

// ViewController.swift

```
import UIKit  
import SceneKit  
import ARKit
```

enum BoxBodyType : Int {

```
    case bullet = 1  
    case barrier = 2
```

}

class viewController: UIViewController, ARSCNViewDelegate, SCNPhysicsContactDelegate {  
 var sceneView: ARSCNView!  
 var lastContactNode: SCNNode! // make sure not getting too many contact requests  
 override func viewDidLoad() {  
 super.viewDidLoad()  
 self.sceneView = ARSCNView(frame: self.view.frame)  
 self.sceneView.debugOptions = [ARSCNDebugOptions.showFeaturePoints, ARSCNDebugOptions.showWorldOrigin]  
 self.view.addSubview(self.sceneView)  
 sceneView.delegate = self  
 sceneView.showsStatistics = true  
 let scene = SCNScene()

let box1 = SCNBox(width: 0.1, height: 0.1, length: 0.1, chamferRadius: 0)

let material1 = SCNMaterial()

material1.diffuse.contents =UIColor.red

box1.materials = [material1]

let box1Node = SCNNode(geometry: box1) // share the same geometry

let box2Node = SCNNode(geometry: box1)

let box3Node = SCNNode(geometry: box1)

box1Node.name = "Barrier1"

↓ similar ones

box1Node.physicsBody = SCNPhysicsBody(type: .static, shape: nil)

↓ similar ones

box1Node.physicsBody?.categoryBitMask = BoxBodyType.barrier.rawValue

↓ similar ones

box1Node.position = SCNVector3(0, 0, -0.4)

box2Node.position = SCNVector3(-0.2, 0, -0.4)

box3Node.position = SCNVector3(0.2, 0.2, -0.4)

scene.rootNode.addChildNode(box1Node)

↓ similar ones

sceneView.scene = scene

self.sceneView.scene.physicsWorld.contactDelegate = self

register GestureRecognizers()

3

```

func physicsWorld(_ world: SCNPhysicsWorld, didBegin contact: SCNPhysicsContact) {
    var contactNode: SCNNode!
    if contact.nodeA.name == "Bullet" {
        contactNode = contact.nodeB
    } else {
        contactNode = contact.nodeA
    }
    if self.lastContactNode != nil && self.lastContactNode == contactNode {
        return
    }
    self.lastContactNode = contactNode
    // let material = SCNMaterial()
    // material.diffuse.contents = UIColor.green // will cause all the boxes change to green because all use box1
    // self.lastContactNode.geometry?.materials = [material]

    let box1 = SCNBox(width: 0.1, height: 0.1, length: 0.1, chamferRadius: 0)
    let material1 = SCNMaterial()
    material1.diffuse.contents = UIColor.green
    box1.materials = [material1]
    self.lastContactNode.geometry = box1
}

private func registerGestureRecognizers() {
    let tapGestureRecognizer = UITapGestureRecognizer(target: self, action: #selector(shoot))
    self.sceneView.addGestureRecognizer(tapGestureRecognizer)
}

@objc func shoot(recognizer: UIGestureRecognizer) {
    guard let currentFrame = self.sceneView.session.currentFrame else {
        return
    }
    var translation = matrix_identity_float4x4
    translation.columns.3.z = -0.3
    let box = SCNBox(width: 0.05, height: 0.05, length: 0.05, chamferRadius: 0)
    let material1 = SCNMaterial()
    material1.diffuse.contents = UIColor.yellow
    let boxNode = SCNNode(geometry: box)
    boxNode.name = "Bullet"
    boxNode.physicsBody = SCNPhysicsBody(type: .dynamic, shape: nil)
    boxNode.physicsBody?.categoryBitMask = BoxBodyType.bullet.rawValue
    boxNode.physicsBody?.contactTestBitMask = BoxBodyType.barrier.rawValue // which types the boxNode collide with trigger notification
    boxNode.physicsBody?.isAffectedByGravity = false // orient the box to make it face the current frame
    boxNodesimdTransform = matrix_multiply(currentFrame.camera.transform, translation)
    let forceVector = SCNVector3(boxNode.worldFront.x * 2, boxNode.worldFront.y * 2,
                                boxNode.worldFront.z * 2)
    boxNode.physicsBody?.applyForce(forceVector, asImpulse: true)
    self.sceneView.scene.rootNode.addChildNode(boxNode)
}

```