

Prof. Me. Hélio Esperidião

Apresentação de textos em Unity - Componente "Text"

Adicionar um componente de texto

GameObject >> UI >> Text

🚭 Unity 2017.4.40f1 Personal (64bit) - aula01.unity - 1_bimestre_Aulas - PC, Mac & Linux Standalone* <D



Canvas unity

- O Canvas do Unity é um componente gráfico que fornece um espaço de tela para que os elementos da interface do usuário (UI) sejam renderizados.
- Ele é usado para criar a interface do usuário do jogo, incluindo botões, textos, imagens e outros elementos interativos.



Mova o canvas para dentro do personagem

• Mover o canvas para dentro do personagem permite que o canvas se movimente junto com o personagem.



Propriedades de texto

- Selecione o componente de texto.
 - No inspector altere propriedades de texto como cor, negrito e itálico, alinhamento e etc.

Hierarchy	Inspector	Services	
Create * Q*All	👕 🗹 Text		🗌 Static 🔻
Gaula01* →=	Tag Untage	ged 🕴 Laver UI	
V Personagem	-90		
v Canvas Text	V OB Rect Ira		Pos 7
Main Camera		200 62	0
▼ cena		Width Height	
chão		160 30	
Diamond_Azuul_1	Anchors		
Diamond_Azuul_2 Diamond_Azuul_3	Pivot	X 0.5 Y 0.5	
Diamond_Azuul_4	Rotation	X 0 Y 0	Z 0
Diamond_Azuul_5	Scale	X 1 Y 1	Z 1
Diamond_Vermelho_1	O Canvas	Renderer	a .
Diamond_Vermelho_2	Text (Se	rint)	
Diamond_Vermelho_3	Text		
Diamond_vermeino_4	New Text		
Diamond_vermeino_3			
	Character		
	Font	Arial	0
	Font Style	Normal	\$
	Font Size	14	
	Line Spacing	1	
	Rich Text	I	
	Paragraph		
	Alignment		
	Align By Geo	ometry 🗌	
	Horizontal O	verflow Wrap	+
	Vertical Ove	rflow Truncate	+
	Best Fit		

Altere a renderização de câmera

 Clique no canvas posicionado na hierarquia do projeto, no inspector mude a propriedade render mode para "Screen Space - Camera"

The Edit Assets GameObject Component Window Help					
🖑 💠 🔀 🛄 💓 🔍 🔍 Center 📽 Local					
≡ Hierarchy					
Create * Q*All					
	1 Ipspector Services				<u> </u>
▼ Personagem	Canvas				🗌 Static 🔻
▼ Canvas	Tag Untagged	+ L	ayer UI	′	;;
Text	Rect Transform				D 0.
Main Camera	Some values driven by Canvas.				
▼ cena		Dec V	Dec V	Dec 7	
chão		717.49	258.26	0	
Diamond_Azuul_1		Width	Height		
Diamond_Azuul_2		1418	516		E R
Diamond_Azuul_3	Anchors				
Diamond_Azuul_4	Pivot	X 0.5	Y 0.5		
Diamond_Azuul_5	Rotation	x O	Y O	7 0	
Diamond_Vermelho_1	Scale	X 1	Y 1	7 1	
Diamond_Vermelho_2					A
Diamond_Vermelho_3	Render Mode	Screen Space -	Overlay		1 m
Diamond_Vermelho_4	Pixel Perfect		o la la la		
Diamond_Vermelho_5	Sort Order	 Screen Sp 	ace - Overlay		
	Target Display	Screen Sp	ace - Camera		+
	Additional Shader Channels	World Sp	ace		;
	▼ 🖾 🗹 Canvas Scaler (Script)				[2] \$,

🚭 Unity 2017.4.40f1 Personal (64bit) - aula01.unity - 1_bimestre_Aulas - PC, Mac &

File Edit Assets GameObject Component Window Help

Render Camera

 Clique duas vezes na propriedade "Render Camera" e selecione "Main Camera" com dois cliques.

	Inspector Services			≟ .=
	🍞 🗹 Canvas			🗌 Static 🔻
	Tag Untagged	‡ Laye	r UI	\$
1	▼ ∂ ∂ Rect Transform			🔃 \$,
	Some values driven by Canvas.			
		Pos X	Pos Y	Pos Z
		1.07	0.72	90
		Width	Height	
		1418	516	[] R
,	Anchors			
	Pivot	X 0.5	Y 0.5	
	Rotation	X 0	Y 0	Z 0
	Scale	X 0.03012007	Y 0.03012007	Z 0.03012007
	▼ Canvas			[🖉 🔅,
	Render Møde	Screen Space - Camera		
	Pixel Perfect			
	Render Camera	📹 Main Camera (Camera)		
	Plane Distance	Select Camera	×	
	Sorting Layer			*]
	Order in Layer	Q		
	Additional Shader Channels	Assets Scene		*]
	▼ 🕅 🗹 Canvas Scaler (Script) UI Scale Mode	None		🔊 🌣, ;

Configure os componentes

- Posicione livremente o componente de texto no cenário.
- Ao tocar em "pontos1" o personagem ganha 1 ponto. Ao tocar em "pontos2" o personagem ganha 2 pontos.
- Configurações:
 - Diamante azul
 - Colisor
 - Tag: "pontos1"
 - Diamante vermelho
 - Colisor
 - Tag: "pontos2"



Altere o nome do componente

- Para identificar o componente de texto na programação é necessário modificar o seu nome.
- O exemplo modifica o nome de "Text" para "TXT_QTD_PONTOS".

'≔ Hierarchy	<u> </u>	O Inst	🗄 Hierarchy 🔒 📲
Create * Q*All			Create * Q*All
▼ 🚭 aula01*	*≡	1	▼ 🚭 aula01* -=
▼ Personagem		Τi	▼ Personagem
▼ Canvas		VAR	▼ Canvas
Text	<u>_</u>	1.00	TXT_QTD_PONTOS
Main Camera	Сору		Main Camera
▼ cena	Paste		▼ cena
chão			chão
Diamond Azuul	Rename	t	Diamond_Azuul_1
Diamond Azuul	Duplicate		Diamond_Azuul_2
Diamond Azuul	D L .	1	Diamond_Azuul_3
Diamond Azuul	Delete	ł	Diamond_Azuul_4
Diamond Azuul	Colort Deefah		Diamond_Azuul_5
Diamond Verm	Select Prelab	-	Diamond_Vermelho_1
Diamond Verm	Create Empty		Diamond_Vermelho_2
Diamond Verm	create Empty		Diamond_Vermelho_3
Diamond Verm	3D Object	>	Diamond_Vermelho_4
Diamond Verm	2D Object	>	Diamond_Vermelho_5

Biblioteca

using UnityEngine; //Biblioteca utilizada para cenários using UnityEngine.SceneManagement; //Biblioteca utilizada para trabalhar com o UI using UnityEngine.UI;

<mark>Atributos</mark>

public class tempBimestre2 : MonoBehaviour {

// Observe que antes do tipo da variável há a palavra "public", isso significa
// que esse atributo pode ser acessado de outros lugares
// e é possível visualizar e inicializar esse atributo na interface do inspector.

public Text UITextPontos;



Configure

- A inicialização do atributo
 UITextPontos pode ser diferente quando público
- Para inicializar, arraste do componente "TXT_QTD_PONTOS" para dentro de UITextPontos no inspector

	Collab • 🛆 Ac	count • Layers • Layout •
'≔ Hierarchy 🔒 📲	Inspector	<u></u> = +≡
Create * Q*All	Circle	Static 🔻
▼ 🚭 temp* -=		t Laver Default t
Square	Tag Oncagged	· Layer Delaute ·
Square (1)	▼ 🙏 Transform	[] \$,
Square (2)	Position	X -4.14 Y -1.07 Z 0
Square (3)	Rotation	X 0 Y 0 Z 0
Square (4)	Scale	X 0.4959133 Y 0.4959133 Z 0.4959133
Main Camera	🔻 🔍 🗹 Sprite Renderer	💽 \$.
T Canvas	Sprite	Circle 0
TXT OTD PONTOS	Color	1
Diamond	Flip	
Square (5)	Material	Sprites-Default 0
EventSystem	Draw Mode	Simple \$
	Sorting Layer	Default +
	Order in Layer	0
	Mask Interaction	None +
	▼ 🕼 🗹 Temp Bimestre 2 (Script)	
	Script	€ tempBimestre2 O
	Ui Text Pontos 🔷 🔪	TXT_QTD_PONTOS (Text) 0
	🔻 🌒 🗹 Circle Collider 2D	
		B Edit Collider
	Material	None (Physics Material 2D) O
	Is Trigger	
	Used By Effector	
	Offset	X 0 Y 0
	Radius	0.5
	▶ Info	

```
void Start () {
        OtdPontos = 0;
        TagTriggerStay=""
                                                                                   Método Start()
        TagTriggerEnter=""
       TagTriggerExit=""
       TagObjetoStay=""
       TagObjetoExit="";
       MaterialSemAtrito = new PhysicsMaterial2D();
       TotalPulos = 1;
       ContadorPulos = 0;
       Renderer = GetComponent<SpriteRenderer> ();
        PegouChave01 = false;
        TagObjetoTocado = "";
       ApertouBotaoPular = false; // inicia a variavel com false;
        EstaTocandoAlgumColisor = false; // inicializa a variavel como falso;
       VelocidadePuloSimples = 10.0f; // determina o valor da velocidade do Pulo;
       // Obtém o componente Rigidbody2D do objeto ao qual este script está anexado
       CorpoRigidoPersonagem = GetComponent<Rigidbody2D> ();
       Colisor2dPersonagem = GetComponent<Collider2D> ();
       // Define a escala da gravidade para o Rigidbody2D (1 é o valor padrão)
       CorpoRigidoPersonagem.gravityScale = 3.0f;
       // Congela a rotação do Rigidbody2D para evitar que o personagem gire ao colidir com algo
       CorpoRigidoPersonagem.freezeRotation = true;
       // Inicializa as variáveis de velocidade
       VelocidadeX = Of; // Começa parado no eixo X
       VelocidadeY = Of; // Comeca parado no eixo Y
       VelocidadeHorizontalMaxima = 5.0f; // Define a velocidade máxima no eixo X
       DirecaoHorizontal = Of; // Começa sem direção definida
       VelocidadeVerticalMaxima = 5.0f; // Define a velocidade maxima no eixo y
       DirecaoVertical = 0f;
                                        // Define a direcão do movimento na vertical
       // Cria um vetor de velocidade inicial para o personagem
       VetorVelocidadePersonagem = new Vector2 (VelocidadeX, VelocidadeY);
       // Aplica o vetor de velocidade ao Rigidbody2D para mover o personagem
       CorpoRigidoPersonagem.velocity = VetorVelocidadePersonagem;
```

OnCollisionEnter2D()

void OnCollisionEnter2D(Collision2D objetoTocado){

```
TagObjetoTocado = objetoTocado.gameObject.tag;
```

```
if (TagObjetoTocado == "pontos1" ) {
    QtdPontos=QtdPontos+1; //soma 1 em QtdPontos
    //Converte o valor de QtdPontos para o tipo string(texto)
    string textoPontos= QtdPontos.ToString();
    //Armazena o texto com os pontos na propriedade
    //Text de UITextPontos
    UITextPontos.text = textoPontos;
    //destrói pontos1
    Destroy (objetoTocado.gameObject);
}
```

Transforma um número em texto Pois: UiTextPontos.text só é capaz de receber variáveis do tipo texto(string)

```
if (TagObjetoTocado == "pontos2" ) {
    QtdPontos=QtdPontos+2; //soma 2 em QtdPontos
    //Converte o valor de QtdPontos para o tipo string(texto)
    string textoPontos= QtdPontos.ToString();
    //Armazena o texto com os pontos na propriedade
    //Text de UITextPontos
    UITextPontos.text = textoPontos;
    Destroy (objetoTocado.gameObject);//destrói pontos2
}
```

Text é uma propriedade de UITextPontos que contem o texto que será apresentado.