

```

/*
https://www.youtube.com/watch?v=R1XHWyprFVk
*/

import UIKit
import RealityKit
import PlaygroundSupport

//let modelEntity = try ModelEntity.load(named:
    "toy_biplane.usdz")
let anchorEntity = AnchorEntity(plane: .horizontal)
let modelEntity = createCube()
modelEntity.generateCollisionShapes(recursive:true)

//let modelEntity = try ModelEntity.load(named: "Plane.reality")
anchorEntity.name = "CubeAnchor"
anchorEntity.addChild(modelEntity)

print("load")
let arView = ARView(frame: UIScreen.main.bounds , cameraMode: .ar,
    automaticallyConfigureSession: true)

//addCube(arView: arView)
arView.scene.addAnchor(anchorEntity)

let alert = UIAlertController(title: "My Alert", message: "This is
    an alert.", preferredStyle: .alert)
alert.addAction(UIAlertAction(title: NSLocalizedString("OK",
    comment: "Default action"), style: .default, handler: { _ in
        NSLog("The \"OK\" alert occured.")
    })))

```

```
arView.installGestures( [.translation, .rotation, .scale] , for:
    modelEntity)
arView.enableObjectRemoval()
```

```
extension ARView {
    func enableObjectRemoval() {
        let longPressGestureRecognizer =
            UILongPressGestureRecognizer(target: self, action:
                #selector(handleLongPress(recognizer:)))
        self.addGestureRecognizer(longPressGestureRecognizer)
    }
    @objc func handleLongPress(recognizer:
        UILongPressGestureRecognizer){
        print("long press..")
        //vc.present(alert, animated: true, completion: nil)
        let location = recognizer.location(in: self)
        if let entity = self.entity(at: location) {
            print("entity")
            if let anchorEntity = entity.anchor, anchorEntity.name
                == "CubeAnchor" {
                anchorEntity.removeFromParent()
                print("remove anchor with name: " +
                    anchorEntity.name)
            }
        }
    }
}
```

```
//PlaygroundPage.current.needsIndefiniteExecution = true
```

```
//var vc = MyController()
```

```
//PlaygroundPage.current.liveView = vc
```

```
PlaygroundPage.current.setLiveView(arView)
```