

ERBASE – Escola Regional de Computação Bahia-Alagoas-Sergipe
Aracaju, 22 de Agosto de 2018
Laboratório de Entretenimento Digital Aplicado (LEnDA) - UEFS
prof. D.Sc. Victor Sarinho

Minicurso – Introdução ao Godot 3 (1ª parte)

Atividade 1 – Contador Temporal;

- 1- Crie um novo projeto;
- 2- Adicione um Panel na cena inicial;
- 3- Adicione um Label e um Timer ao Panel da cena principal;
- 4- Ative a propriedade Autostart do Timer no Inspector;
- 5- Adicione um script ao Label;
- 6- Adicione o seguinte código ao script do Label:

```
extends Label
```

```
var count=0
```

```
func _ready():  
    get_node("/root/Panel/Timer").connect("timeout",self,"_on_Timer_timeout")
```

```
func _on_Timer_timeout():  
    print(count)  
    count += 1  
    self.set_text(str("count = ",count))
```

- 7- Execute o projeto (salve a cena antes) e veja o resultado.

Atividade 2 – Trabalhando com Sprites;

- 1- Crie um novo projeto;
- 2- Adicione um Sprite na cena inicial;
- 3- Selecione a propriedade Texture do Sprite no Inspector e carregue uma nova imagem para o sprite;
- 4- Adicione um script ao Sprite;
- 5- Adicione o seguinte código ao script do Sprite:

```
extends Sprite
```

```
func _ready():  
  
    print("viewport:"+str(get_viewport_rect()))  
    #translate to center of the parent, in this case, the viewport  
    var newPos = Vector2(get_viewport_rect().size.x/2, get_viewport_rect().size.y/2)  
    self.set_position(newPos)
```

```
#rotate by 90 degrees. set_rot takes radians so we need to convert using in-built function  
self.set_rotation(deg2rad(90))
```

```
#scale by 2x  
self.set_scale(Vector2(2,2))
```

```
func _draw():  
    # each frame draw a bounding rect to show boundaries clearer  
    self.draw_rect(self._edit_get_rect(), Color(0,0,1,0.2))
```

6- Execute o projeto (salve a cena antes), veja o resultado;

7- Substitua o script do Sprite pelo script abaixo, execute o programa e veja o resultado:

```
extends Sprite
```

```
func _ready():  
    self.set_process(true)
```

```
func _process(delta):  
    var cur_pos = self.get_position()  
    cur_pos.x += 100 * delta  
  
    # wrap around screen  
    if(cur_pos.x > get_viewport_rect().size.x + self.get_region_rect().size.x/2):  
        cur_pos.x = -self.get_region_rect().size.x/2  
    self.set_position(cur_pos)
```

8- Execute o projeto (salve a cena antes), veja o resultado;

9- Adicione um Timer (vide atividade anterior) ao Sprite e adapte o script para mover o Sprite com o passar do tempo do Timer.

Atividade 3 – Entradas do Usuário;

- 1- Crie um novo projeto;
- 2- Adicione um Panel na cena inicial;
- 3- Adicione um script ao Panel;
- 4- Adicione o seguinte código ao script do Panel:

```
extends Panel
```

```
func _ready():  
    self.set_process(true)
```

```
func _process(delta):  
    if(Input.is_key_pressed(KEY_ESCAPE)):  
        if(Input.is_key_pressed(KEY_SHIFT)):  
            get_tree().quit()
```

5- Execute o projeto (salve a cena antes), veja o resultado;

6- Acrescente o código abaixo na função `_process` e veja o resultado:

```
if (Input.is_mouse_button_pressed(BUTTON_LEFT)):
    print(str("Mouse at location:",get_viewport().get_mouse_position(), " moving at speed: ",
Input.get_last_mouse_speed()));
```

Atividade 4 – Inputs & Sprites;

- 1- Crie um novo projeto;
- 2- Adicione um Sprite na cena inicial;
- 3- Selecione a propriedade Texture do Sprite no Inspector e carregue uma nova imagem para o sprite;
- 4- Adicione um script ao Sprite;
- 5- Adicione o seguinte código ao script do Sprite:

extends Sprite

```
func _ready():
    set_process_input(true)

func _input(event):
    # if user left clicks
    if(event is InputEventMouseButton):
        if(event.button_index == 1):
            self.set_position(Vector2(event.position.x,event.position.y))

    # on keyboard cursor key
    if(event is InputEventKey):
        var curPos = self.get_position()

        if(event.scancode == KEY_RIGHT):
            curPos.x+= 10
            self.set_position(curPos)

        if(event.scancode == KEY_LEFT):
            curPos.x-= 10
            self.set_position(curPos)
```

- 6- Execute o projeto (salve a cena antes), veja o resultado.

- 7- Selecione /Project/Project Settings no menu inicial;
- 8- Selecione a aba InputMap e adicione uma nova ação denominada MOVE_RIGHT;
- 9- Acrescente um novo Sprite ao Panel e adicione um novo script ao mesmo;
- 10- Utilize um dos métodos abaixo para manipular a ação MOVE_RIGHT no novo Sprite criado:

extends Sprite

```
func _ready():
    set_process(true)

func _process(delta):
    if(Input.is_action_pressed("MOVE_RIGHT")):
```

```

var cur_pos = self.get_position()
cur_pos.x += 1
self.set_position(cur_pos)

```

ou

extends Sprite

```

func _ready():
    set_process_input(true)

func _input(event):
    if(event.is_action("MOVE_RIGHT")):
        var cur_pos = self.get_position()
        cur_pos.x += 1
        self.set_position(cur_pos)

```

Atividade 5 – Efeitos Sonoros;

- 1- Faça o download de arquivos .WAV ou .OGG na web (<http://www.freesound.org/> por exemplo)
- 2- Crie um novo projeto;
- 3- Adicione um Panel na cena inicial;
- 4- Adicione um AudioStreamPlayer ao Panel da cena principal;
- 5- Selecione a propriedade Stream do AudioStreamPlayer no Inspector e carregue um arquivo .WAV ou .OGG;
- 6- Selecione a aba InputMap e adicione uma nova ação denominada PLAYER_JUMP;
- 7- Adicione um script ao AudioStreamPlayer e aplique o código abaixo:

extends AudioStreamPlayer

```

func _ready():
    set_process_input(true)

func _input(event):
    if event.is_action_pressed("PLAYER_JUMP"):
        play()

```

- 8- Execute o projeto (salve a cena antes), veja o resultado;
- 9- Adicione um Button ao Panel da cena principal;
- 10- Adicione um script ao Button e aplique o código abaixo:

extends Button

```

func _ready():
    get_node("/root/Panel/Button").connect("pressed",self,"_on_Button_pressed")

func _on_Button_pressed():
    get_tree().get_root().get_node("Panel").get_node("AudioStreamPlayer").play()

```

- 11- Execute o projeto (salve a cena antes), veja o resultado.

