Pinning and Rotating Physics Bodies

Pin a node so it's free to rotate about a certain point on its parent node.

Overview

A common task in a physics simulation is to pin bodies together. Pinning locks the position of an object but allows it to rotate—for example, to fix a wheel to a vehicle. SpriteKit offers two options for pinning physics bodies—either using the `pinned` property or creating an `SKPhysicsJointPin` joint.

The `pinned` property offers convenience because it requires less code. If you wish to simulate friction, the physics body's `angularDamping` property is analogous to the joint's `frictionTorque` property. You must use a joint, however, if you want to control the rotation speed. In addition, use a joint to control the anchor point or limit the rotation angles—you do that by using `lowerAngleLimit` and `upperAngleLimit`.

Pin a Node Using the Pinned Property

The following code demonstrates using the pinned property. The code creates a sprite node with an image of a pendulum (two circles of different sizes joined by a center rod). A physics body is created based on the generated texture, has its pinned property set to true, and is assigned to the node. The node is rotated 90 degrees—to align horizontally—so that gravity pulls the heavier end toward the ground.

```
let pinned = SKSpriteNode(imageNamed: "pendulum")

pinned.physicsBody = SKPhysicsBody(texture: pinned.texture!,
                                   size: pinned.size)
pinned.position = CGPoint(x: 320,
                          y: 320)
pinned.zRotation = CGFloat.pi / 2
pinned.physicsBody?.pinned = true

scene.addChild(pinned)
```

Because the node's physics body is pinned, it stays in position but rotates around its center.

Pin a Node Using a Joint

The following code demonstrates using a physics joint pin. In this example, a static anchor is created and added to the scene. The sprite node is added as a child of the anchor node and a SKPhysicsJointPin joins the two together.

```
let anchor = SKNode()

anchor.position = CGPoint(x: 320,
                          y: 320)
anchor.zRotation = CGFloat.pi / 2
anchor.physicsBody = SKPhysicsBody()
anchor.physicsBody?.isDynamic = false

let jointed = SKSpriteNode(imageNamed: “pendulum”)
jointed.physicsBody = SKPhysicsBody(texture: pinned.texture!,
                                    size: pinned.size)

scene.addChild(anchor)
anchor.addChild(jointed)

let pinJoint = SKPhysicsJointPin.joint(withBodyA: anchor.physicsBody!,
                                       bodyB: jointed.physicsBody!,
                                       anchor: anchor.position)
scene.physicsWorld.add(pinJoint)
```