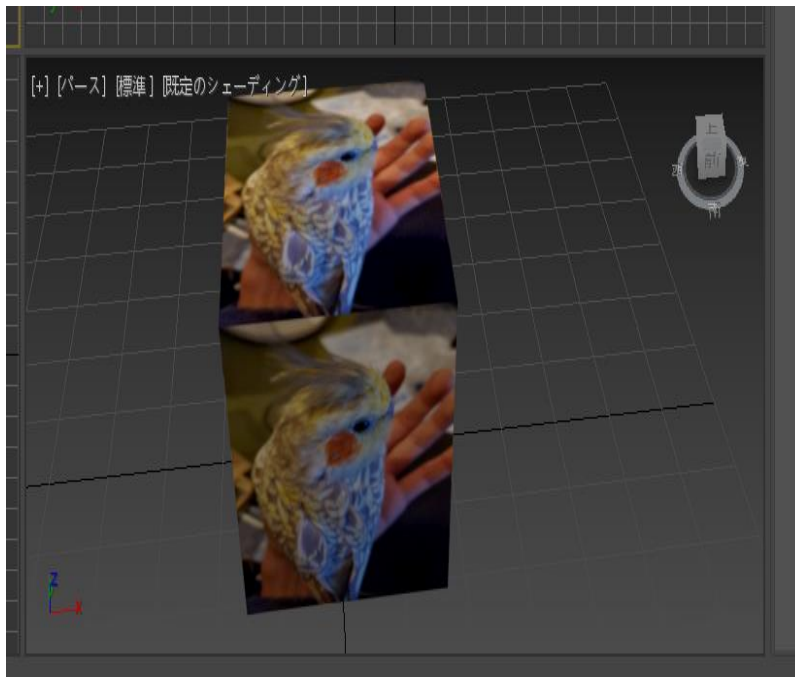
コ ピ ー し た  
GameTemplate の  
Game フォルダの  
Game.sln をダブル  
クリックすると、  
Visual Studio が立ち  
上がる。

### 3. 3D モデル出力

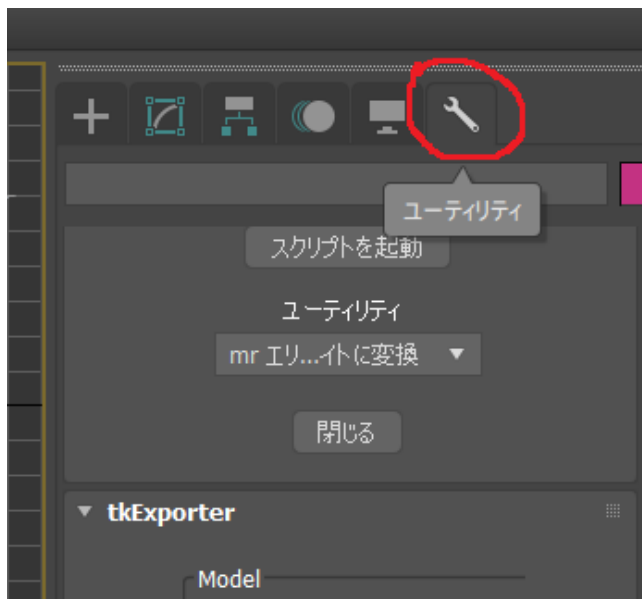
3dsMax から k2Engine で 3D モデルを使用するため  
の、「tkm ファイル」出力までを説明する



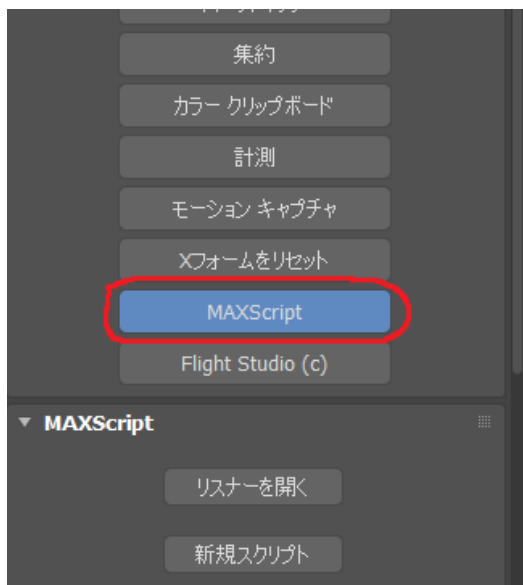
### 3.1 tkExporter を起動



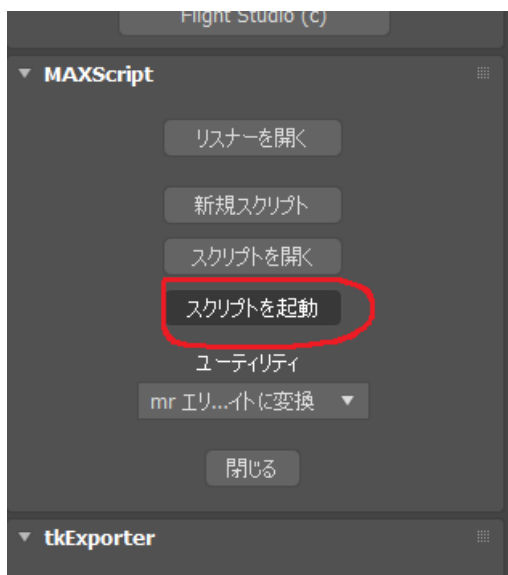
3dsMax でモデリングを行う  
(テクスチャの貼りは、**5. K2Engine 用 3D モデルの max マテリアルの設定**の項目を参照)



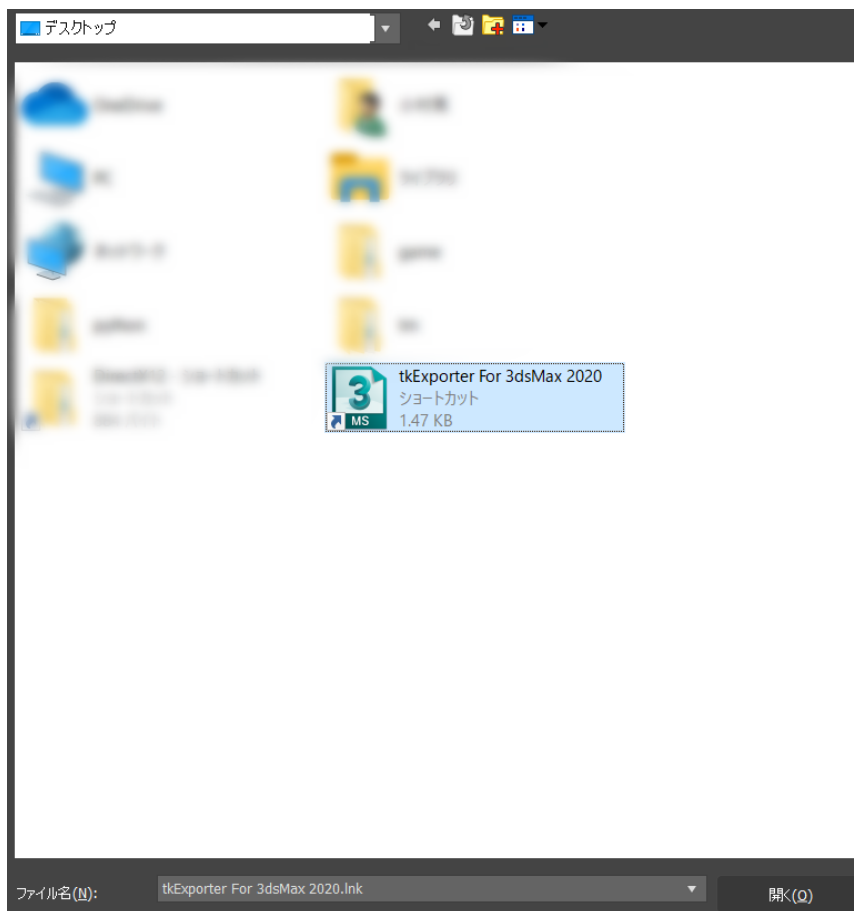
「ユーティリティ」を選択



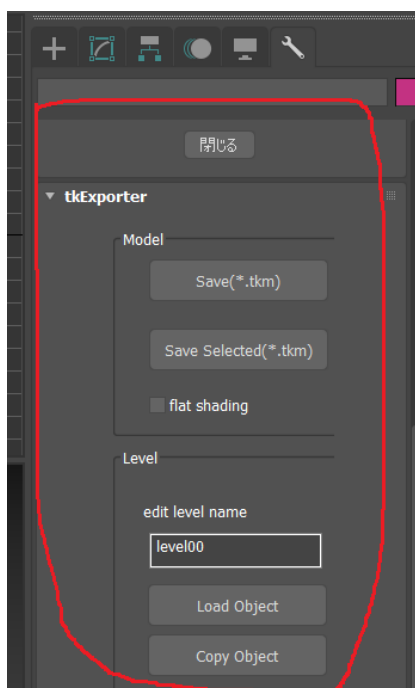
「MAXScript」を選択



「スクリプトを起動」を選択

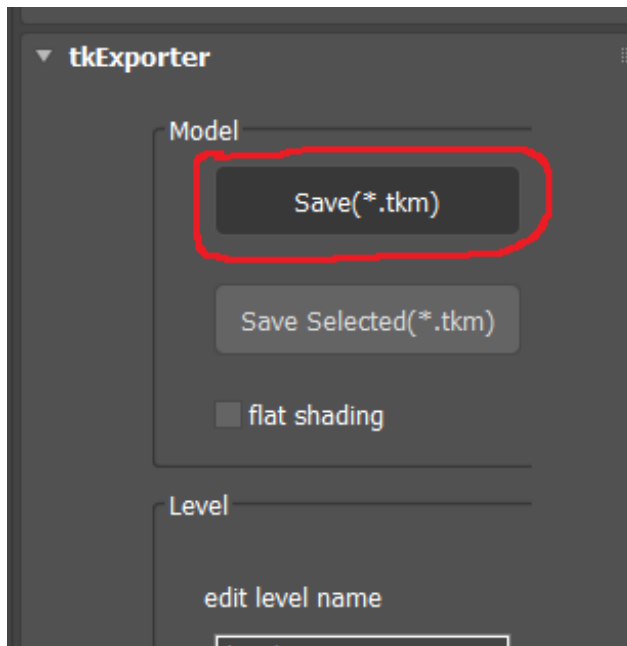


デスクトップ  
の「tkExporter  
For 3dsMax」を  
選択

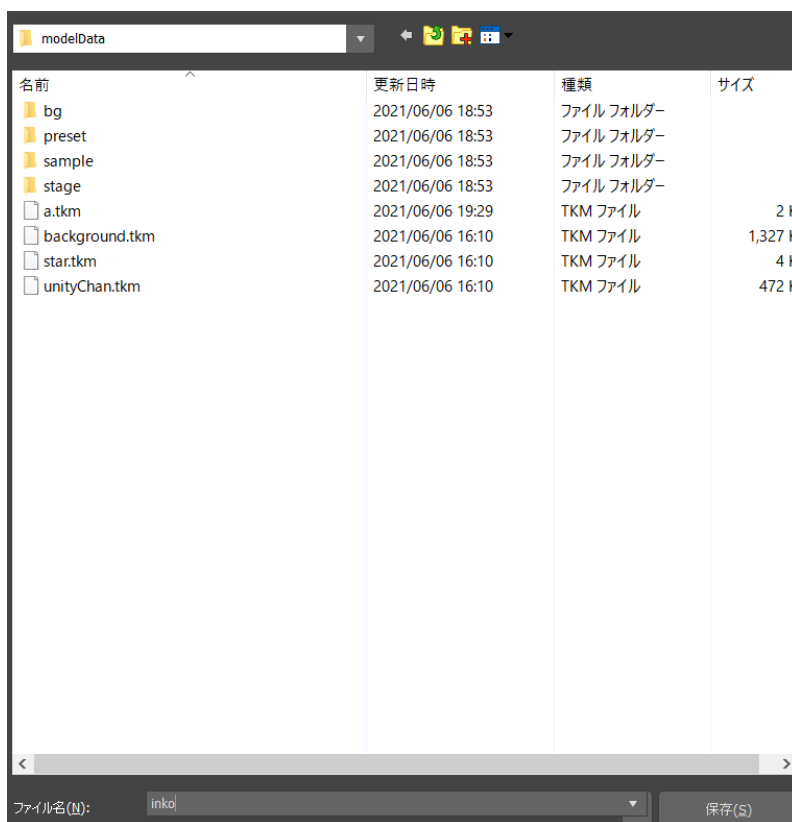


ユーティリティにこのような項目が追加される

## 3.2 「tkm ファイル」を出力



追加された項目の  
「Save(\*.tkm)」を選択する

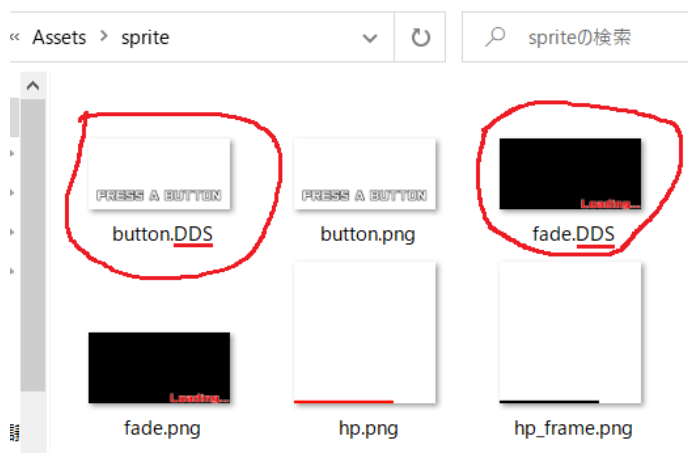


ファイルパスを設定して、出力！！  
(テクスチャ等のファイルパスに漢字や空白が入っていると失敗するかも)

## 4. 画像表示

K2Engine で画像を表示する際の、画像ファイルについて説明する

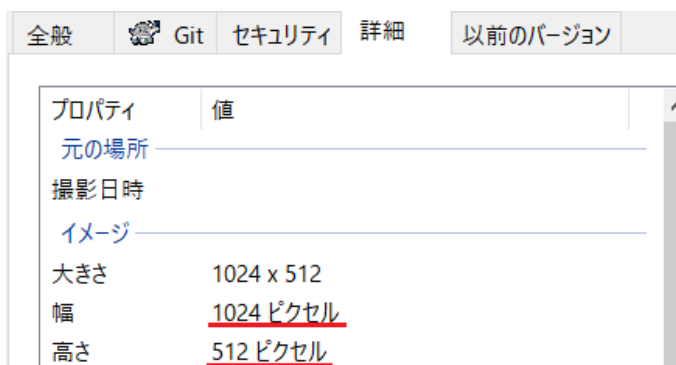
### 4.1 dds ファイル



K2Engine で画像を表示したい場合は、png や jpeg といった画像ファイルを「**dds ファイル**」に変換する必要がある



**dds ファイル**への変換については、**web サイト**や **k2SLEditor** を使用する、「**dds 変換**」等でググる

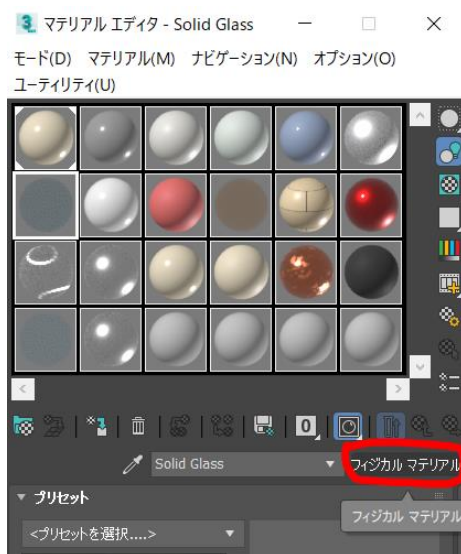


\* 画像は 200\*500 等、横縦のピクセル数が偶数になるようにする

## 5. K2Engine 用 3D モデルの max マテリアルの設定

K2Engine 用の 3D モデルを 3ds Max で作る際の、マテリアルの設定について説明する

### 5.1 マテリアルの種類を「DirectX Shader」に設定



マテリアルを選択した状態で、「**フィジカルマテリアル**」を選択する

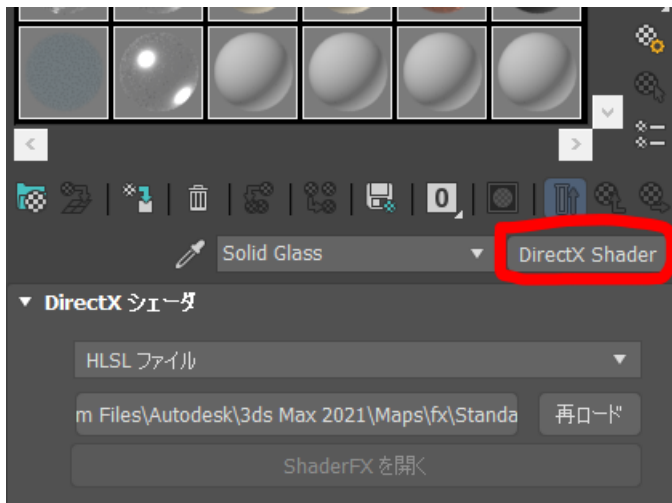
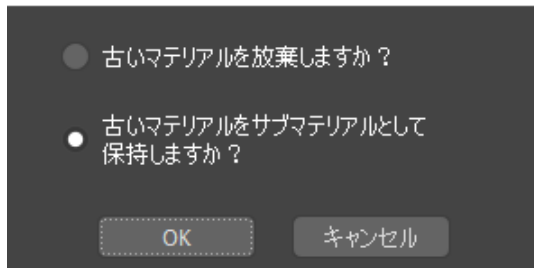


× マテリアル/マップ  
ブラウザが開くので、  
「**DirectX Shader**」を  
選択する

マテリアルを置換

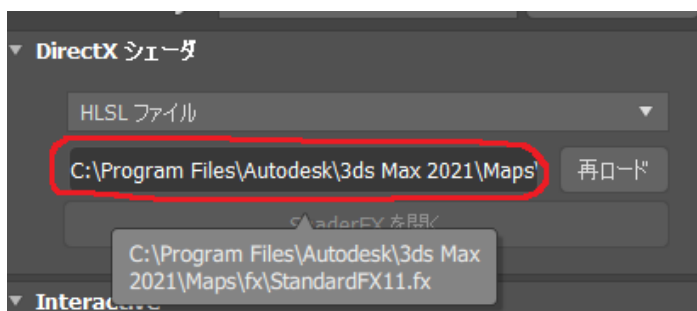
×

\* どちらでもよい

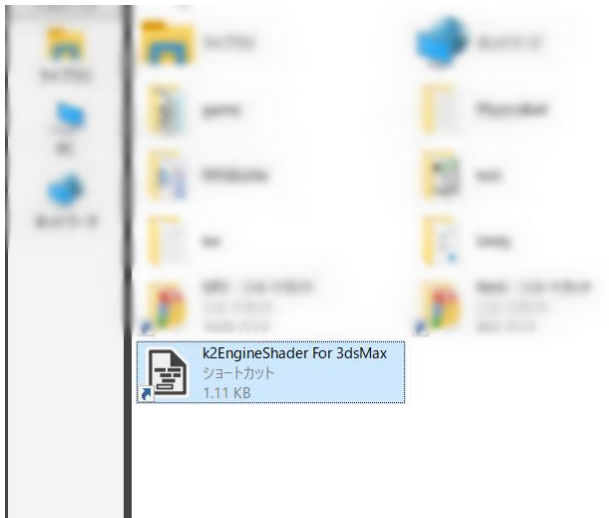


マテリアルの種類が「DirectX Shader」に置き換わっていることを確認する

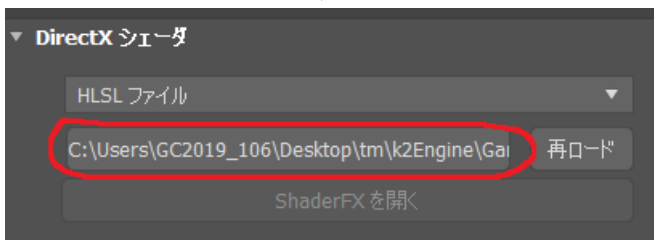
## 5.2 マテリアルの DirectX シェーダに「k2EngineShader.fx」を設定



HLSL ファイルにデフォルトでファイルが設定されているので、

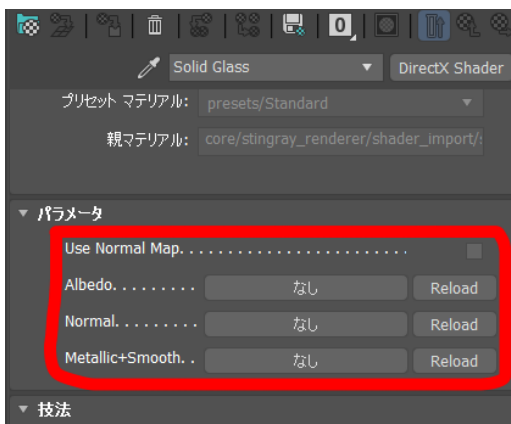


デスクトップ上の  
「**k2EngineShader  
For 3dsMax**」を選択する



変更した後

## 5.3 マテリアルにテクスチャを設定



パラメータの欄にテクスチャ等を設定する

4つの項目がある

- User Normal Map
- Albedo
- Normal
- Metallic+Smooth



### 5.3.1 User Normal Map

後述する NormalMap を使用するかどうかを設定する項目

### 5.3.2 Albedo

ベースカラーテクスチャを設定する項目

### 5.3.3 Normal

NormalMap(法線テクスチャ)を設定する項目

User Normal Map を設定しておかないと有効にならない

### 5.3.4 Metallic+Smooth

Metallic(金属度)と Smooth(滑らかさ)のテクスチャを設定する項目

それぞれ

Metallic はテクスチャの R(赤)

Smooth はテクスチャの  $\alpha$  値(不透明度)

を参照する

\* 各テクスチャの縦横のピクセル数は 2 の累乗にする

2、4、8、16、32、64、128、256、512、1024…

### 5.3.5 Metallic(金属度)

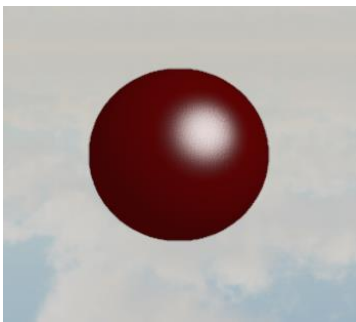
鏡面反射で発生する光の色が変わる

高いほど Albedo テクスチャの色

低いほど白色

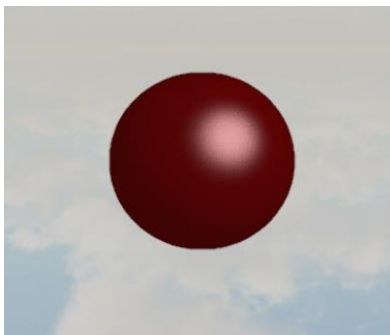
になる

以下に実例を示す



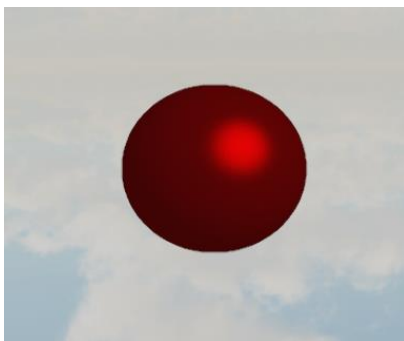
Metallic : 0.0

白い鏡面反射光



Metallic : 0.5

少し赤みがかかった鏡面反射光



Metallic : 1.0

赤い鏡面反射光

### 5.3.6 Smooth(滑らかさ)

写り込みのしやすさ

値が高いほど写り込みがハッキリする

以下に実例を示す



Smooth : 0.0

写り込みが全く発生していない



Smooth : 0.5

写り込みが発生している



Smooth : 1.0

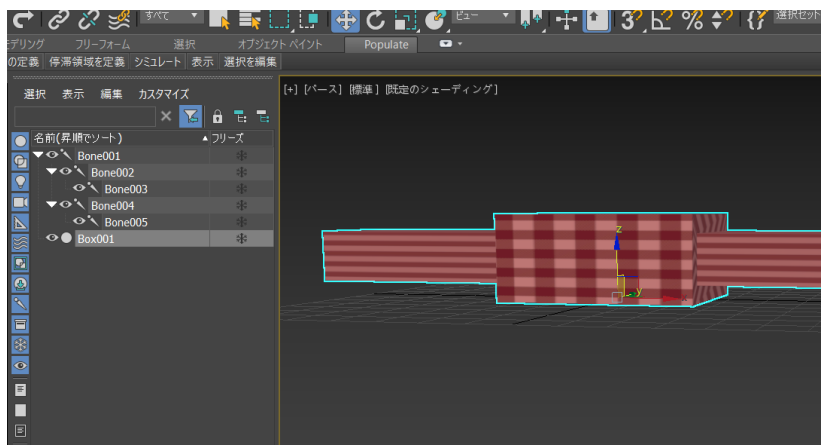
鏡のようにハッキリと写り込みが発生している

## 6. 3D モデルのアニメーションの出力

K2Engine でアニメーションを再生する際に必要な、「tka ファイル」の出力方法について説明する

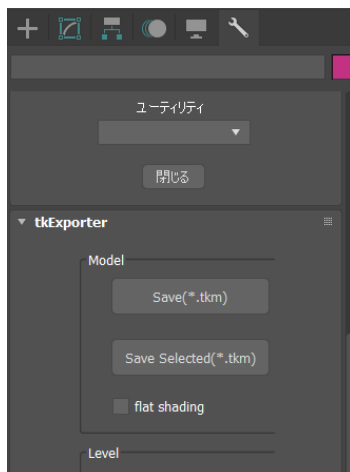
### 6.1 3D モデルにボーンとアニメーションを設定

ボーン、アニメーションについては、[こちらの動画](#)を参照

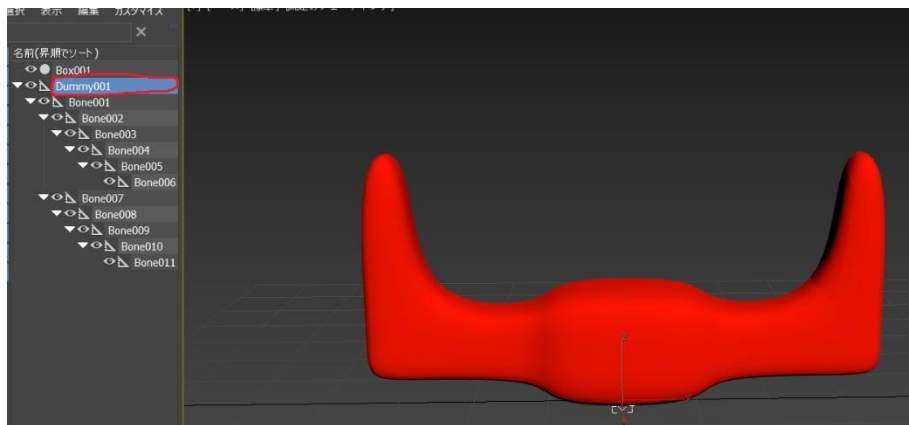


ボーンとアニメーションを設定し終わったの図

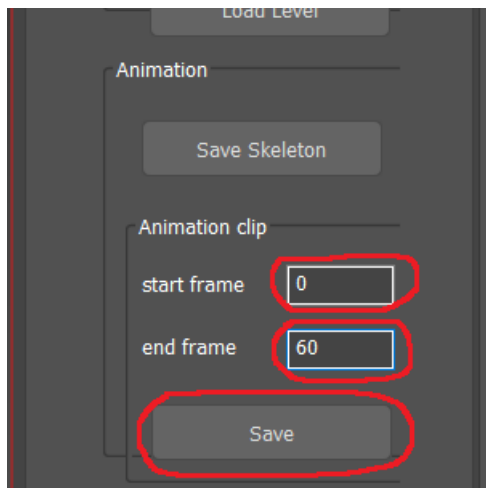
### 6.2 tkExporter を起動し、出力するアニメーションのフレームを設定



tkExporter を起動したの図

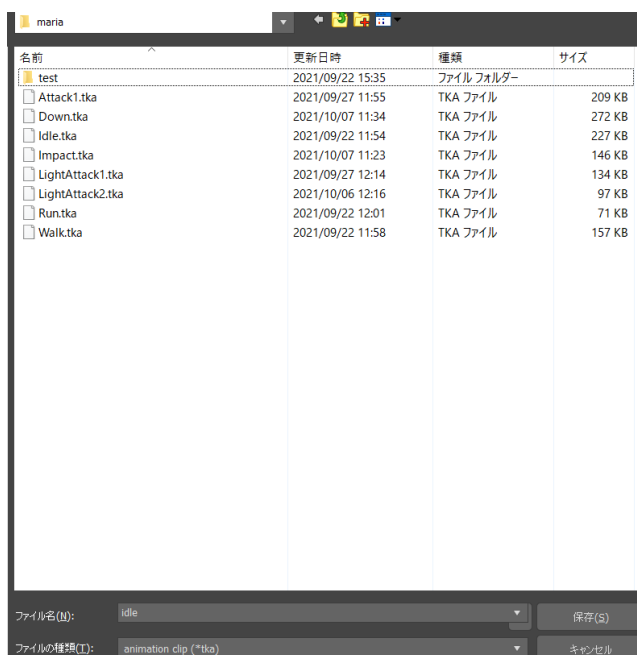


「ルートボ  
ーン」を選  
択



tkExporter の項目に「**Animation**」と  
いう項目がある

出力するアニメーションの「**最初のフ  
レーム**」と「**最後のフレーム**」を設定  
し、「**Save**」を選択する



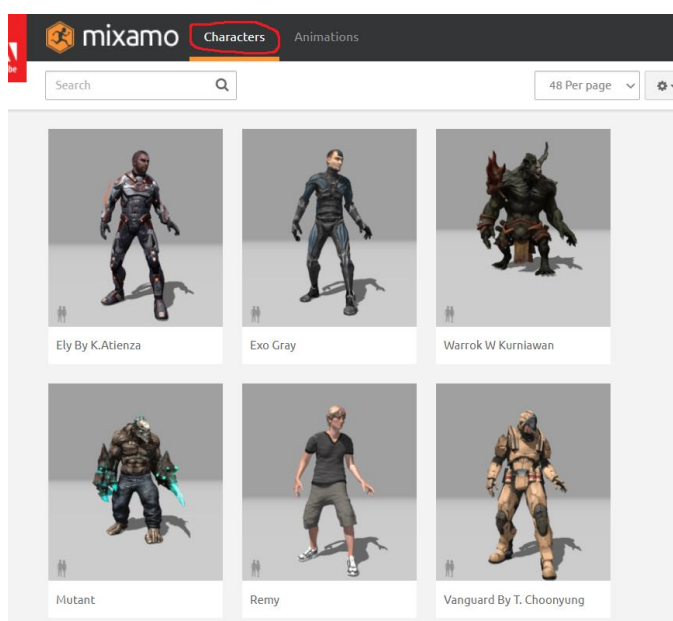
ファイルパスを設定し、「**tka  
ファイル**」を出力

## 7. Mixamo の紹介

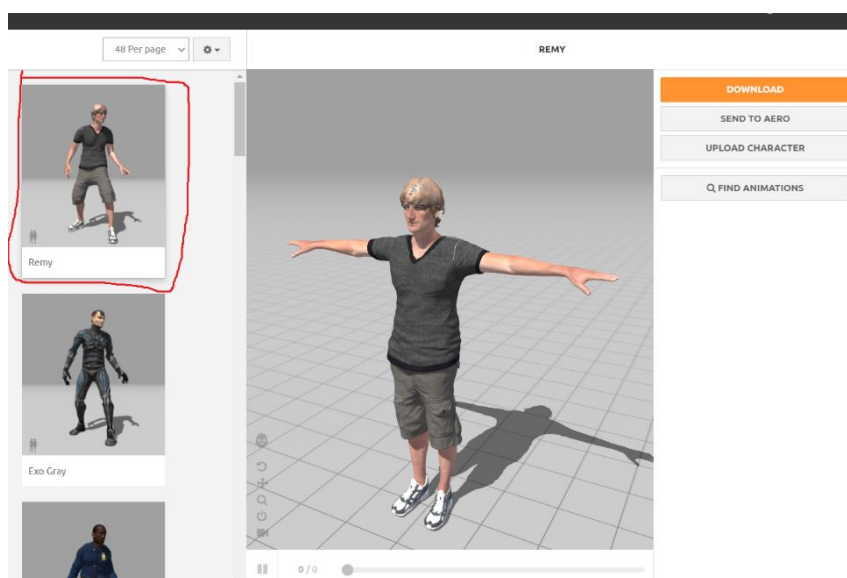
Adobe の [Mixamo というサイト](#) について説明する  
3D モデルのダウンロード、既存のアニメーションの適用ができる

\* Adobe のアカウントが必要である

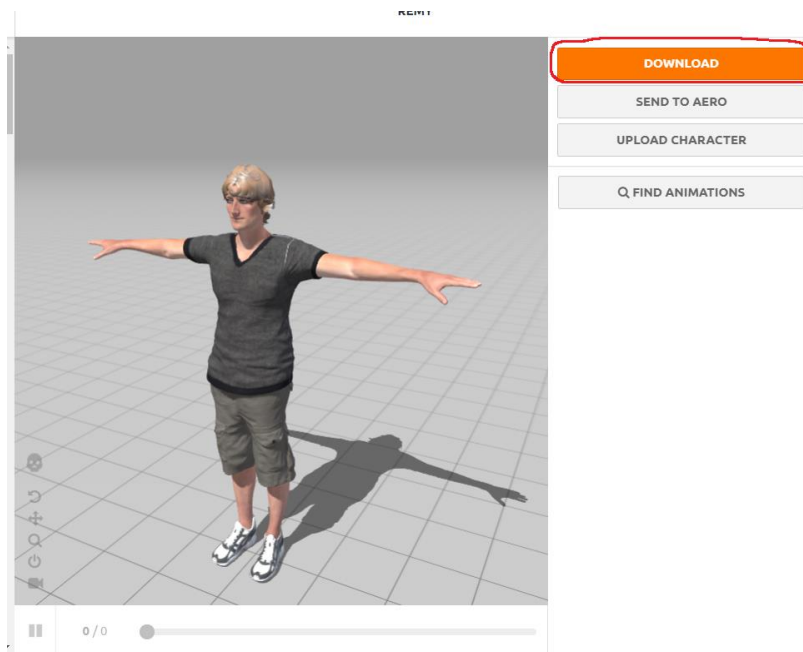
### 7.1 モデルのダウンロード



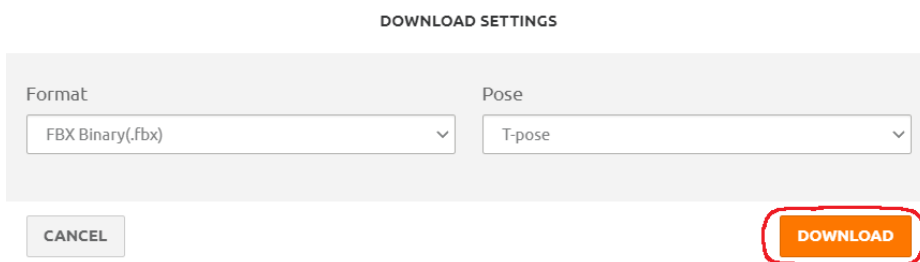
Mixamo を開いたら、サイト上部の「**Characters**」を選択する  
すると、利用可能なモデル一覧が表示される



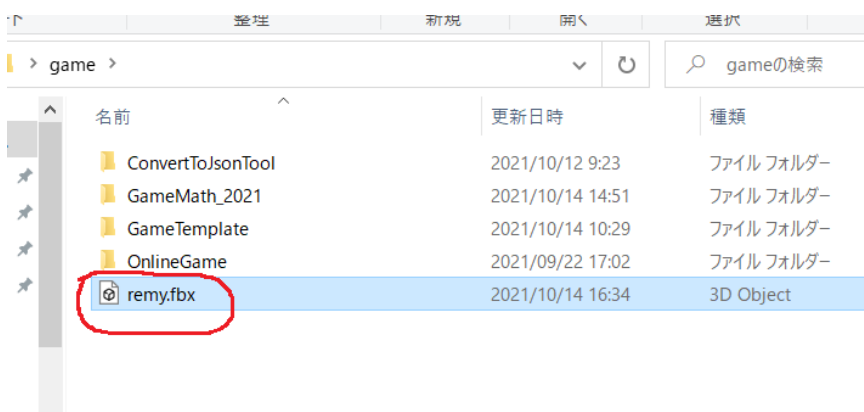
モデルを選択すると、サイト右側にモデルが表示される



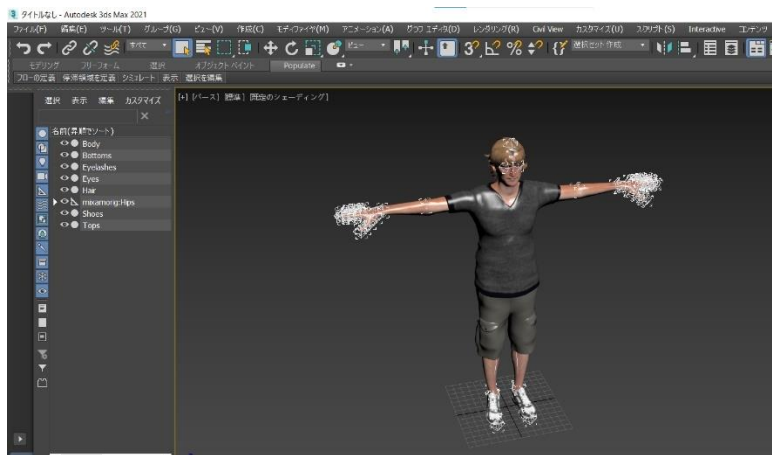
「**DOWNLOAD**」  
でモデルをダウン  
ロードできる



ダウンロー  
ド

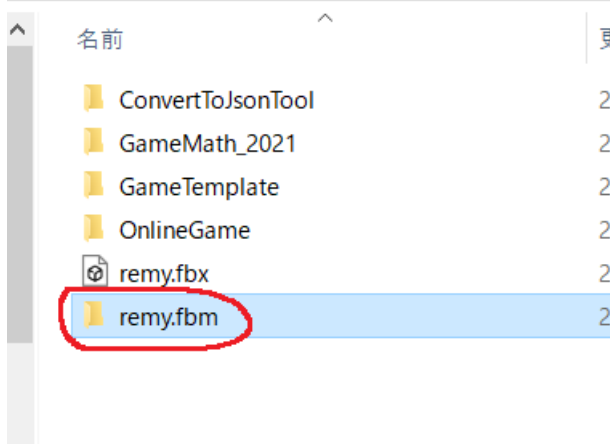


ダウンロード  
した **fbx** ファ  
イル

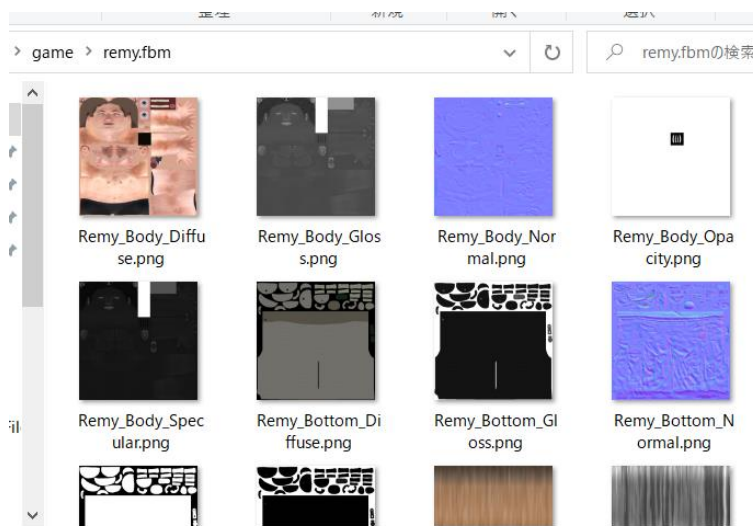


ダウンロードした  
fbx ファイルを **3ds  
Max** で読み込むと、  
(ボーン設定済み)

game



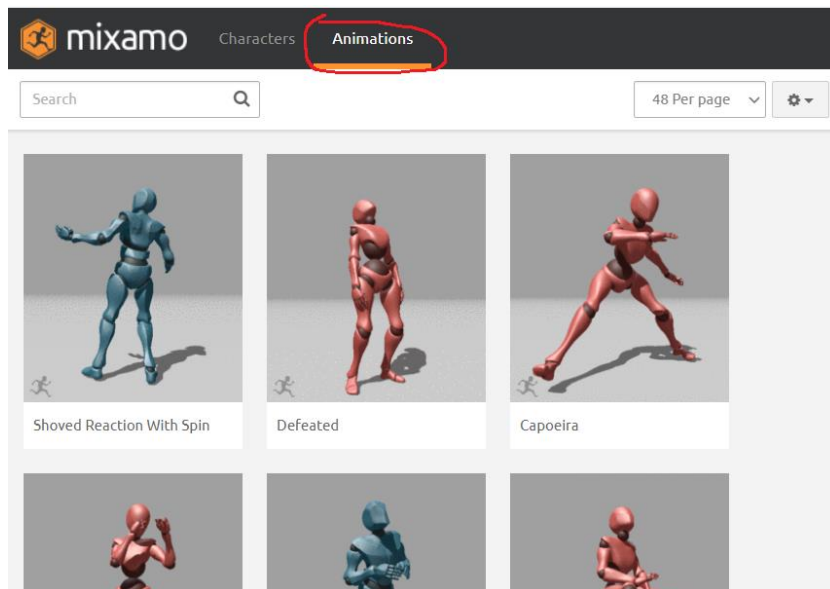
テクスチャのフォルダが作成  
されている



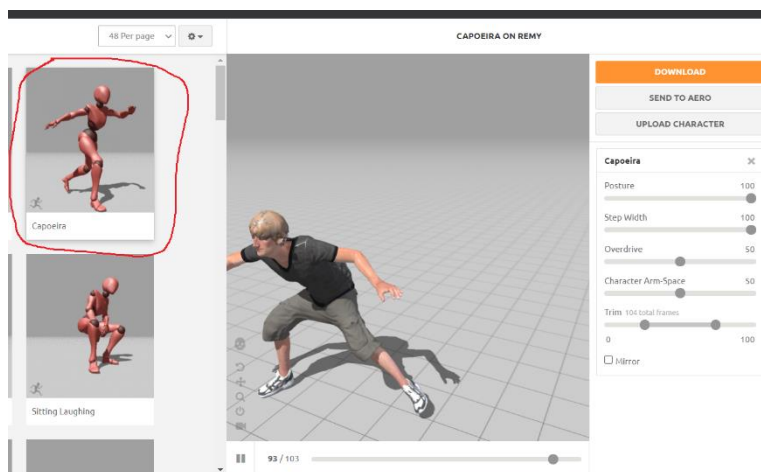
↑ の remy.fbm の中身



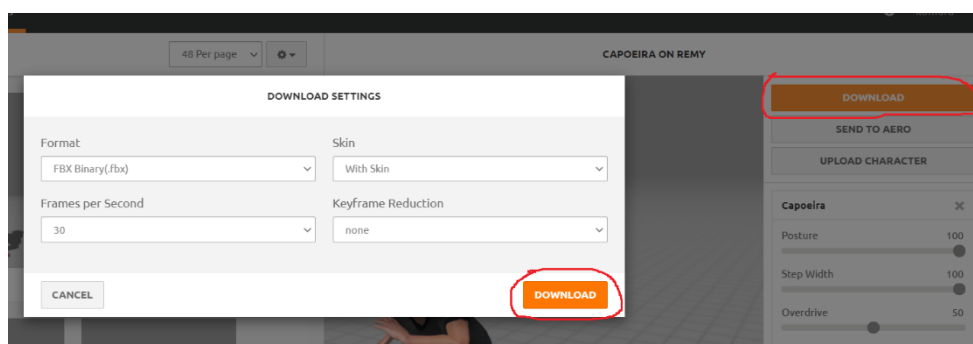
## 7.2 アニメーションの適用とダウンロード



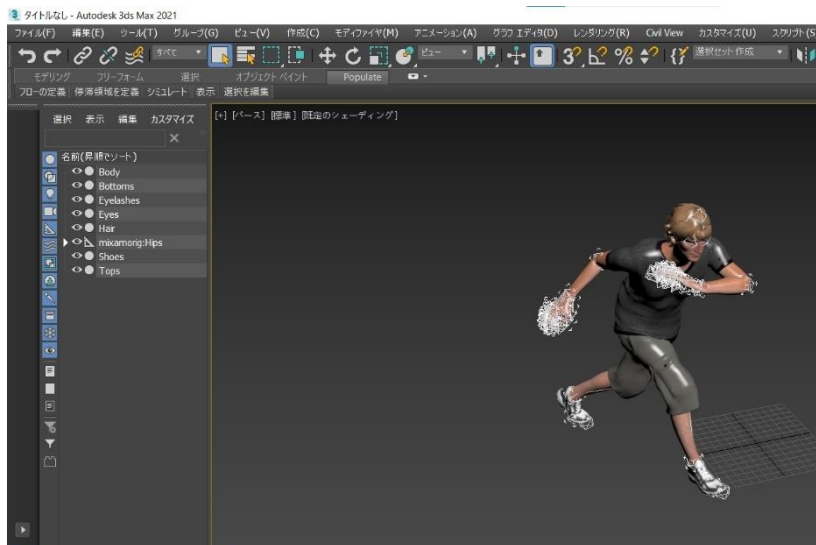
「**Animations**」  
タブで、アニメーション一覧が表示される



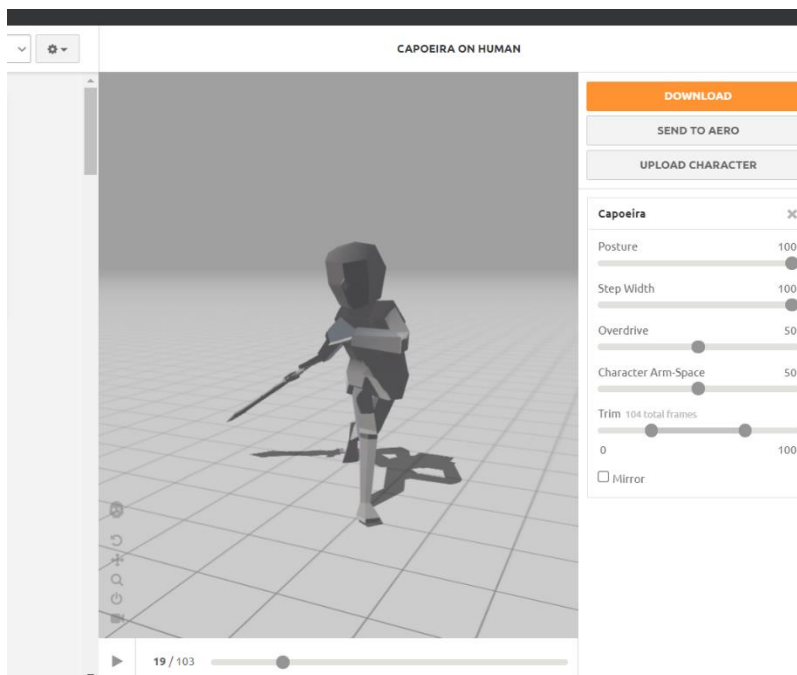
アニメーションを選択すると、右側のモデルにアニメーションが適用される



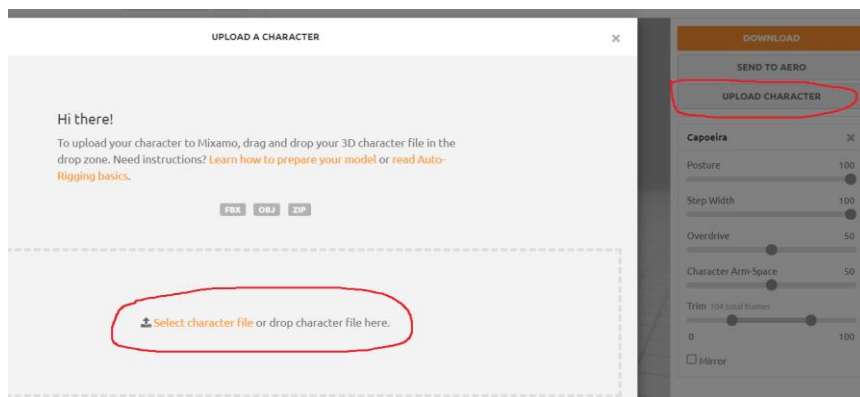
ダウンロード



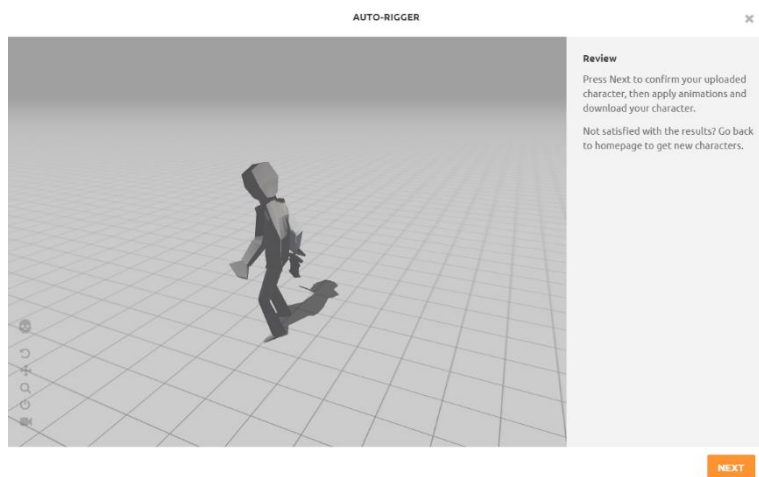
アニメーション  
付きの fbx がダウ  
ンロードされる



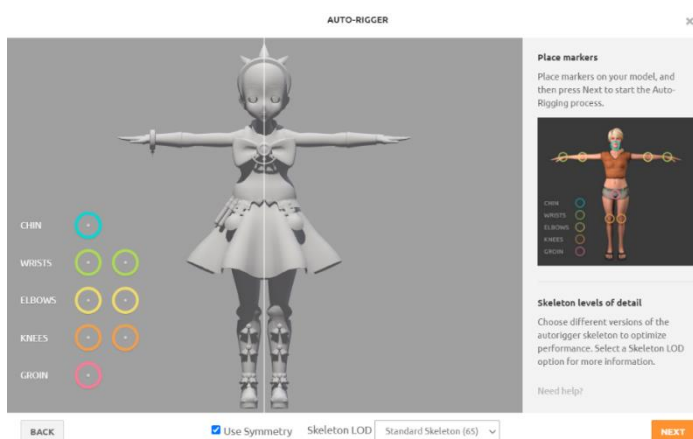
Mixamo のモデル  
以外にもアニメー  
ションを適用でき  
る



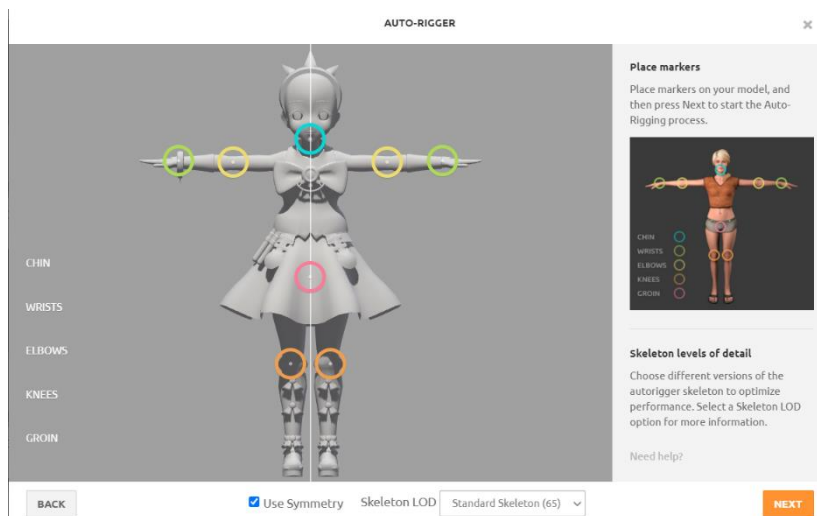
「UPLOAD CHARACTER」  
→ 「Select character File」  
を選択



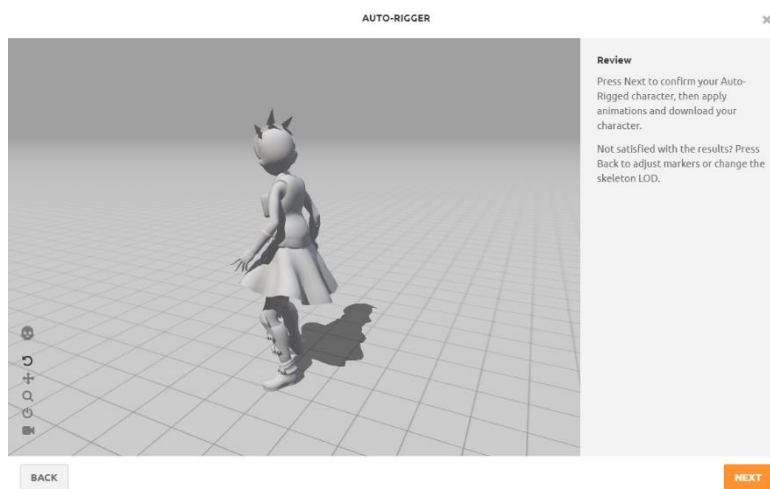
ボーン付き、Tポーズのモデルをアップロードするとこのようになる



ボーンが無いモデルをアップロードすると、顎、肘、手首、膝、股間の位置を設定するように要求される  
(ボーンを付けるため)



それぞれの位置  
を設定すると



自動でボーンを割り  
当ててくれる

この状態でモデルを  
ダウンロードすると、  
ボーン付モデルが入  
手できる

## 8. ナビゲーションメッシュと A\* アルゴリズム

### 8.1 ナビゲーションメッシュ、A\* アルゴリズムとは？

「ナビゲーションメッシュ」と「A\* アルゴリズム」

この 2 つを使用すれば、オブジェクトを「障害物を避けて」目的地に移動させることができる

<https://youtu.be/c3qxCHnbIKg>

↑ の動画で、Unity ちゃんが「黄色い弾」に向かって、「壁を避けて」移動しているのが分かるだろうか

- ・ ナビゲーションメッシュ

[詳しくはこちら](#)

要約すると、キャラクターが移動する「地形」や「地図」のデータである

- ・ A\* スターアルゴリズム

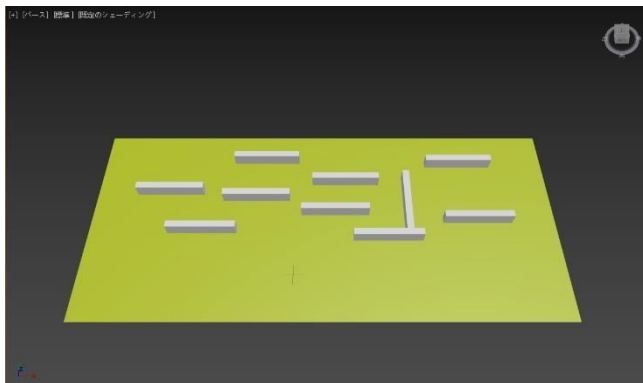
[詳しくはこちら](#)

要約すると、探索アルゴリズムの一種であり、出発地点から目的地までの最短経路を計算するアルゴリズムである

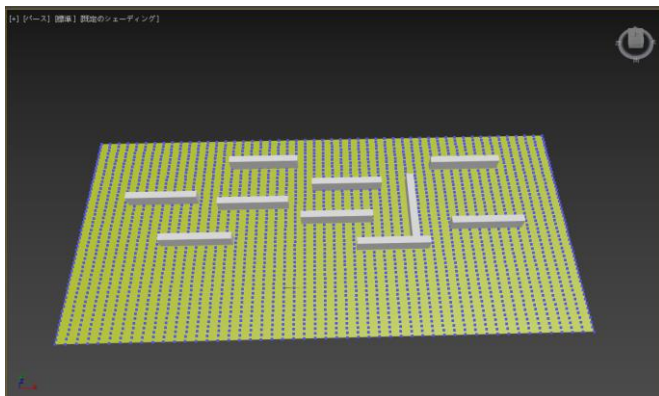
ナビゲーションメッシュデータを使用して、A\* アルゴリズムで目的地までの経路を計算するイメージである

## 8.2 tkn ファイル出力

上記の動画のような動きをさせるには、ナビゲーションメッシュデータである、「**tkn ファイル**」を用意する必要がある



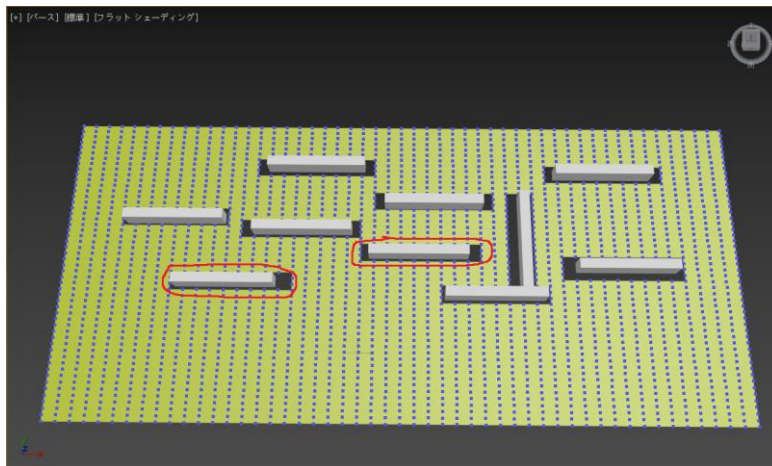
3dsMax で地形データを用意する



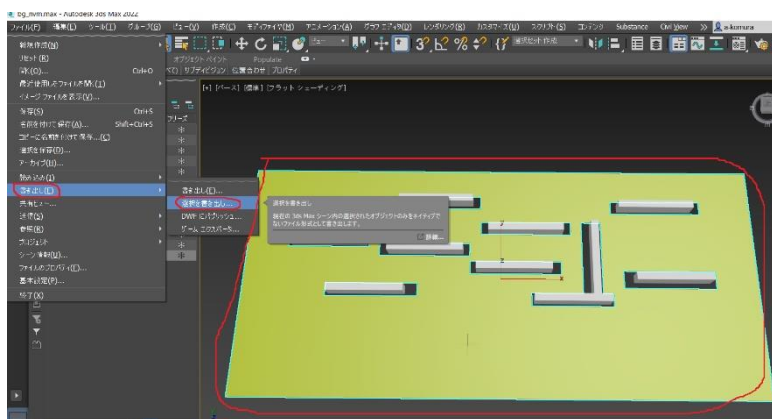
「**地面**」の部分に当たるモデルの**頂点数**を調整する

\* 頂点数が多ければ、より自然な動きをするが、処理が重くなる

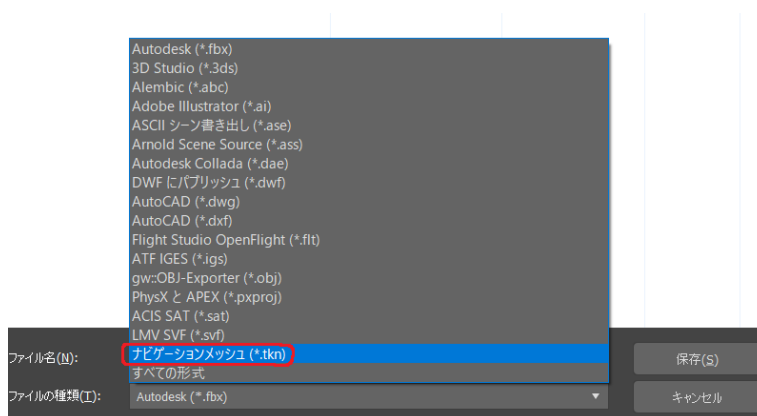
\* 「MultiRes」というモディファイヤを使用すると○



「地面」に当たる部分のモデルの「障害物」がある部分のポリゴンを切り抜く  
 \* 障害物がある部分に移動しないようにするため



「地面」に当たるモデルを選択し、  
 ファイル→書き出し→選択を書き出し



k2Engine のセットアップが終了してれば、ファイルの種類に「ナビゲーションメッシュ (\*.tkn)」が追加されているのでこれを選択し、出力

## 8.3 出力した tkn ファイルを使用して、移動処理を実装する

詳しくは、[Sample\\_25](#) を参照

以下、[Sample\\_25](#) のコードについて解説をする

```
// ナビメッシュを構築。  
m_nvmMesh.Init("Assets/nvm/test.tkn");
```

tkn ファイルを読み込んで、ナビゲーションメッシュデータを構築

```
// パス検索  
m_pathFinding.Execute(  
    m_path,                // 構築されたパスの格納先  
    m_nvmMesh,             // ナビメッシュ  
    m_position,            // 開始座標  
    m_targetPointPosition, // 移動目標座標  
    PhysicsWorld::GetInstance(), // 物理エンジン  
    50.0f,                 // AIエージェントの半径  
    200.0f                 // AIエージェントの高さ。  
);
```

上記で構築したナビゲーションメッシュデータを参照し、

[A\\*アルゴリズム](#)を用いて、出発地から目的地までの経路を構築、

m\_path に経路データが構築される

[処理に時間が掛かる](#)ので、Update 関数にそのまま書いて毎フレーム実行させるのは非推奨



```
// パス上を移動する。  
m_position = m_path.Move(  
    m_position, |  
    10.0f,  
    isEnd  
);
```

**m\_path**(経路データ)を使用して、移動処理を行う

