Unity2D　3分間プログラミング

ボタンのつくりかた

今回やりたいことは　これ！

①画面にボタンをおく

②透明のキャラをつくる

③ボタンを押すと HP がへる

④HP を表示する

画面設定

Unity をひらいて、２Dで新しいプロジェクトをつくります。

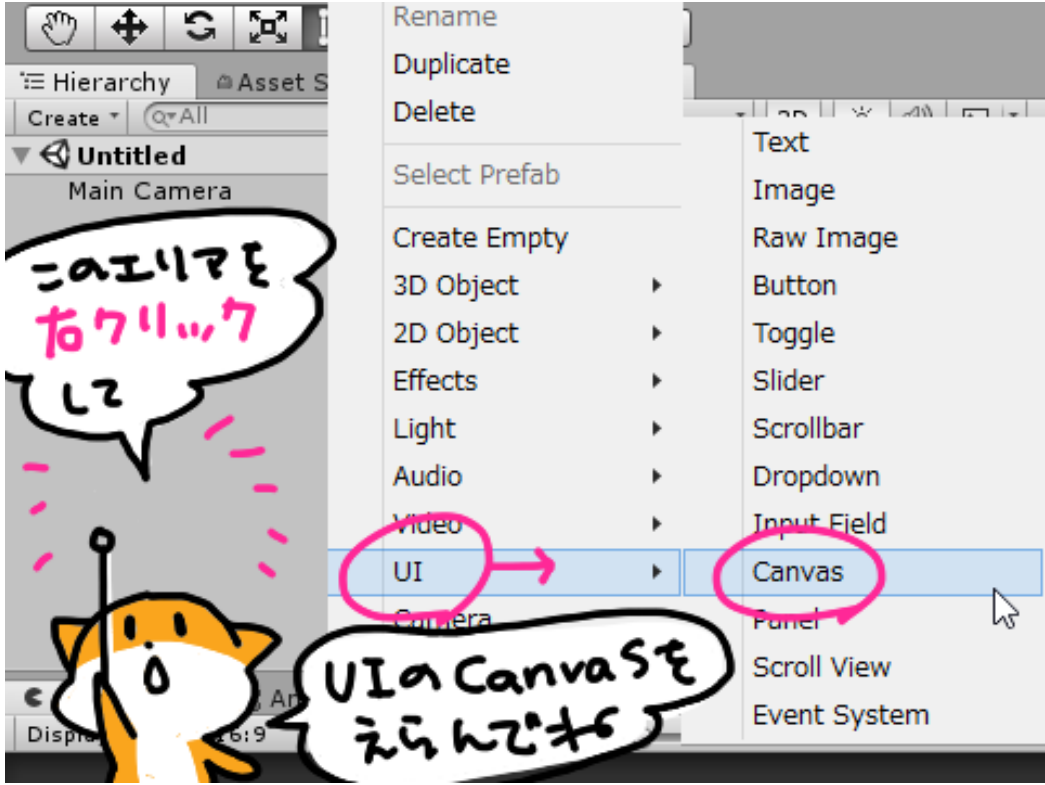
画面比を16：9に

みためにこだわりがなければそのままでOKです

キャンバスをおこう

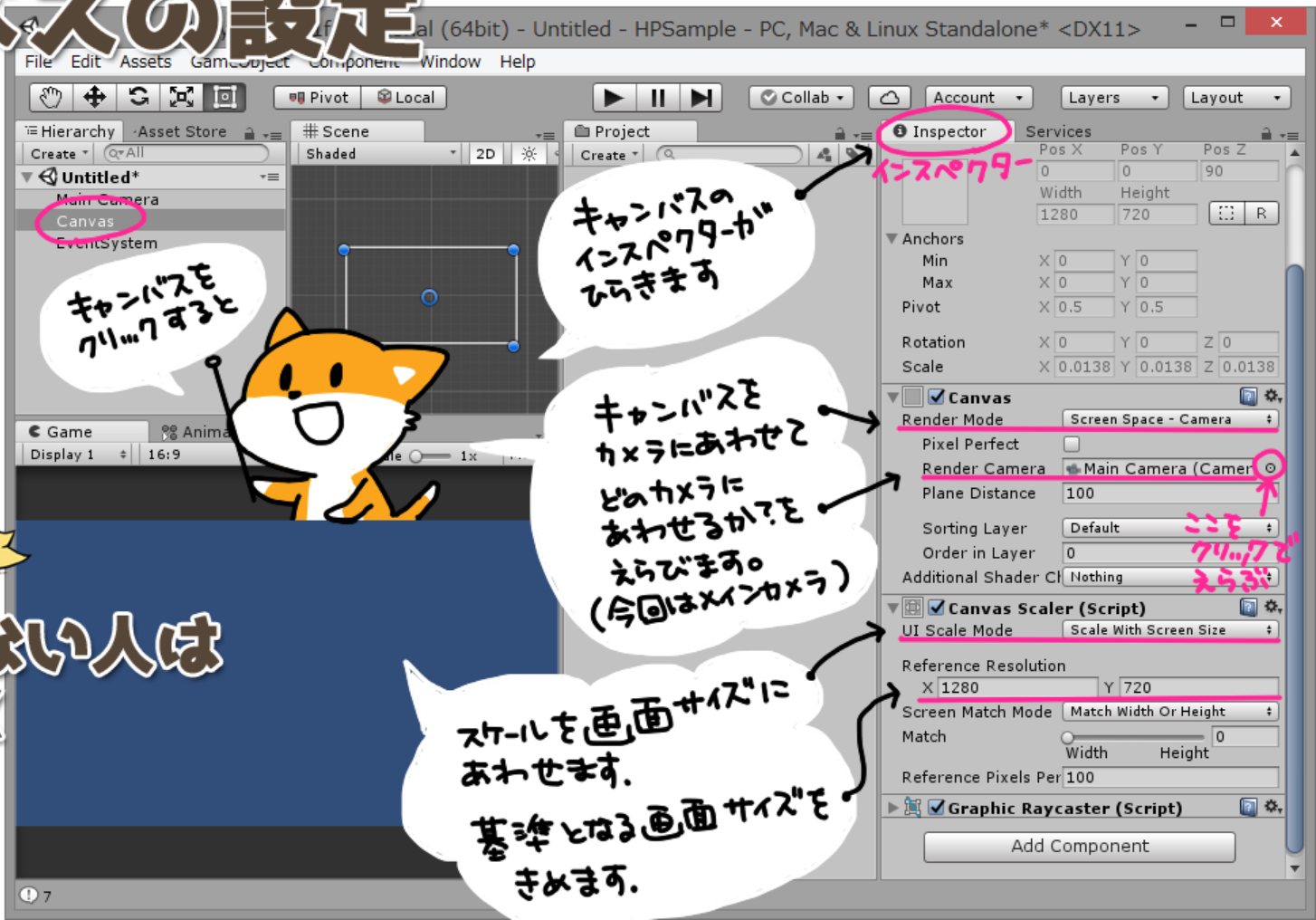
ボタンや文字を表示するのに
キャンバスというのがいります

ヒラエルキーを右クリで
UI⇨Canvasをえらびます

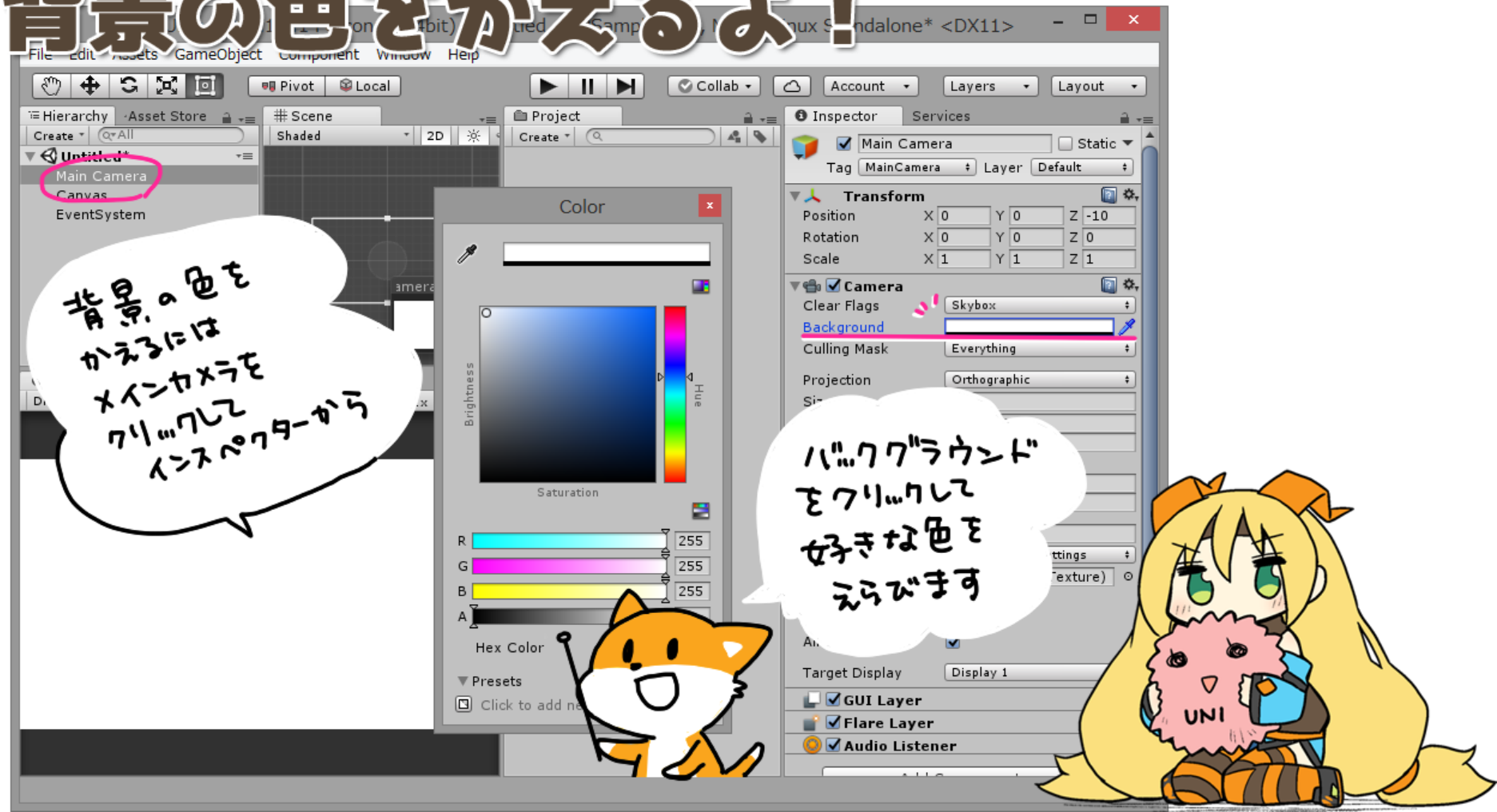


キャンバスの設定

こだわりがない人は
とばしてOK

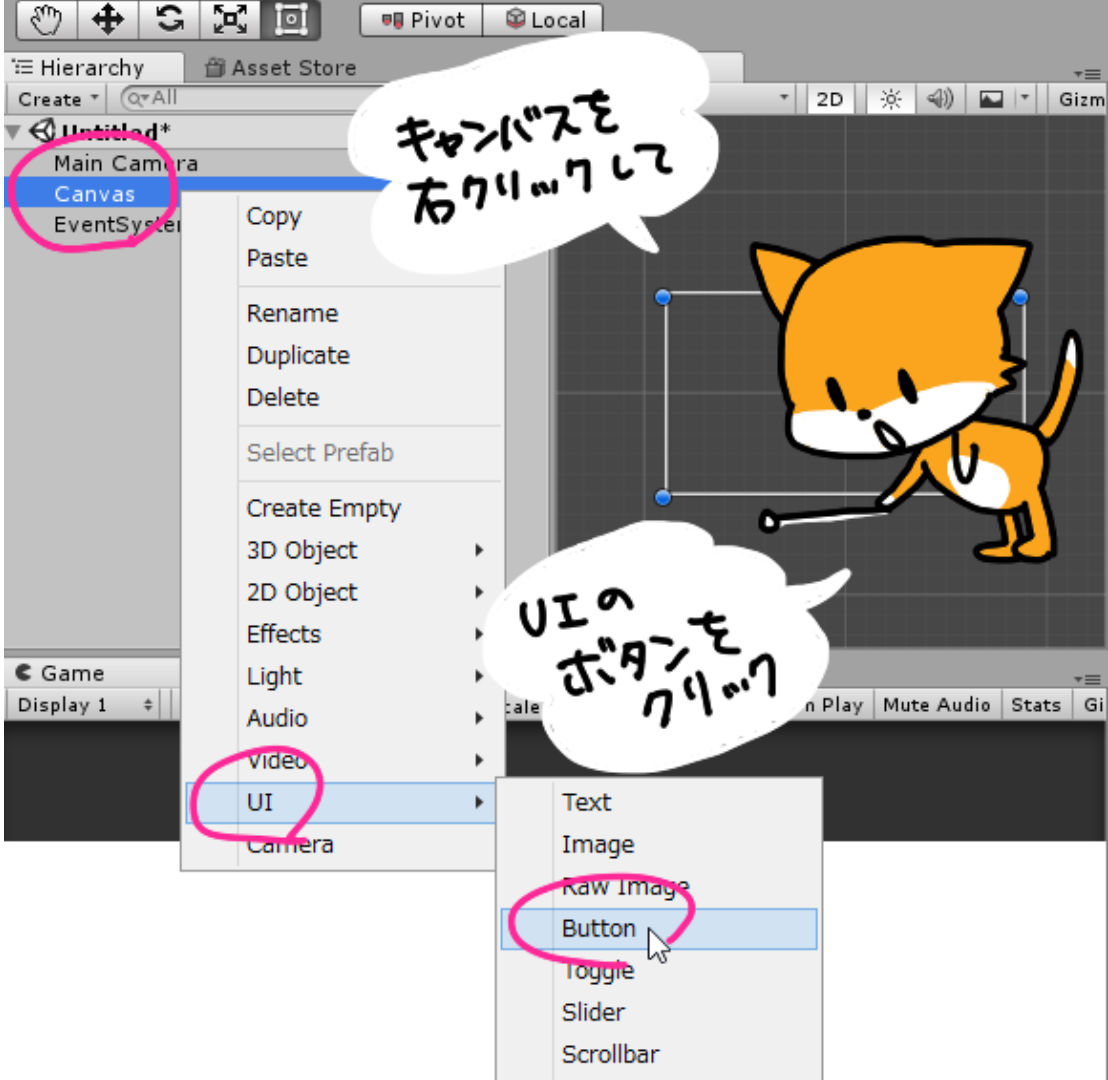


背景の色をかえるよ！

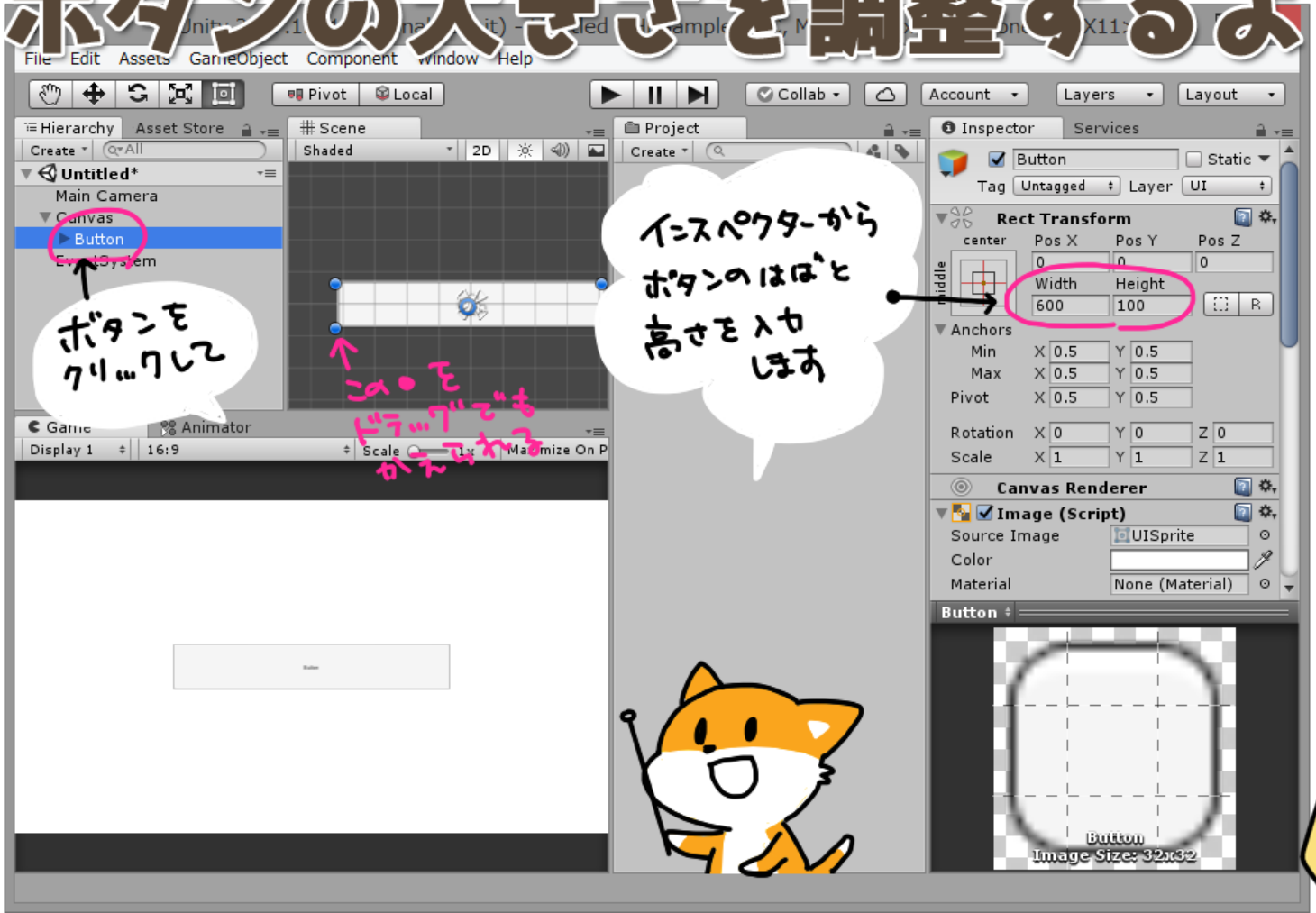

Hierarchy
Asset Store
Create
Main Camera
Canvas
EventSystem
Scene
Shaded
2D
Project
Create
Inspector
Services
Main Camera
Static
Tag MainCamera
Layer Default
Transform
Position X 0 Y 0 Z -10
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1
Camera
Clear Flags Skybox
Background
Culling Mask Everything
Projection Orthographic
Target Display Display 1
GUI Layer
Flare Layer
Audio Listener
Color
Brightness
Hue
Saturation
R 255
G 255
B 255
A
Hex Color
Presets
背景の色を
かえるには
メインカメラを
クリックして
インスペクターから
バックグラウンド
をクリックして
好きな色を
えらびます
UNI

ボタンをおこう

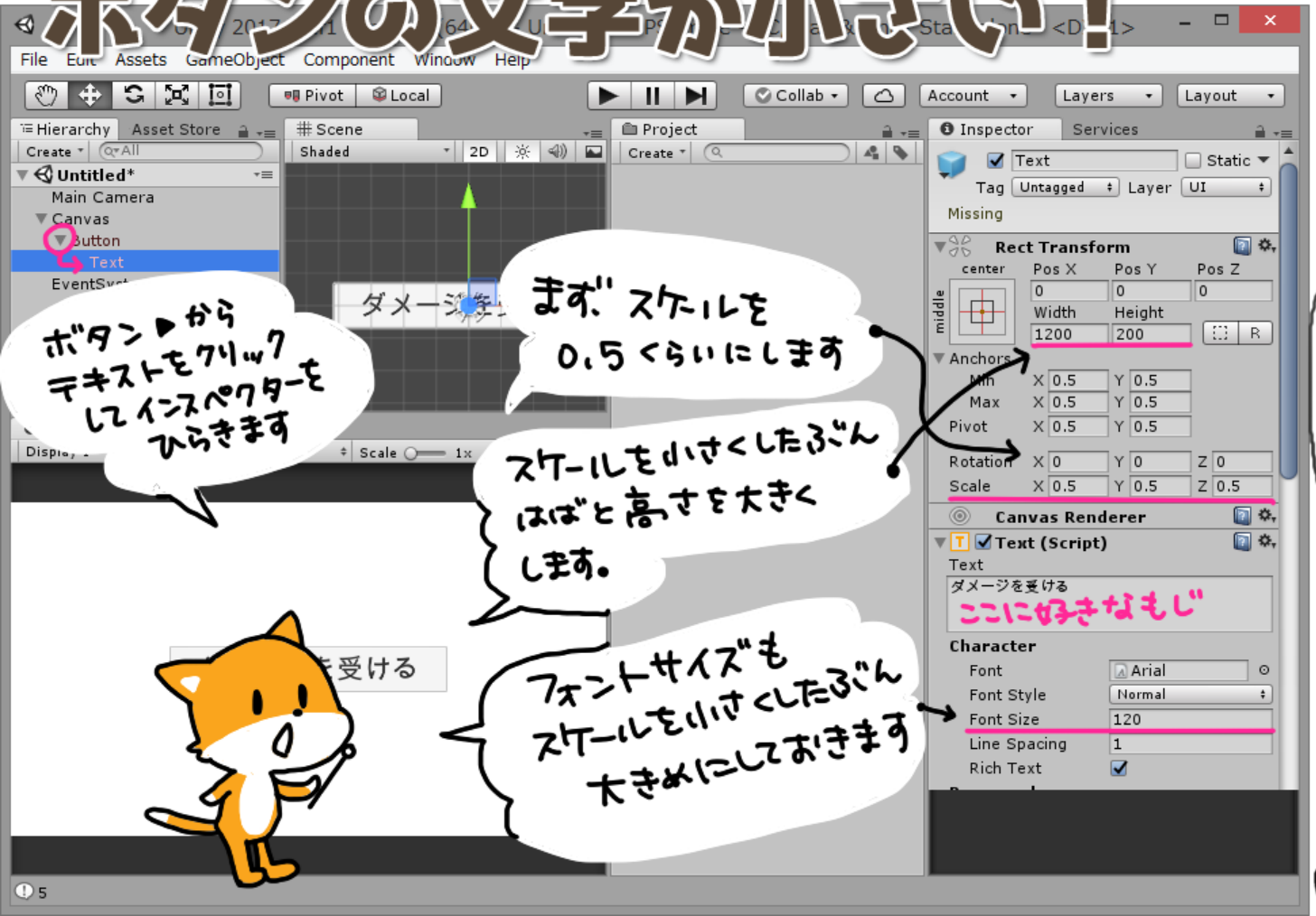
キャンバスの子にしたいので
キャンバスを右クリから
UI⇨ボタンをえらびます



ボタンの大きさを調整するよ

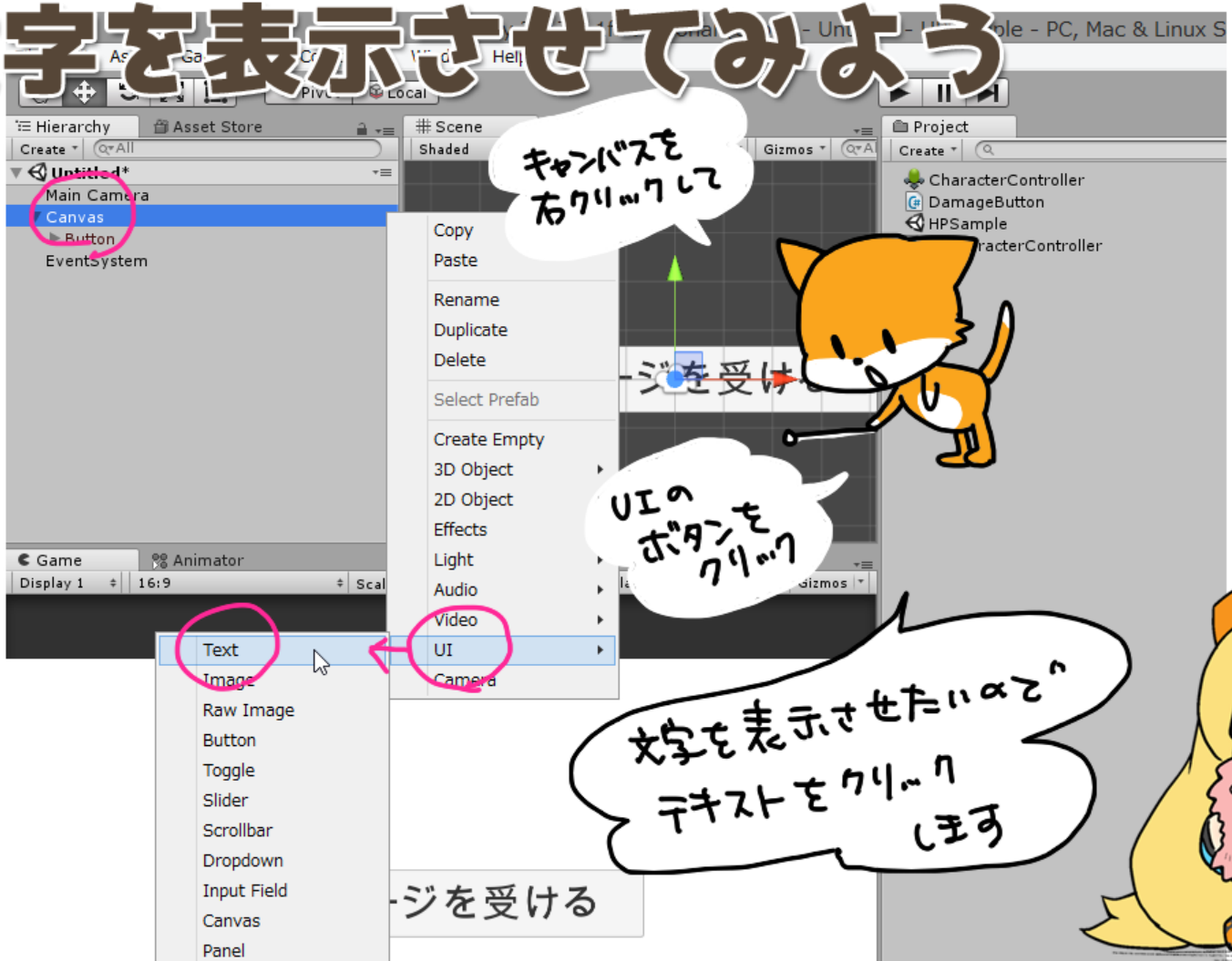


ボタンの文字が小さい！

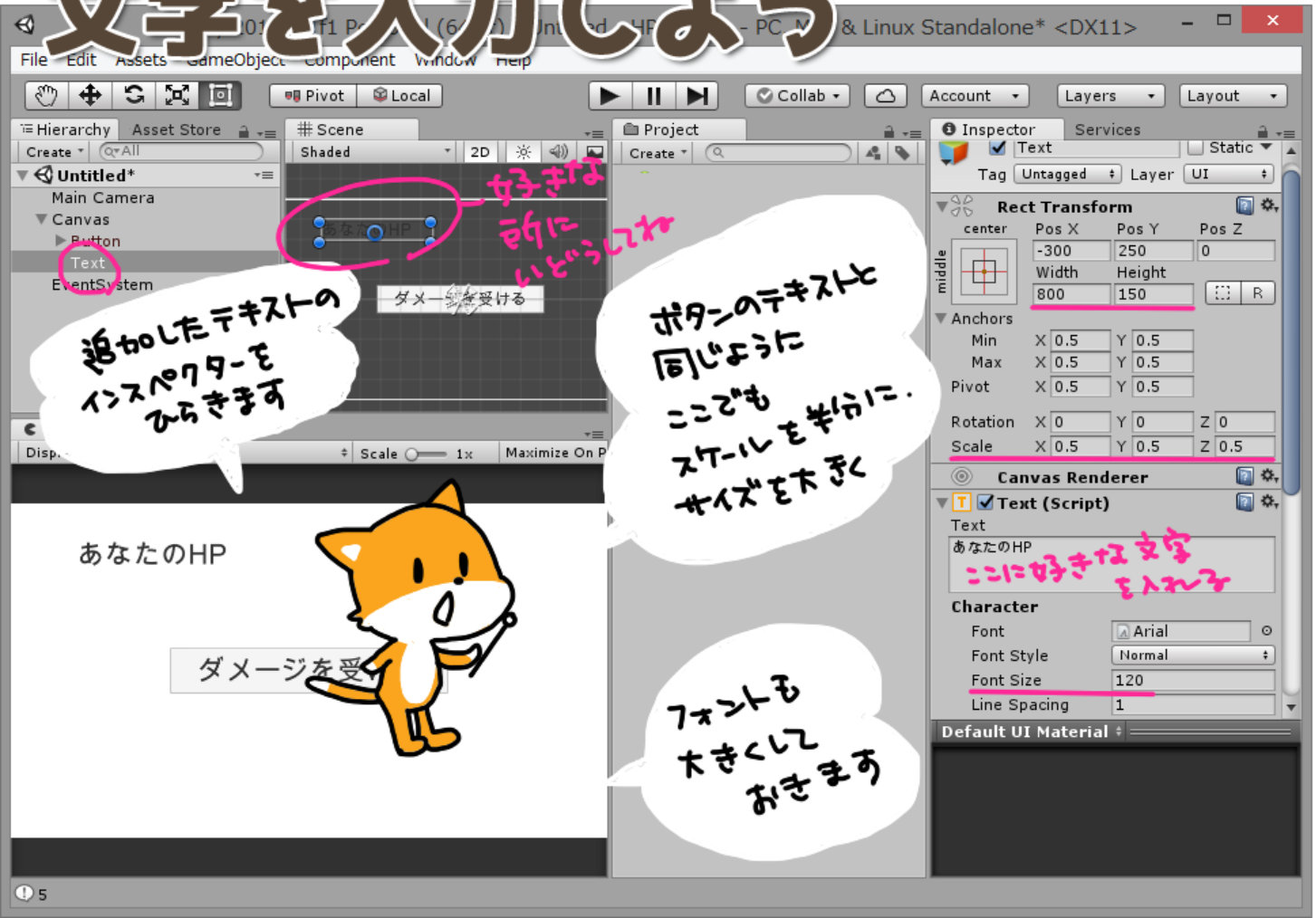

ボタンをクリックしてインスペクターをひらきます
まずスケールを0.5くらいにします
スケールを小さくしたぶんはばと高さを大きくします。
フォントサイズもスケールを小さくしたぶん大きめにしておきます
Hierarchy
Asset Store
Scene
Project
Inspector
Services
Main Camera
Canvas
Button
Text
Rect Transform
Width
Height
1200
200
Anchors
Pivot
Rotation
Scale
Canvas Renderer
Text (Script)
ダメージを受ける
ここに好きなもじ
Character
Font
Arial
Font Style
Normal
Font Size
120
Line Spacing
Rich Text


スケールの設定は
文字のにじみ対策
フォントサイズのみの
変更でもいいよ

文字を表示させてみよう

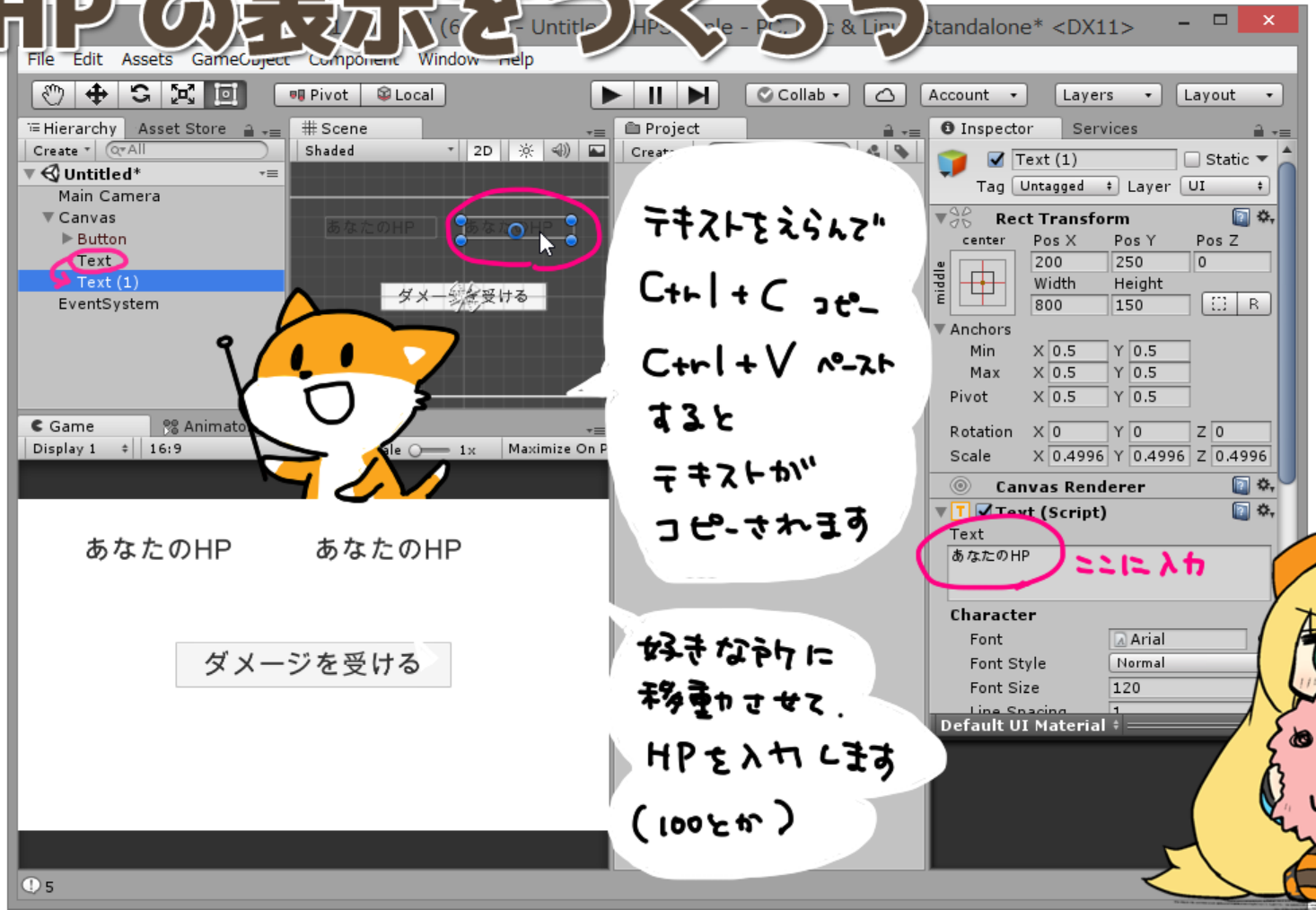


文字を入力しよう

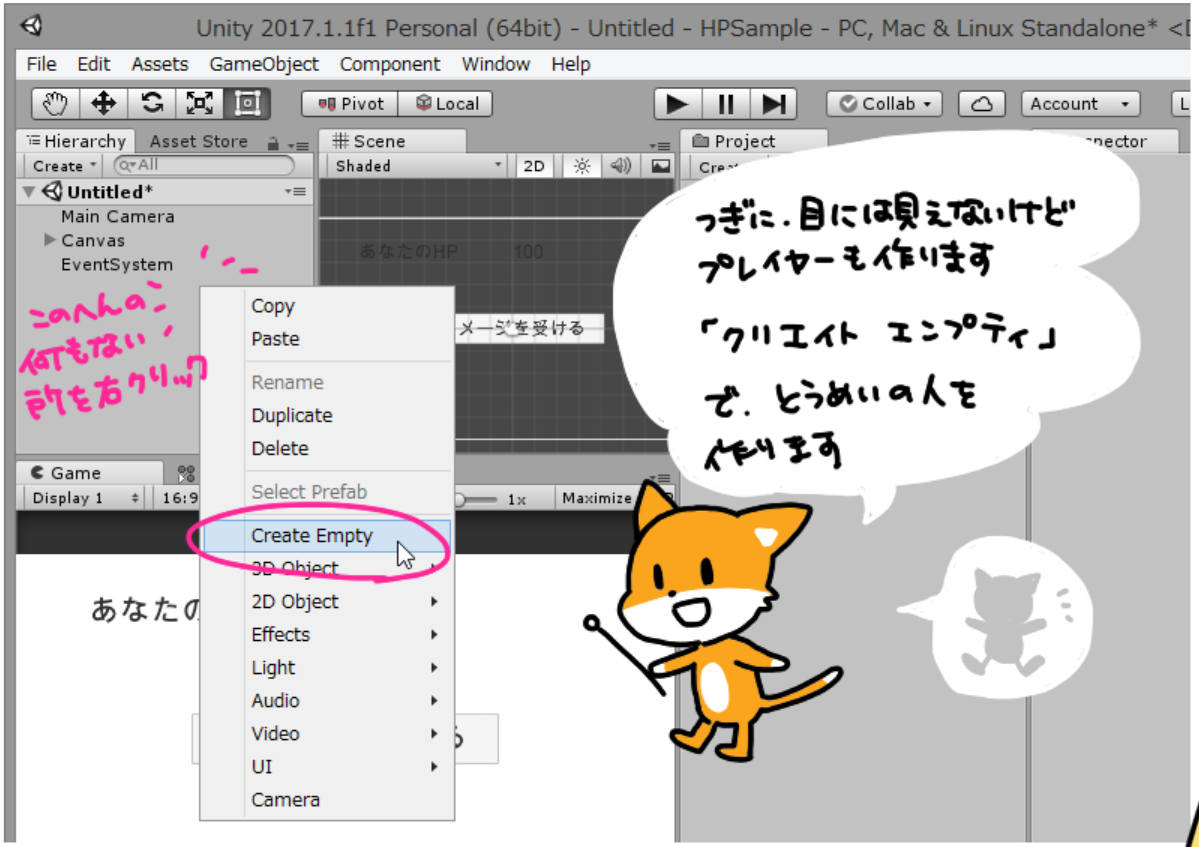

File Edit Assets GameObject Component Window Help
Pivot
Local
Collab
Account
Layers
Layout
Hierarchy
Asset Store
Create
Untitled*
Main Camera
Canvas
Button
Text
EventSystem
Scene
Shaded
2D
ダメージを受ける
Project
Create
Inspector
Services
Text
Static
Tag
Untagged
Layer
UI
Rect Transform
center
middle
Pos X
Pos Y
Pos Z
-300
250
0
Width
Height
800
150
Anchors
Min X 0.5 Y 0.5
Max X 0.5 Y 0.5
Pivot X 0.5 Y 0.5
Rotation X 0 Y 0 Z 0
Scale X 0.5 Y 0.5 Z 0.5
Canvas Renderer
Text (Script)
Text
あなたのHP
ここに好きな文字を入れる
Character
Font
Arial
Font Style
Normal
Font Size
120
Line Spacing
1
Default UI Material
好きな所にいどうしてね
追加したテキストのインスペクターをひらきます
ボタンのテキストと同じようにここでもスケールを半分に．サイズを大きく
あなたのHP
ダメージを受ける
フォントも大きくしておきます


このスケールの設定も文字のにじみ対策なのでなくてもOKだよ

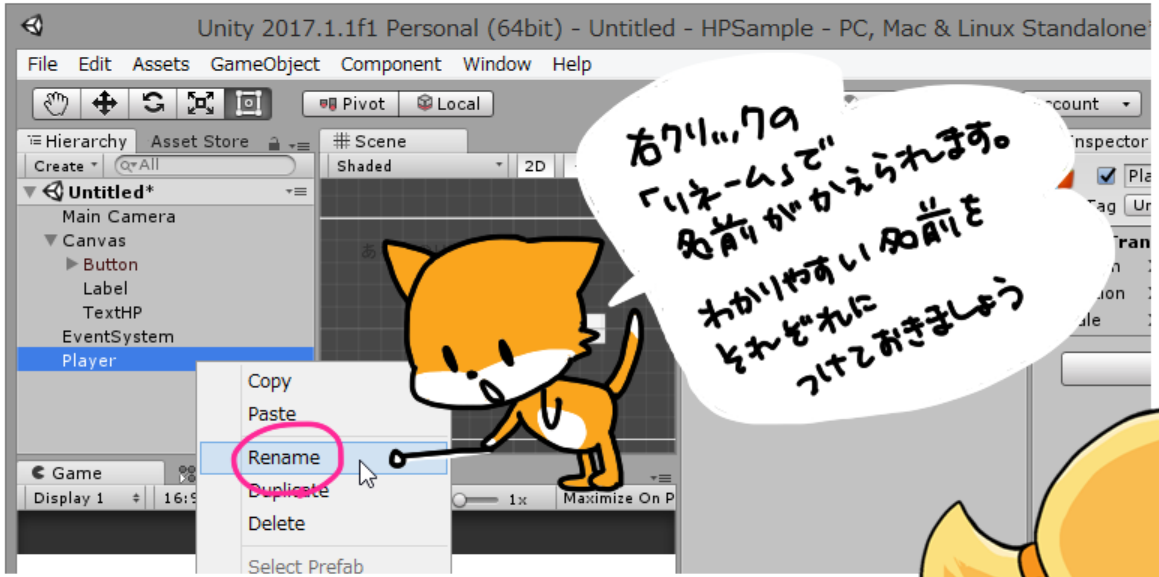
HPの表示をつくろう


File Edit Assets GameObject Component Window Help
Pivot Local
Collab Account Layers Layout
Hierarchy Asset Store
Create
Untitled*
Main Camera
Canvas
Button
Text
Text (1)
EventSystem
Scene
Shaded 2D
あなたのHP
ダメージを受ける
Project
Inspector Services
Text (1)
Static
Tag Untagged Layer UI
Rect Transform
center
Pos X Pos Y Pos Z
200 250 0
Width Height
800 150
middle
Anchors
Min X 0.5 Y 0.5
Max X 0.5 Y 0.5
Pivot X 0.5 Y 0.5
Rotation X 0 Y 0 Z 0
Scale X 0.4996 Y 0.4996 Z 0.4996
Canvas Renderer
Text (Script)
Text
あなたのHP
ここに入力
Character
Font Arial
Font Style Normal
Font Size 120
Default UI Material
Game
Display 1 16:9
あなたのHP あなたのHP
ダメージを受ける
テキストをえらんで
Ctrl+C コピー
Ctrl+V ペースト
すると
テキストが
コピーされます
好きな所に
移動させて、
HPを入力します
(100とか)
UNI

プレイヤーを作ろう

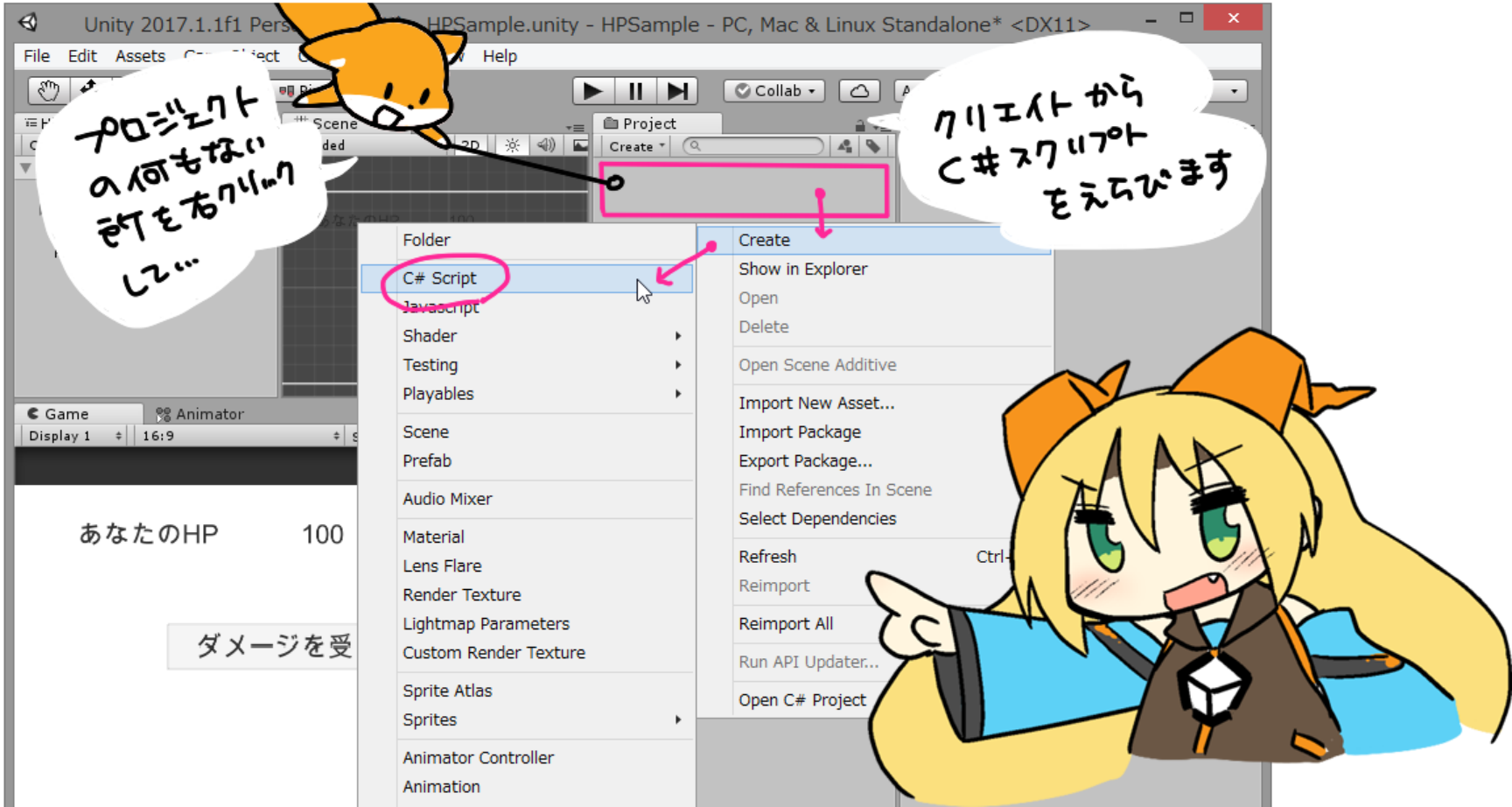

Unity 2017.1.1f1 Personal (64bit) - Untitled - HPSample - PC, Mac & Linux Standalone*
File Edit Assets GameObject Component Window Help
Pivot Local
Collab Account
Hierarchy Asset Store
Create All
Untitled*
Main Camera
Canvas
EventSystem
Scene
Shaded 2D
Project
あなたのHP 100
メージを受ける
このへんの何もない所を右クリック
Copy
Paste
Rename
Duplicate
Delete
Select Prefab
Create Empty
3D Object
2D Object
Effects
Light
Audio
Video
UI
Camera
Game
Display 1 16:9
Maximize
あなたの
つぎに、目には見えないけど
プレイヤーも作ります
「クリエイト エンプティ」
で、とうめいの人を
作ります

プレイヤーやボタンに名前を付けよう

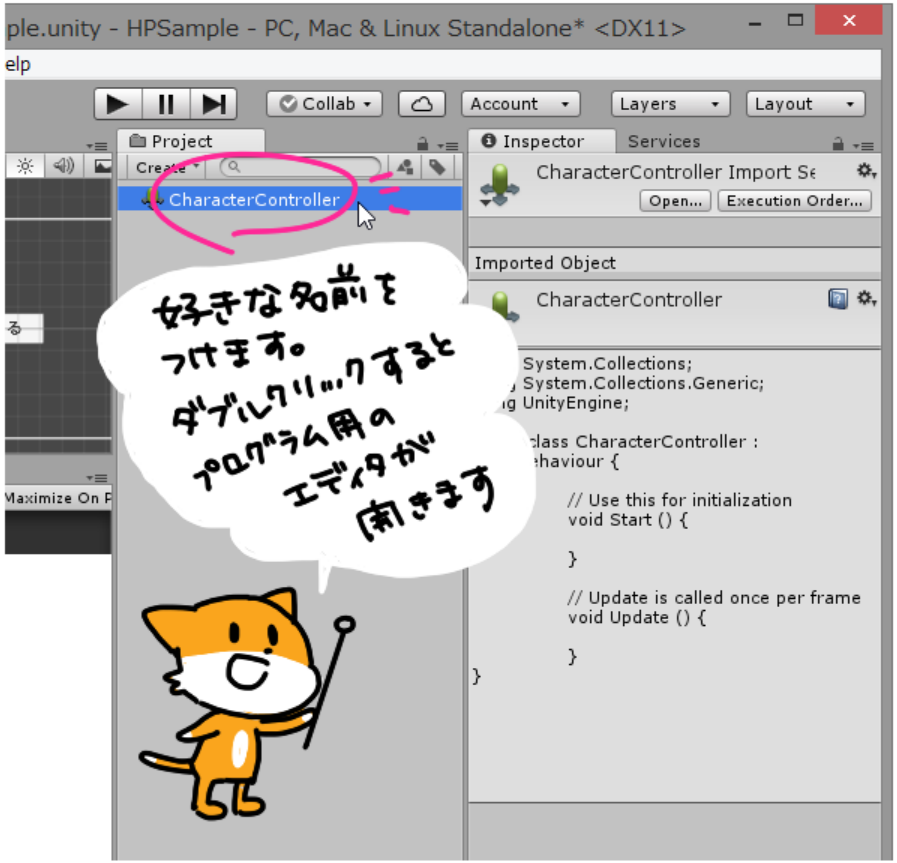



Unity 2017.1.1f1 Personal (64bit) - HPSample.unity - HPSample - PC, Mac & Linux Standalone <DX11>
File Edit Assets GameObject Component Window Help
Collab
Account
Layers
Layout
Hierarchy
Create
HPSample
Main Camera
Canvas
EventSystem
Player
Project
Inspector
Services
ダメージを受ける
Game
Animator
Display 1
16:9
Scale
あなたのHP 100
ダメージを受ける
これで HPの表示、ボタン キャラクター ができました
次はこれらに プログラムして いきましょう！
UNI

プログラムファイルを作ろう

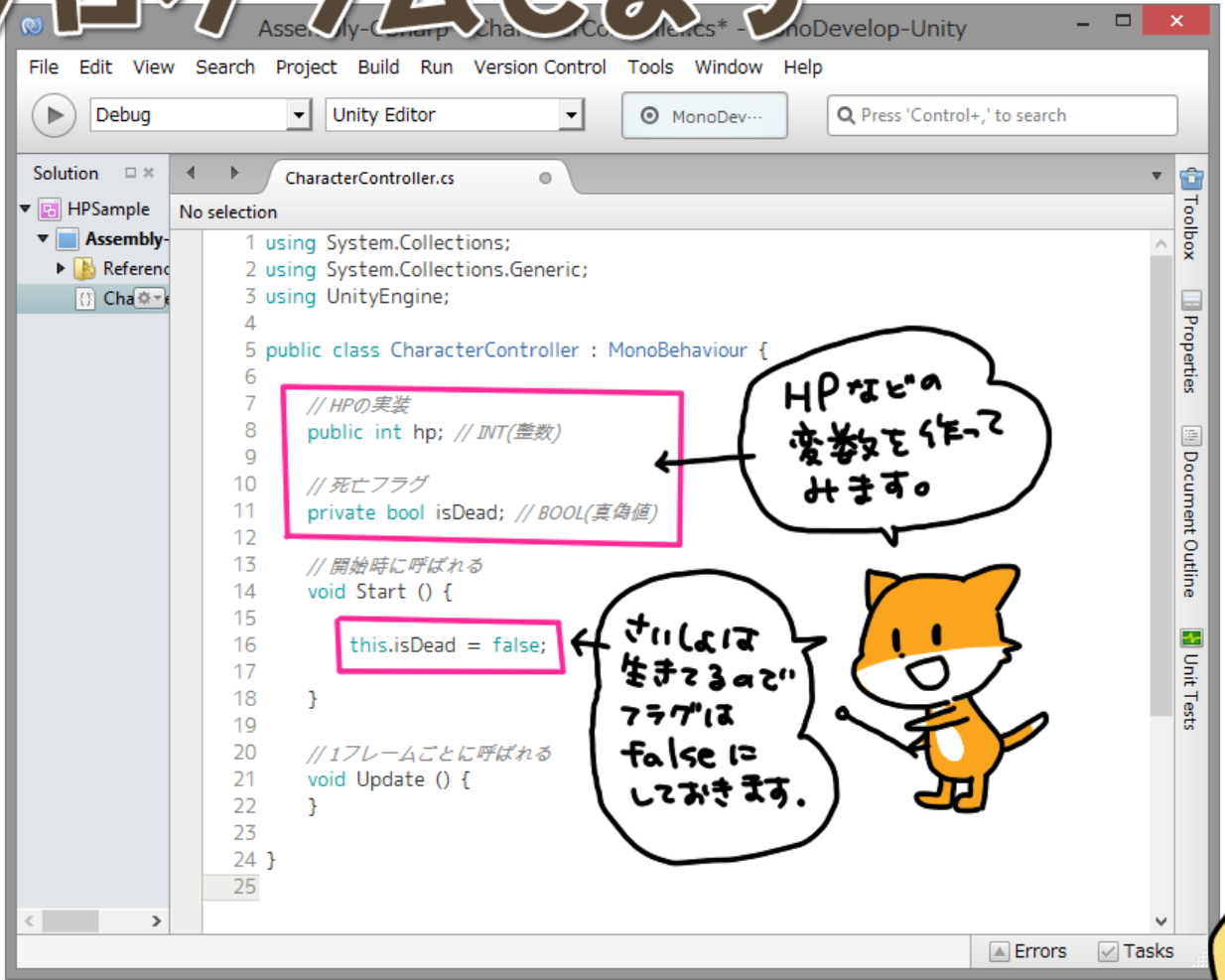


ファイルに名前を付けよう



あとで
プレイヤーにアタッチする
プログラムだから
プレイヤー hogehoge
とかがいいかも

プログラムしよう


File Edit View Search Project Build Run Version Control Tools Window Help
Debug
Unity Editor
Press 'Control+,' to search
Solution
HPSample
CharacterController.cs
No selection
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class CharacterController : MonoBehaviour {
6
7 // HPの実装
8 public int hp; // INT(整数)
9
10 // 死亡フラグ
11 private bool isDead; // BOOL(真偽値)
12
13 // 開始時に呼ばれる
14 void Start () {
15
16 this.isDead = false;
17
18 }
19
20 // 1フレームごとに呼ばれる
21 void Update () {
22 }
23
24 }
25
HPなどの変数を作ってみます。
さいしょは生きてるのでフラグはfalseにしておきます。
Toolbox
Properties
Document Outline
Unit Tests
Errors
Tasks


変数をパブリックにしておくと外からなかみをかえられるよ

プログラムしよう

CharacterController.cs

CharacterController ▸ Start ()

```
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  public class CharacterController : MonoBehaviour {
6
7      // HPの実装
8      public int hp; // INT(整数)
9
10     // 死亡フラグ
11     private bool isDead; // BOOL(真偽値)
12
13     public bool IsDead() {
14         return this.isDead;
15     }
16
17     // 開始時に呼ばれる
18     void Start () {
19         this.isDead = false;
20     }
21
22     // 1フレームごとに呼ばれる
23     void Update () {
24     }
25
26 }
```

ついか

関数も作ってみます
「死んでる？」と
聞かれたら、今の状態を
おしえてくれます。

プライベートな
変数にアクセス
するので
「アクセサ」とも
いいます

フツーはこうやって
外からカッテに変数を
さわれないように
しとくのだ

プログラムしよう

CharacterController.cs

CharacterController ▸ No selection

```
7   // HPの実装
8   public int hp; // INT(整数)
9
10  // 死亡フラグ
11  private bool isDead; // BOOL(真偽値)
12
13  public bool IsDead() {
14      return this.isDead;
15  }
16
17  public void Damage( int value ) {
18
19      // ダメージ値を減算
20      this.hp -= value;
21
22      // 死んでる？
23      if( this.hp <= 0 ) {
24
25          // 0以下にはしない
26          this.hp = 0;
27
28          // 死亡フラグを立てる
29          this.isDead = true;
30
31      }
32
```

ついか

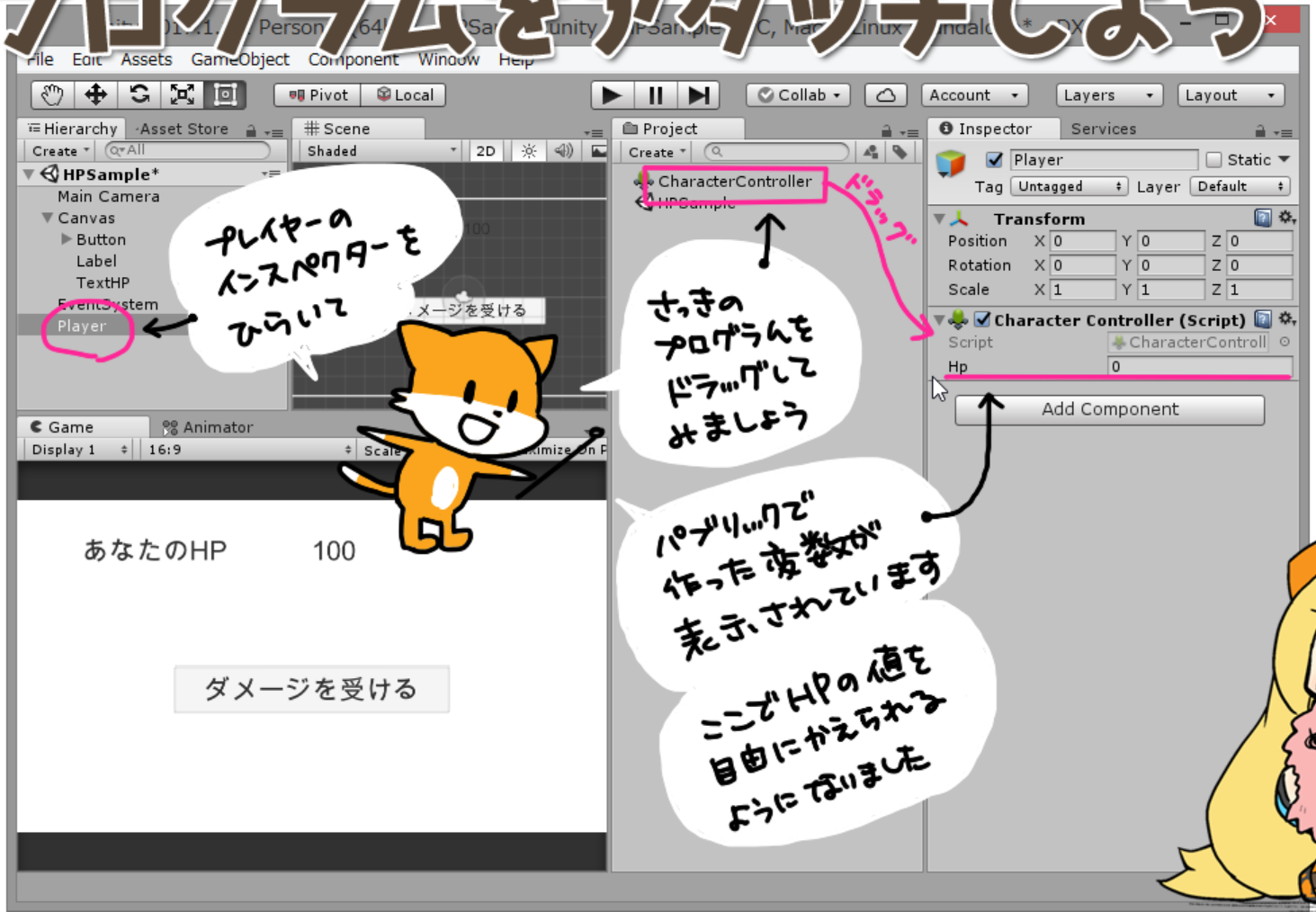
ダメージを受ける用の関数も作ります。

ダメージ値をわたすとHPをへらします。

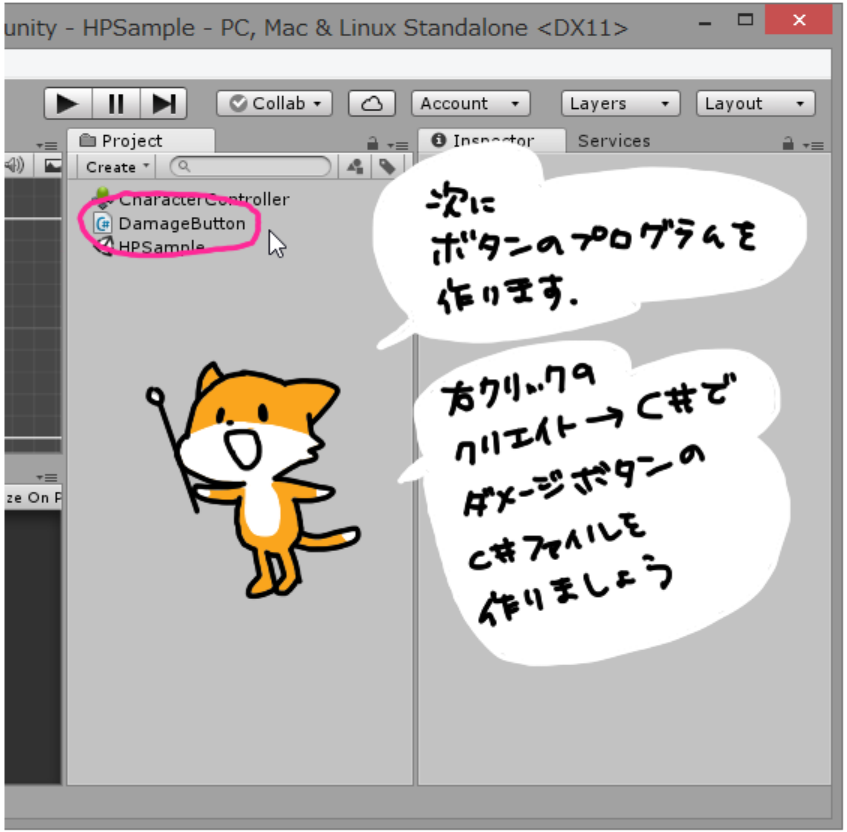
死んだ時のフラグもここでします。

HPも外でカッテにへらされたらヤダからこんなかんじにするよ

プログラムをアタッチしよう



ボタンのプログラムをつくろう



ボタンのプログラムをつくろう

File Edit View Search Project Build Run Version Control Tools Window Help

Debug | Unity Editor | Solution I··· | Press 'Control+,' to search

Solution

HPSample

Assembly-

Referenc

Characte

Dan

CharacterController.cs | DamageButton.cs

No selection

```
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  public class DamageButton : MonoBehaviour {
6
7      public CharacterController character;
8
9      public void OnClick() {
10
11         // ダメージ値
12         int damage;
13
14         // ランダム
15         System.Random rnd = new System.Random();
16         damage = rnd.Next(10); // 0～9までの乱数
17         Debug.Log( damage );
18
19         // ダメージを与える
20         character.Damage(damage);
21
22     }
23
24 }
25
```

パブリックにすると Unity 上でキャラを指定できます

ほんとはおんましゃかない

わかりやすくするため．

Toolbox | Properties | Document Outline | Unit Tests

Errors | Tasks

あとから
Unity 上で指定するよ

ボタンのプログラムをつくろう

File Edit View Search Project Build Run Version Control Tools Window Help

Debug | Unity Editor | Solution I··· | Press 'Control+,' to search

Solution

HPSample

Assembly-

Referenc

Characte

Dan

CharacterController.cs | DamageButton.cs

No selection

```
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  public class DamageButton : MonoBehaviour {
6
7      public CharacterController character;
8
9      public void OnClick() {
10
11         //ダメージ値
12         int damage;
13
14         //ランダム
15         System.Random rnd = new System.Random();
16         damage = rnd.Next(10); //0~9までの乱数
17         Debug.Log( damage );
18
19         //ダメージを与える
20         character.Damage(damage);
21
22     }
23
24 }
25
```

オンクリックで ボタンが押された時の 処理をかきます

変数をつくって

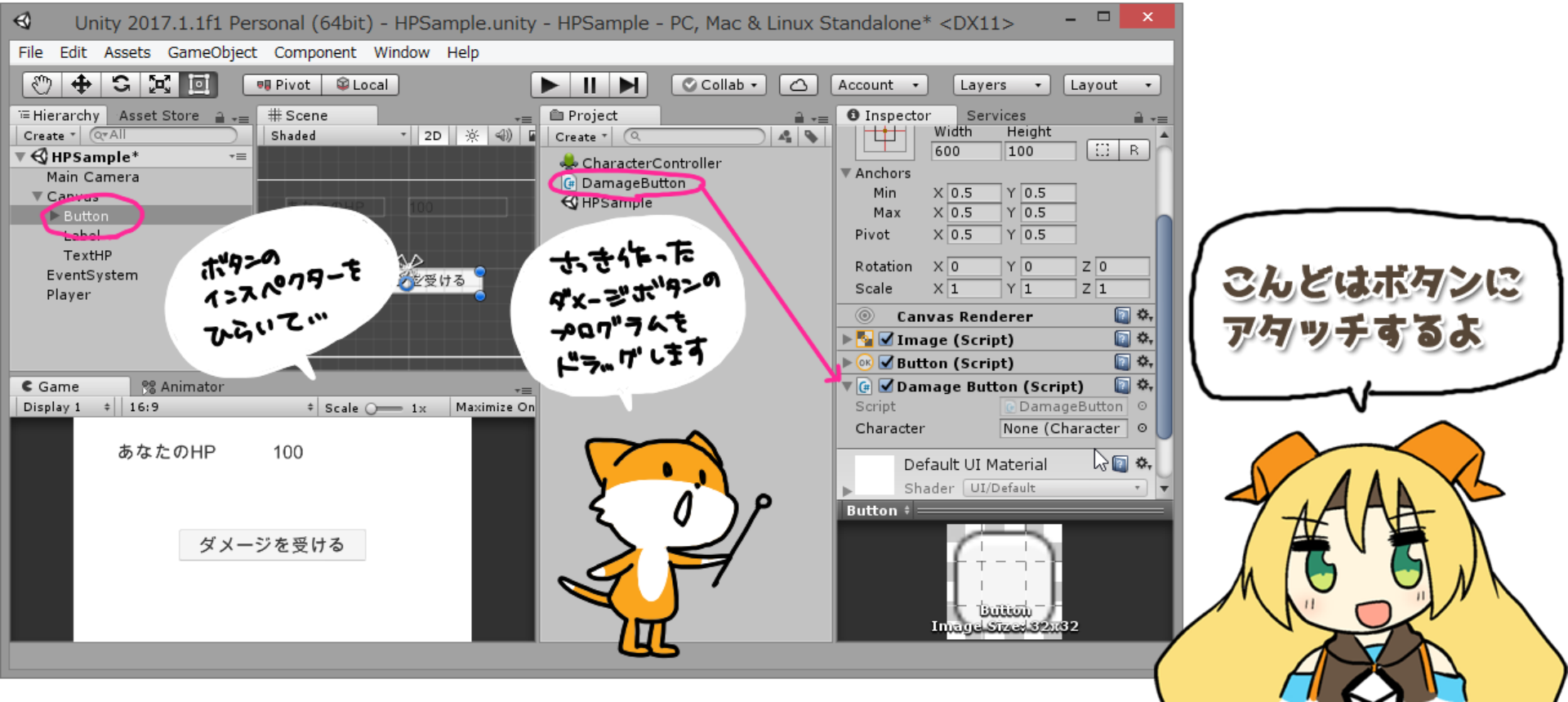
変数に ランダム値を入れます

次に キャラ側に 作ったダメージ関数をつかって キャラにダメージをあたえます.

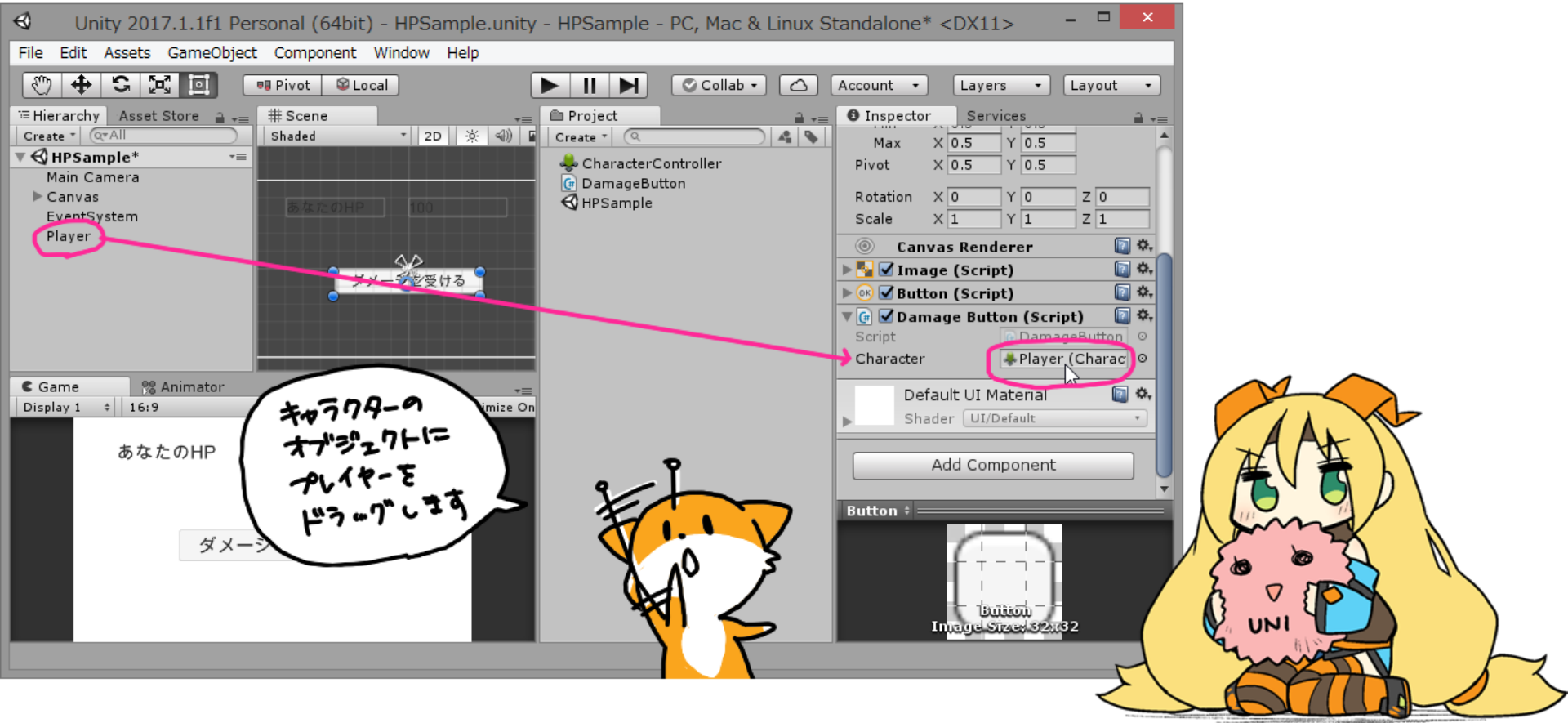
Toolbox | Properties | Document Outline | Unit Tests

Errors | Tasks

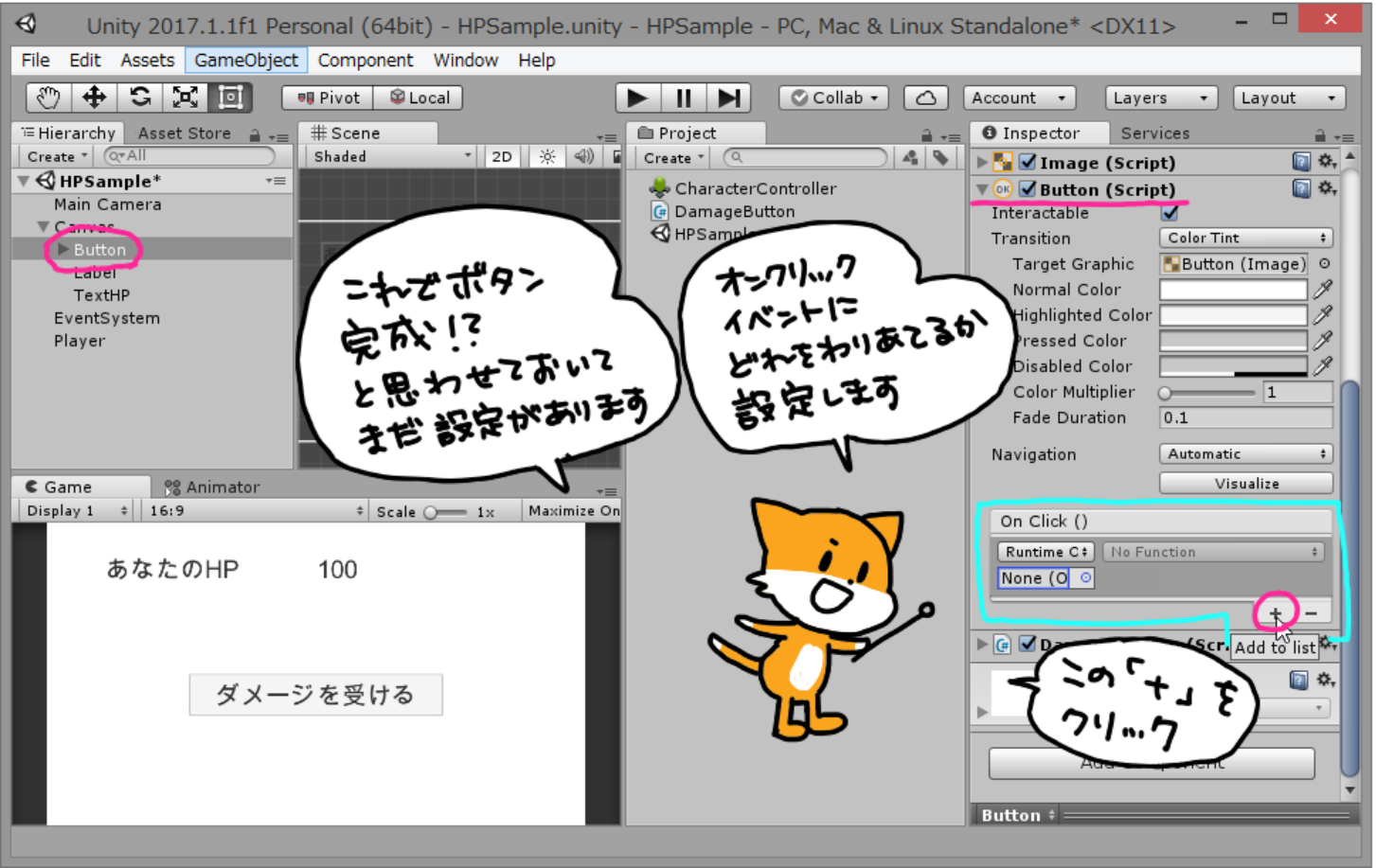
プログラムをアタッチしよう



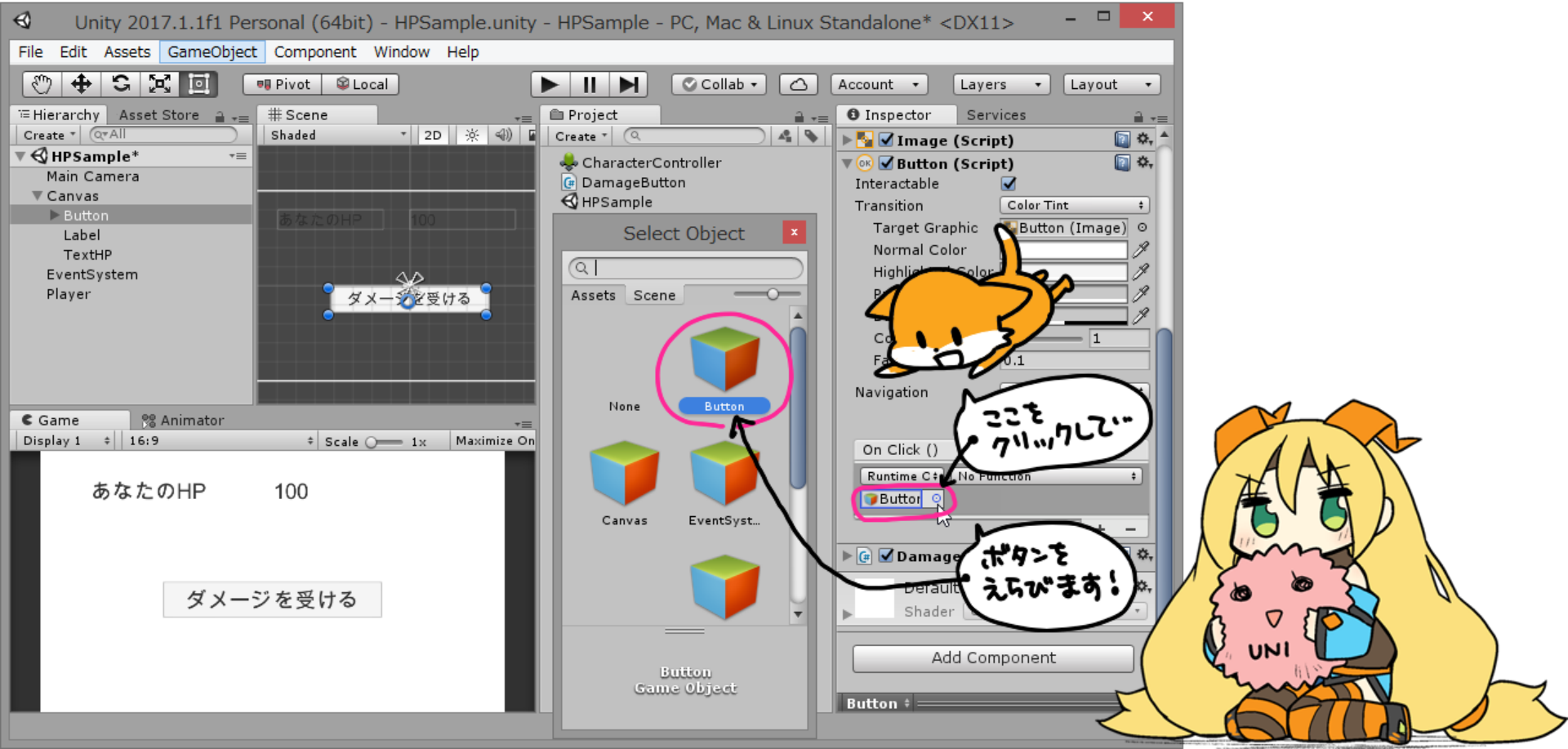
プログラムをアタッチしよう



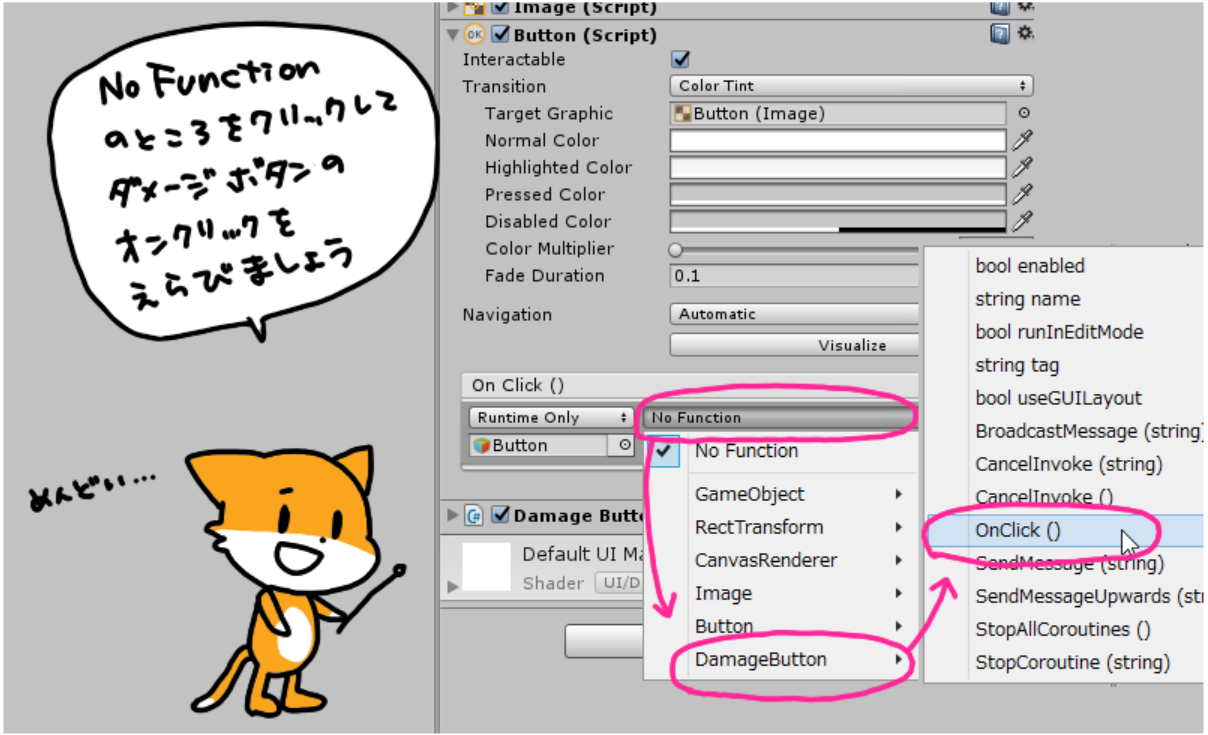
ボタンを押されたときの設定をしよう



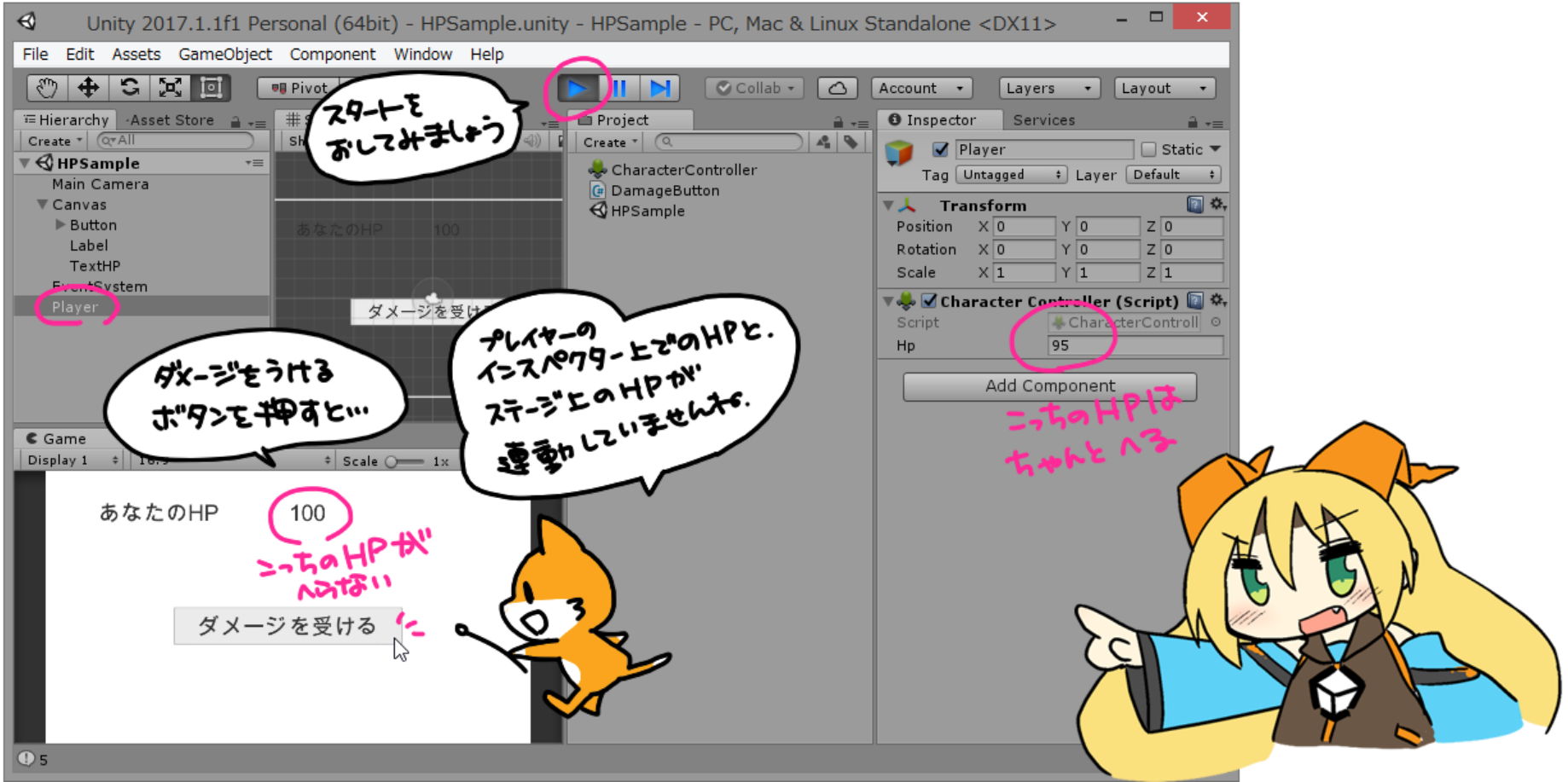
ボタンを押されたときの設定をしよう



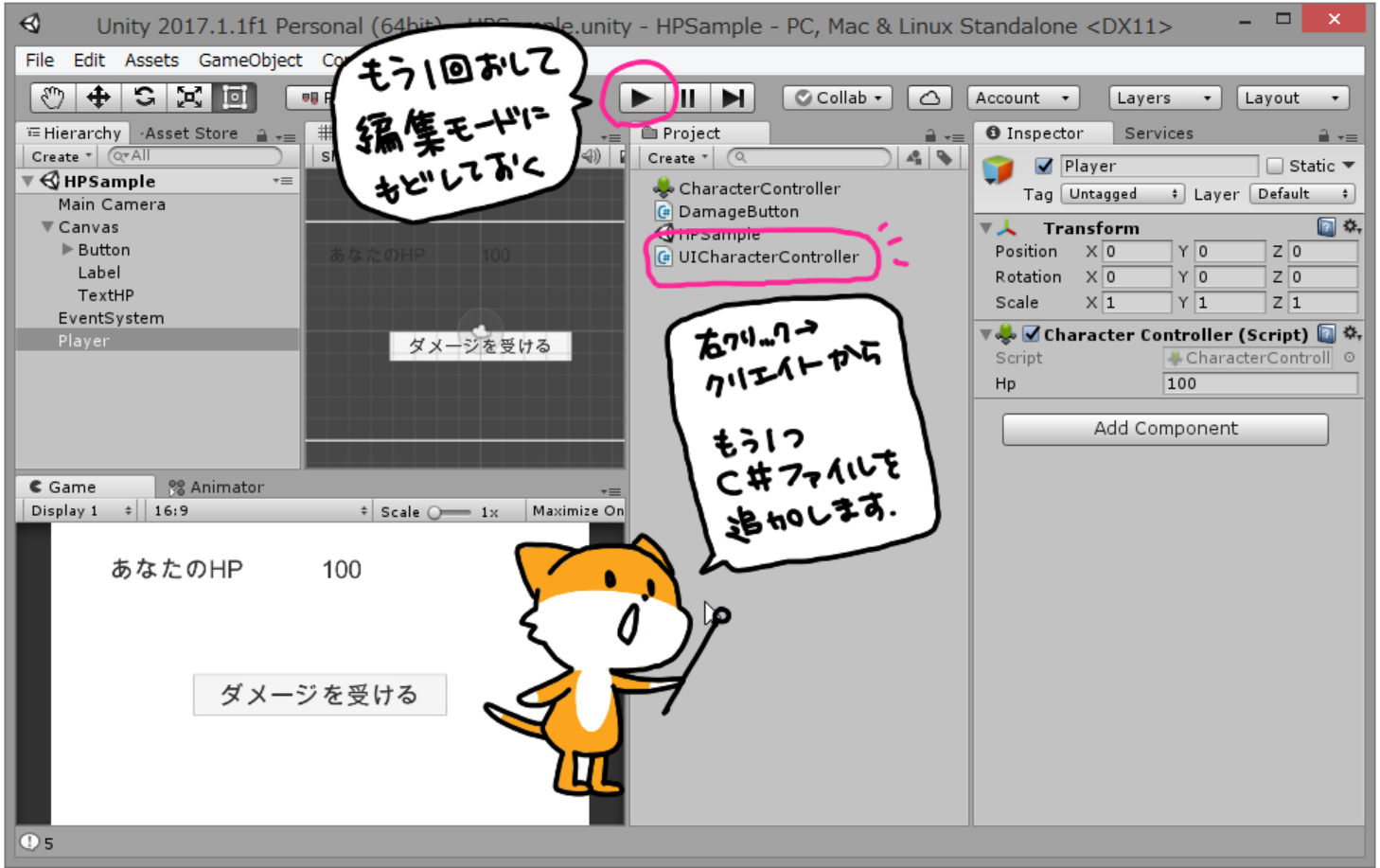
ボタンを押されたときの設定をしよう



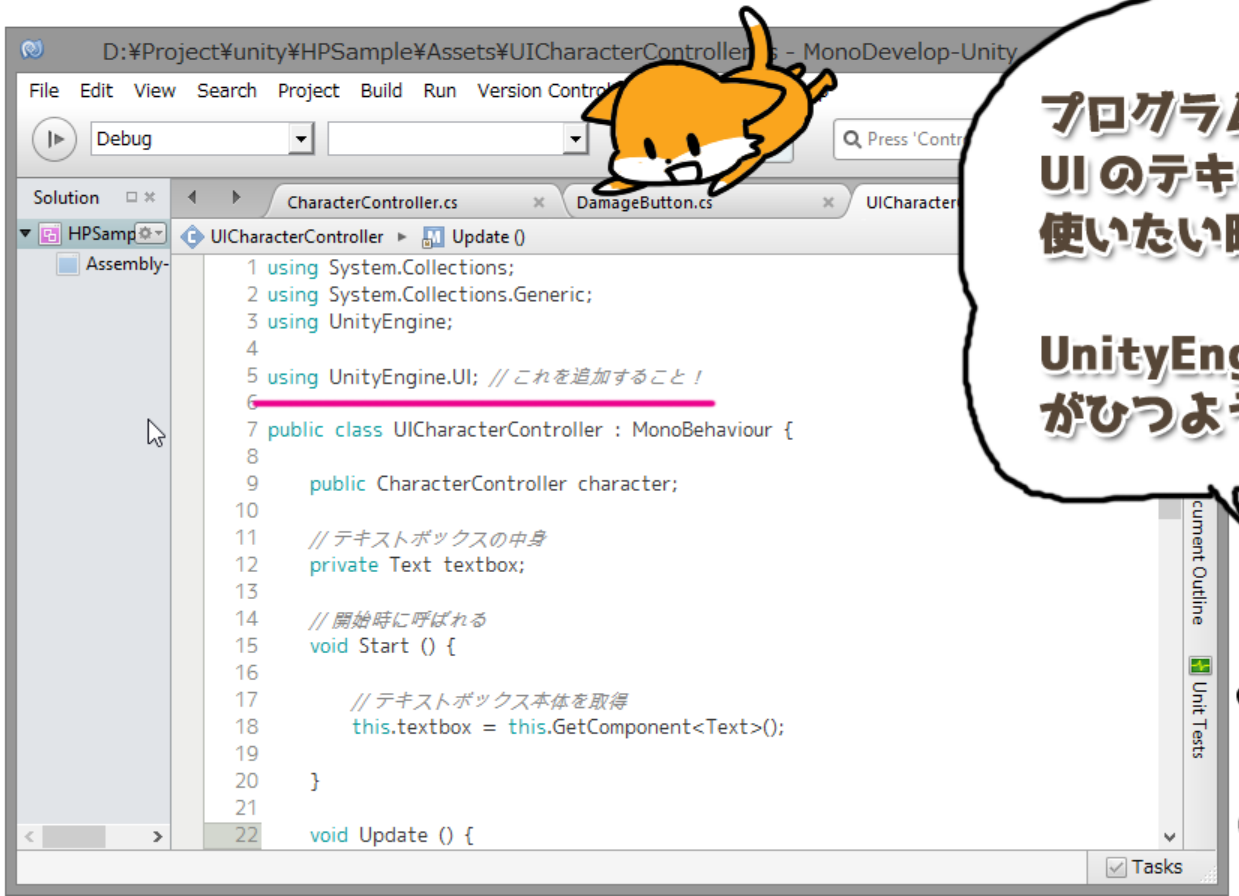
プレイしてみよう


Unity 2017.1.1f1 Personal (64bit) - HPSample.unity - HPSample - PC, Mac & Linux Standalone <DX11>
File Edit Assets GameObject Component Window Help
スタートを
おしてみましょう
Hierarchy
Asset Store
Create
HPSample
Main Camera
Canvas
Button
Label
TextHP
EventSystem
Player
Project
CharacterController
DamageButton
HPSample
Inspector
Services
Player
Static
Tag Untagged
Layer Default
Transform
Position X 0 Y 0 Z 0
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1
Character Controller (Script)
Script CharacterControll
Hp 95
Add Component
こっちのHPは
ちゃんと へる
ダメージをうける
ボタンを押すと…
プレイヤーの
インスペクター上でのHPと、
ステージ上のHPが
連動していません。
Game
Display 1
Scale 1x
あなたのHP
100
こっちのHPが
へらない
ダメージを受ける
5

表示用のプログラムをつくろう



表示用のプログラムをつくろう



プログラムで
UIのテキストを
使いたい時は

UnityEngine.UI
がひつようだよ

表示用のプログラムをつくろう

D:¥Project¥unity¥HPSample¥Assets¥UICharacterController.cs - MonoDevelop-Unity

File Edit View Search Project Build Run Version Control Tools Window Help

Debug

MonoDev···

Press 'Control+,' to search

Solution

HPSamp

Assembly-

CharacterController.cs | DamageButton.cs | UICharacterController.cs

UICharacterController ▸ Update ()

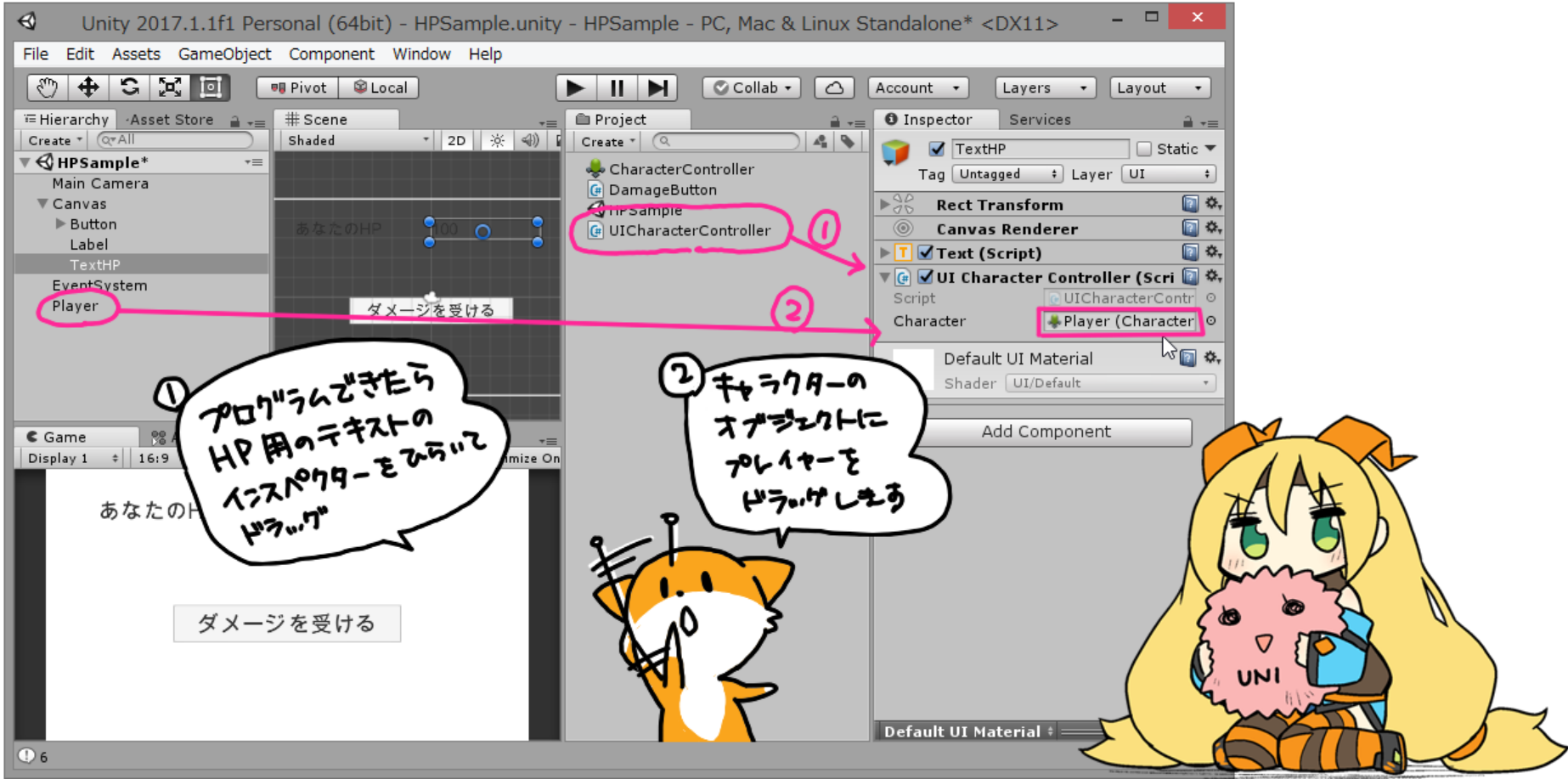
```
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  using UnityEngine.UI; // これを追加すること！
6
7  public class UICharacterController : MonoBehaviour {
8
9      public CharacterController character;
10
11     // テキストボックスの中身
12     private Text textbox;
13
14     // 開始時に呼ばれる
15     void Start () {
16
17         // テキストボックス本体を取得
18         this.textbox = this.GetComponent<Text>();
19
20     }
21
22     void Update () {
```

Toolbox Properties Document Outline

Tasks

表示用のプログラムをつくろう

テキストにプログラムをアタッチ



かんせい！


Unity 2017.1.1f1 Personal (64bit) - HPSample.unity - HPSample - PC, Mac & Linux Standalone* <DX11>
File Edit Assets GameObject
スタートボタンを押して…
Hierarchy
HPSample*
Main Camera
Canvas
Button
Label
TextHP
EventSystem
Player
Project
CharacterController
DamageButton
HPSample
UICharacterController
Inspector
Services
Player
Tag Player
Layer Default
Transform
Position
Rotation
Scale
Character Controller (Script)
Hp 62
Add Component
おつかれさまー！
ボタンを押すとHPがへるようになりました！
Game
Animator
Display 1
16:9
Scale 1x
Maximize On
あなたのHP 62
ダメージを受ける
mog
mog
mog
Oh! my Croquette