

Class 05-1



SpriteKit is Apple's exciting 2D game framework that was first released in September 2013 with iOS 7.

Sprite Kit

Spritekit Basics

- SKNode
 - ◆ SKNode是所有內容的基礎類別(包含SKScene)
 - ◆ SKLabelNode、SKSpriteNode、SKVideoNode 等皆是SKNode的子類別
- SKScene
 - ◆ 在Scene中，所有元素皆屬於樹狀的資料結構，而SKScene是根節點(root)，每個節點的位置會由父節點的座標系決定。
 - ◆ 一個遊戲可能會需要建立多個SKScene的子類別
(e.g. 遊戲主選單、遊戲場景、遊戲結束場景)
- SKAction
 - ◆ 此類別用來對SKNodes 設定動作 (e.g. 移動 縮放 旋轉 播放音訊)
- SKPhysicsBody
 - ◆ 此類別將物理特性添加到SKNodes，使其可以模擬出複雜的變化(碰撞 重力 摩擦力)

Spritekit Basics

- **SKView:** This is a subclass of the UIView. SKView is the view in which all of the SpriteKit contents are added and displayed on the screen.
- **SKNode:** This is the building block of SpriteKit. It is just an empty node. No visual elements. On a SKNode we can add child nodes such as SKSpriteNode. These nodes can be visual elements or a scene.
- **SKScene:** An SKScene is a subclass of the SKNode. It represents a single scene from the game. SKScene acts as a root node of the SKNode. SKScene provides the content to the SKView. You just need to call the function presentScene over the SKView and pass the scene instance.
- **SKAction:** An SKAction is used to set actions such as animations/movements on SKNodes.
- **SKPhysicsBody:** This class adds the physics components to the SKNodes. Components such as gravity, collision are commonly used in games.

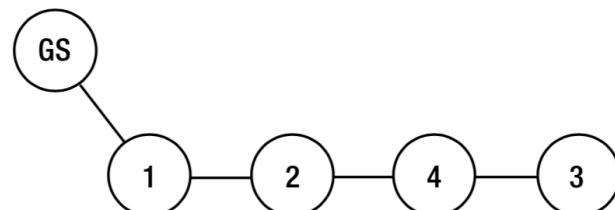
SpriteKit Node Tree

Method	Purpose
addChild()	The <code>addChild(_:)</code> method adds a node to the end of the receiver's collection of child nodes.
insertChild(_:at:)	The <code>insertChild(_:at:)</code> method inserts a child node at a specific position in the receiver's collection of child nodes.
removeFromParent()	<code>removeFromParent()</code> removes the receiving node from its parent.

```
var node1 = SKSpriteNode()
var node2 = SKSpriteNode()
var node3 = SKSpriteNode()
```

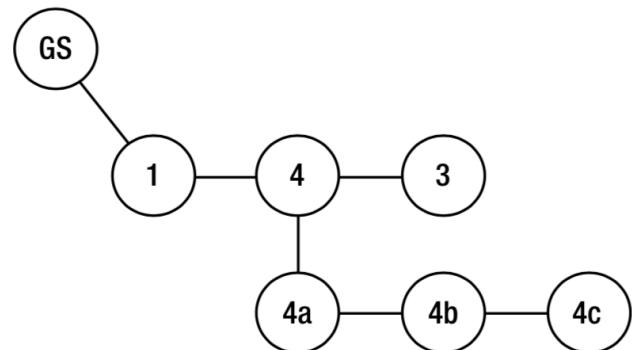
```
gameScene.addChild(node1)
gameScene.addChild(node2)
gameScene.addChild(node3)
```

```
var node4 = SKSpriteNode()
gameScene.insertChild(node4, at: 2)
```

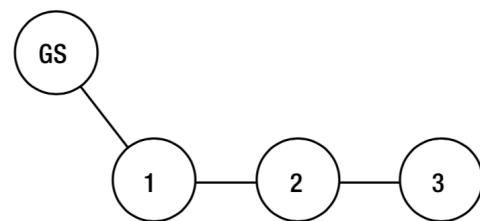
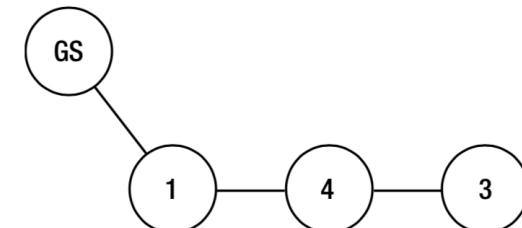


```
var node4a = SKSpriteNode()
var node4b = SKSpriteNode()
var node4c = SKSpriteNode()
```

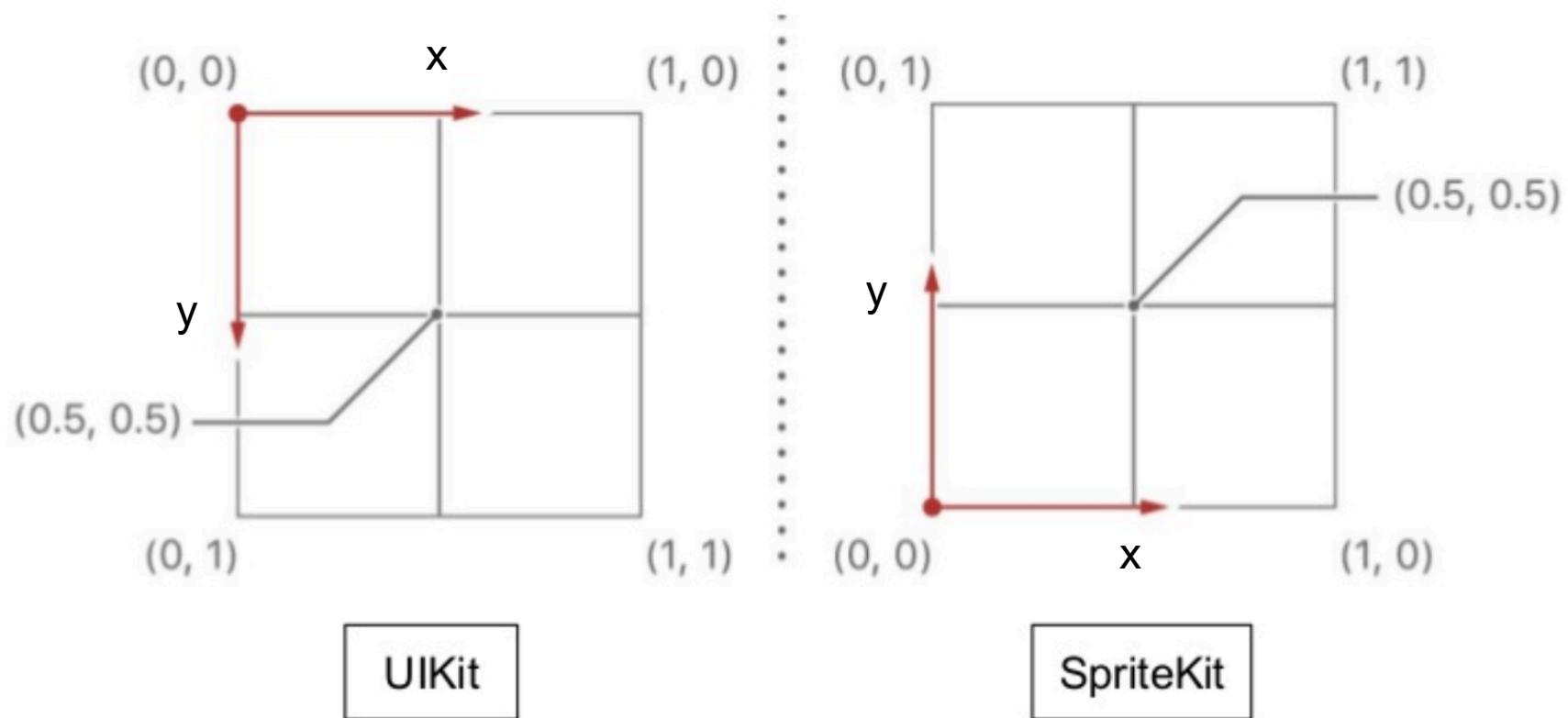
```
node4.addChild(node4a)
node4.addChild(node4b)
node4.addChild(node4c)
```



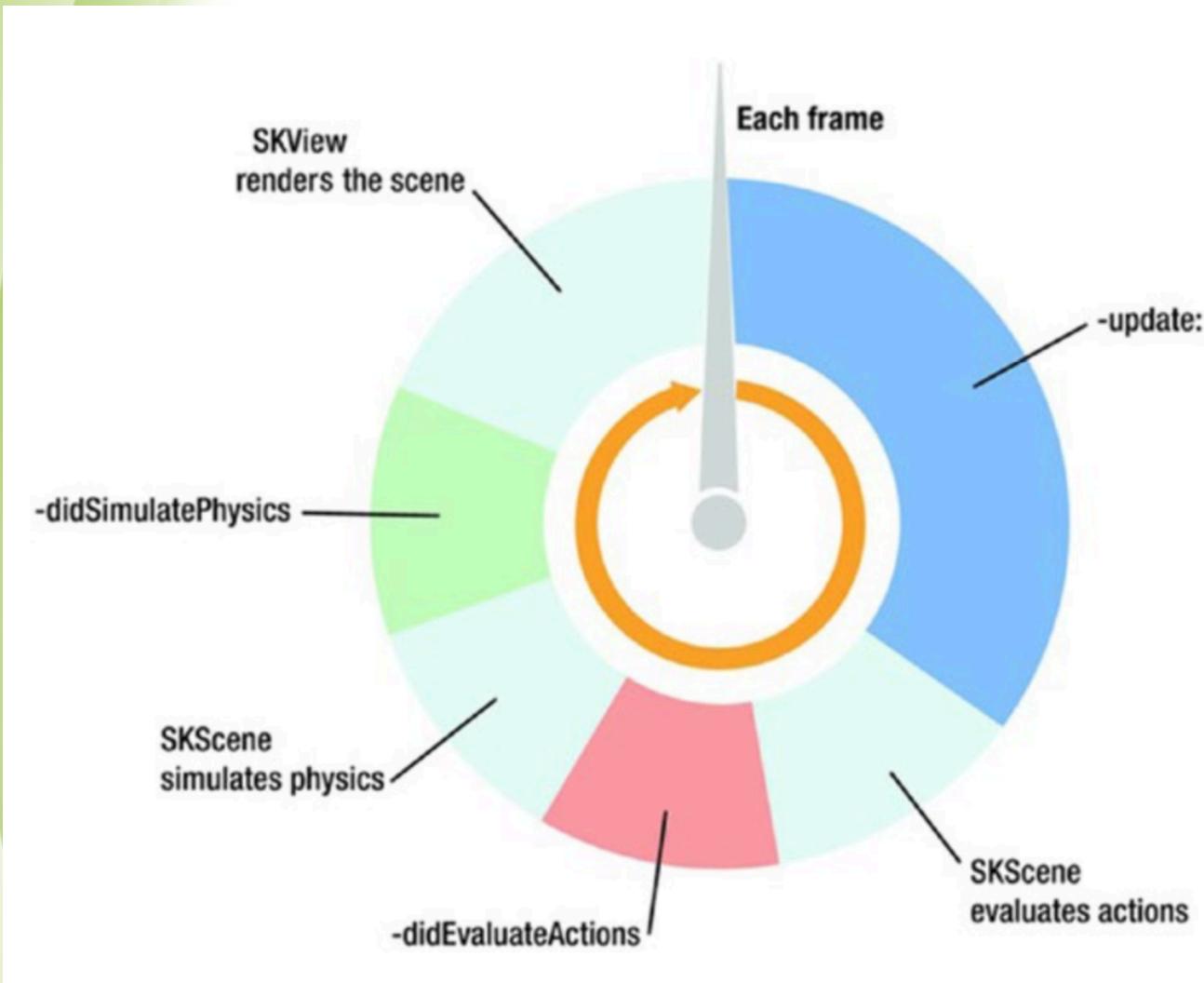
`node2.removeFromParent()`



SpriteKit Coordinate



Spritekit rendering loop

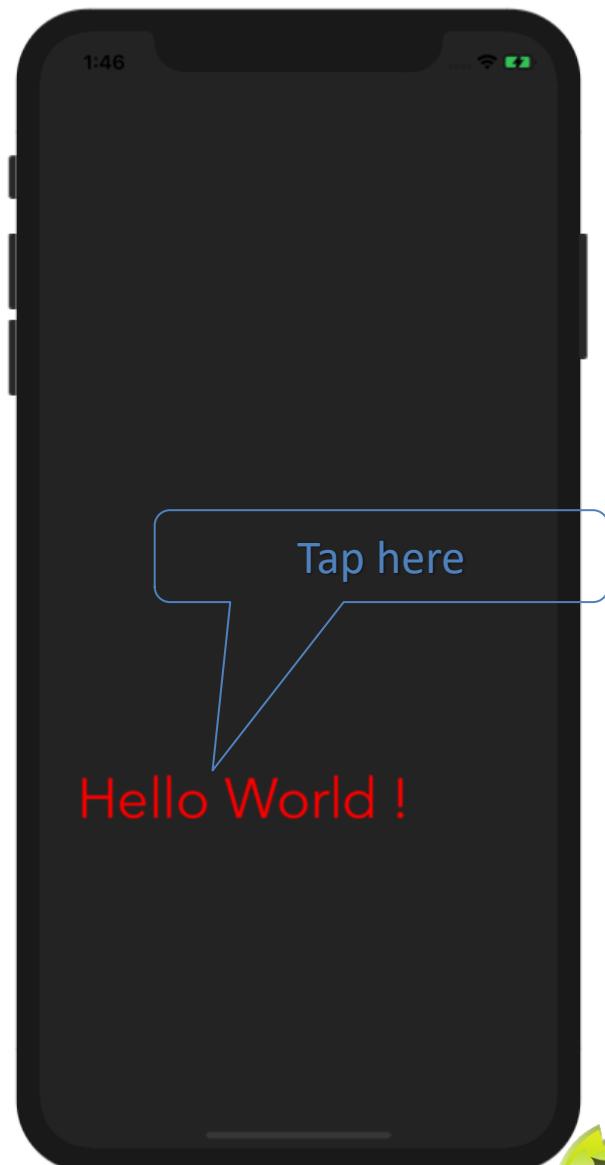


SpriteKit Example

Hello World !



iPhone 11 — 13.3



iPhone 11 — 13.3



Welcome to Xcode

Version 11.3.1 (11C504)



Get started with a playground

Explore new ideas quickly and easily.



Create a new Xcode project

Create an app for iPhone, iPad, Mac, Apple Watch, or Apple TV.



Clone an existing project

Start working on something from a Git repository.



Show this window when Xcode launches



Firspsrite

~/Desktop



cpractice

~/Desktop



MyPlayground

~/Desktop



hellosprite

~/Desktop/2016 iOS/pr6 4h



hw1

~/Desktop



2-1

~/Desktop



Calculator

~/Desktop

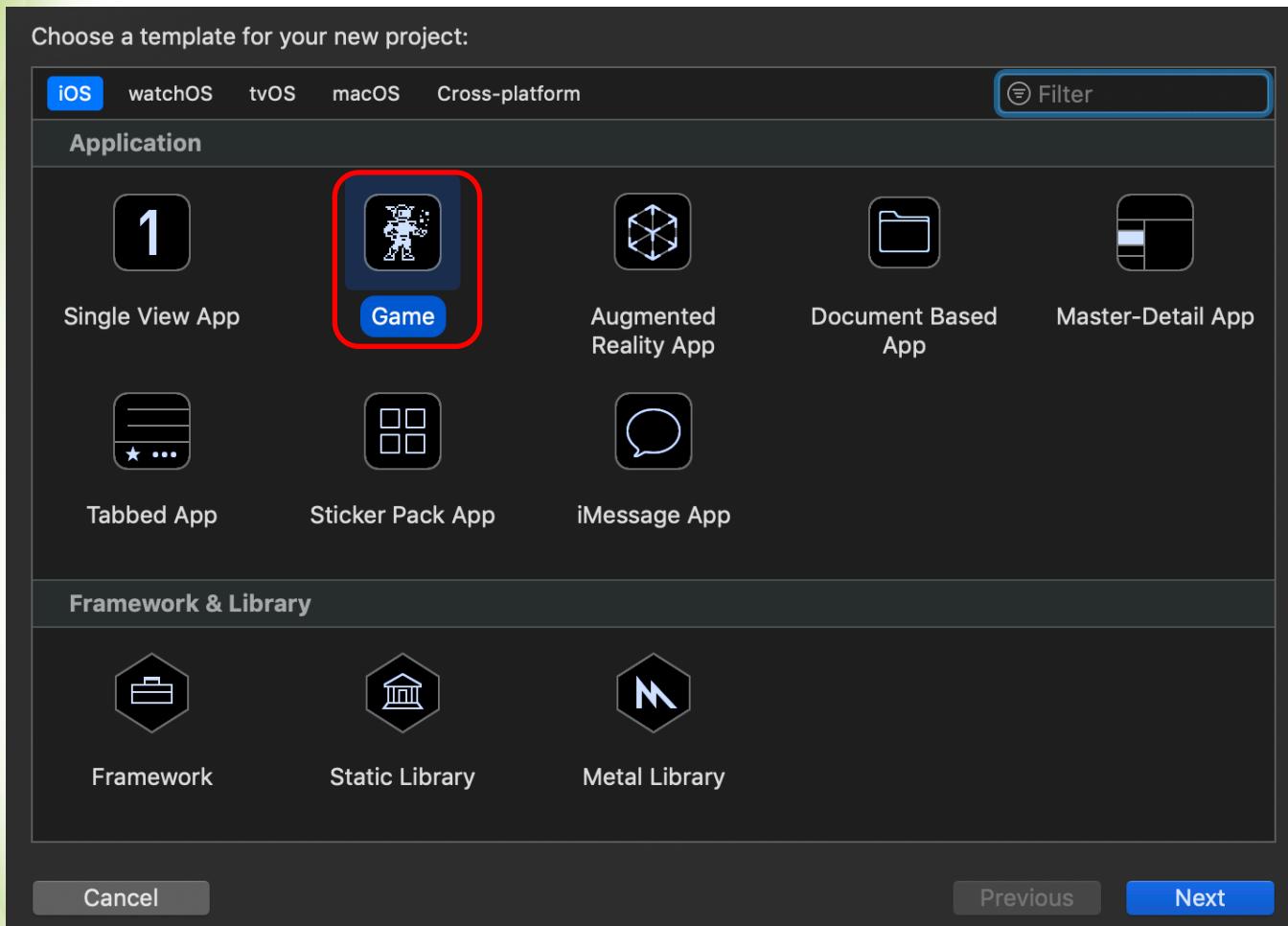


Show this window when Xcode launches

Open another project...

Hello Sprite Kit!

- Create Sprite Kit Game



- Named project

Choose options for your new project:

Product Name: Firstsprite

Team: Add account...

Organization Name: hpc

Organization Identifier: hpc

Bundle Identifier: hpc.Firstsprite

Language: Swift

Game Technology: SpriteKit

Integrate GameplayKit

Include Unit Tests

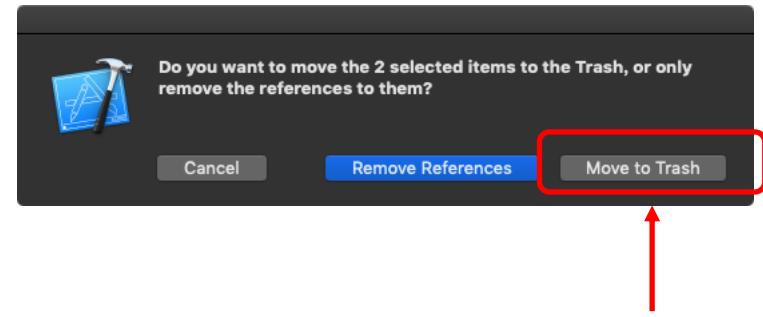
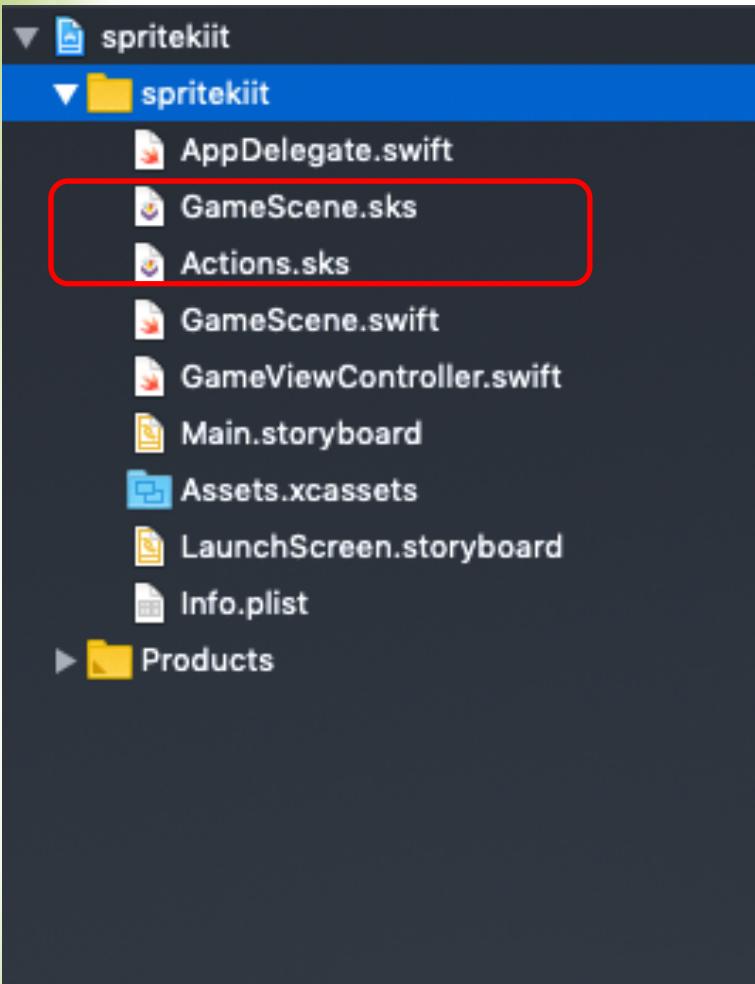
Include UI Tests

Cancel Previous Next

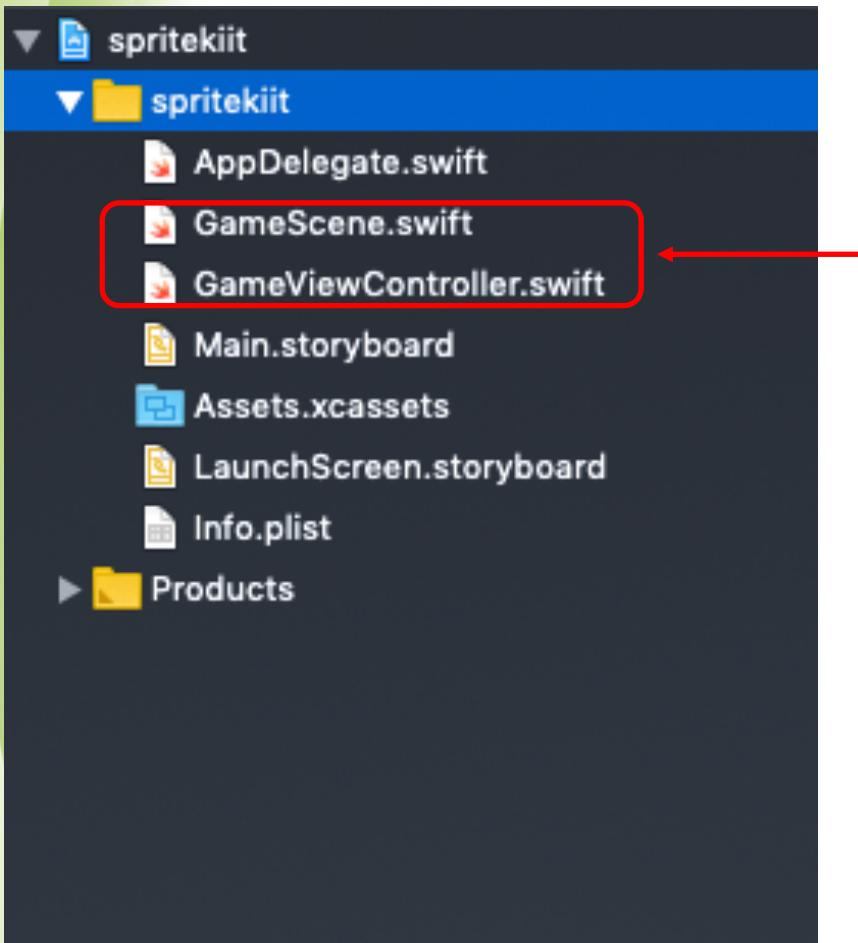


Run it and dragging your mouse to make some rectangles





Make sure you permanently
move the files to the trash



Delete everything inside
Not the entire file !

GameViewController.swift

```
1 import SpriteKit  
2  
3  
4 class GameViewController: UIViewController {  
5     override func viewDidLoad() {  
6         let scene = GameScene(size: view.frame.size)  
7         let skView = view as! SKView  
8         skView.presentScene(scene)  
9     }  
10 }  
11 }
```

The func Runs after the viewcontroller is loaded into memory

force downcast the controller's view to a SpriteKit view

GameScene.swift

Change to your style

#selector is from objective-c syntax

```
import SpriteKit

class GameScene: SKScene {
    let label = SKLabelNode(text: "Hello World !")

    override func didMove(to view: SKView) {
        label.position = CGPoint(x: view.frame.width / 2, y: view.frame.height / 2)
        label.fontSize = 45
        label.fontColor = SKColor.red
        label.fontName = "Avenir"
        label.speed = 5

        let recognizer = UITapGestureRecognizer(target: self, action: #selector(tap))
        view.addGestureRecognizer(recognizer)
        addChild(label) //add the label as a child node to the scene's node tree.
    }
}
```

Create a tap gesture recognizer
and add it to the scene's view.

Select which
method do you
want to call

GameScene.swift

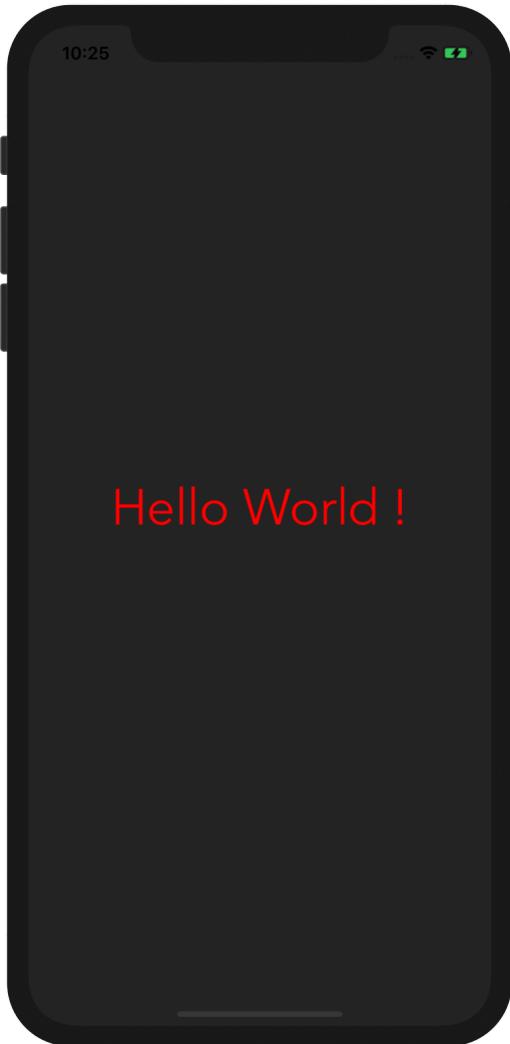
Not in func didMove

convertPoint(fromView:) Converts a point from the coordinate system of a given view to that of the receiver.

```
22 @objc func tap(recognizer: UIGestureRecognizer) {  
23     let viewLocation = recognizer.location(in: view)  
24     let sceneLocation = convertPoint(fromView: viewLocation)  
25     let moveToAction = SKAction.move(to: sceneLocation, duration: 1)  
26     label.run(moveToAction)  
27 }
```

This is the expected duration of an action's animation.

Run it



```

1 import SpriteKit
2
3 class GameScene: SKScene {
4     let label = SKLabelNode(text: "Hello World !")
5     var txtchange :Bool = false
6
7     override func didMove(to view: SKView) {
8         label.position = CGPoint(x: view.frame.width / 2, y:
9             view.frame.height / 2)
10        label.fontSize = 45
11        label.fontColor = SKColor.red
12        label.fontName = "Avenir"
13        label.speed = 5
14
15        let recognizer = UITapGestureRecognizer(target: self, action:
16            #selector(tap))
17        let dtaprecognizer = UITapGestureRecognizer(target: self,
18            action: #selector(doubletap))
19        dtaprecognizer.numberOfTapsRequired = 2
20        view.addGestureRecognizer(recognizer)
21        view.addGestureRecognizer(dtaprecognizer)
22        addChild(label) //add the label as a child node to the scene's
23                      //node tree.
24    }
25
26    @objc func doubletap(recognizer: UIGestureRecognizer) {
27        if txtchange == false {
28            label.text = "I ❤️ SpriteKit"
29            txtchange = true
30        }
31        else {
32            label.text = "Hello World !"
33            txtchange = false
34        }
35    }
36}
37

```

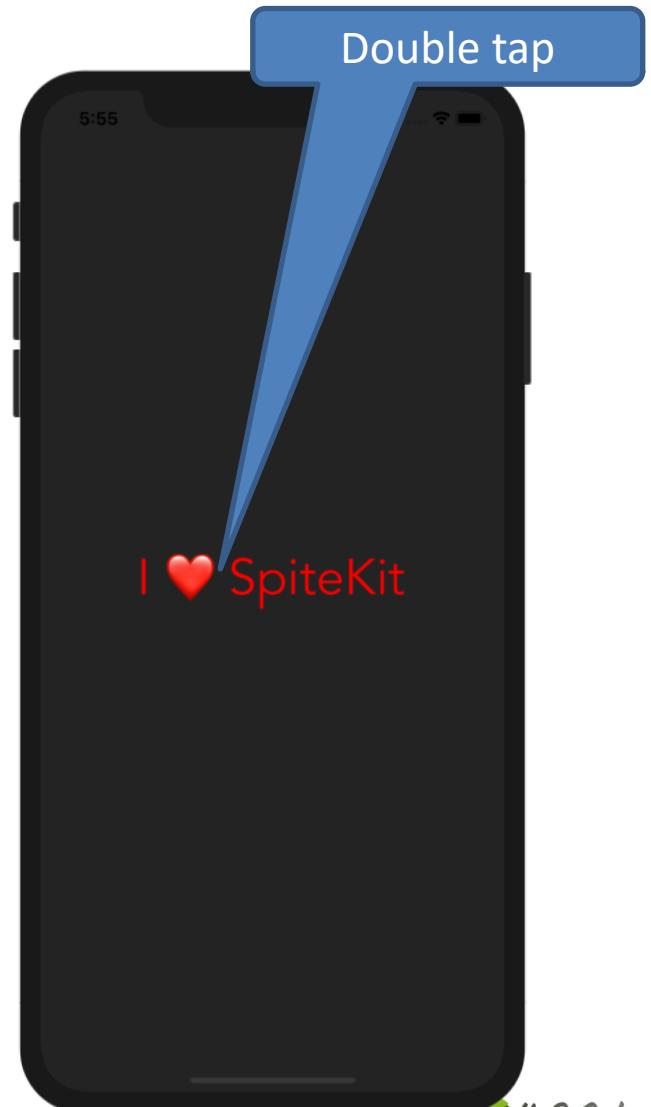
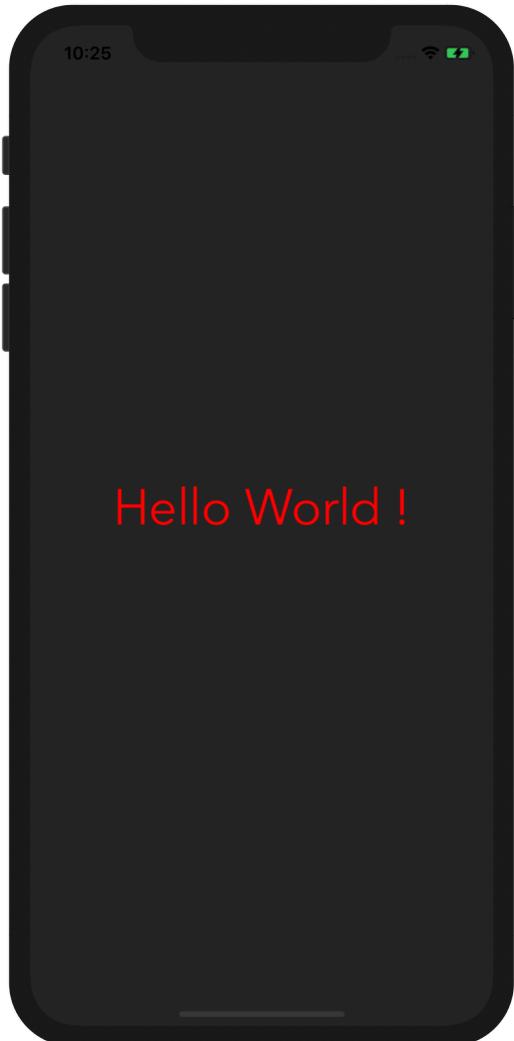
Add the double tap
recognizer

```

28    @objc func doubletap(recognizer: UIGestureRecognizer) {
29        if txtchange == false {
30            label.text = "I ❤️ SpriteKit"
31            txtchange = true
32        }
33        else {
34            label.text = "Hello World !"
35            txtchange = false
36        }
37    }

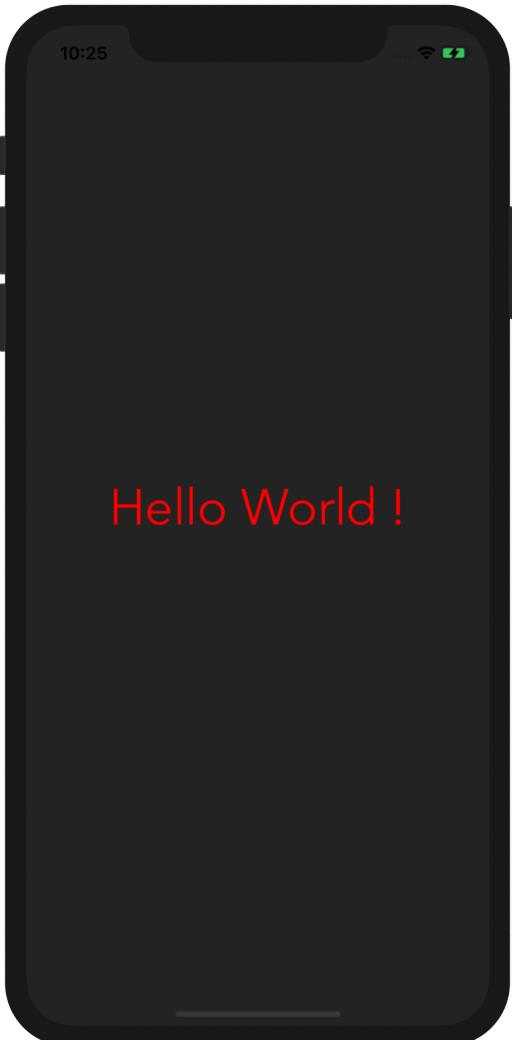
```

Run it

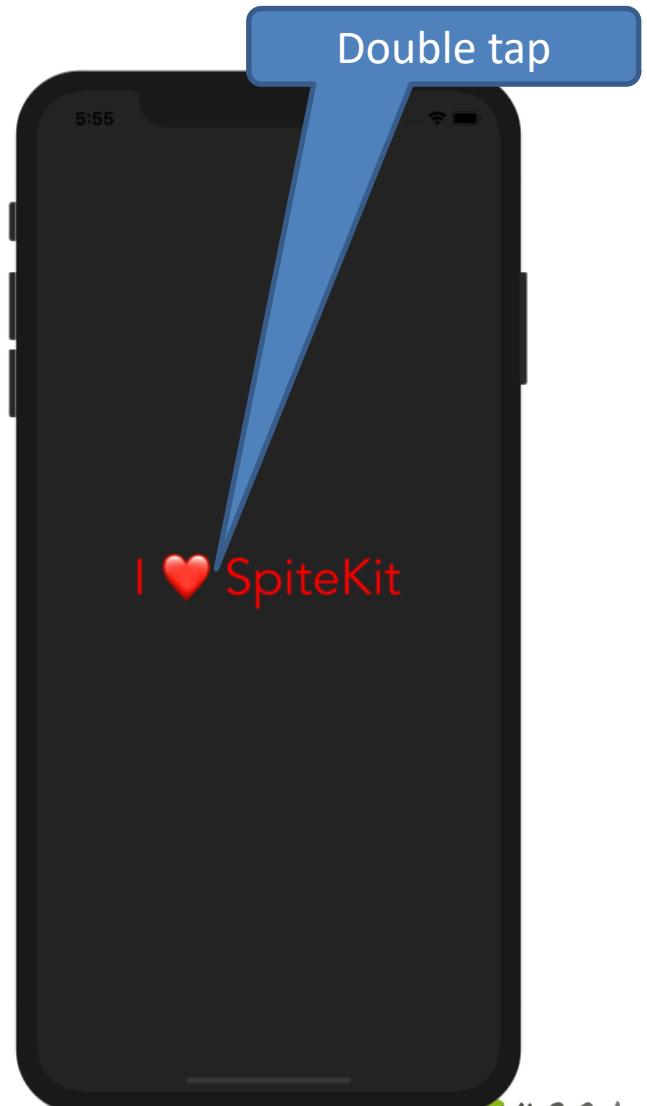


Run it

So, we can add
more gestures you
want...



iPhone 11 — 13.3



iPhone 11 — 13.3

Create A Game Using SpriteKit



iPhone 11 Pro Max — 13.3



iPhone 11 Pro Max — 13.3



iPhone 11 Pro Max — 13.3



Welcome to Xcode

Version 11.3.1 (11C504)



Get started with a playground

Explore new ideas quickly and easily.



Create a new Xcode project

Create an app for iPhone, iPad, Mac, Apple Watch, or Apple TV.



Clone an existing project

Start working on something from a Git repository.



Show this window when Xcode launches



Firstsprite
~/Desktop



cpractice
~/Desktop



MyPlayground
~/Desktop



hellosprite
~/Desktop/2016 iOS/pr6 4h



hw1
~/Desktop



2-1
~/Desktop



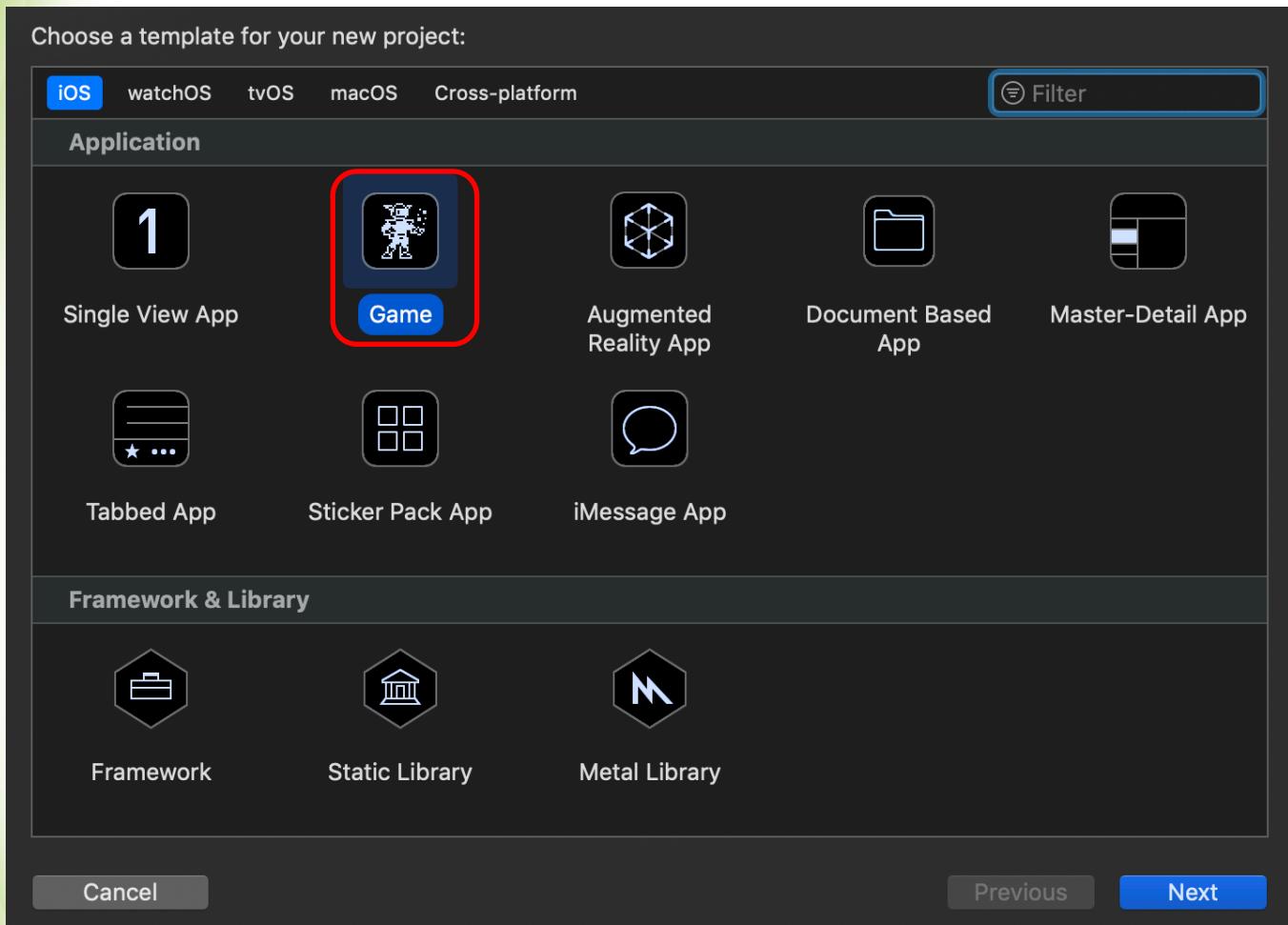
Calculator
~/Desktop



Show this window when Xcode launches

Open another project...

- Create Sprite Kit Game



Choose options for your new project:

Product Name: Myspritekit

Team: Add account...

Organization Name: hpc

Organization Identifier: hpc

Bundle Identifier: hpc.Myspritekit

Language: Swift

Game Technology: SpriteKit

Integrate GameplayKit

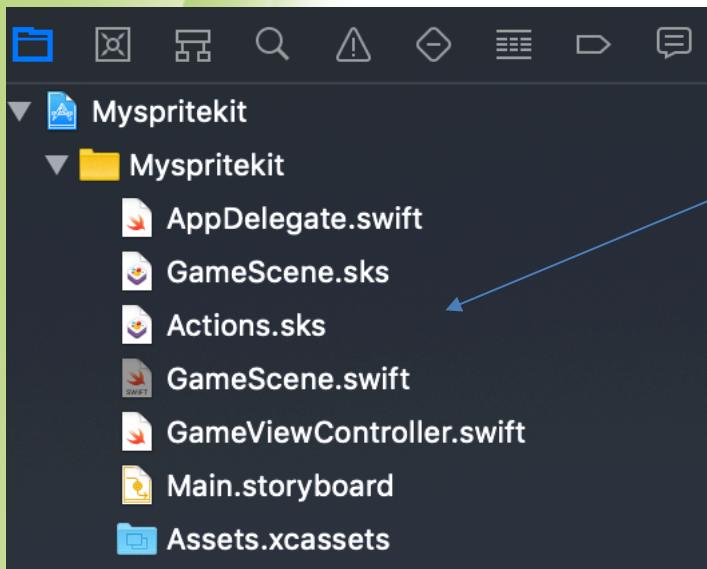
Include Unit Tests

Include UI Tests

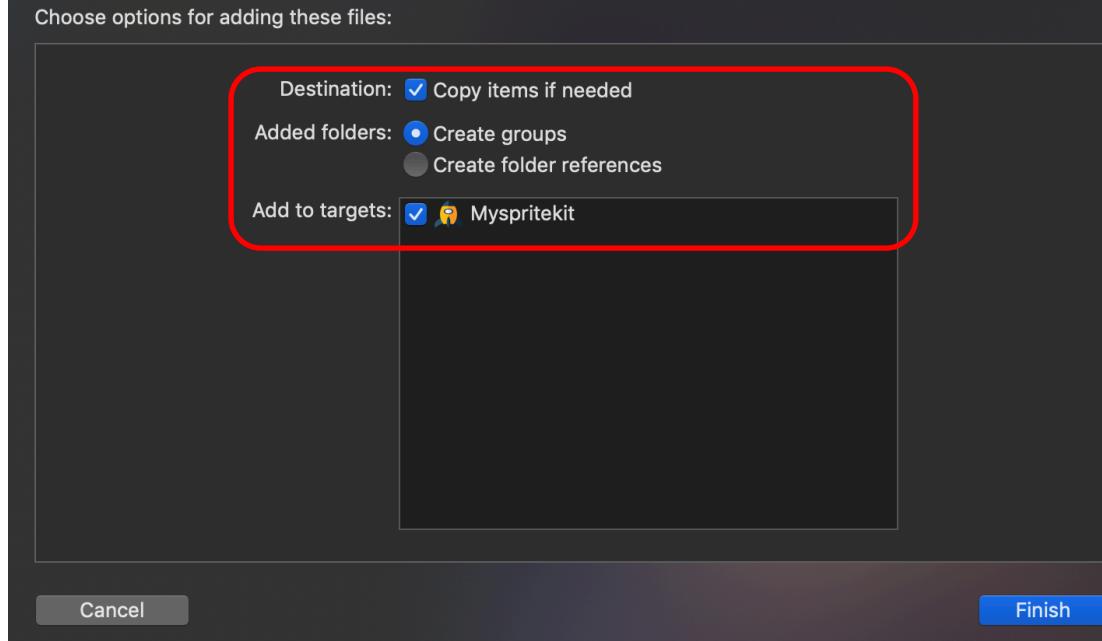
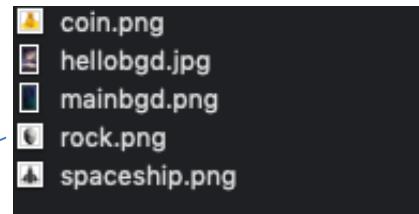
Cancel

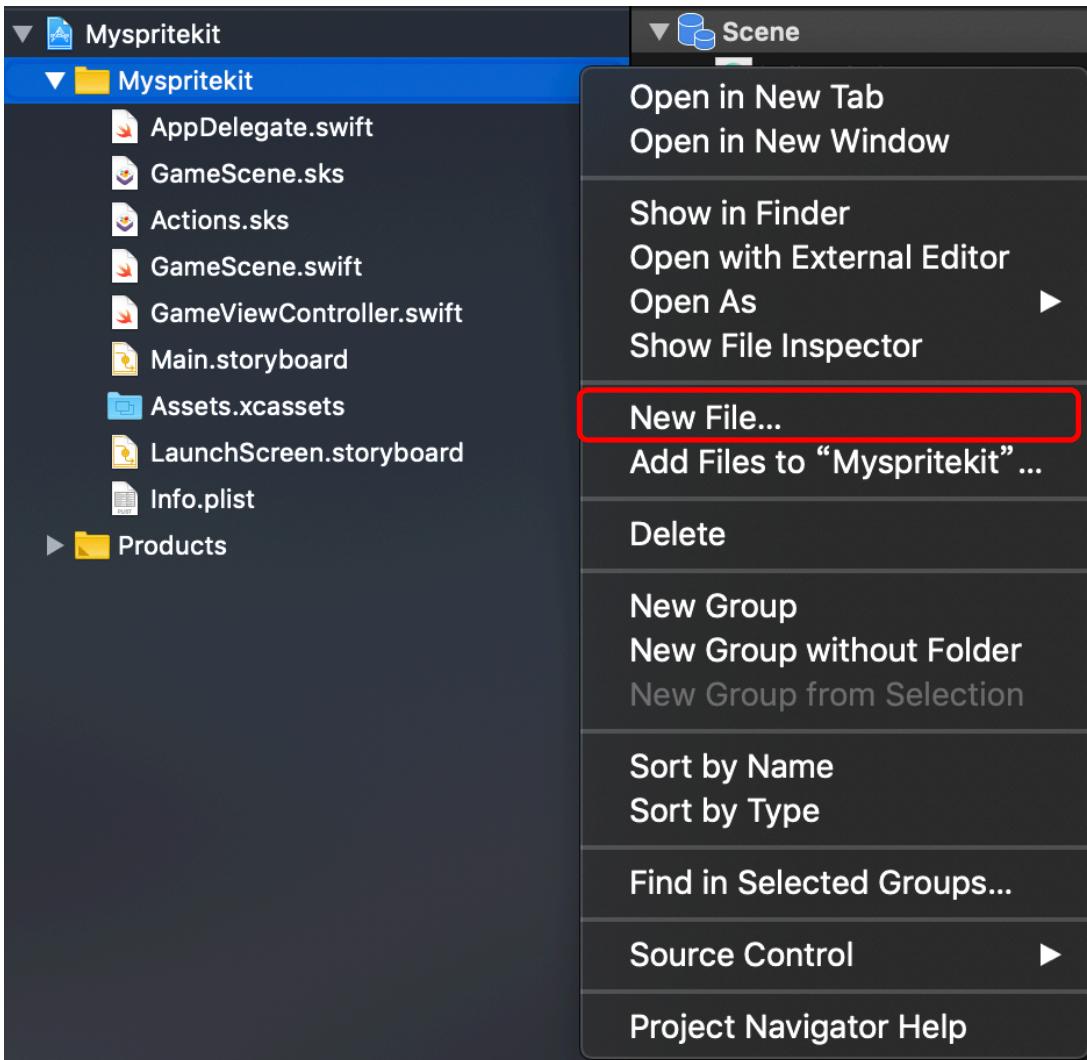
Previous

Next



insert





Choose a template for your new file:

iOS watchOS tvOS macOS Filter

Source

 Cocoa Touch Class	 UI Test Case Class	 Unit Test Case Class	 Swift File	 Objective-C File
 Header File	 IIG File	 C File	 C++ File	 Metal File
User Interface				
 Storyboard	 SwiftUI View	 View	 Empty	 Launch Screen

Cancel Previous Next

Choose options for your new file:

Class: HelloScene

Subclass of: SKScene

Also create XIB file

Language: Swift

Cancel

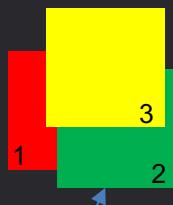
Previous

Next

GameViewController.swift

將UIView 強制轉為SKView

```
1 import UIKit
2 import SpriteKit
3
4 class GameViewController: UIViewController {
5
6     override func viewDidLoad() {
7         super.viewDidLoad()
8
9         if let view = self.view as! SKView? {
10             let scene = HelloScene(size: view.bounds.size)
11             scene.scaleMode = .aspectFill
12             view.presentScene(scene)
13             view.ignoresSiblingOrder = true
14             view.showsFPS = true
15             view.showsNodeCount = true
16         }
17     }
18 }
```



True: You need to set their zPosition .
False: Means that when multiple nodes share the same zPosition

HelloScene.swift

```
1 import UIKit
2 import SpriteKit
3
4 class HelloScene: SKScene {
5     override func didMove(to view: SKView) {
6         createScene()
7     }
8     func createScene(){
9         let bgd = SKSpriteNode(imageNamed: "hellobgd.jpg")
10        bgd.size.width = self.size.width
11        bgd.size.height = self.size.height
12        bgd.position = CGPoint(x:self.frame.midX, y:self.frame.midY)
13        bgd.zPosition = -1
14
15
16        let hellolabel = SKLabelNode(text: "Space 🚀 Adventure ")
17        hellolabel.name = "label"
18        hellolabel.position = CGPoint(x: self.frame.midX, y: self.frame.midY)
19        hellolabel.fontName = "Avenir-Oblique"
20        hellolabel.fontSize = 28
21
22        self.addChild(bgd)
23        self.addChild(hellolabel)
24
25 }
```

Add child node to
root(HelloScene)

HelloScene.swift

Override the function in class

When one or more touches occurred

```
27  override func touchesBegan(_ touches: Set<UITouch>, with event: UIEvent?) {  
28      let labelNode = self.childNode(withName: "label")  
29      let moveup = SKAction.moveBy(x: 0, y: 200, duration: 1)  
30      let zoomin = SKAction.scale(to: 3.0, duration: 1)  
31      let pause = SKAction.wait(forDuration: 0.5)  
32      let zoomout = SKAction.scale(by: 0.5, duration: 0.25)  
33      let fadeaway = SKAction.fadeOut(withDuration: 0.25)  
34      let remove = SKAction.removeFromParent()  
35      let movesequece =  
36          SKAction.sequence([moveup, zoomin, pause, zoomout, pause, fadeaway, remove])  
37      labelNode?.run(movesequece)  
38      /* You can also group these actions , try it */  
39      /*  
40          let rotate = SKAction.rotate(byAngle: CGFloat.pi * 2, duration:3)  
41          let groupAction = SKAction.group([zoomout,rotate,moveup])  
42          labelNode?.run(groupAction)  
43      */
```

Run it and see what happens

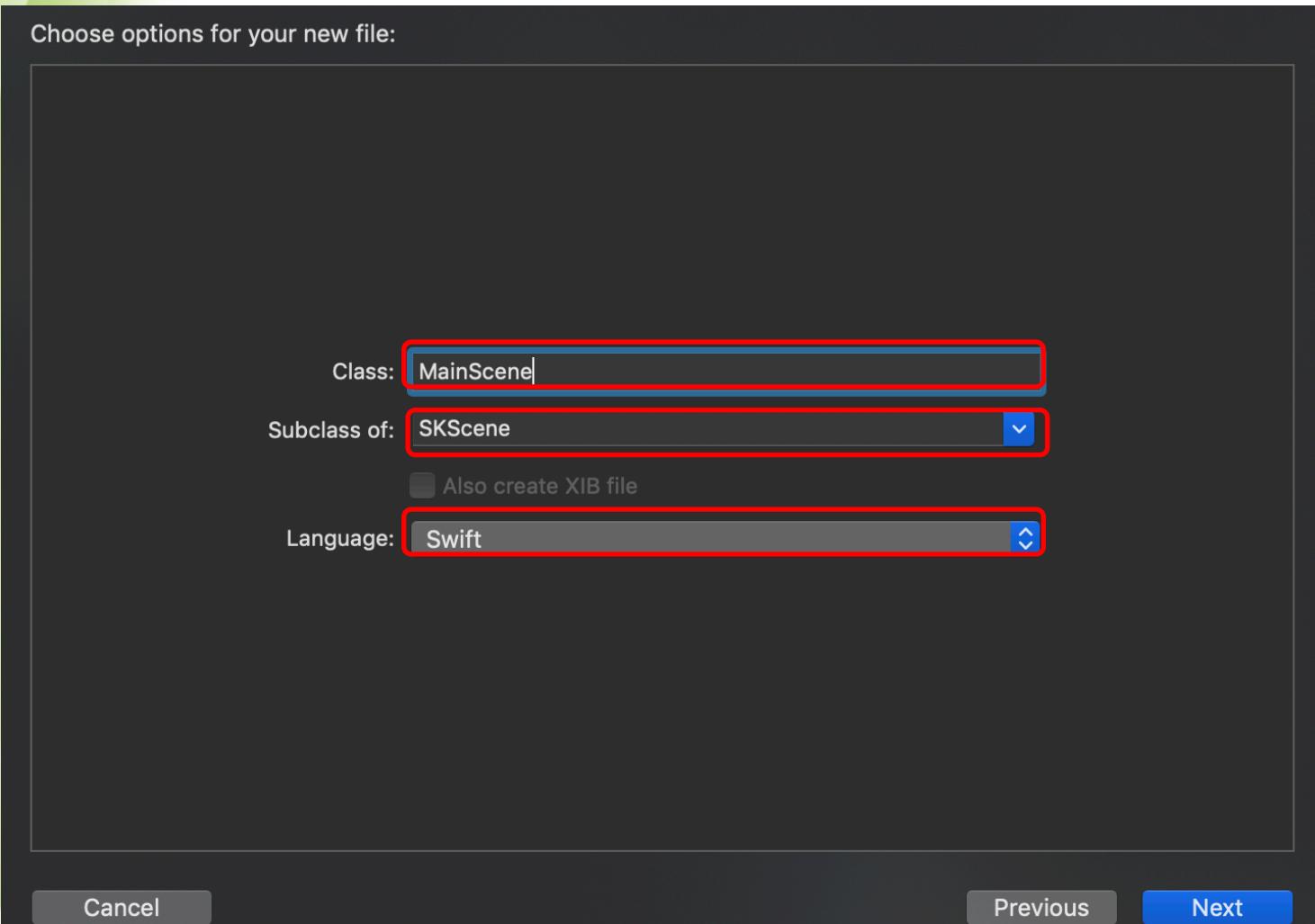


iPhone 11 Pro Max — 13.3



iPhone 11 Pro Max — 13.3

We need more Scene,
Create new SKScene



HelloScene.swift

In func touchesBegan

```
19     override func touchesBegan(_ touches: Set<UITouch>, with event: UIEvent?) {  
20         let labelNode = self.childNode(withName: "label")  
21         let moveup = SKAction.moveBy(x: 0, y: 200, duration: 2)  
22         let zoomin = SKAction.scale(to: 3.0, duration: 2)  
23         let pause = SKAction.wait(forDuration: 0.5)  
24         let zoomout = SKAction.scale(by: 0.5, duration: 0.25)  
25         let fadeaway = SKAction.fadeOut(withDuration: 0.25)  
26         let remove = SKAction.removeFromParent()  
27         let movesequece = SKAction.sequence([moveup,zoomin,pause,zoomout,pause,fadeaway,remove])  
28         labelNode?.run(movesequece) delete  
29         /* You can also group the actions , try it */  
30         /*  
31         let rotate = SKAction.rotate(byAngle: CGFloat.pi * 2, duration:3)  
32         let groupAction = SKAction.group([zoomout,rotate,moveup])  
33         labelNode?.run(groupAction)  
34         */  
35     }
```

Insert the transition scene code

```
labelNode?.run(movesequece, completion: {  
    let mainScene = MainScene(size: self.size)  
    let doors = SKTransition.doorOpenVertical(withDuration: 0.5)  
    self.view?.presentScene(mainScene, transition: doors)  
})
```

When movesequence is complete

used to perform an animated transition to a new scene.

MainScene.swift

```
8     override func didMove(to view: SKView) {  
9         createScene()  
10    }
```

Need to define

Line 19

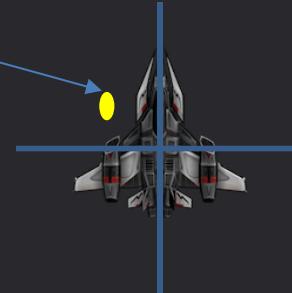
```
12    func createScene() {  
13        let mainbgd = SKSpriteNode(imageNamed: "mainbgd.png")  
14        mainbgd.size.width = self.size.width  
15        mainbgd.size.height = self.size.height  
16        mainbgd.position = CGPoint(x:self.frame.midX ,y:self.frame.midY)  
17        mainbgd.zPosition = -1  
18  
19        let spaceship = newSpaceship()  
20        spaceship.position = CGPoint(x: self.frame.midX, y: self.frame.midY-150)  
21  
22        self.addChild(mainbgd)  
23        self.addChild(spaceship)
```

MainScene.swift

```
53 func newSpaceship() ->SKSpriteNode {  
54     let ship = SKSpriteNode(imageNamed: "spaceship.png")  
55     ship.size = CGSize(width: 75, height: 75)  
56     ship.name = "ships"  
57  
58     let leftlight = newLight()  
59     leftlight.position = CGPoint(x:-20, y: 6)  
60     ship.addChild(leftlight)  
61  
62     let rightlight = newLight()  
63     rightlight.position = CGPoint(x:20, y: 6)  
64     ship.addChild(rightlight)  
65  
66     ship.physicsBody = SKPhysicsBody(circleOfRadius: ship.size.width / 2)  
67     ship.physicsBody?.usesPreciseCollisionDetection = true  
68     ship.physicsBody?.isDynamic = false  
69  
70     return ship  
71 }
```

Line 67

True : The simulation performs a more precise and more expensive calculation to detect these collisions. This property should be set to true on small, fast moving bodies.

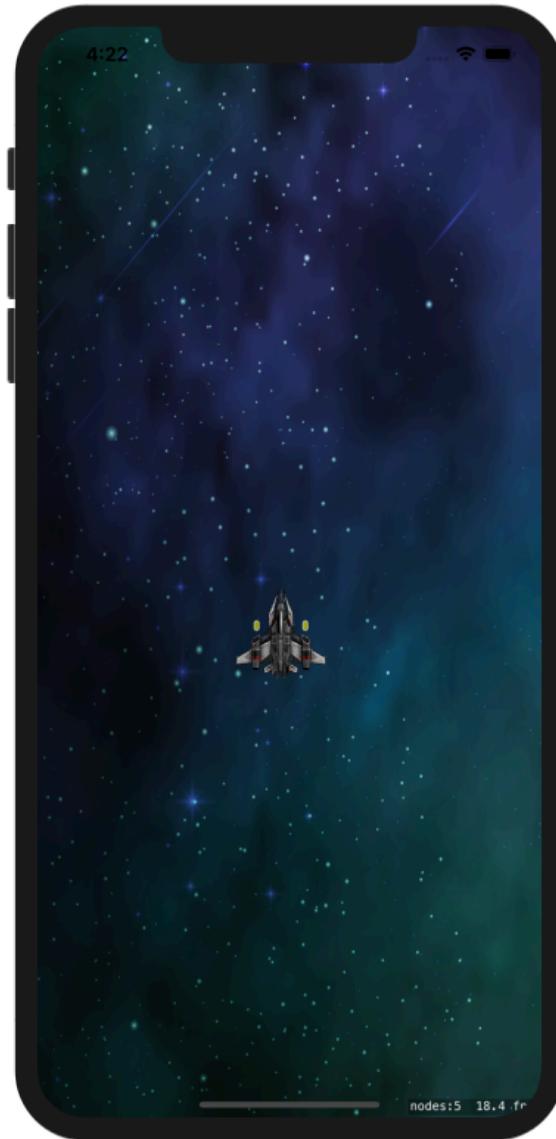


A Boolean value that indicates whether the physics body is moved by the physics simulation.

creates a path of an rectangle.
CGRect : the rectangle to add

```
76 func newLight() -> SKShapeNode {  
77     let light = SKShapeNode()  
78     light.path = CGPath(rect: CGRect(x:-2,y:-4,width:4,height:8), transform: nil)  
79     light.strokeColor = SKColor.white  
80     light.fillColor = SKColor.yellow →   
81  
82     let blink = SKAction.sequence([  
83         SKAction.fadeOut(withDuration: 0.25),  
84         SKAction.fadeIn(withDuration: 0.25)  
85     ])  
86     let blinkForever = SKAction.repeatForever(blink)  
87     light.run(blinkForever)  
88     return light  
89 }
```

Run it
We have a cool spaceship with
cool lights now !

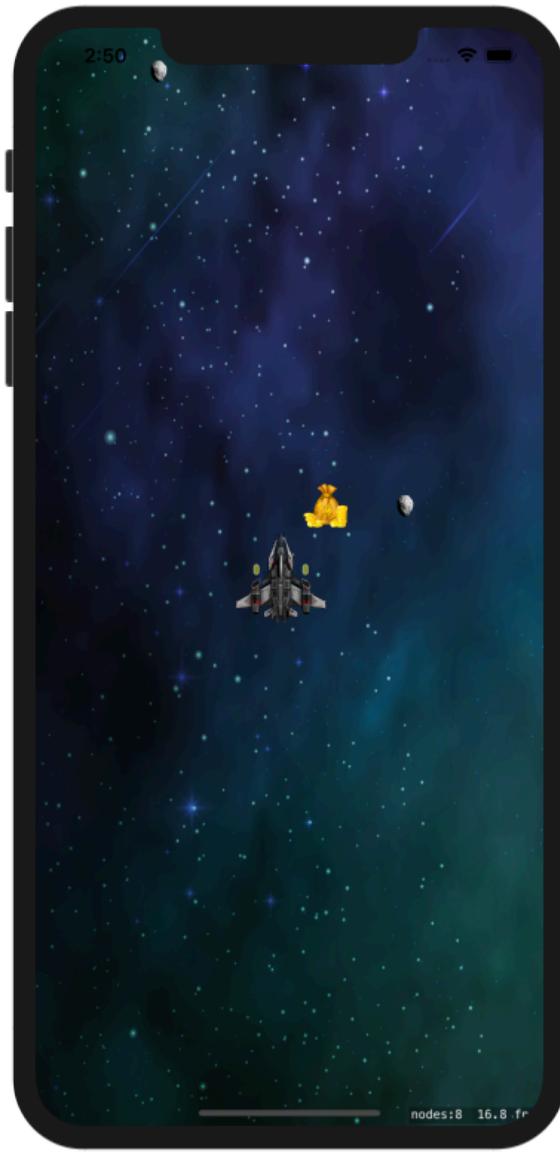


iPhone 11 Pro Max — 13.3

Run it
We have a cool spaceship with
cool lights now !

Let's add some nodes

4/1



Do nothing for 3s and
then remove from parent

MainScene.swift

```
62 @objc func newRock()  {
63     let rock = SKSpriteNode(imageNamed: "rock.png")
64     rock.size = CGSize(width: 40 , height: 40)
65     let remove = SKAction.sequence([SKAction.wait(forDuration: 3),SKAction.removeFromParent()])
66     let w = self.size.width
67     let h = self.size.height
68     let x = CGFloat(Float(arc4random()).truncatingRemainder(dividingBy: w))
69     rock.position = CGPoint(x: x , y: h)
70     rock.name = "rocks"
71     rock.physicsBody = SKPhysicsBody(circleOfRadius: 4)
72     rock.physicsBody?.usesPreciseCollisionDetection = true
73     rock.run(remove)
74     self.addChild(rock)
75 }
```

MainScene.swift

```
77 @objc func newCoin()  {
78     let coin = SKSpriteNode(imageNamed: "coin.png")
79     coin.size = CGSize(width: 30, height: 30)
80     let remove = SKAction.sequence([SKAction.wait(forDuration: 3),SKAction.removeFromParent()])
81     let w = self.size.width
82     let h = self.size.height
83     let x = CGFloat(Float(arc4random()).truncatingRemainder(dividingBy: w))
84     coin.position = CGPoint(x: x , y: h)
85     coin.name = "coins"
86     coin.physicsBody = SKPhysicsBody(circleOfRadius: 4)
87     coin.physicsBody?.usesPreciseCollisionDetection = true
88     coin.run(remove)
89     self.addChild(coin)
90 }
```

MainScene.swift

In function createScene()

```
12 func createScene() {
13     let mainbgd = SKSpriteNode(imageNamed: "mainbgd.png")
14     mainbgd.size.width = self.size.width
15     mainbgd.size.height = self.size.height
16     mainbgd.position = CGPoint(x:frame.size.width / 2, y: frame.size.height / 2)
17     mainbgd.zPosition = -1
18
19     let spaceship = newSpaceship()
20     spaceship.position = CGPoint(x: self.frame.midX, y: self.frame.midY-150)
21
22     self.addChild(mainbgd)
23     self.addChild(spaceship)
24
25     Timer.scheduledTimer(timeInterval: 0.5, target: self, selector: #selector (newRock), userInfo: nil,
26                           repeats: true)
27     Timer.scheduledTimer(timeInterval: 2, target: self, selector: #selector(newCoin), userInfo: nil,
28                           repeats: true)
29 }
```

Insert Timer

MainScene.swift

```
1 import UIKit
2 import SpriteKit
3
4 class MainScene: SKScene {
5
6     override func didMove(to view: SKView) {
7         createScene()
8         let panrecognizer = UIPanGestureRecognizer(target: self, action: #selector(handpan))
9         view.addGestureRecognizer(panrecognizer)
10    }
```

declare a recognizer

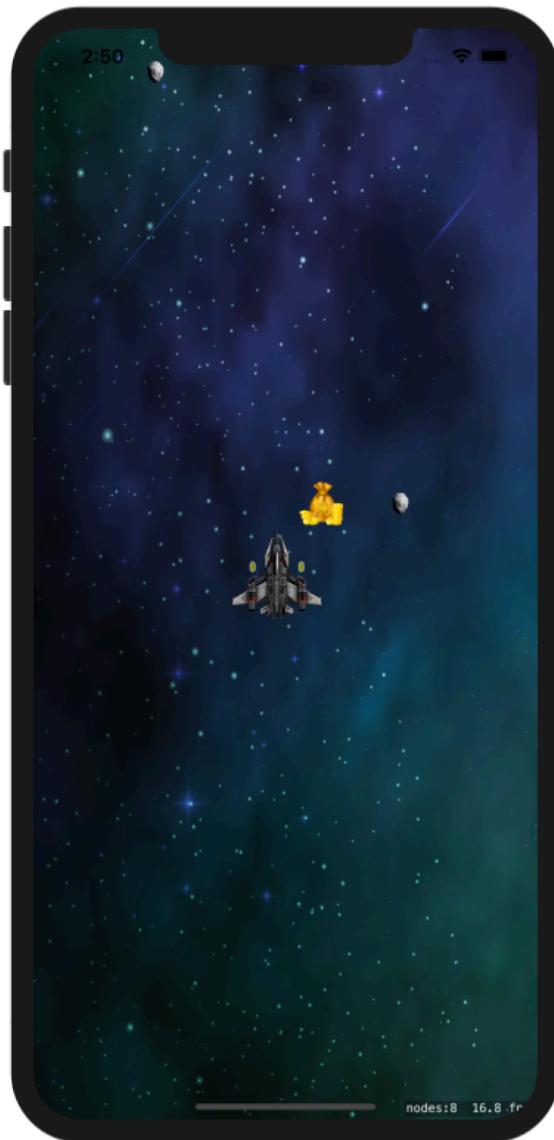
call which function
when happen

Target: MainScene

MainScene.swift

```
123 @objc func handpan(recognizer: UIPanGestureRecognizer) {
124     let viewLocation = recognizer.location(in: view)
125     let sceneLocation = convertPoint(fromView: viewLocation)
126     let moveAction = SKAction.moveTo(x: sceneLocation.x, duration: 0.1)
127     self.addChild(node: "ships")!.run(moveAction)
128 }
```

Run it



iPhone 11 Pro Max — 13.3

An object that implements the SKPhysicsContactDelegate protocol can respond when two physics bodies with overlapping contactTestBitMask values are in contact with each other in a physics world.

```
4 class MainScene: SKScene, SKPhysicsContactDelegate {
```

```
8     override func didMove(to view: SKView) {  
9         self.physicsWorld.contactDelegate = self
```

Func newSpaceship()

```
ship.physicsBody?.categoryBitMask = 0x1 << 1  
ship.physicsBody?.contactTestBitMask = 0x1 << 2
```

Type: UInt32 , Defines the collision categories to which a physics body belongs.

Determines which categories this physics body makes contact with

We don't know which one is spaceship, bodyA ? or bodyB ?



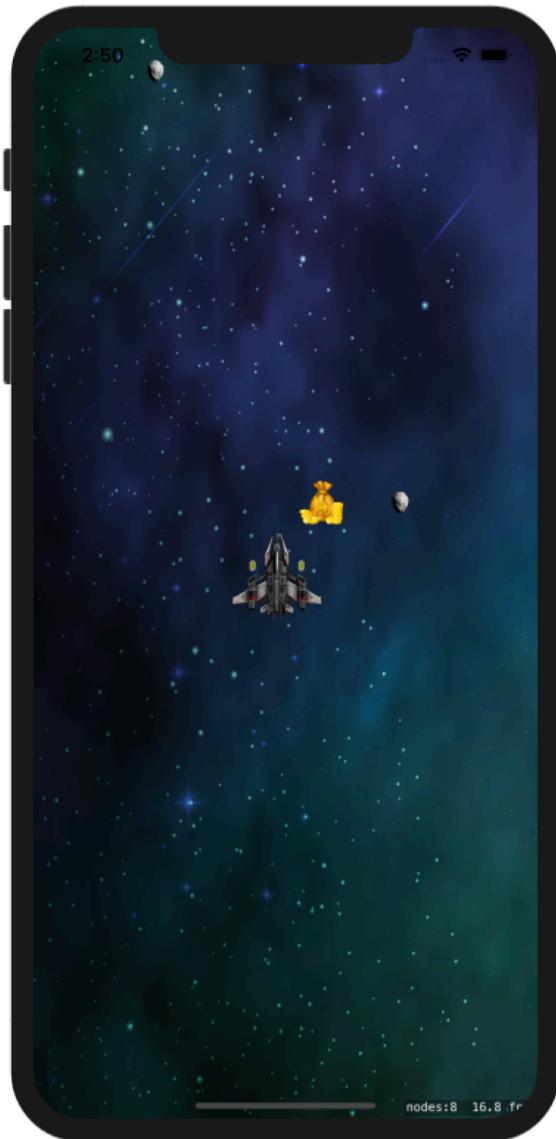
Determine which nodes collided

```
130 func didBegin(_ contact: SKPhysicsContact) {  
131     var firstBody = SKPhysicsBody()  
132     var secondBody = SKPhysicsBody()  
133     if contact.bodyA.node?.name == "ships"{  
134         firstBody = contact.bodyA  
135         secondBody = contact.bodyB  
136     } else {  
137         firstBody = contact.bodyB  
138         secondBody = contact.bodyA  
139     }  
140     if (firstBody.node?.name == "ships" && secondBody.node?.name == "rocks") {  
141         print("You lose !\n")  
142     }  
143     else if (firstBody.node?.name == "ships" && secondBody.node?.name == "coins"){  
144         contact.bodyB.node?.removeFromParent()  
145         print("Get point 100 !\n")  
146     }  
147     /* if (secondBody.node?.name == "rocks"){  
148         print("You lose !\n")  
149     }  
150     else{  
151         contact.bodyB.node?.removeFromParent()  
152         print("Get point 100 !\n")  
153     }  
154 */  
155 }
```

```
Get point 100 !  
You lose !  
Get point 100 !  
Get point 100 !  
Get point 100 !  
Get point 100 !
```

All Output ◊

Filter



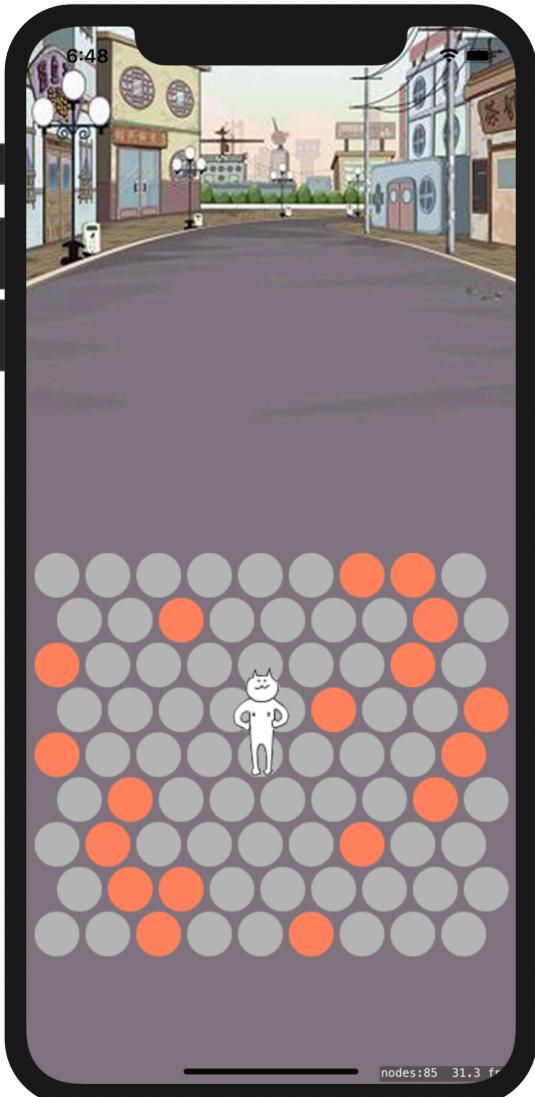
iPhone 11 Pro Max — 13.3

Class 05-2



Create A Game Using SpriteKit

Cat



Create the game project

The screenshot shows the Xcode welcome screen. At the top center is a blue blueprint icon with a hammer and a pencil. Below it is the text "Welcome to Xcode" and "Version 11.3.1 (11C504)". On the left, there are three main options: "Get started with a playground", "Create a new Xcode project", and "Clone an existing project". The "Create a new Xcode project" option is highlighted with a red rectangle. At the bottom left is a checkbox for "Show this window when Xcode launches". On the right side, there is a sidebar listing recent projects:

- MySpritekit
~/Desktop/MySpritekit
- Myspritekit
~/Desktop/xcode
- Angelica Fighti
~/Desktop/Legend-Wings-master
- Firstsprite
~/Desktop/xcode
- Myspritekit02
~/Desktop/xcode
- Cat
...16 iOS/範例程式/CHT/第8章 程式碼/8.7 零路
- hash-table
~/Desktop/xcode
- 4-1-2
~/Desktop/xcode
- MyPlayground
~/Desktop/xcode
- 4-1
~/Desktop/xcode

At the bottom right is a link "Open another project...".

Choose options for your new project:

Product Name: **spritekit_cat**

Team: [Add account...](#)

Organization Name: **hpc**

Organization Identifier: **hpc**

Bundle Identifier: **hpc.spritekit-cat**

Language: **Swift**

Game Technology: **SpriteKit**

Integrate GameplayKit

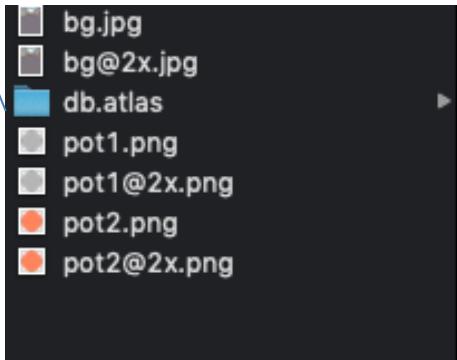
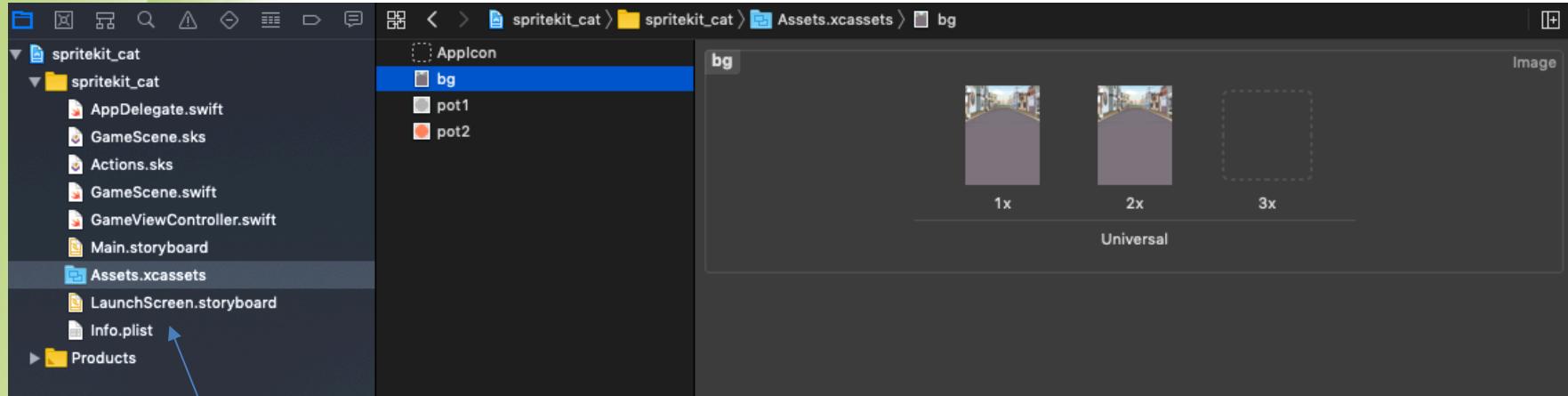
Include Unit Tests

Include UI Tests

[Cancel](#)

[Previous](#)

[Next](#)



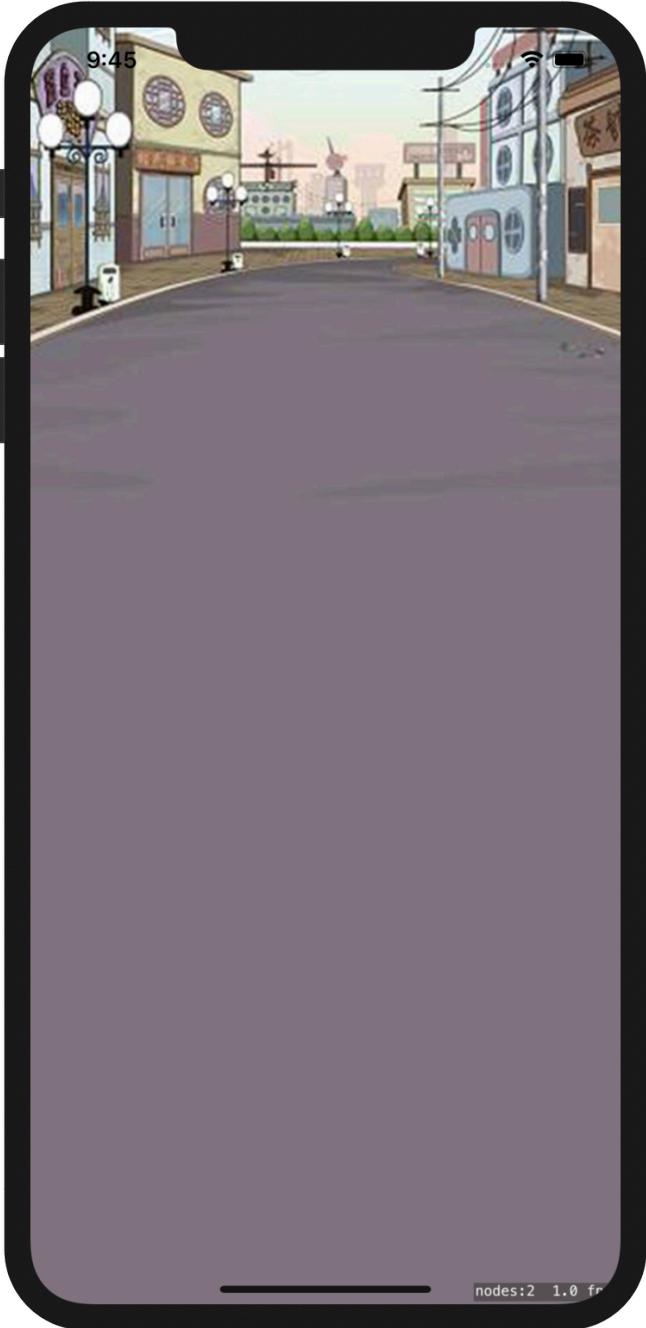
將db.atlas直接拖入專案，
其餘對應的放入Assets

GameViewController.swift

```
1 import UIKit
2 import SpriteKit
3
4
5 class GameViewController: UIViewController {
6
7     override func viewDidLoad() {
8         super.viewDidLoad()
9
10        if let view = self.view as! SKView? {
11            let scene = GameScene(size: view.bounds.size)
12            view.ignoresSiblingOrder = true
13            view.showsFPS = true
14            view.showsNodeCount = true
15            view.presentScene(scene)
16        }
17    }
18 }
```

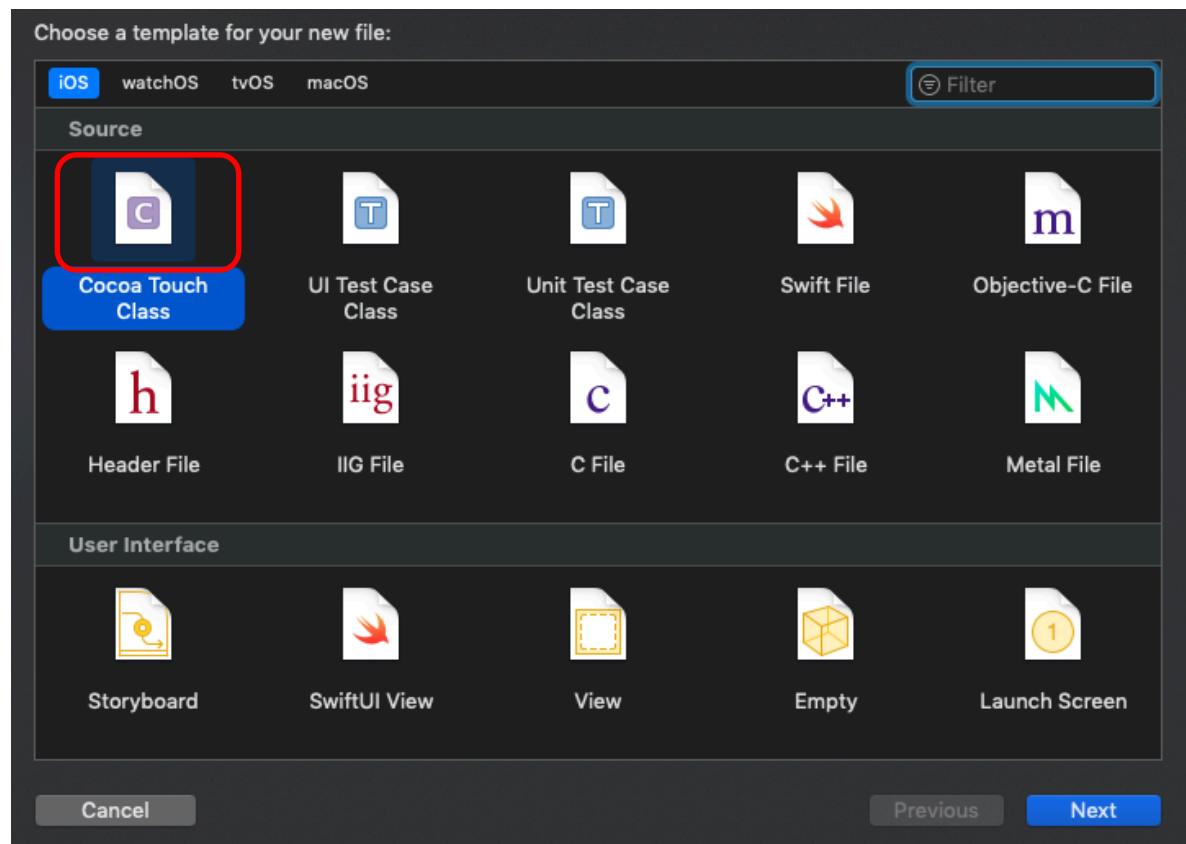
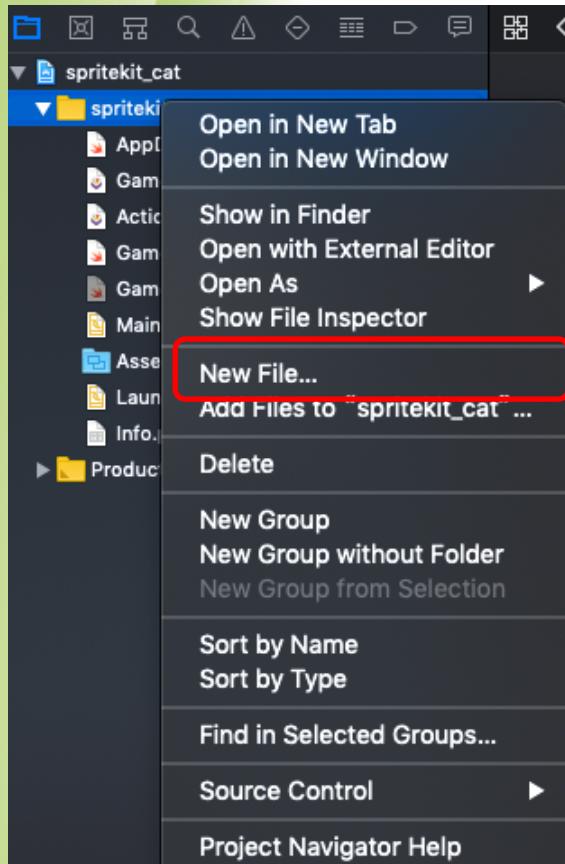
GameScene.swift

```
1 import SpriteKit
2
3 class GameScene: SKScene {
4
5     override func didMove(to view: SKView) {
6         let background = SKSpriteNode(imageNamed: "bg")
7         background.position = CGPoint(x:self.frame.midX,y:self.frame.midY)
8         addChild(background)
9     }
10    override func touchesBegan(_ touches: Set<UITouch>, with event: UIEvent?) {
11
12    }
13 }
```

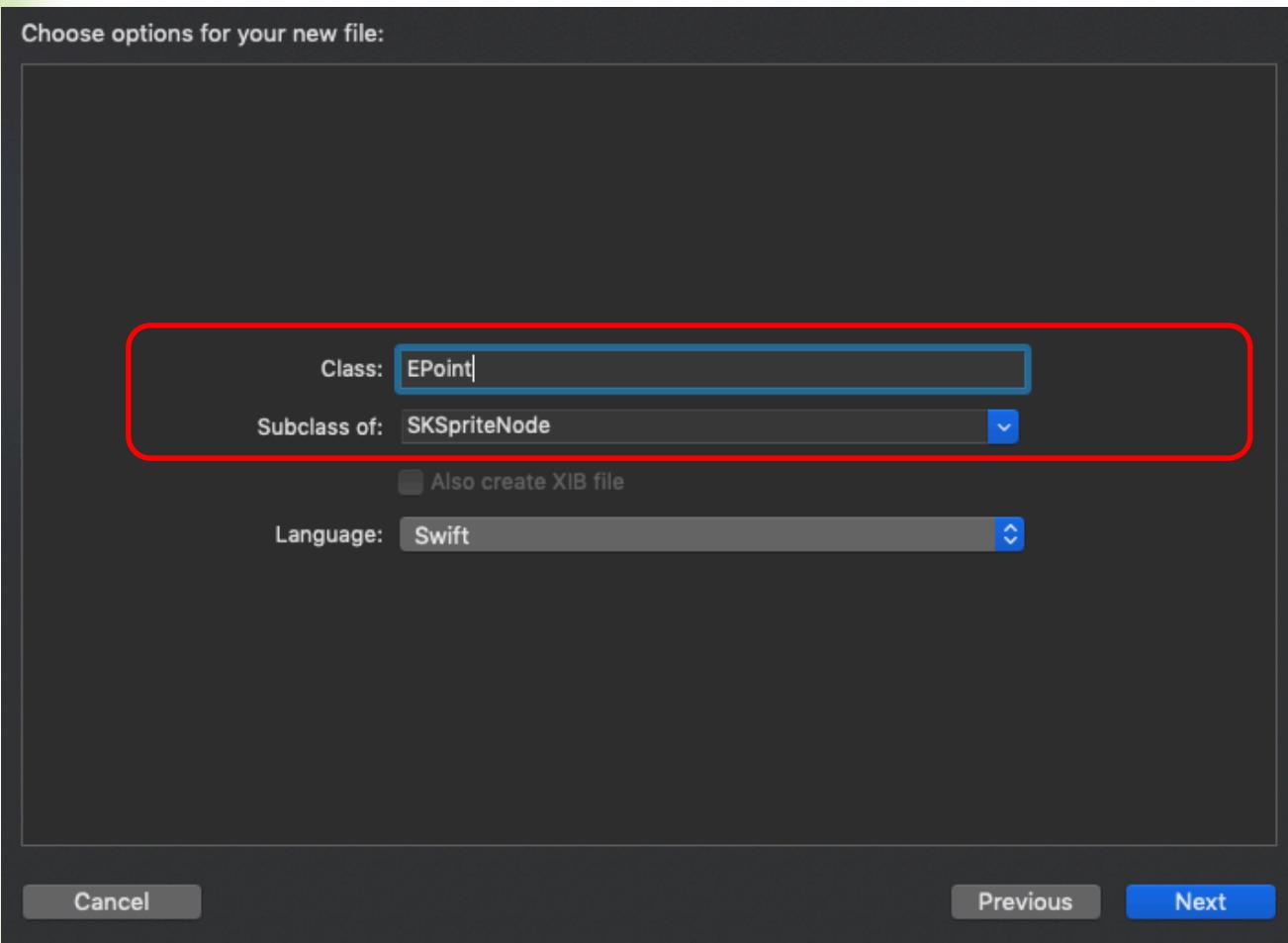


iPhone 11 Pro Max — 13.3

- New File



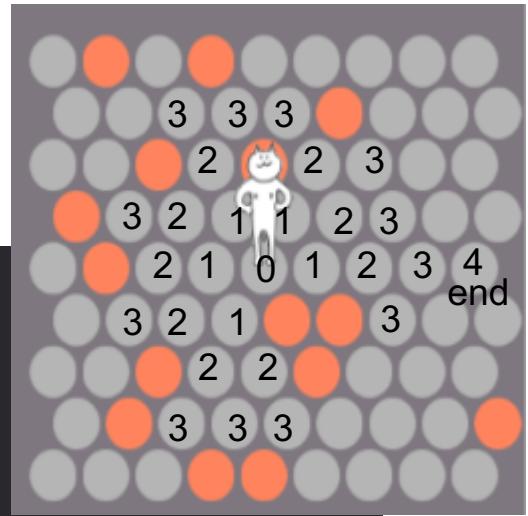
- New File



Class EPoint

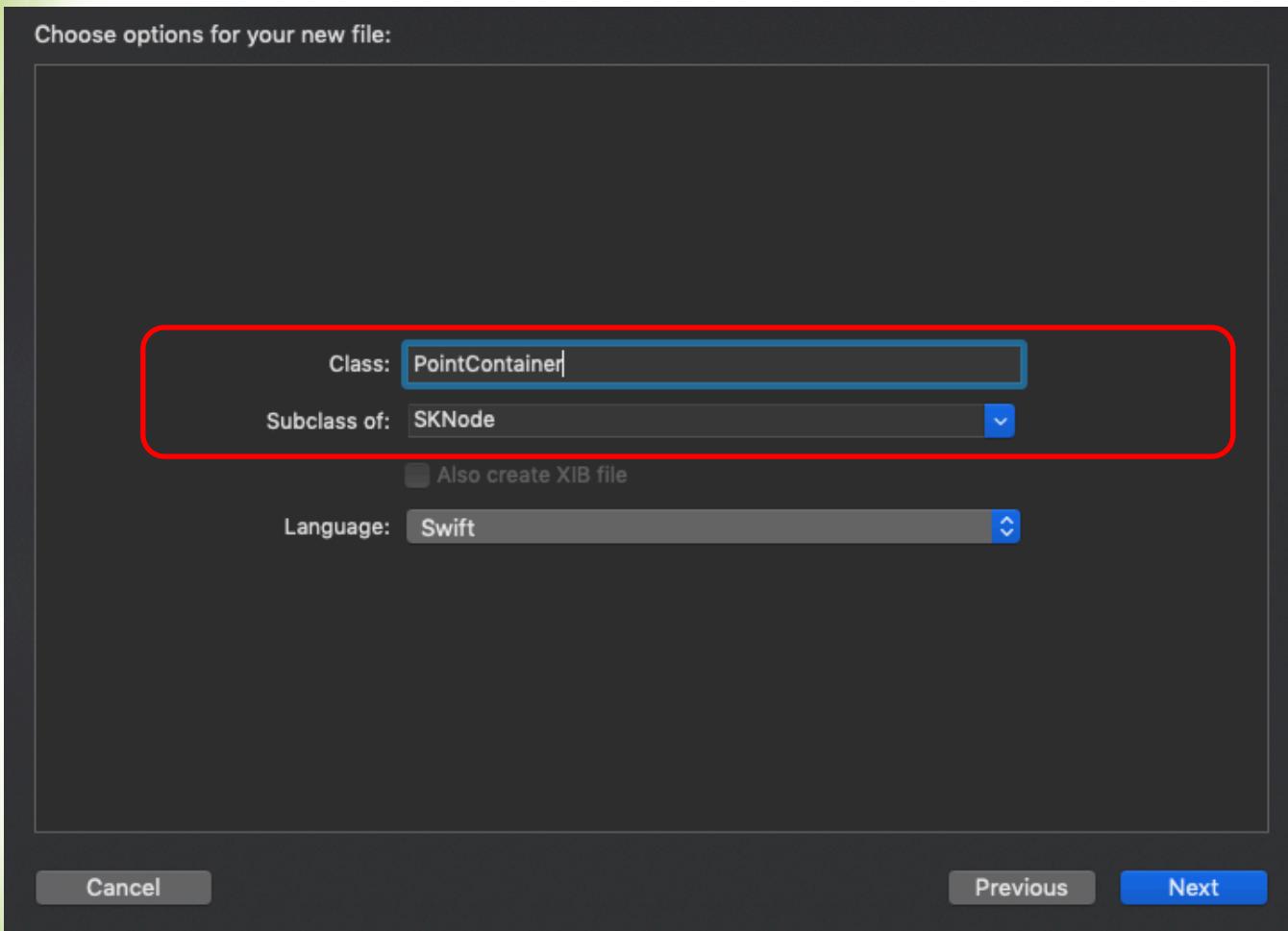
```
1 import SpriteKit  
2  
3 enum pointtype : Int {  
4     case gray = 0  
5     case red = 1  
6 }  
7 class EPoint: SKSpriteNode {  
8     var prePointIndex = -1  
9     var aroundPoint = [Int]()  
10    var step = 99  
11    var index = 0  
12    var type = pointtype.gray  
13    var isEdge = false  
14 }
```

此點是否為邊緣點



用陣列來儲存周圍的點

- New File



SKTexture 是一個 object 用於渲染
SKShapeNode 與 SKSpriteNode。
Texture 通常是由專案裡的圖集創建，
當被載入到記憶體後，它將保存到你
刪除它為止

PointContainer.swift

```
1 import SpriteKit
2
3 class PointContainer: SKNode {
4     let textPoint1 = SKTexture(imageNamed: "pot1")
5     let textPoint2 = SKTexture(imageNamed: "pot2")
6     var arrPoint = [EPoint]()
7     let startIndex = 40
8     var currIndex = 40
9     var isFind = false
10    var stepNum = 0
11    var arrNext = [Int]()
12 }
```

代表下一步可以走到的點
貓要從此陣列選取下一步

代表貓找到了邊緣點，遊戲失敗

此函式來初始化共81個點

```
15     func onInit(){
16         for i in 0...80 {
17             let point = EPoint(texture: textPoint1)
18             let row = Int(i/9)
19             let col = i%9
20             var gap = 0
21             //奇偶行判斷,0為奇數行,1為偶數行
22             if Int(row % 2) == 1 {
23                 gap = 19
24             }else {
25             }
26             //取得紋理寬度,用來計算位置
27             let width = Int(textPoint1.size().width)
28             let x = col * (width+5) - (9*width)/2 + gap
29             let y = row * width - (9*width)/2
30             point.position = CGPointMake(x:CGFloat(x),y:CGFloat(y))
31             //判斷是否為邊緣點
32             if row == 0 || row == 8 || col == 0 || col == 8 {
33                 point.isEdge = true
34             }
35             point.index = i
36             point.zPosition = 10
37             addChild(point)
38             arrPoint.append(point)
39         }
40     }
```

將此點加入到PointContainer
與點陣列中

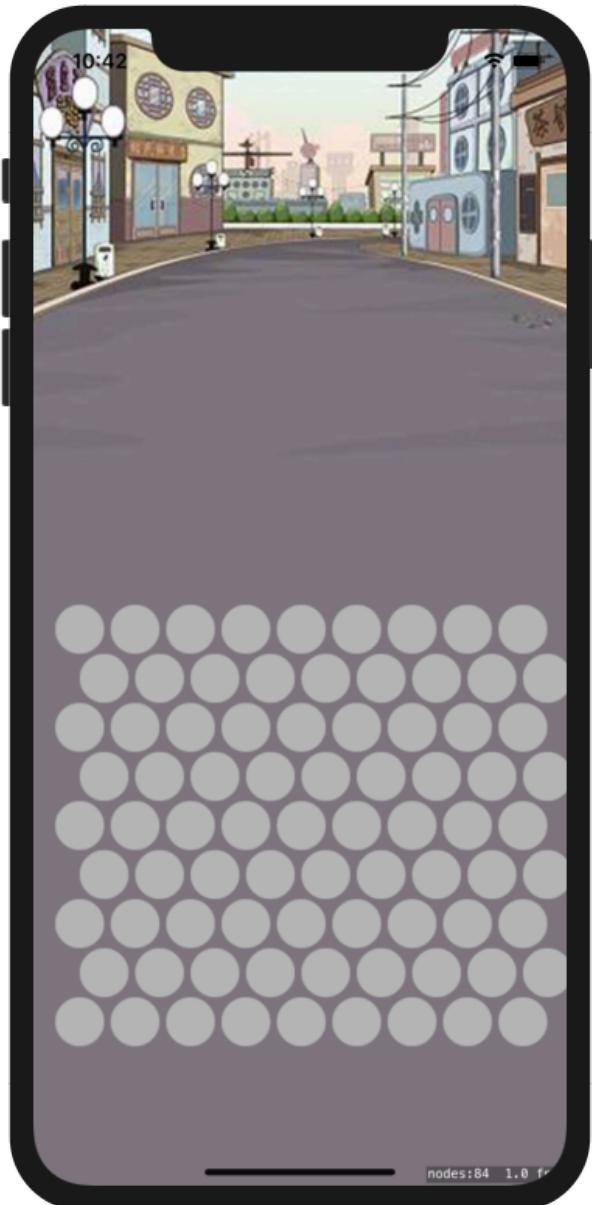
GameScene.swift

宣告一個PointContainer
物件

設定在Scene中的位置，
並呼叫onInit函式來初始化
所需的Nodes

```
1 import SpriteKit
2
3 class GameScene: SKScene {
4     let pcontainer = PointContainer()
5
6     override func didMove(to view: SKView) {
7         let background = SKSpriteNode(imageNamed: "bg")
8         background.position = CGPoint(x:self.frame.midX,y:self.frame.midY)
9         addChild(background)
10
11         pcontainer.position = CGPoint(x: self.frame.midX, y: self.frame.midY-150)
12         addChild(pcontainer)
13         pcontainer.onInit()
14     }
15     override func touchesBegan(_ touches: Set<UITouch>, with event: UIEvent?) {
16
17     }
18 }
```

Run it



iPhone 11 — 13.3

PointContainer.swift

此函式將arrPoint陣列裡的每個點之周圍點都記錄下來，邊緣點不需紀錄

觀察其規律，判斷奇偶行

```
43 func onData(){
44     for point in arrPoint {
45         let row = Int( point.index / 9 )
46         if Int(row % 2) == 1 {
47             if point.index - 1 >= 0 && point.index - 1 <= 80 {
48                 point.aroundPoint.append(point.index - 1)
49             }
50             if point.index + 9 >= 0 && point.index + 9 <= 80 {
51                 point.aroundPoint.append(point.index + 9)
52             }
53             if point.index + 10 >= 0 && point.index + 10 <= 80 {
54                 point.aroundPoint.append(point.index + 10)
55             }
56             if point.index + 1 >= 0 && point.index + 1 <= 80 {
57                 point.aroundPoint.append(point.index + 1)
58             }
59             if point.index - 8 >= 0 && point.index - 8 <= 80 {
60                 point.aroundPoint.append(point.index - 8)
61             }
62             if point.index - 9 >= 0 && point.index - 9 <= 80 {
63                 point.aroundPoint.append(point.index - 9)
64             }
65 }
```

PointContainer.swift in Function onData()

```
66         else {
67             if point.index - 1 >= 0 && point.index - 1 <= 80 {
68                 point.aroundPoint.append(point.index - 1)
69             }
70             if point.index + 8 >= 0 && point.index + 8 <= 80 {
71                 point.aroundPoint.append(point.index + 8)
72             }
73             if point.index + 9 >= 0 && point.index + 9 <= 80 {
74                 point.aroundPoint.append(point.index + 9)
75             }
76             if point.index + 1 >= 0 && point.index + 1 <= 80 {
77                 point.aroundPoint.append(point.index + 1)
78             }
79             if point.index - 9 >= 0 && point.index - 9 <= 80 {
80                 point.aroundPoint.append(point.index - 9)
81             }
82             if point.index - 10 >= 0 && point.index - 10 <= 80 {
83                 point.aroundPoint.append(point.index - 10)
84             }
85         }
86     }
87 }
```

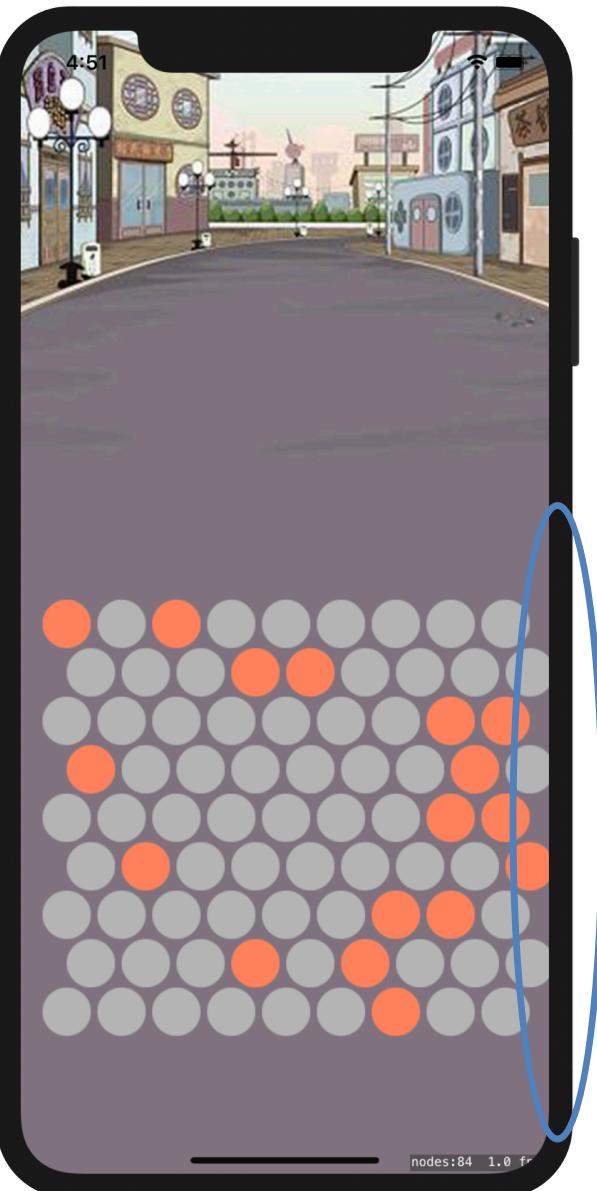
PointContainer.swift

```
89     //改變圓點類型
90     func onSetRed(index:Int){
91         arrPoint[index].type = pointtype.red
92         arrPoint[index].texture = textPoint2
93     }
94     //隨機生成紅色點
95     func onCreateRed(){
96         for i in 0...8 {
97             let r1 = Int(arc4random() % 9) + i * 9
98             let r2 = Int(arc4random() % 9) + i * 9
99             if r1 != startIndex {
100                 onSetRed(index: r1)
101             }
102             if r2 != startIndex {
103                 onSetRed(index: r2)
104             }
105         }
106     }
```

PointContainer.swift In func onInit()

```
29         // 判斷是否為邊緣點
30         if row == 0 || row == 8 || col == 0 || col == 8 {
31             point.isEdge = true
32         }
33         point.index = i
34         point.zPosition = 10
35         addChild(point)
36         arrPoint.append(point)
37     }
38     onData()
39     onCreateRed()
40 }
```

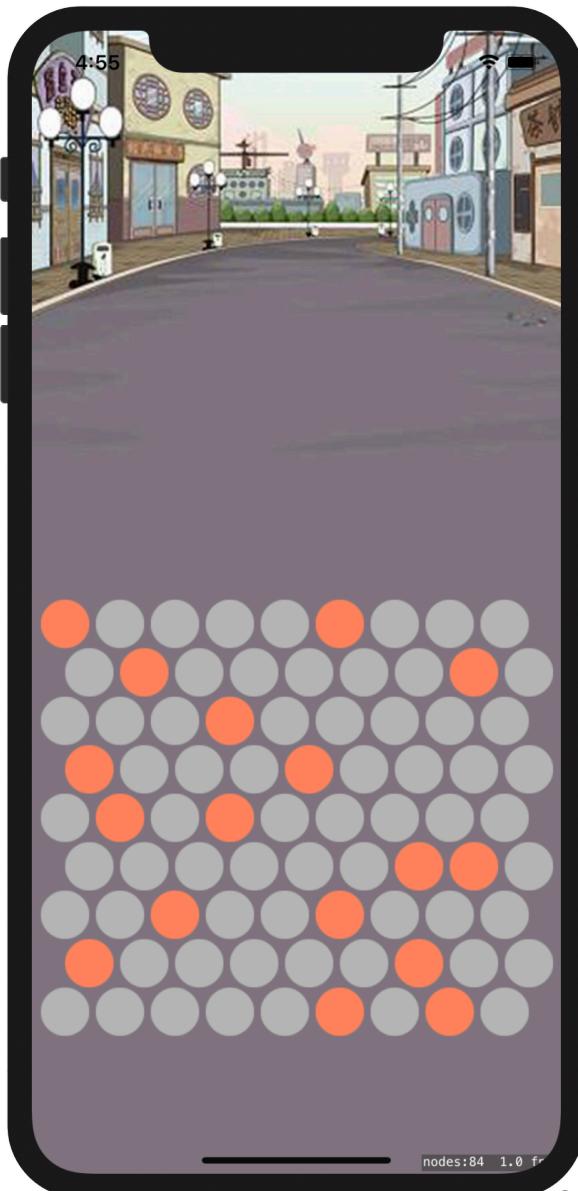
Run it



iPhone 11 — 13.3

GameScene.swift

```
11     pcontainer.position = CGPoint(x: self.frame.midX-10, y:  
12         self.frame.midY-150)  
13     addChild(pcontainer)  
14     pcontainer.onInit()  
    }
```



iPhone 11 — 13.3