

Blender + Python



Agenda

- **¿Qué es Blender?**
- **Proceso de animación**
- **Python en Blender**

¿Qué es Blender?

- **1995 → Ton Roosendaal (NeoGeo)**
- **1998 → NaN**
- **2002 → Blender Foundation (100mil EU)**



¿Qué es Blender?

- **Software para modelado, iluminación, renderizado, animación 3D y 2D.**
- **Software para composición digital: edición de vídeo, escultura y pintura.**
- **Software para vídeo juegos**
- **Software libre con licencia GPLv2**

¿Qué es Blender?

- **Servicios de la Blender Foundation**

- <https://www.blender.org/id/login>
- <https://cloud.blender.org/>
 - <https://cloud.blender.org/p/spring> (Blender 2.8)
- <https://www.blendernetwork.org/>
- <https://www.blender.org/conference/>
- <https://store.blender.org/>
- <https://code.blender.org/> (Development)

Proceso de animación

- **Pre-producción**

- Conceptualización: historia/storyline, personajes, espacios, música, ambientación, Gameplay (en los juegos)

- **Producción**

- Diseño: artístico/gráficos, sonido, mecánico,
- Implementación_ modelado, rigging, materiales, texturas, animación, iluminación, renderizado
- Pruebas: jugabilidad (en los juegos)

- **Post-producción**

- Voces, música, efectos (FX)

Python en Blender

- **Python 3.6**
- **En GNU/Linux lanzar Blender desde la consola**
- **DEMO**

```
for x in range(0, 5):
```

```
    for y in range(0, 5):
```

```
        for z in range(0, 5):
```

```
            bpy.ops.mesh.primitive_cube_add(view_align=False,  
enter_editmode=False, location=(x*3, y*3, z*3))
```

Python en Blender

```
import bpy
```

```
positions = (0,3,2), (4,1,3), (3,-5,1), (3,10,1), (1,8,1)  
start_pos = (0,0,0)
```

```
ob = bpy.data.objects["Sphere"]
```

```
frame_num = 0
```

```
for position in positions:
```

```
    bpy.context.scene.frame_set(frame_num)
```

```
    ob.location = position
```

```
    ob.keyframe_insert(data_path="location", index = -1)
```

```
    frame_num += 20
```



```

• import bpy
• import math
•
• def create_wall():
•     for x in range(0, 5):
•         for y in range(0, 5):
•             for z in range(0, 5):
•                 bpy.ops.mesh.primitive_cube_add(view_align=False, enter_editmode=False, location=(x*3, y*3, z*3))
•
• def degree_to_radian(degree):
•     return math.radians(degree)
•
• #create_wall()
• camera = bpy.data.objects['Camera']
• camera.rotation_mode = 'XYZ' #XYZ Euler
•
• positions = (-6, -36, 0.2), (-3, -12, 1.2), (-2,-8, 1.2), (-1,-4, 1.2), (-0, -0, 1.2), (0, 1, 1.2)
• start_pos = (-6, -36, 0.2)
• x_rotation = 90
•
• frame_num = 0
•
• for position in positions:
•     bpy.context.scene.frame_set(frame_num)
•     camera.location = position
•     camera.rotation_euler[0] = degree_to_radian(x_rotation)
•     camera.keyframe_insert(data_path="location", index = -1)
•     x_rotation += 0.4
•     frame_num += 20

```

Python en Blender

```
import bpy

# diguja el boton
class ToolsPanel(bpy.types.Panel):
    bl_label = "Crear esferas"
    bl_space_type = "VIEW_3D"
    bl_region_type = "TOOLS"

    def draw(self, context):
        self.layout.operator("crear.geoda")

# funcionalidad del boton
class OBJECT_OTGEO(bpy.types.Operator):
    bl_idname = "crear.geoda"
    bl_label = "Crear esfera geoda"

    def execute(self, context):
        bpy.ops.mesh.primitive_ico_sphere_add(subdivisions=2, size=1)

        return {'FINISHED'}

# registrar la funcionalidad
bpy.utils.register_module(__name__)
```

Referencias

- <https://es.wikipedia.org/wiki/Blender>
- <https://elbinario.net/2018/01/31/scripting-con-blender-y-python/>
- <https://www.youtube.com/watch?v=HQlTqiBdvk>
- <https://www.youtube.com/watch?v=pywbPQD9vYU>
(Grease Pencil 2D animation - Blender 2.8)