



MxE SDK向け  
PSDファイル作成ガイド  
MxE チュートリアル #2 ボタングループ

2023.07 MxEサービスドキュメント

# 目次

- [文書の目的](#)
- [対象とするGUIObjectコントロール](#)
- [サンプルファイルについて](#)
- [事前準備](#)
- PSDファイル作成手順
  - [サマリー](#)
  - [Step 1 : プラグインによるレイヤー群の生成](#)
  - [Step 2 : 各レイヤーへの絵柄の配置](#)
  - [Step 3 : タップ判定領域の配置](#)
  - [Step 4 : ラスタライズとレイヤーの整理](#)
  - [Step 5 : 状態ごとの表示切り替えの設定](#)
  - [Step 6 : 動作パラメータの編集](#)
- [SDKへの取り込み結果](#)

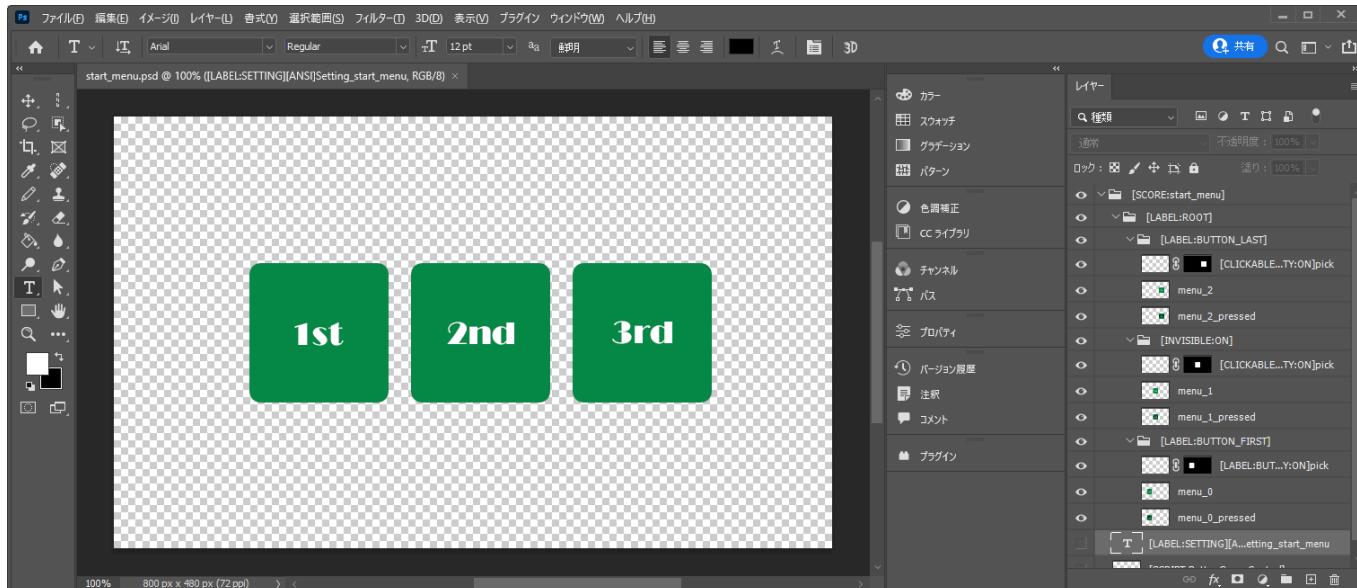
# 文書の目的

---

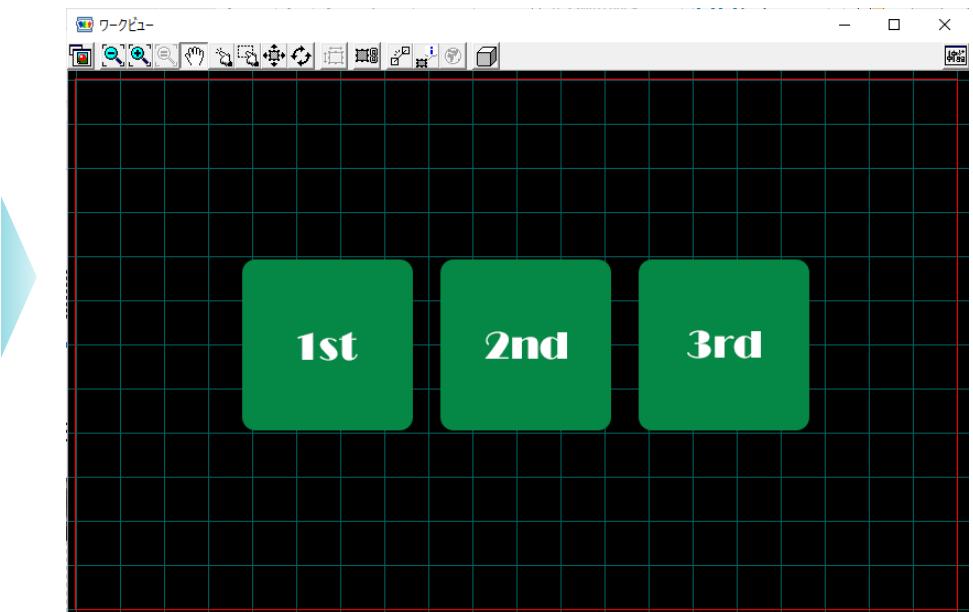
- IoT用HMIデザインツール「MatrixEngine（略称「MxE」）SDK」では、Photoshopデザインデータを画面要素（「GUIObject」）として取り込むことができます。
- MxEへの取り込みに適した構成のPSDファイルは、Photoshop用プラグイン「GUIObject Constructor」を利用すると効率的に作成できます。
- 本書では「GUIObject Constructor」の利用手順を含めた、MatrixEngine SDKへの取り込みに適したPhotoshop PSDファイルの作成手順、およびSDKへの取り込み結果を図示します。
  - 上記目的以外のPhotoshop操作や用語については、Photoshopのヘルプを参照ください。

# 本書が対象とするGUIObjectコントロール

- 本書ではButtonGroupコントロールを作成する一連の手順を紹介します。
- 800x480のサイズの画面中央に表示される、3つ連続するボタンのグループを作るものとします。



Photoshop上データ



SDKに取り込んだ状態

# サンプルファイルについて

---

- 本書に従って作成されるPSDファイルは次の通りです。
  - ファイル名 : start\_menu.psd

# 事前準備

---

- 事前にPhotoshop用プラグイン「GUIObject Constructor」を導入ください。
  - MatrixEngine SDK およびプラグインの導入手順については、別資料「インストール及び環境構築ガイド」を参照ください。
  - MatrixEngine SDKおよび各ツール群については別資料「MatrixEngine概要」を参照ください。

# HMIデザインを容易にするMatrixEngineツール群

- MatrixEngine SDK
  - GUIコンテンツオーサリングツール本体です。
- PSD Importer
  - MatrixEngine SDK向けのプラグインです。
  - Photoshopで作成した素材（PSDファイル）をMatrixEngine SDKにインポートできます。
  - Photoshop向けプラグインと合わせてご利用いただくと、デザイン制作との協働効率がより向上します。
- Photoshop Plugin GUIObject Constructor
  - Photoshop向けのプラグインです。
  - MatrixEngine SDKに取り込むためのPSDファイル（レイヤー構造）を簡単・効率的に作成できます。
- Outline Builder
  - MatrixEngine SDK向けのプラグインです。
  - 一定パターンでコードを自動生成できる便利ツールです。
- GUIObject
  - MatrixEngineスクリプトで記述されたライブラリ群（コントロールモジュールとフレームワーク）です。

# PSDファイル作成手順サマリー

---

Photoshopプラグイン利用 & 設定のアウトライン

# PSDファイル作成手順サマリー (1/2)

---

本書では、800x480サイズの画面中央に表示されるボタングループを作るものと想定し、ButtonGroupコントロールを作成する一連のステップと各操作をご案内します。

## ■ Step1：プラグインによるレイヤー群の生成

- Photoshop用プラグイン「GUIObject Constructor」を使用して、SDK取込時に必要なレイヤーを生成します
- この際、GUIObjectに機能を持たせるための記述としてのレイヤー名はプラグインにより自動で設定されます

## ■ Step2：各レイヤーへの絵柄の配置

- 各レイヤーに必要な絵柄を配置します。

## ■ Step3：タップ判定領域の配置

- タップ機能の判定に必要なレイヤーマスクを設定します。

# PSDファイル作成手順サマリー (2/2)

## ■ Step4：ラスタライズと編集

- SDKに直接取り込めないエフェクト等をラスタライズします。

## ■ Step5：状態ごとの表示切り替えの設定

- SDKへ取り込み後に特定の状態で表示を変えられるよう、各レイヤーの不透明度を変更した組み合わせ（レイヤーカンプ）を設定します。

## ■ Step6：動作パラメータの編集

- SDKへ取り込み後の動作のパラメータを記述したテキストレイヤーを編集します。

## Step1

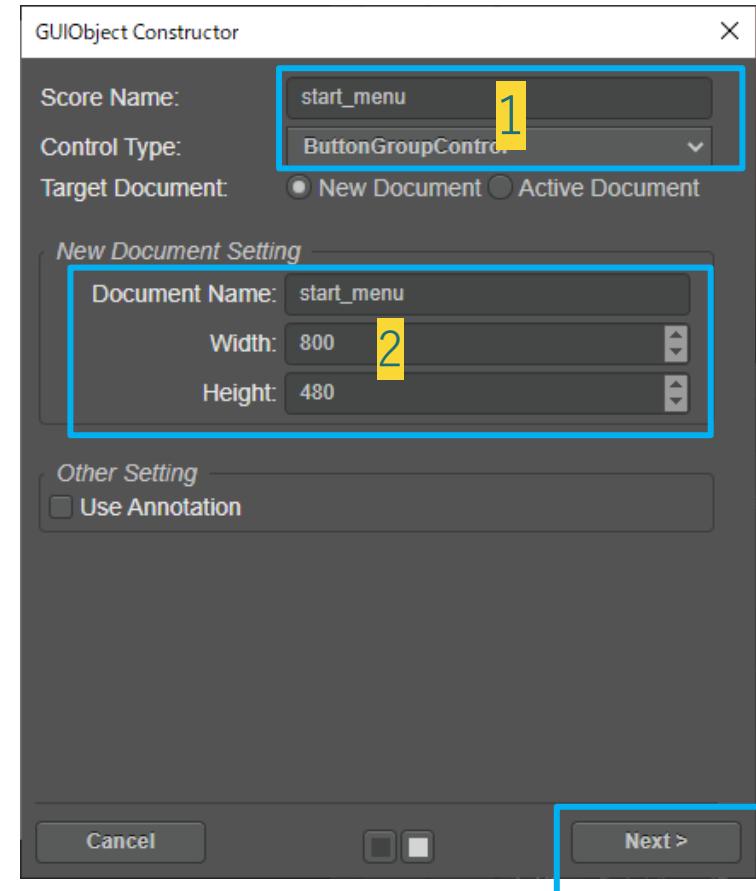
# プラグインによるレイヤー群の生成

---

Photoshop用プラグイン「GUIObject Constructor」を使用して必要なレイヤーを生成する

# Step1-1. 新規ButtonGroupコントロールの設定

- Photoshop用プラグインGUIObject Constructorを起動してください。



- “Score Name”に「start\_menu」と入力し、“Control Type”に「ButtonGroupControl」を選択します。

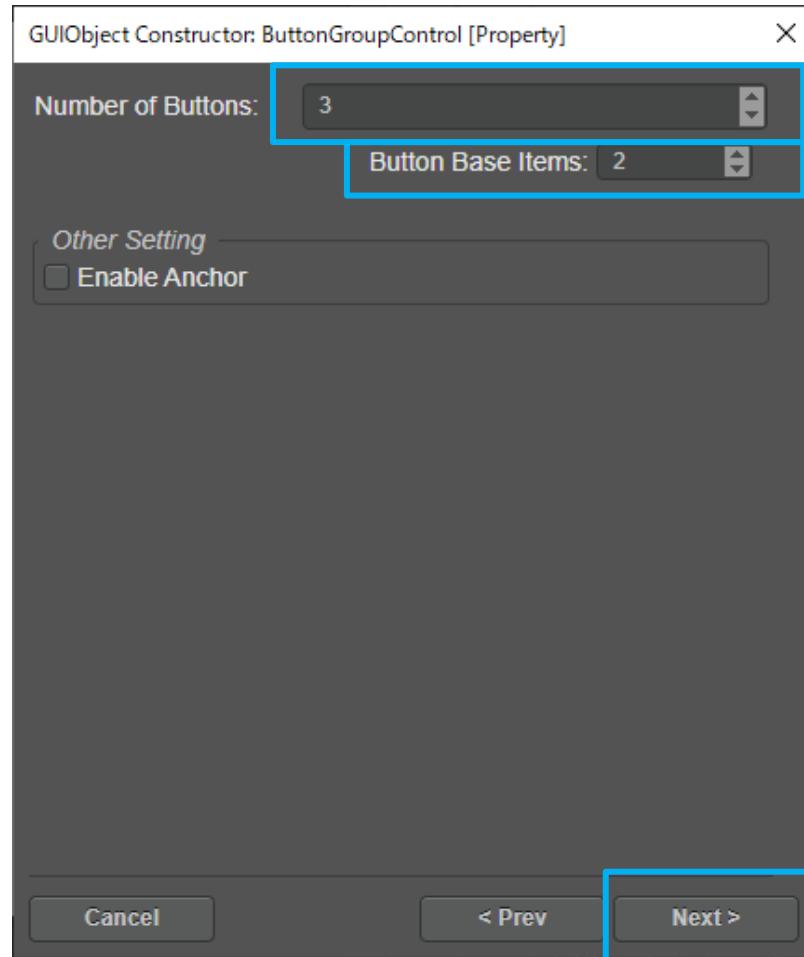
この例では、start画面のメニュー ボタン群という役割を想定して、start\_menuという名前を使用しています。

- “New Document Setting”的 “Width”に「800」、“Height”に「480」を設定します。

**Width**と**Height**はコンテンツ アプリケーションと同じサイズを指定します。この例では、800x480サイズの画面に対し中央に表示されるボタンを作るものと想定しています。

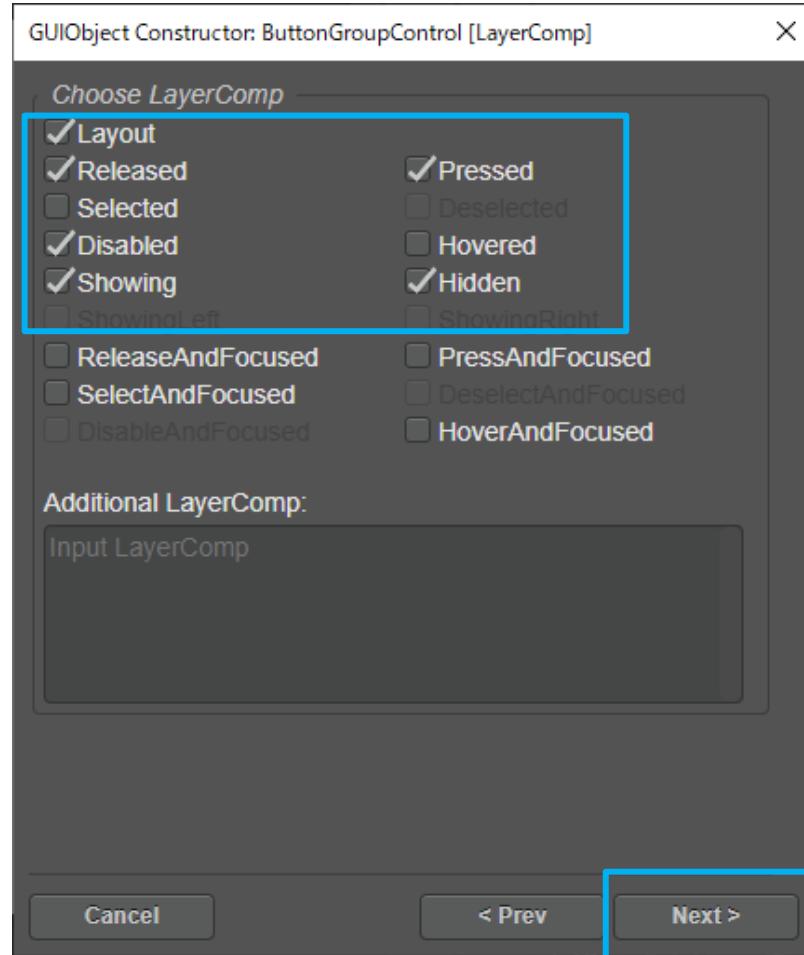
- “Next”ボタンを押下します。

# Step1-2. Property設定



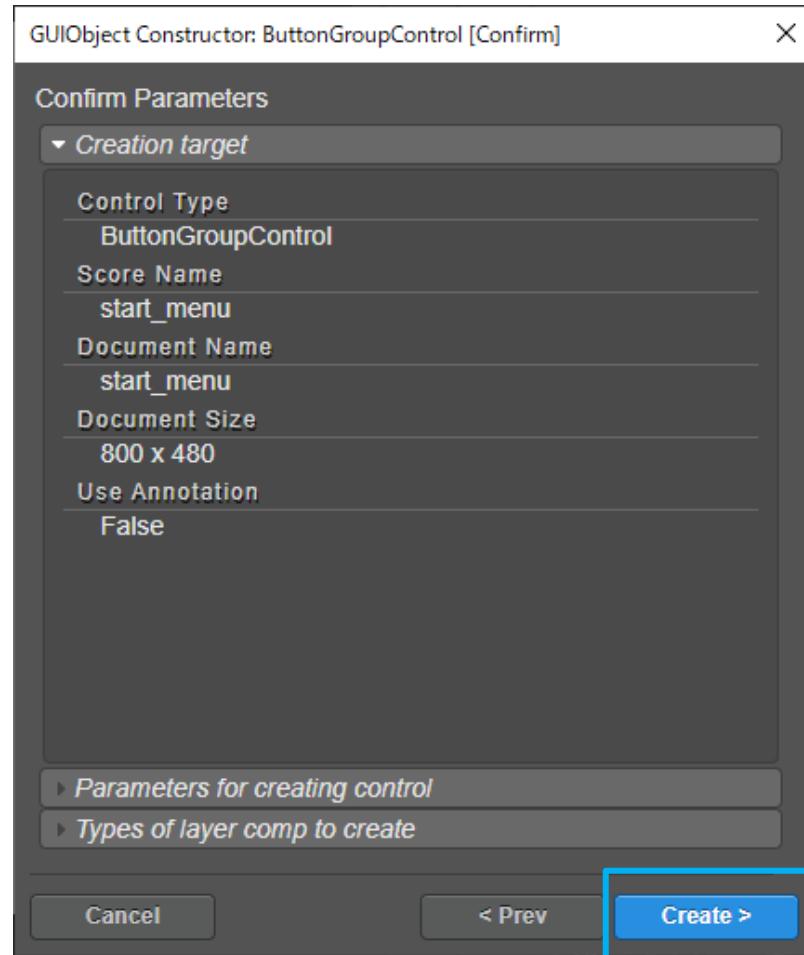
1. “Number of Buttons”に「3」と入力します。  
1つのボタングループが持つボタンの数を設定します。
2. “Button Base Items”に「2」と入力します。  
状態によってボタンの表示を変えたいパターン数を設定します。  
この例では一般的な表示パターンとして「通常時」「タップ時」「非活性時」という3つを想定していますが、非活性時の表示については通常時の表示の透明度を操作したものとするので、「2」としています。
3. “Next”ボタンを押下します。

# Step1-3. LayerComp設定



1. “Choose LayerComp”を左図のようにチェックしてください。  
プラグインの初回起動時には”Disabled”にチェックがありません。  
この例では非活性時の表示も必要なため、 ”Disabled”にチェックを入れてください。
2. “Next”ボタンを押下します。

# Step1-4. 設定内容の確認とレイヤー生成



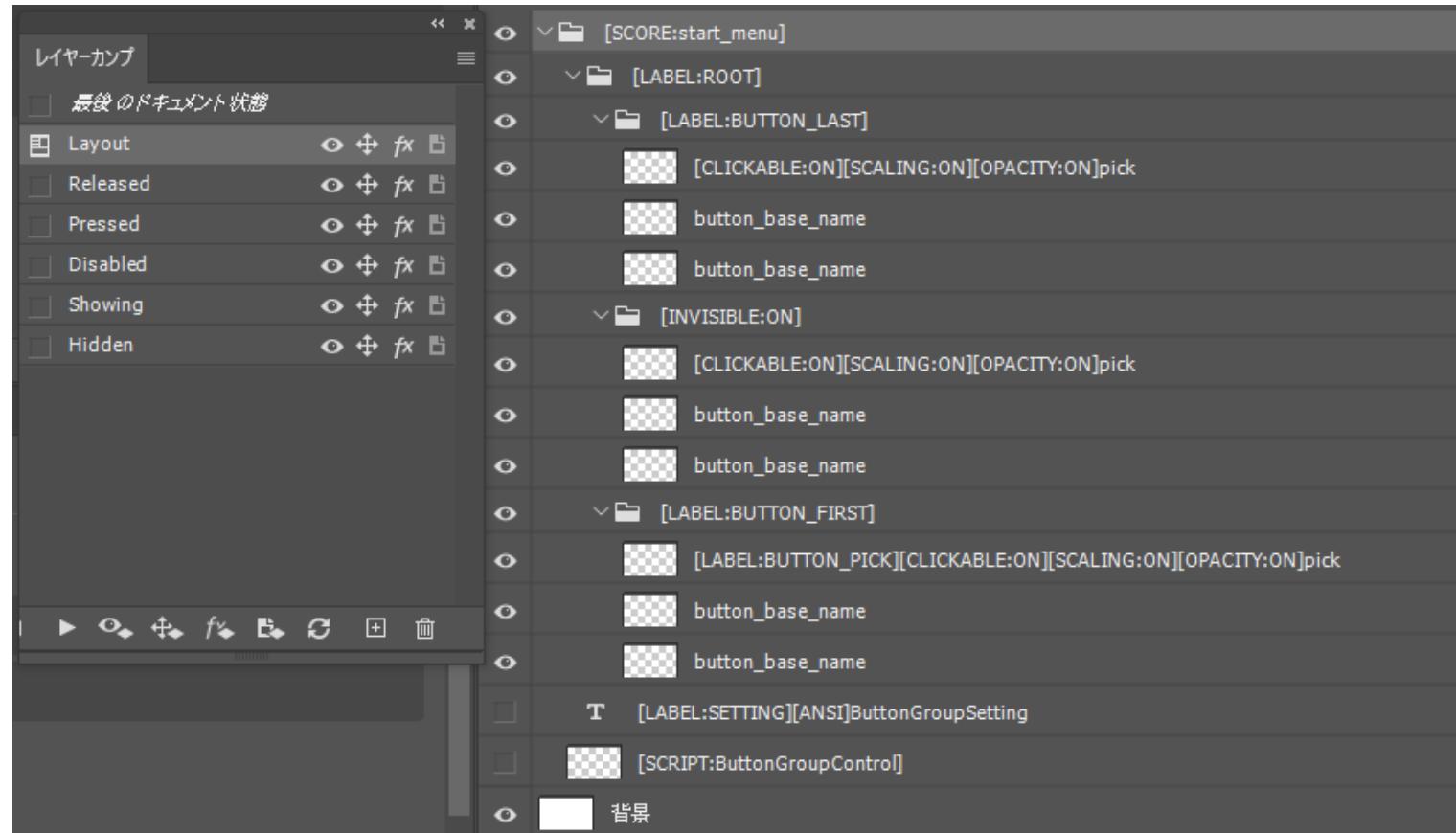
1. レイヤー生成前に、設定した内容を確認してください。
2. “Create>”ボタンを押下します。

設定を変更する場合は”<Prev”ボタンで前画面に戻り、再操作してください

設定内容に基づいてレイヤーが生成されます

# Step1-5. 生成されたレイヤーの確認

1. 次のとおりにレイヤーが生成されたことを確認してください。



選択したレイヤーカンプと、  
GUIObjectのButtonGroupコントロール  
に必要なレイヤー群が生成されます。

Step2

## 各レイヤーへの絵柄の配置

---

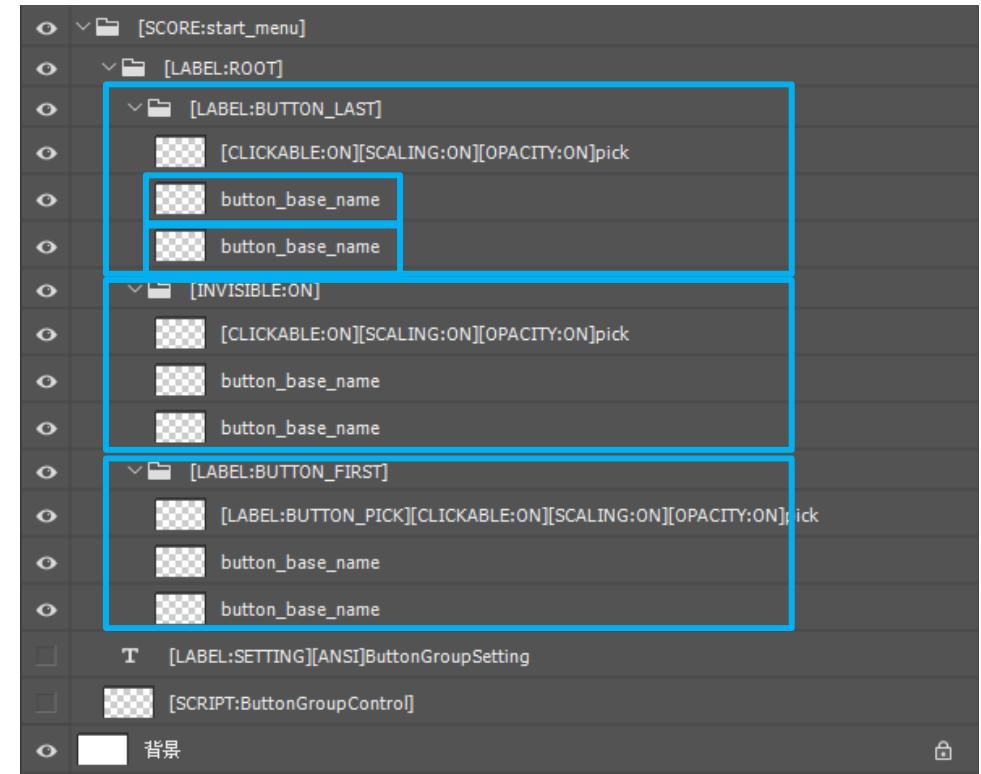
各レイヤーに必要な絵柄を配置する

# Step2-1. ボタン数分のレイヤーグループを確認

- Step2では、各ボタンの2つのレイヤーに「通常時(非活性時)」「タップ時」それぞれの絵柄を置いていきます。

- 生成されたレイヤー群に、"button\_base\_name"という名前のレイヤー2つを含むグループが3つあることを確認してください。

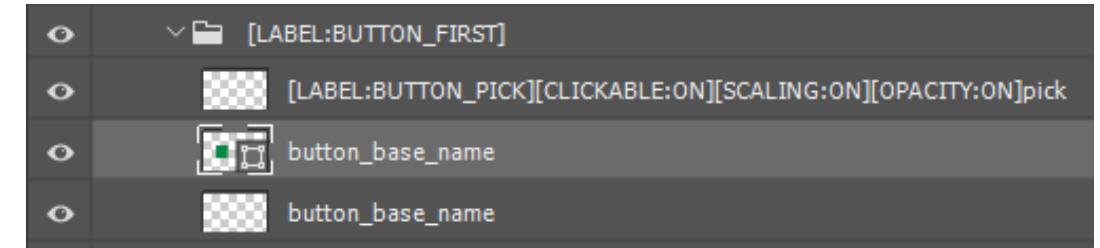
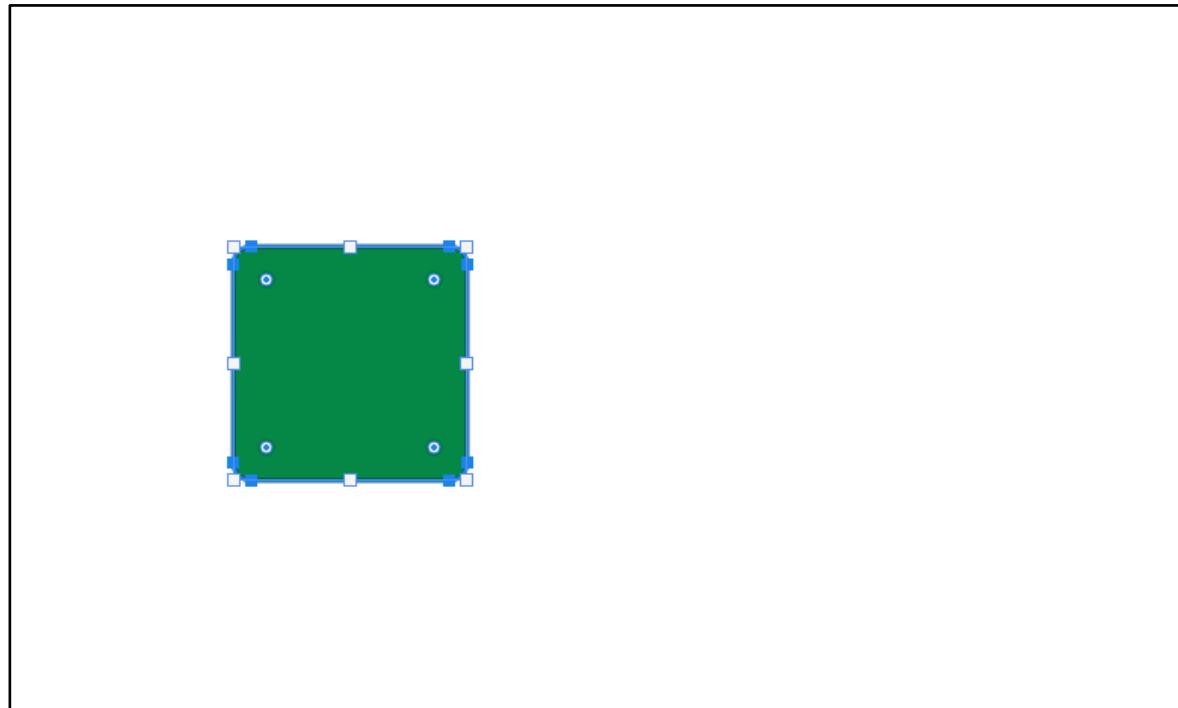
このレイヤーは、**Step1-2で設定した“Number of Buttons”と“Button Base Items”**の数値に応じて生成されます。



## Step2-2. 「通常時」 ボタンをデザインする（1番目）

まずはグループの中の1番目のボタンについて、ベースとなる「通常時」用の物を作成します。

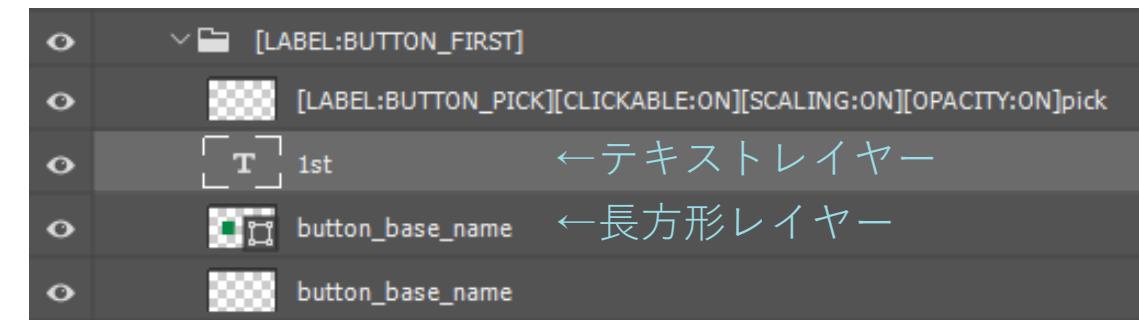
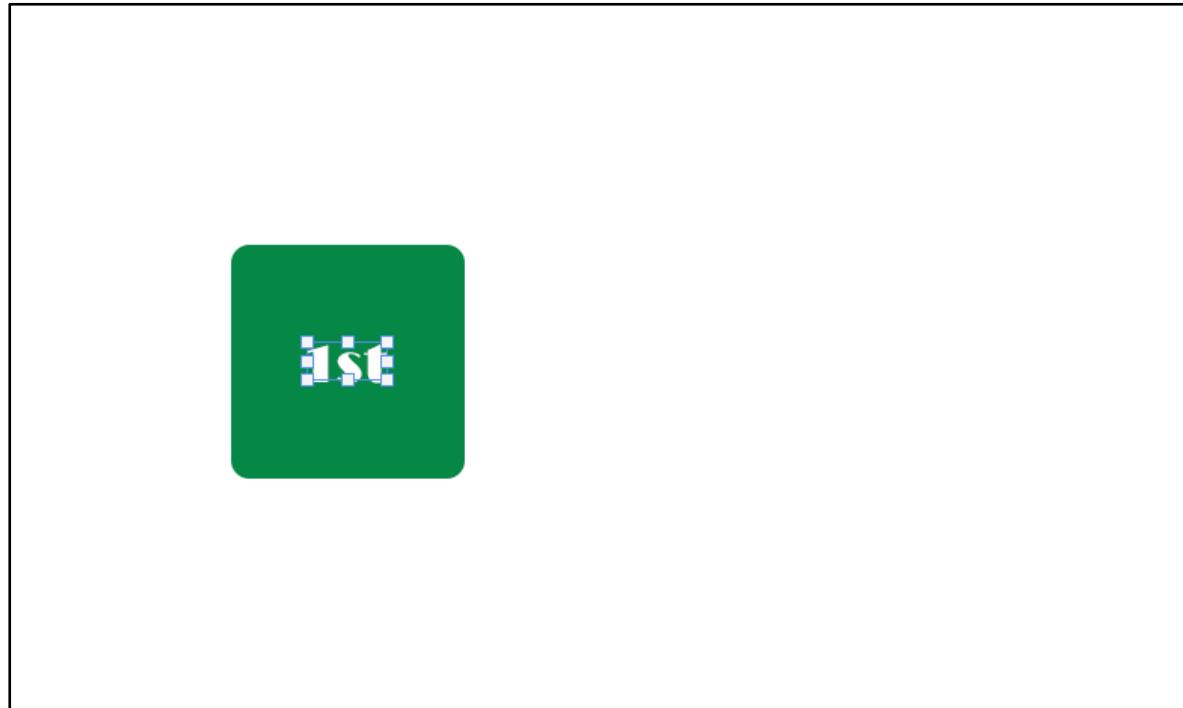
1. “[LABEL:BUTTON\_FIRST]”という名前のレイヤーグループの中、1枚目のレイヤーに155x155（四隅12px丸め）の長方形を画面左側（参考：x:151px, y:163px）に来るよう配置します。



## Step2-2. (続き)

2. 「通常時」ボタンに表示する文言をテキストレイヤーで生成します。

別状態の絵柄用にレイヤーを複製して色を変えられるよう、この段階では長方形と文字レイヤーをラスタライズさせないでおきます。 (【参考】文言："1st"、フォント：Broadway、Regular、36pt)



## Step2-3. 「タップ時」用にレイヤーをコピーする

- 「タップ時」用のレイヤーに「タップ表示」のボタンを配置していきます。  
ここでは作業簡略化のためStep2-2で作成した通常時用レイヤーをコピーして利用します。

1. テキストレイヤーと長方形のレイヤーを複製して順番を揃えてください

レイヤーを右クリックしてコンテキストメニューから「レイヤーを複製…」を実行すると複製できます。

2. プラグインで生成した残りのレイヤーは不要なため、削除します

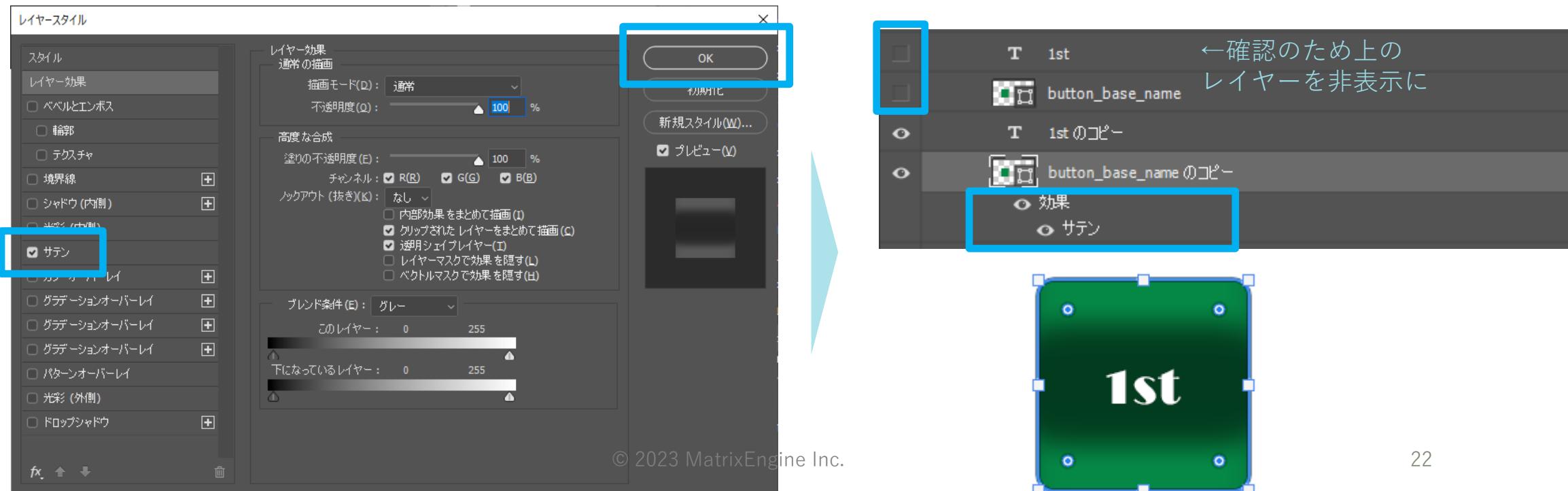
本手順のように、表示パターンのデザイン上レイヤーをコピーして差分を編集することが想定される場合には、Step1-2で Base Itemsを1にしておくと不要レイヤー削除の手間が省けます。



# Step2-4. 「タップ時」ボタンを加工する

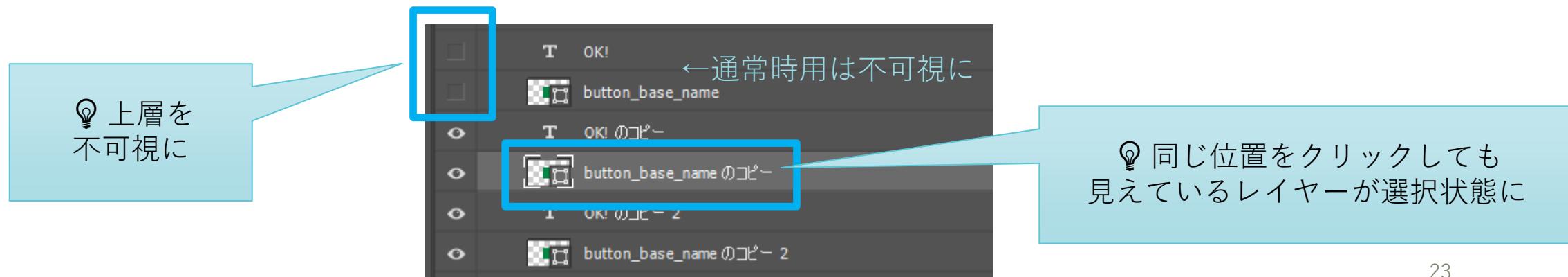
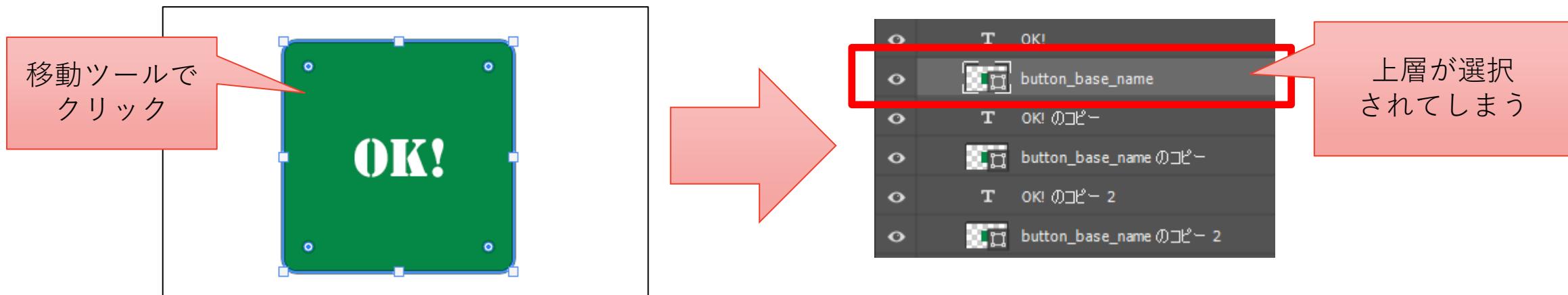
- タップ感を出すため、タップ時用の長方形レイヤーに「サテン」加工を施します。

- タップ時用の長方形レイヤーを右クリックし、"レイヤー効果..."を選択します。
- "レイヤースタイル"ダイアログで"サテン"をチェックします。
- レイヤーにレイヤー効果が付加され、表示が変わったことを確認してください。



## 💡 下層レイヤー編集時は上層レイヤーを不可視に設定する

- 表示画像をクリックした際に上層レイヤーを選択・編集しないように、上層（通常時用）の長方形／テキストレイヤーを不可視にしておくと下層の編集がスムーズになります。



Step3

## タップ判定領域の配置

---

タップ機能の判定に必要なレイヤーマスクを設定する

# Step3-1. 作業レイヤーの確認と準備

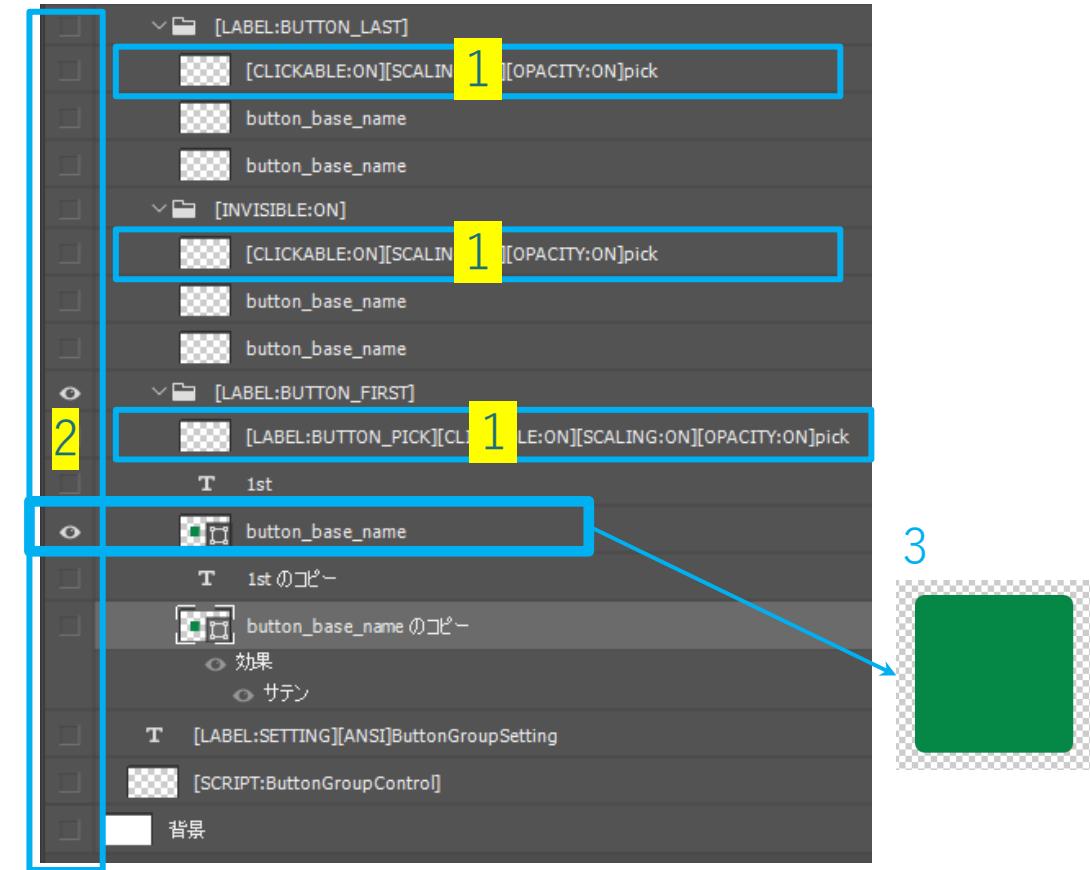
- Step3では、ボタンの「タップ範囲」を示す透明画像をレイヤーマスクを用いて配置していきます。

1. 生成されたレイヤー群で、各レイヤーグループの一番上にタグ（“[LABEL:PICK]”等） + 「pick」 という名前のレイヤーがあることを確認してください。

2. ボタンの位置・サイズに沿ったレイヤーマスクを生成するため、1番目ボタン用「通常時」の長方形レイヤーを除いて不可視にします。右図の通りに設定してください。

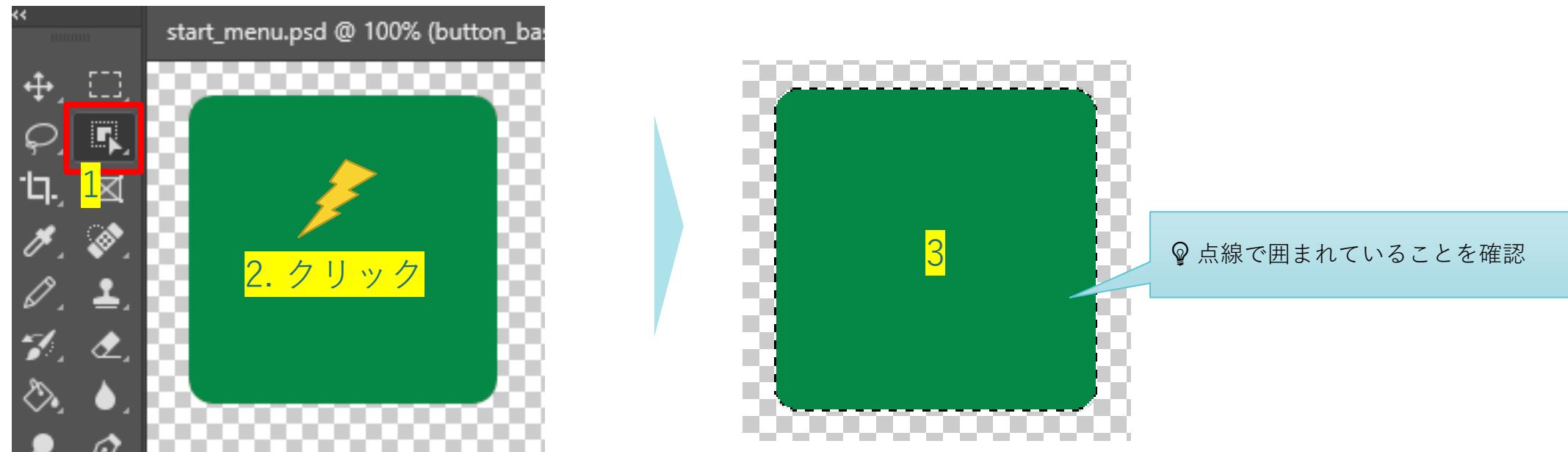
背景は画面サイズフルの範囲を持ち、レイヤーマスクの領域指定の際、邪魔になります。

3. 可視レイヤと画像を確認します。



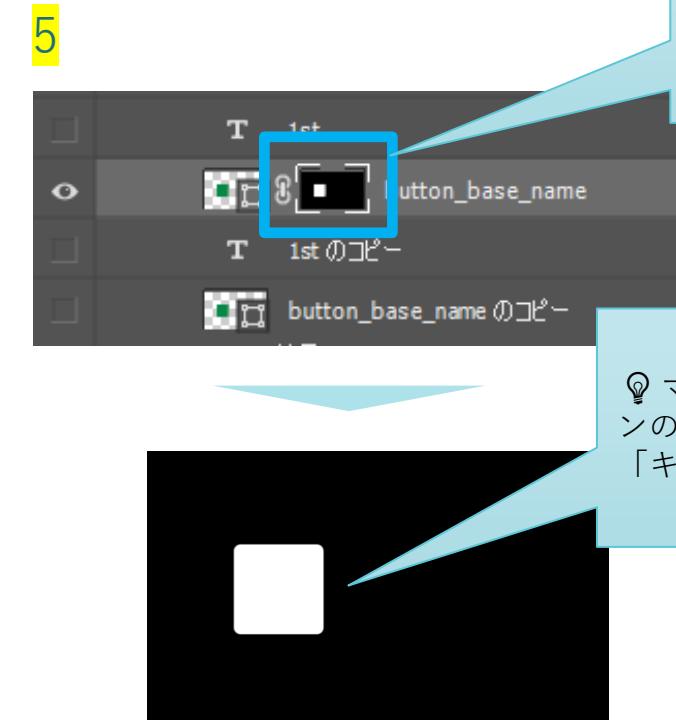
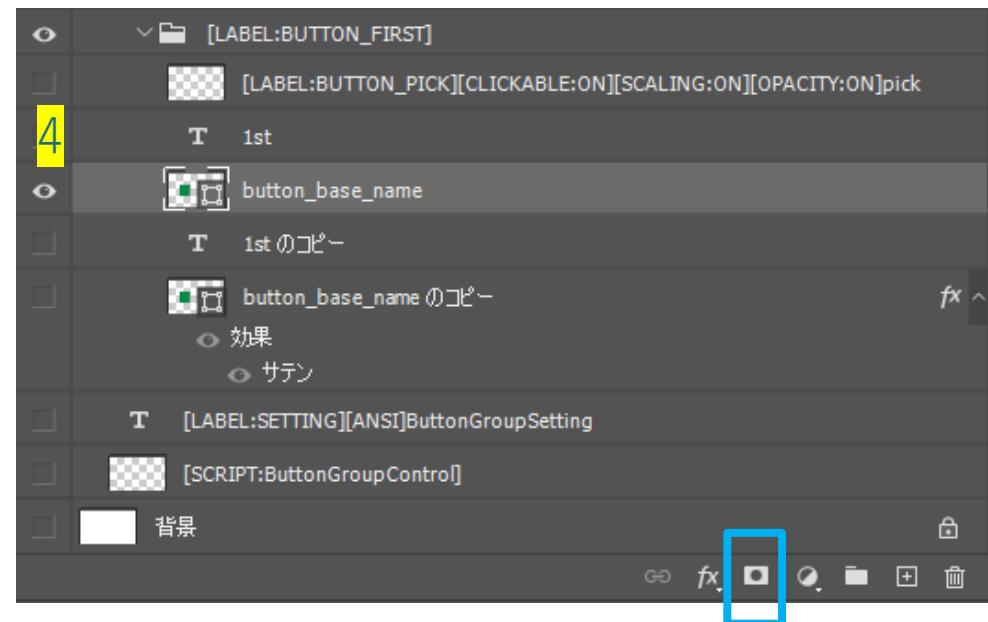
# Step3-2. レイヤーマスクの追加

1. レイヤーマスク生成の元となる長方形を持つレイヤー（通常時ボタンの長方形レイヤー）を選択した状態で”オブジェクト選択ツール”をクリックします。
2. 続いて、レイヤーマスクの作成対象である長方形をクリックします。
3. 長方形オブジェクトの領域が選択中（点線で囲まれる）状態になったのを確認してください。



# Step3-2. レイヤーマスクの追加（続き）

4. 直前の状態のまま、レイヤーリスト下の”レイヤーマスクを追加”をクリックします。
5. 選択中のレイヤーに、同位置・同サイズのレイヤーマスクが追加されます。



💡 レイヤーマスクのアイコンをダブルクリックするとレイヤーマスク画面となり、マスクの位置とサイズが確認できます。

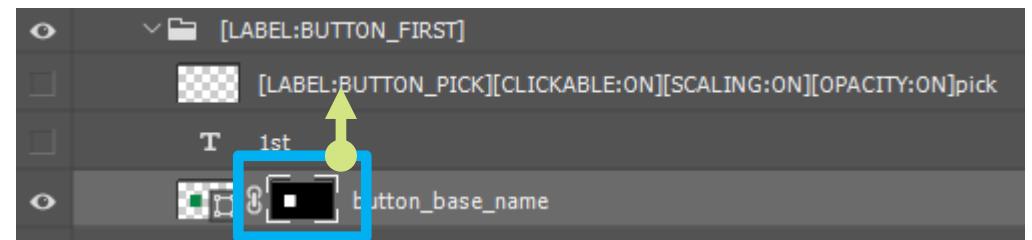
💡 マスクが画面の左側（1番目ボタンの位置）にあることを確認したら「キャンセル」を押下してください。



# Step3-3. レイヤーマスクの移動

- 長方形のレイヤーから生成したレイヤーマスクを、「～pick」という名前が付いた透明レイヤーに移動させます。
- レイヤーマスクを別のレイヤーに移動させるには、レイヤーマスクアイコンをドラッグします。

1. レイヤーマスクアイコンを「～pick」という名前の  
レイヤー上にドラッグ＆ドロップします。



2. レイヤーマスクアイコンが「～pick」レイヤー上に  
あることを確認してください。

タップ判定領域用のレイヤーに、ボタンと同じサイズ・  
位置の透明画素の長方形が出来ました。



Step4

## ラスタライズとレイヤーの整理

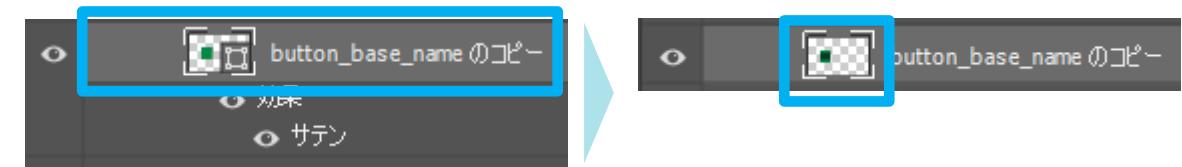
---

SDKに直接取り込めないエフェクト等をラスタライズしてレイヤーをまとめる

# Step4-1. ラスタライズ

- タップ時の表現であるサテン処理とレイヤー効果は、そのままではSDKにインポート出来ません。  
次の処理を加える必要があります。

1. 2番目（タップ時用）の長方形レイヤーを右クリックして、コンテキストメニューから”レイヤースタイルをラスタライズ”を選択してください。



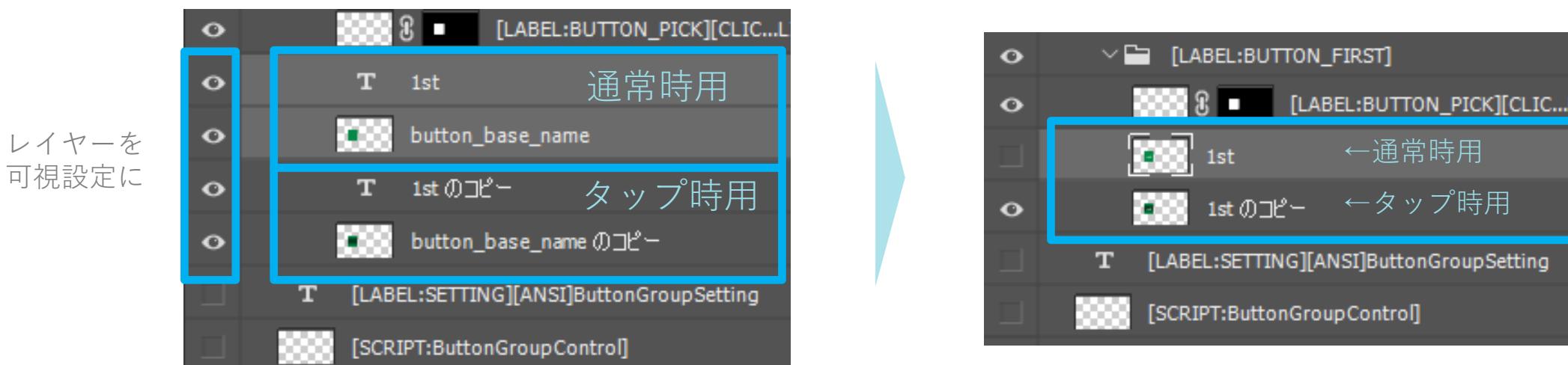
2. 残り長方形のレイヤーも固定します。レイヤーを右クリックして”レイヤーをラスタライズ”を実行してください。



# Step4-2. レイヤーを結合する

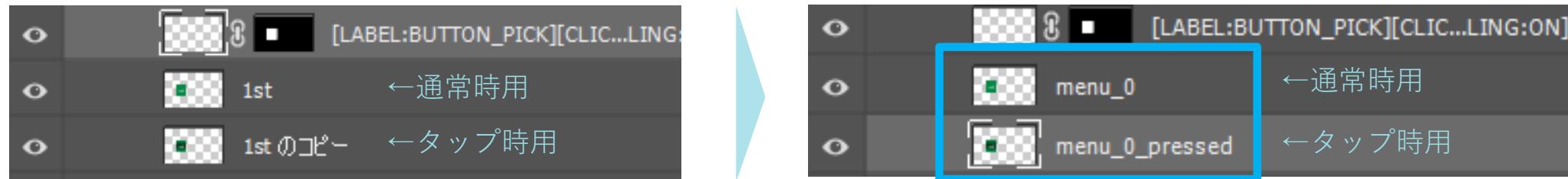
- 「通常時用」「タップ時用」それぞれについて、長方形レイヤーとテキストレイヤーを一つに結合します。
- レイヤーの結合にはコンテキストメニューを利用します。各レイヤーを可視状態にして操作してください。

- 結合するレイヤー2つ（長方形とテキスト）を選択した状態で右クリックして、コンテキストメニューから“レイヤーを結合する”を選択します。通常時用、タップ時用のそれぞれで実行してください。



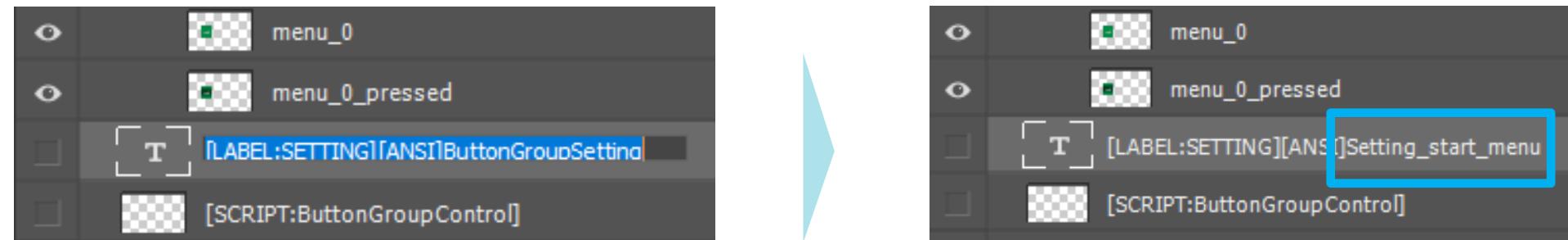
# Step4-3. レイヤー名を編集する①

- ボタンの各状態の区別がしやすいようにレイヤー名を編集します。
  - レイヤー名を編集するにはレイヤー名部分をダブルクリックします。
- レイヤー名をダブルクリックすると、名前変更状態になります。「通常時」「タップ時」の各状態に合わせて、レイヤー名をそれぞれ「menu\_0」「menu\_0\_pressed」に変更してください。  
**【参考】名前ルールの一例：menu\_[インデックス]\_[状態]**



# Step4-4. レイヤー名を編集する②

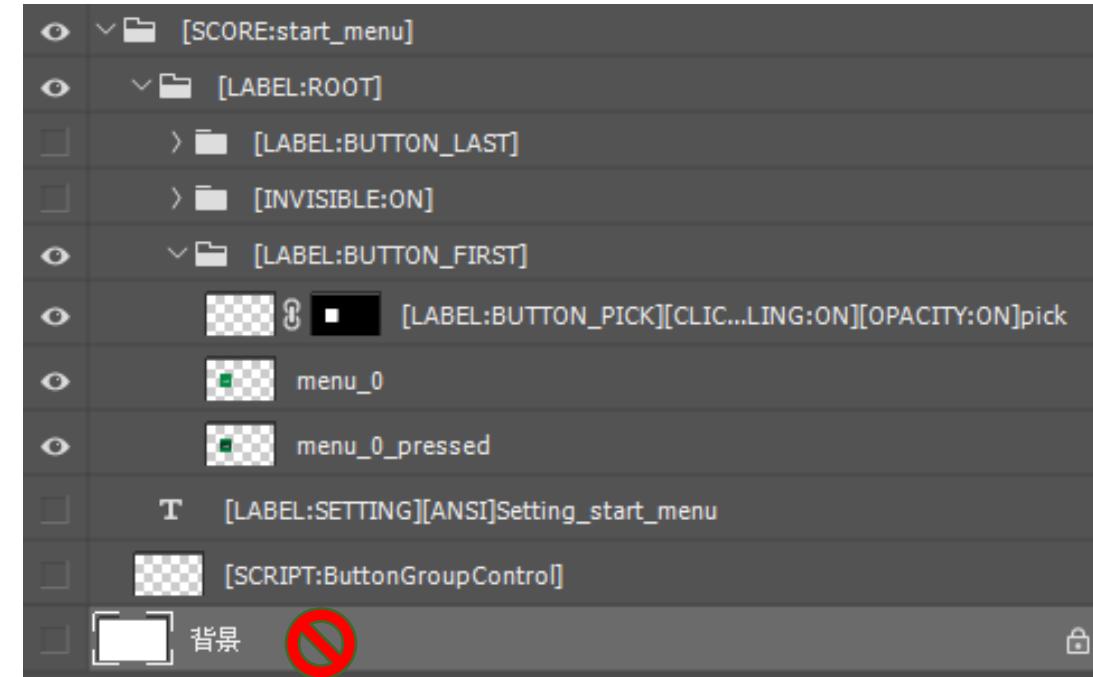
- 自動生成されたレイヤー群には、アニメーション等の設定を予め記述している特別なテキストレイヤーがあります。このレイヤー名をUI名称に合わせて編集します。
  - レイヤー名を編集するにはレイヤー名部分をダブルクリックします。
- 「[LABEL:SETTING][ANSI]ButtonGroupSetting」というレイヤーがあることを確認してください。
  - レイヤー名をダブルクリックすると、名前変更状態になります。レイヤー名を「[LABEL:SETTING][ANSI] Setting\_start\_menu」に変更してください。  
(Constructorでの生成時に"Score Name", "Document Name"として使用した名前(=psdファイル名)を使うと他のコントロールとの重複を回避できます。) 【参考】 [Setting\\_\[スコア名\]](#)



# Step4-5. 不要レイヤーを削除する

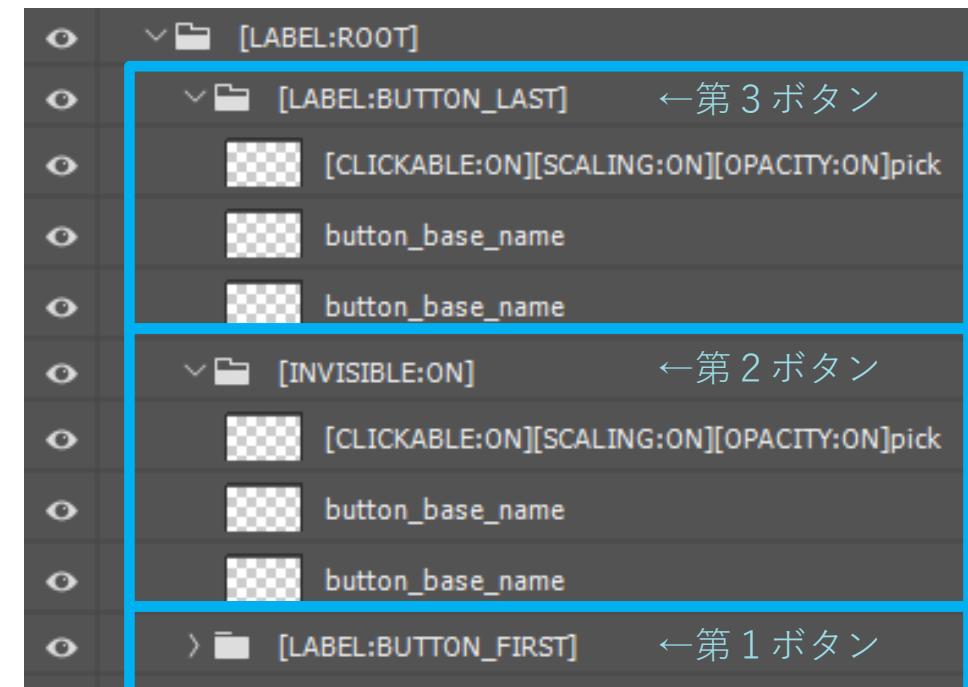
- 不要な背景レイヤーを削除します。レイヤーの削除にはコンテキストメニューを利用します。

1. 「背景」レイヤーを右クリックし、コンテキストメニューから”レイヤーを削除”を選択してください。



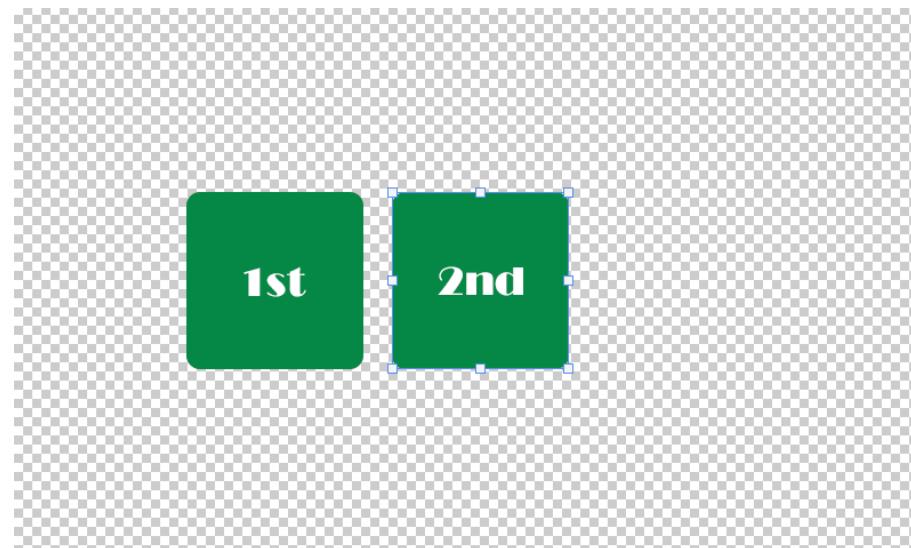
# Step4-6. 第2,第3ボタンを作成する

- Step2-2から4-3までを繰り返して、第2、第3ボタンを作成します。
1. これまでの工程を繰り返して、2番目、3番目のボタンについても、同様に通常画像レイヤー、タップ時用画像レイヤー、レイヤーマスクを作成してください。（それぞれのデータは次ページに続く）



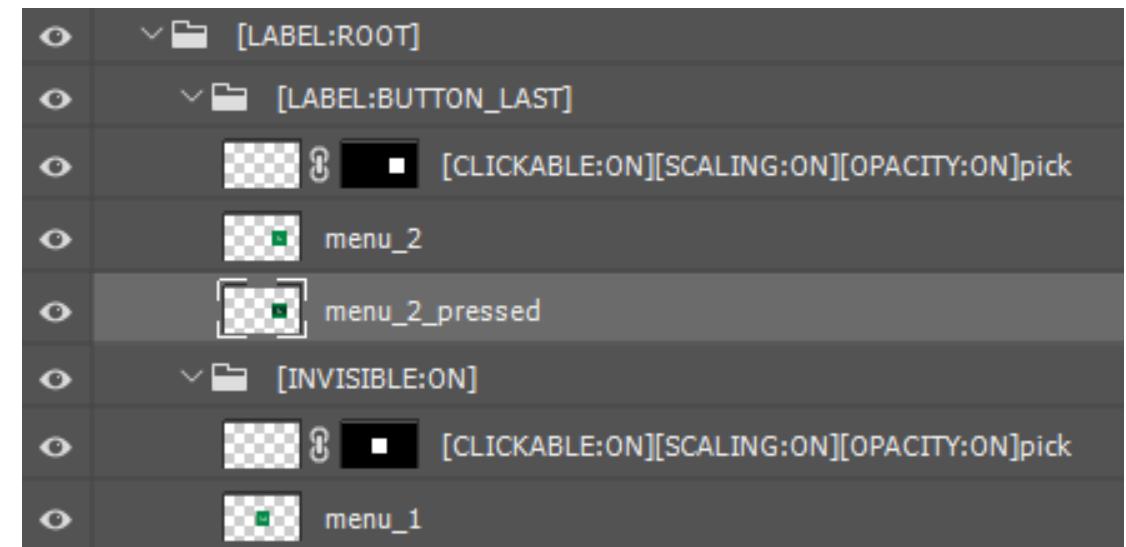
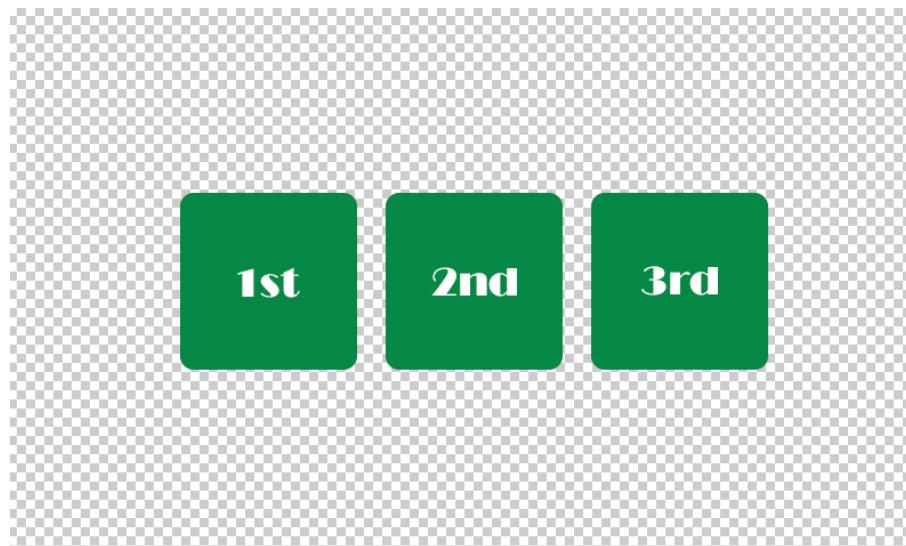
# Step4-6. (続き)

- 第2ボタン：155x155（四隅12px丸め）、位置：x:331px, y:163px  
テキスト：2nd  
レイヤ名：menu\_1[\_pressed]



# Step4-6. (続き)

- 第3ボタン：155x155（四隅12px丸め）、位置：x:511px, y:163px  
テキスト：3rd  
レイヤ名：menu\_2[\_pressed]



Step5

# 状態ごとの表示切り替えの設定

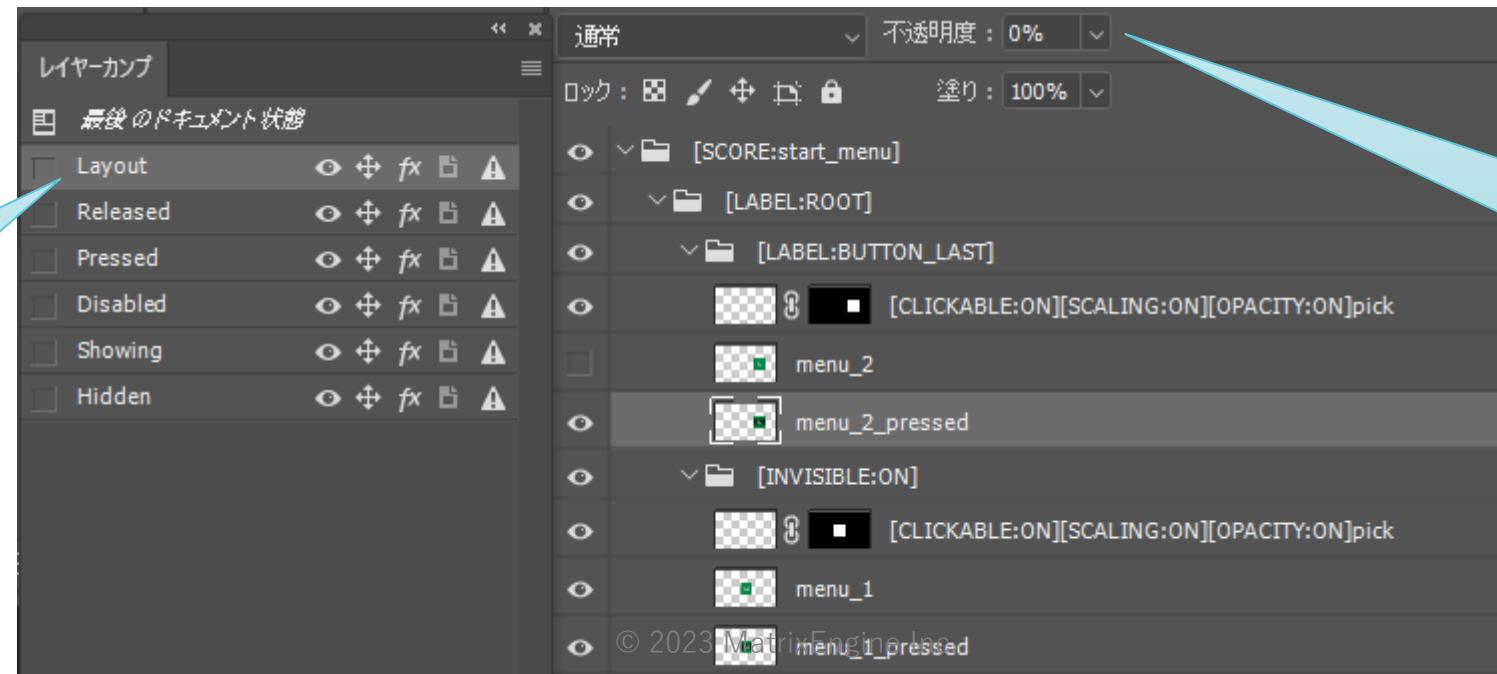
---

ボタンの状態に応じたレイヤーカンプを作成する

# Step5-1. 予備知識（レイヤーカンプ）

- Step5では、ボタンの状態に応じた表示切り替えを実現するために、異なる不透明度のレイヤーの組み合わせ（レイヤーカンプ）を状態の数だけ設定します。

💡 例えば「ボタンが押された状態（タップ時）」では、menu\_[0/1/2]\_pressedレイヤーを不透明に、menu\_[0/1/2]レイヤーを透明にします。このように、状態に応じた表示パターンを作成していきます。



# Step5-2. 通常時の不透明度設定

- 通常表示用に、各レイヤー (menu\_[0/1/2] / menu\_[0/1/2]\_pressed) の不透明度を設定します。

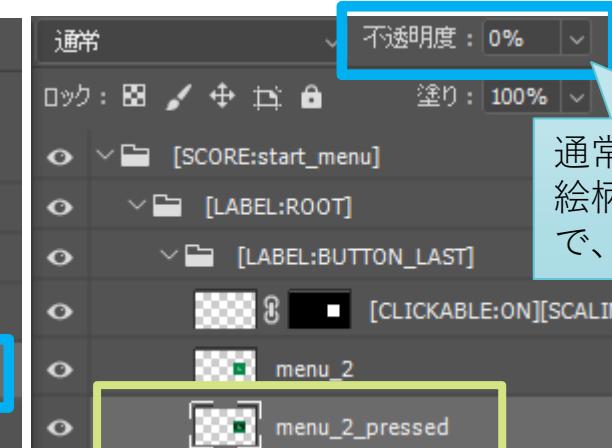
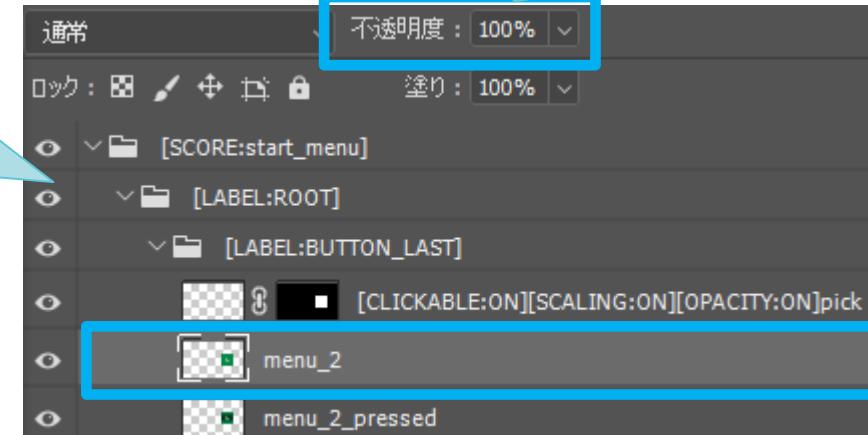
1. レイヤー不透明度を次のように設定してください。

**menu\_[0/1/2] : 100%、menu\_[0/1/2]\_pressed : 0%**

2. [LABEL:SETTING], [SCRIPT:ButtonGroupControl]以外のレイヤーの可視設定をOnにしてください。

レイヤーを選択して設定。通常時用の絵柄を表示したいので、menu\_[0/1/2]レイヤーの不透明度を100%に。

[LABEL:SETTING]  
[SCRIPT:ButtonGroupControl]  
以外のレイヤーの可視設定をOn



通常時に「タップ時」の絵柄は表示したくないので、不透明度を0%に。

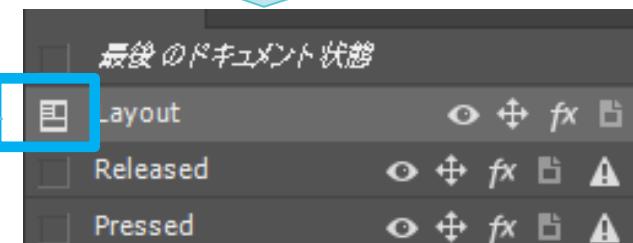
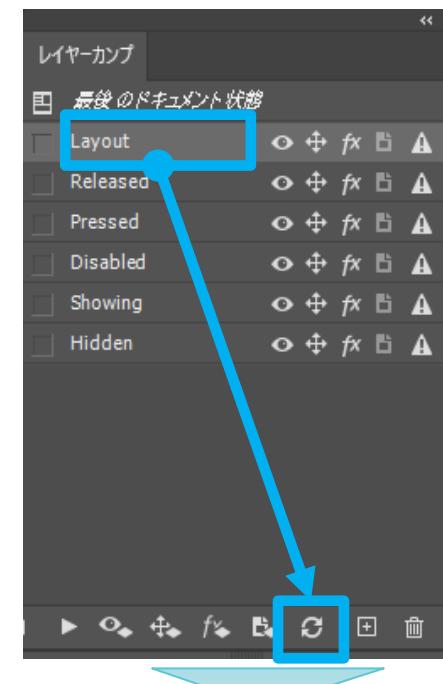
# Step5-3. 通常時のレイヤーカンプ登録

- レイヤーの不透明度の組み合わせパターンをレイヤーカンプとして登録します。

1. Layoutレイヤーカンプの名前の箇所を選択してください  
左端のボックスは選択しないようご注意ください。

2. 下段のリロードマークを選択してください。  
不透明度の組み合わせがLayoutレイヤーカンプとして  
設定されます。  
以降、Layoutの左側のボックスを選択すれば、登録し  
た100%, 0%の組み合わせを呼び出せます。

Layoutレイヤーカンプ（100%, 0%）の呼び出しが可能に



# Step5-3. 続き

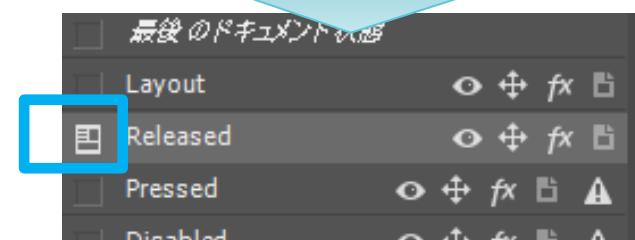
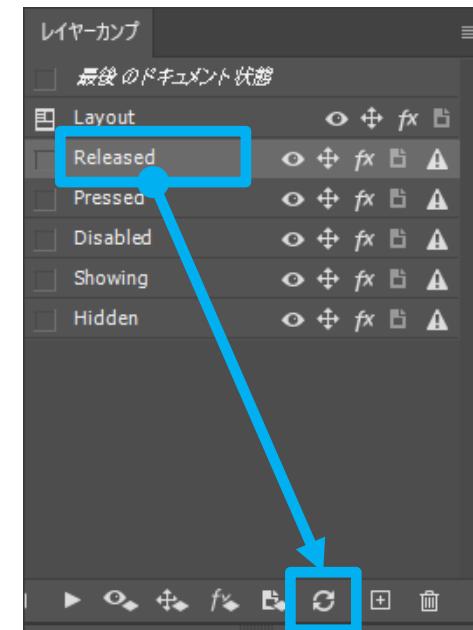
- 一般に（ボタン等）コントロールの通常表示パターンとして、"Layout"" Released"レイヤーカンプを使用します。通常表示状態をReleasedレイヤーカンプにも登録します。

1. Releasedレイヤーカンプの名前の箇所を選択してください

**左端のボックスは選択しないようご注意ください。**

2. 下段のリロードマークを選択してください。

**不透明度の組み合わせがReleasedレイヤーカンプとして設定されます。**



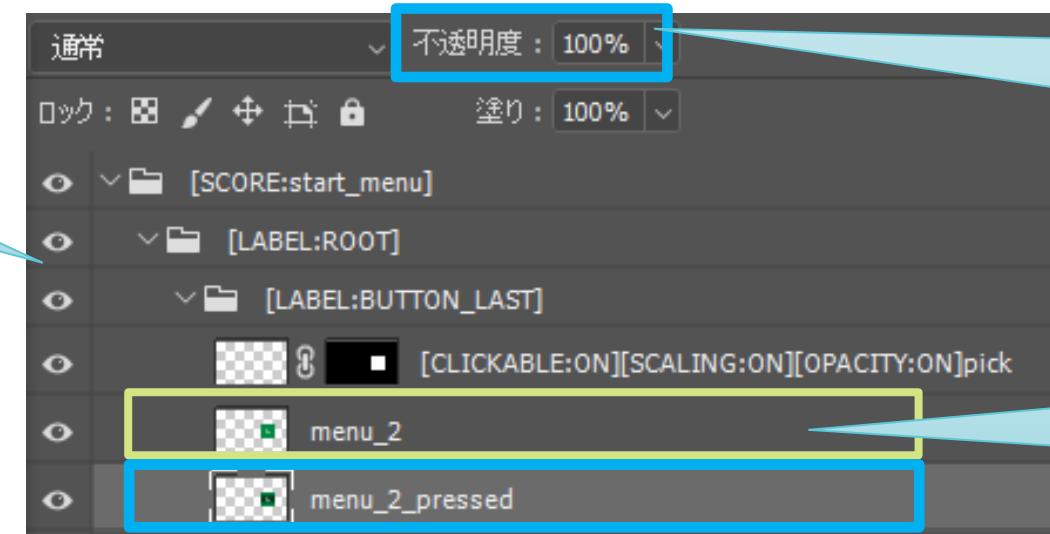
# Step5-4. タップ時の不透明度設定

- タップ時用に、各レイヤー (menu\_[0/1/2] / menu\_[0/1/2]\_pressed) の不透明度を設定します。

1. レイヤー不透明度を次のように設定してください。

**menu\_[0/1/2] : 0%、menu\_[0/1/2]\_pressed : 100%**

2. [LABEL:SETTING], [SCRIPT:ButtonGroupControl]以外のレイヤーの可視設定をOnにしてください。



レイヤーを選択して設定。  
タップ時用の絵柄を表示したいので、  
menu\_[0/1/2]\_presssedレイヤーの  
不透明度を100%に。

タップ時に「通常時」の絵柄は表示  
したくないので、不透明度を0%に。

# Step5-5. タップ時のレイヤーカンプ登録

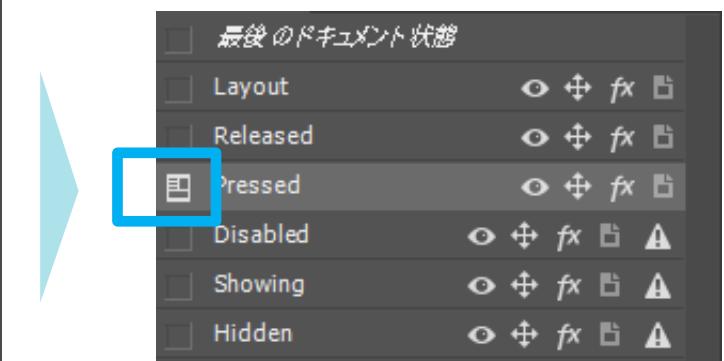
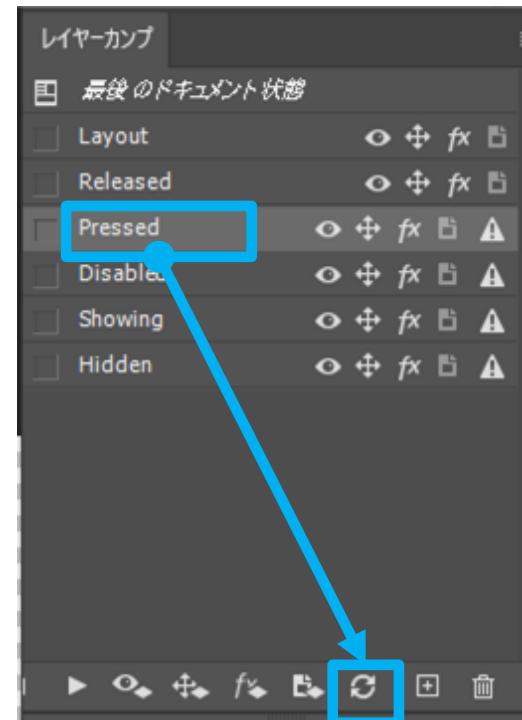
- レイヤーの不透明度の組み合わせパターンをレイヤーカンプとして登録します。

1. Pressedレイヤーカンプの名前の箇所を選択してください

左端のボックスは選択しないようにご注意ください。

2. 下段のリロードマークを選択してください。

不透明度の組み合わせが Pressed レイヤーカンプとして設定されます。



# Step5-6. 非活性時の不透明度設定

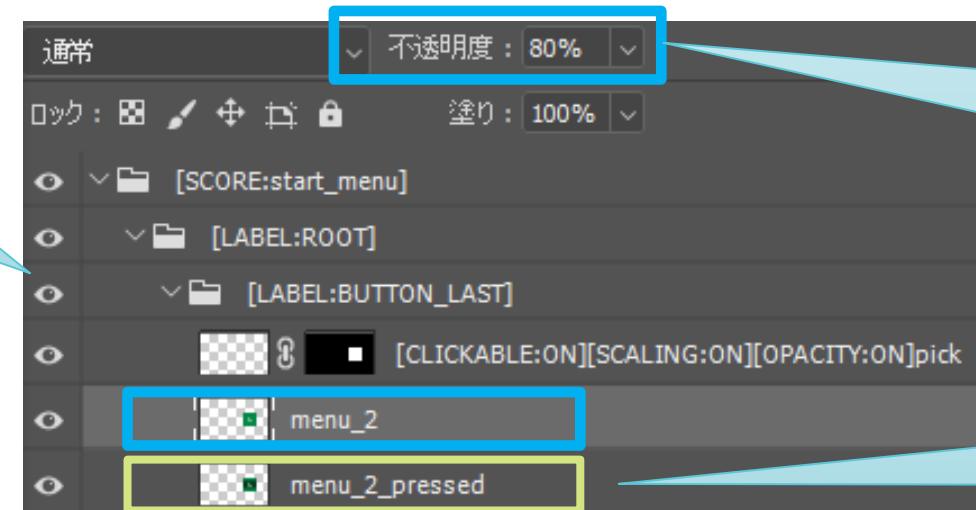
- 非活性時用に、各レイヤー (menu\_[0/1/2] / menu\_[0/1/2]\_pressed) の不透明度を設定します。

1. レイヤー不透明度を次のように設定してください。 (半透明表示で非活性を表現)

**menu\_[0/1/2] : 80% (半透明表示で非活性を表現) 、 menu\_[0/1/2]\_pressed : 0%**

2. [LABEL:SETTING],[SCRIPT:ButtonGroupControl]以外のレイヤーの可視設定をOnにしてください。

[LABEL:SETTING]  
[SCRIPT:ButtonGroupControl]  
以外のレイヤーの可視設定を  
On



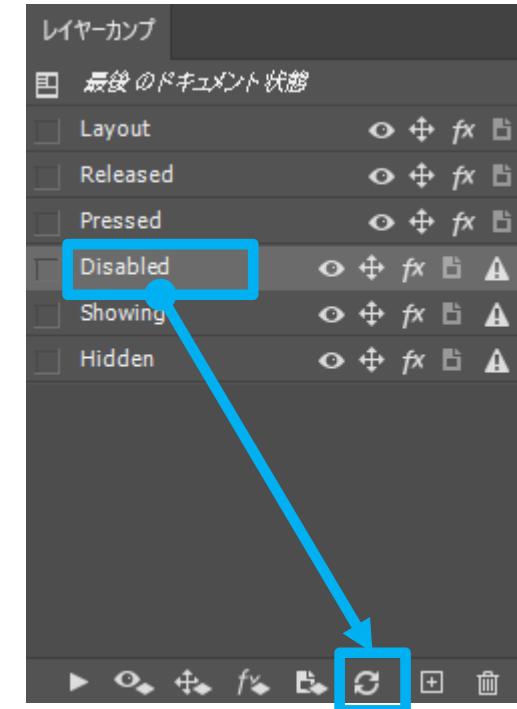
レイヤーを選択して設定。  
半透明表示で非活性を表現したいので、  
menu\_[0/1/2]レイヤーの  
不透明度を80%に。

非活性時に「タップ時」の絵柄は表示  
したくないので、不透明度を0%に。

# Step5-7. 非活性時のレイヤーカンプ登録

- レイヤーの不透明度の組み合わせパターンをレイヤーカンプとして登録します。

- Disabledレイヤーカンプの名前の箇所を選択してください。  
*左端のボックスは選択しないようにご注意ください。*
- 下段のリロードマークを選択してください。  
*不透明度の組み合わせが Disabledレイヤーカンプとして設定されます。*



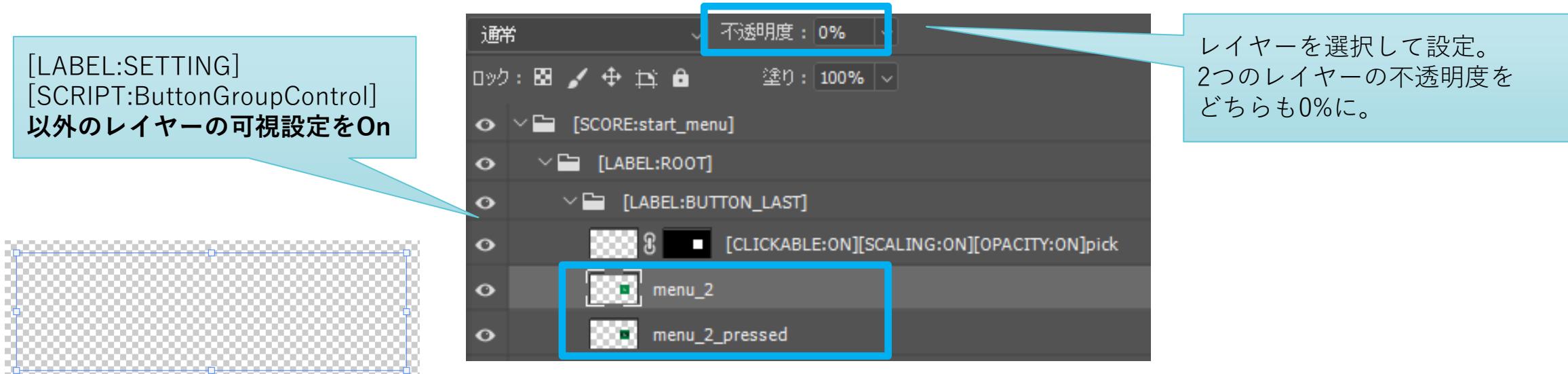
# Step5-8. 非表示時の不透明度設定

- 最後に、出現処理の冒頭と消去処理の最後（つまり非表示の状態）の組み合わせを作成します。

1. レイヤー不透明度を次のように設定してください。

**menu\_[0/1/2] : 0%、menu\_[0/1/2]\_pressed : 0%**

2. [LABEL:SETTING],[SCRIPT:ButtonGroupControl]以外のレイヤーの可視設定をOnにしてください。

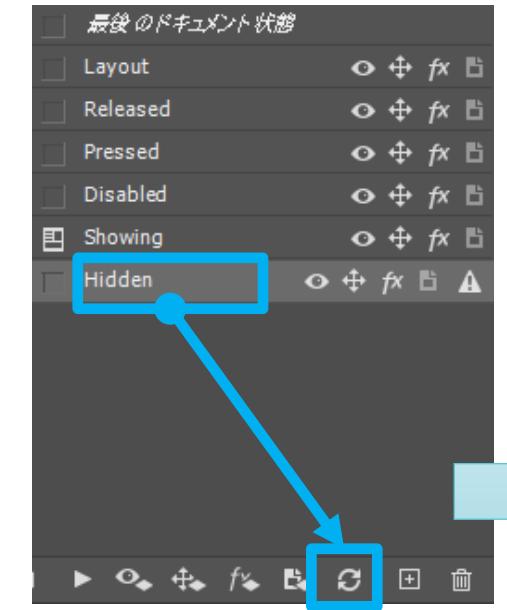
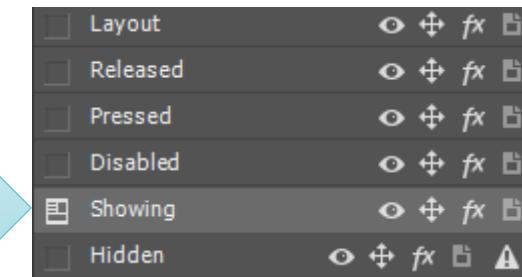
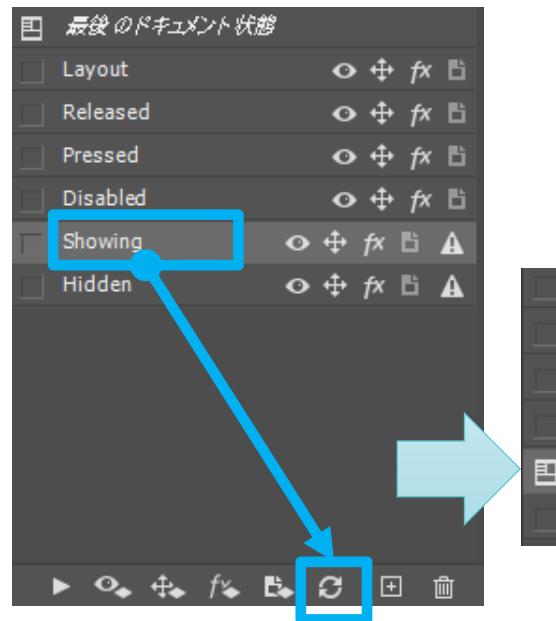


# Step5-9. 非表示時のレイヤーカンプ登録

- レイヤーの不透明度の組み合わせパターンをレイヤーカンプとして登録します。

- Showingレイヤーカンプの名前の箇所を選択して、下段のリロードマークを選択してください。
- 続いて、Hiddenレイヤーカンプの名前の箇所を選択して、下段のリロードマークを選択してください。

不透明度0の組み合わせが ShowingおよびHiddenレイヤーカンプとして設定されました。



Step6

## 動作パラメータの編集

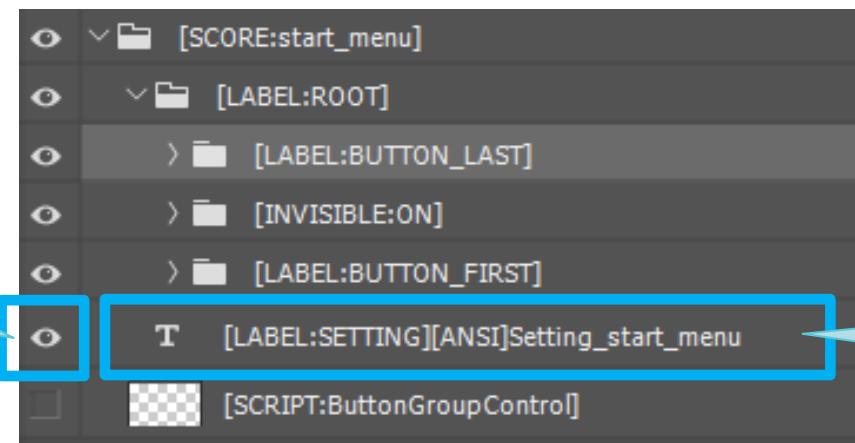
---

SDKへ取り込み後の動作パラメータを記述したテキストレイヤーを編集する

# Step6-1. パラメータ定義用テキストレイヤーの確認

- 自動生成されたレイヤー群には、アニメーション等の設定を予め記述している特別なテキストレイヤーがあります。
- PSDファイルをMxE SDKへインポートする際に、MxE SDKはこのテキストレイヤーに記述されたテキスト内容をオブジェクト（この場合はボタン）の動作パラメータとして読み取り、取り込みます。
- Step6では、テキストの内容を編集していきます。

- [LABEL:SETTING]と名前の付いたテキストレイヤーがあることを確認してください。
- テキストレイヤーを可視状態にします



可視設定をOnに

[LABEL:SETTING]～という名前のテキストレイヤー  
(Step4-4で名前変更を行ったレイヤーです)

# Step6-2. テキスト内容をエディタへコピーする

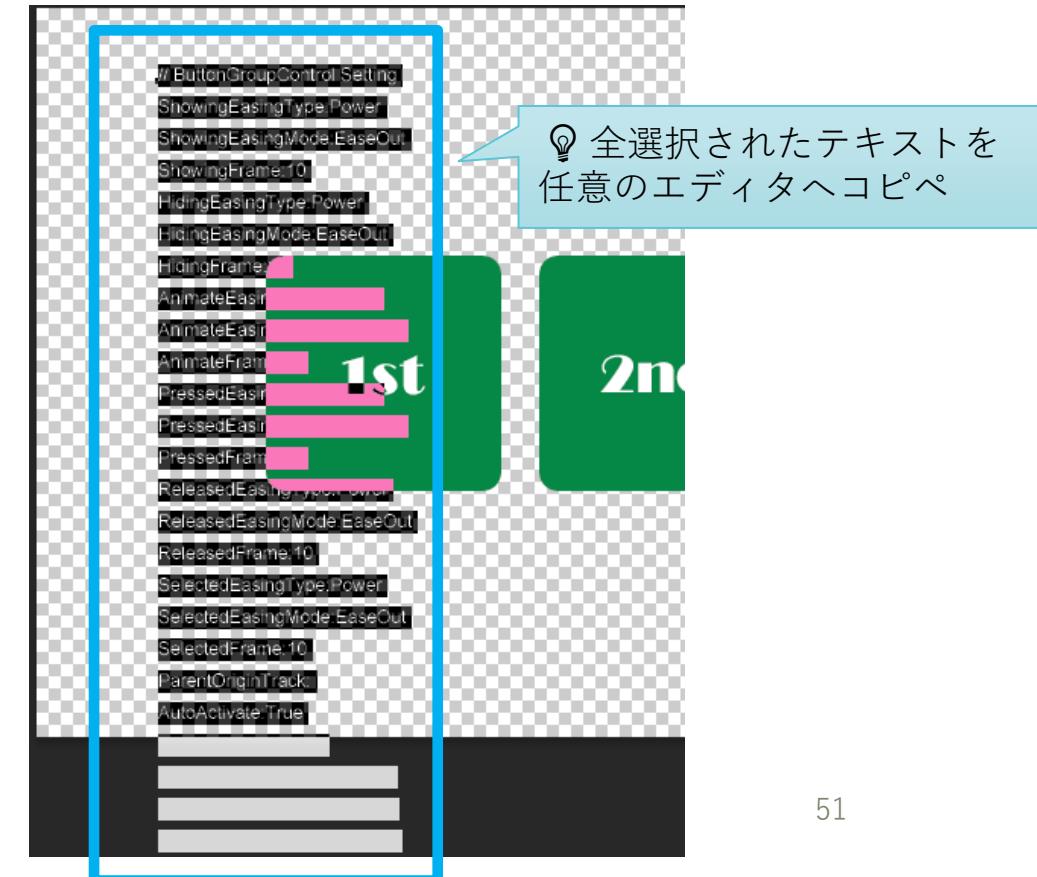
- パラメータの編集は任意のテキストエディタで行います。次の手順で、テキストレイヤーのテキスト内容をコピーできます。

- [LABEL:SETTING]と名前の付いたテキストレイヤーを選択し、"横書き文字ツール"をクリックします。
- 横書き文字ツールを有効にした状態で「Ctrl+A」を実行してください。テキスト内容が全選択されます。
- 全選択したテキストを、任意のテキストエディタへコピー＆ペーストします。



💡 レイヤーを選択

💡 "横書き文字ツール"選択後「Ctrl+A」実施



# Step6-3. 動作パラメータを編集する

- パラメータには様々な項目がありますが、ここでは常時表示設定のOn/Offを制御するパラメータを編集します。

- エディタ上で「AutoActivate」の設定値を「False」に書き換えてください。  
編集後のテキストはパラメータ定義用のテキストレイヤーに戻します。特にテキストファイルに保存する必要はありません。
- エディタ上で「Ctrl+A」を実行してテキストを全選択します。その後「Ctrl+C」を実行して全テキストをコピーしてください。

```
1 // ButtonGroupControl Setting  
2 ShowingEasingType:Power  
3 ShowingEasingMode:EaseOut  
4 ShowingFrame:10  
5 HidingEasingType:Power  
19 SelectedFrame:10  
20 ParentOriginTrack:  
21 AutoActivate:False  
22 FocusEnabled:False  
23 FocusPrevNextEnabled:True
```

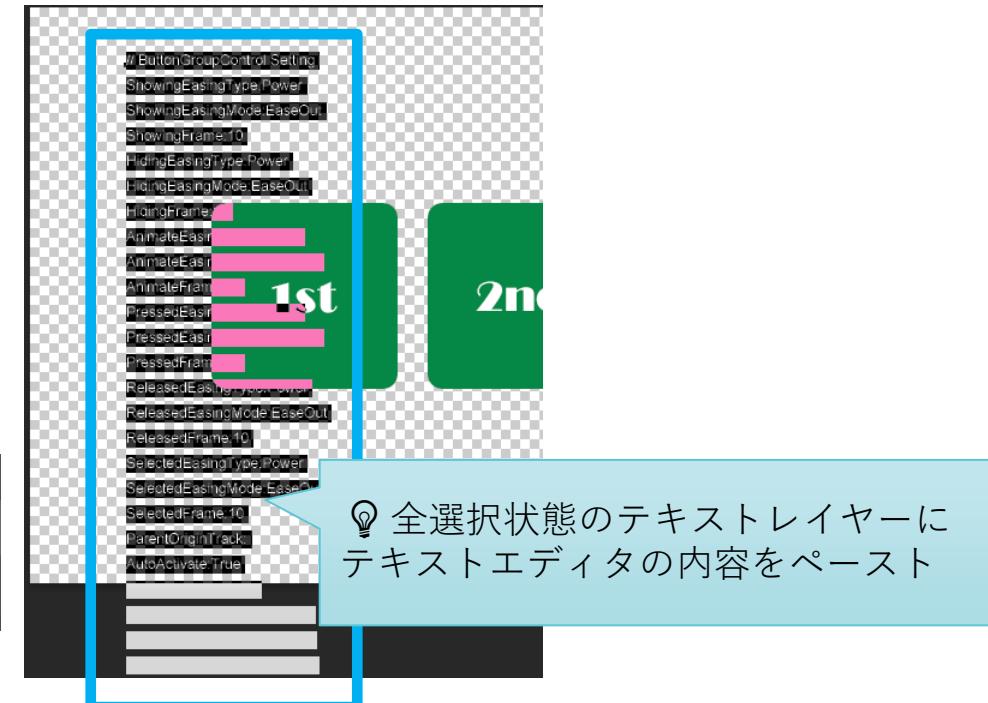
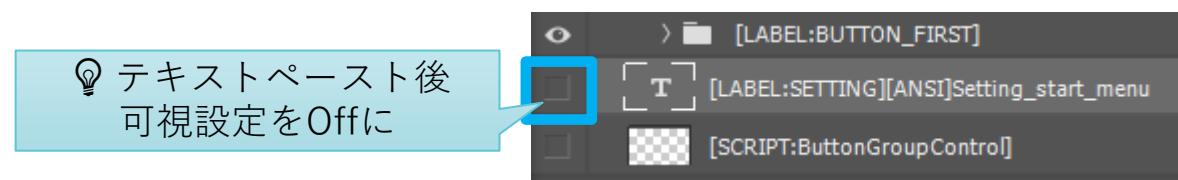
💡 「False」に

💡 書き換えたテキストを全選択し、  
テキストレイヤーにコピペします。

## Step6-4. 変更したテキストをレイヤーへコピーする

- Photoshopに戻り、テキストレイヤーの内容を変更後のパラメータテキストで上書きします。

- 全選択状態 ([Step6-2](#)) のテキストレイヤーへ、[Step6-3](#)で変更したテキストをペーストしてください。
- テキストレイヤーを不可視にします。



以上でMxE SDK取り込み用のPSDファイル完成です。任意の名前でファイルを保存してください。

# SDKへの取り込み結果

---

PSD Importerによる取り込み結果イメージ

# 取り込み結果イメージ

- PSD Importerを利用してMxE SDKにPSDファイルを取り込んだ結果です。スコアのフレームを見ると、「Step5」で設定したレイヤーカンプに基づいた表示状態の切替を確認できます。

