Software Studio
軟體設計與實驗

Physics Tutorial

Hung-Kuo Chu
Department of Computer Science
National Tsing Hua University

CS2410

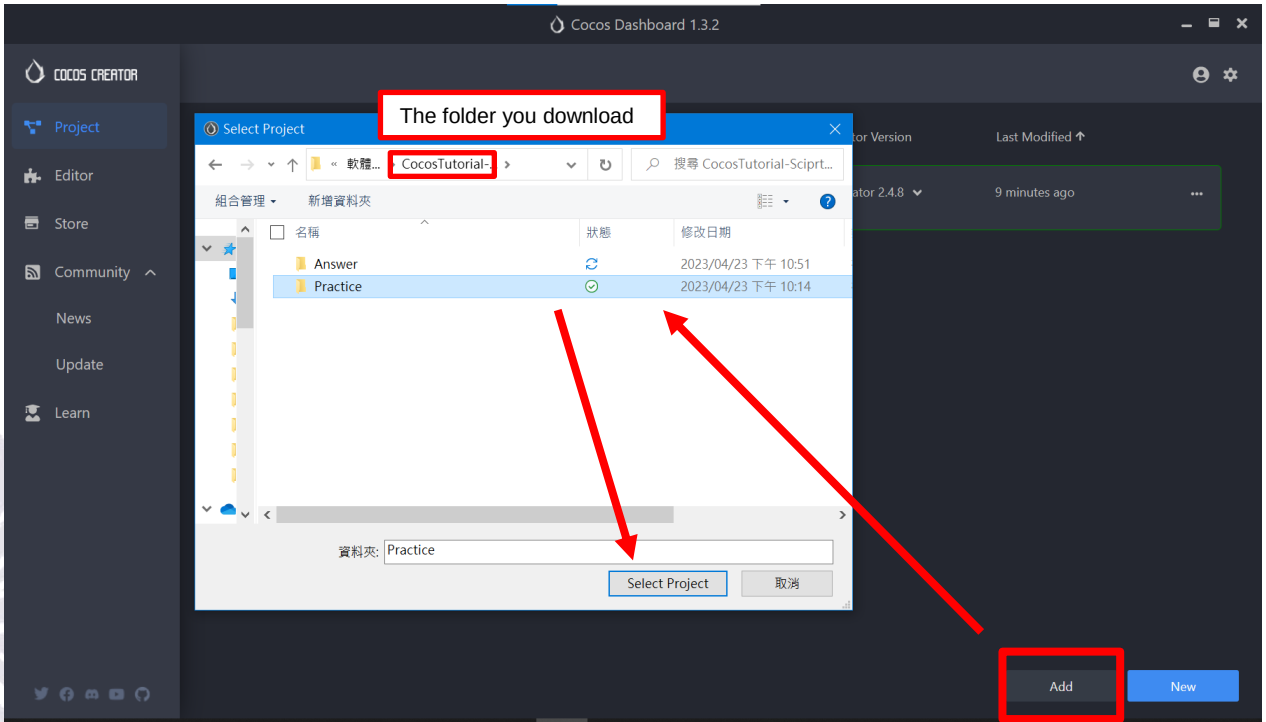
Goal

Contents

- Update score
- Contact with pigs
- Trigger with game item
- What you can do more:
  - Add Effect Sound.
  - Add more effect.

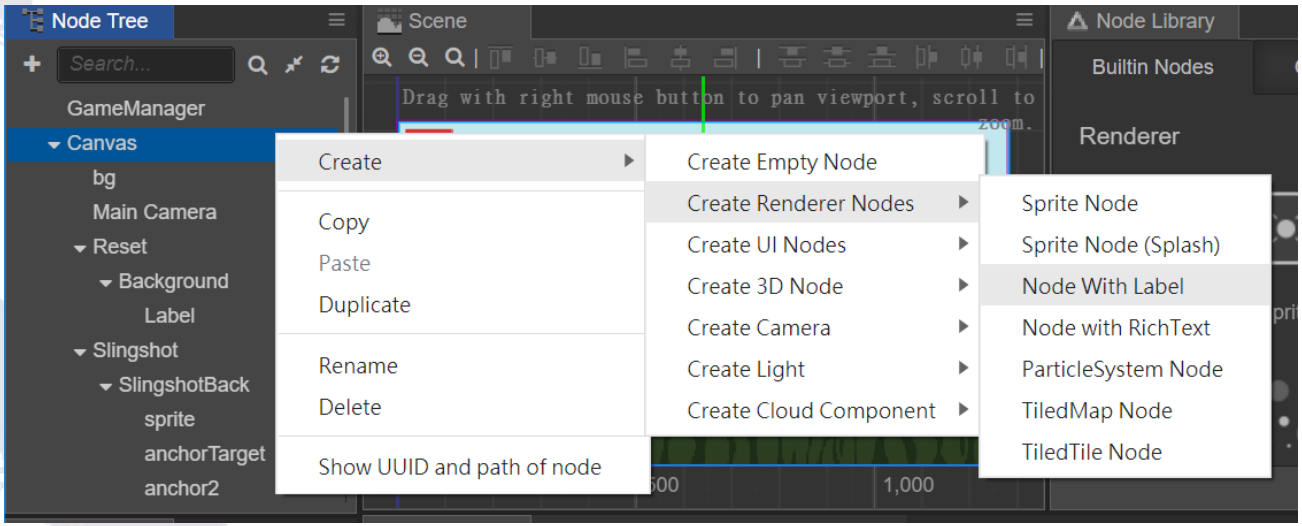
Open the Project

- Step1. Download project from eeclass or GoogleDrive and unzip
  - https://reurl.cc/VzRmmZ
- Step2. Add the Practice folder to Cocos



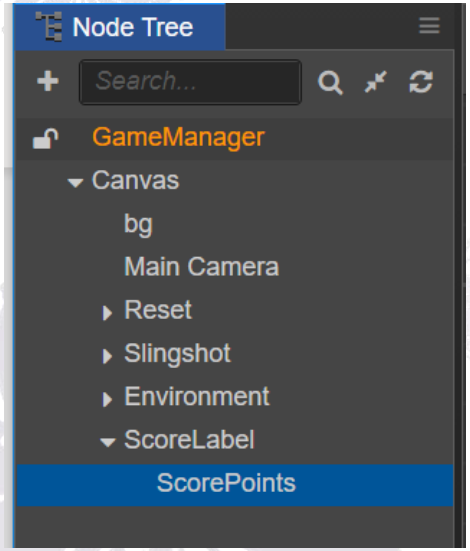
Create a Score Node

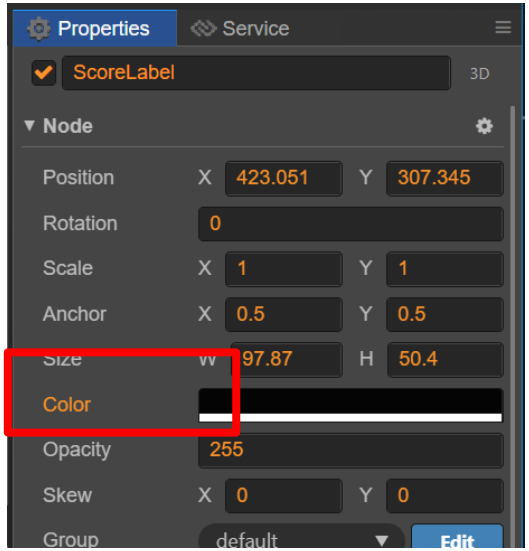
- Step1. Switch to main scene
  - assets/Scenes/main
- Step2. Create two Label node to show score
  - Right Click Canvas
  - Create/Create Renderer Nodes/Node With Label

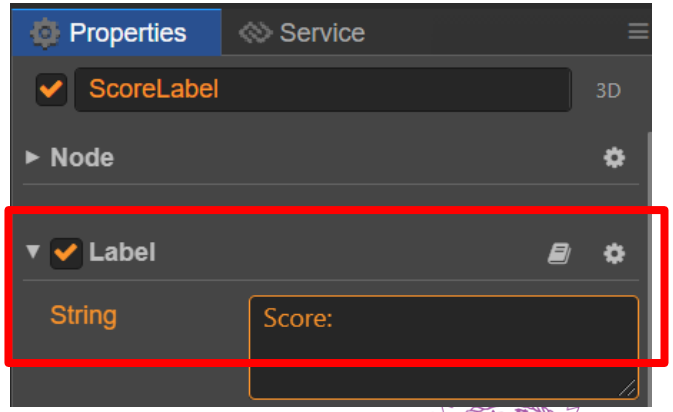


Create a Score Node

- Step3. Rename one node to “ScoreLabel” and the other to “ScorePoints”
- Step4. Drag ScorePoints to ScoreLabel as child
- Step5. Find a good position and color
- Step6. Set the label string







Set up Score Property

- Step1. In bird.ts, define score for record the score point number and scorePoints for the ScorePoints node (TODO 1.1)
  - assets/Scripts/bird.ts
- Step2. Initialize the score to 0 when start (TODO 1.2)
  - assets/Scripts/bird.ts start()

```
24      // ========== TODO 1.1 ==========
25      // 1. Define score related property.
26      score: number;
27
28      @property(cc.Node)
29      scorePoints = null;
30      // ==============================
```

```
46      start() {
47          this.initProperties();
48          this.initResetButton();
49          // ========== TODO 1.2 ==========
50          // 1. Initiallize the score to 0 when start.
51          this.score = 0;
52          // ==============================
53      }
```

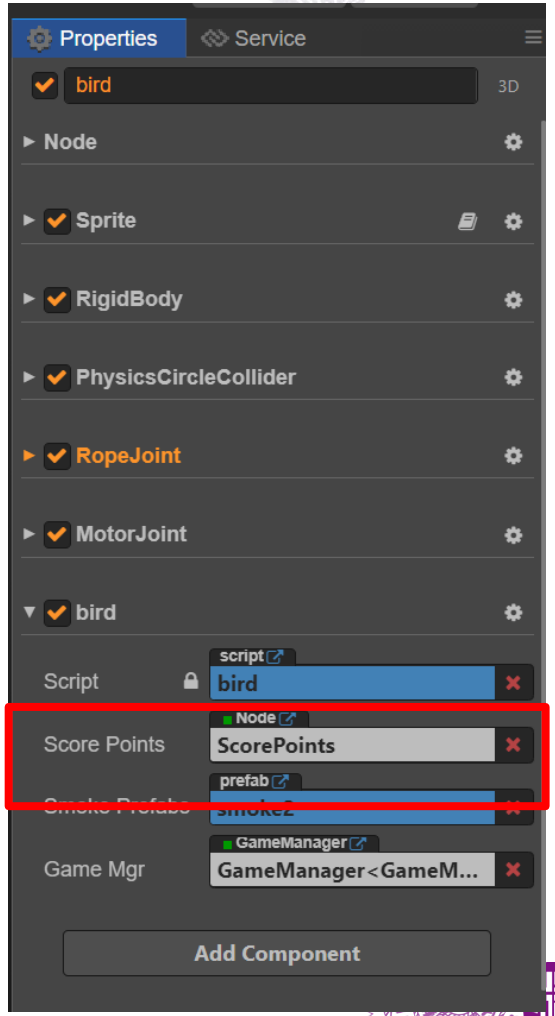
Set up Update function

- Step1. Create a function updateScore() to update the score points (TODO 1.3)

```
197     // ========== TODO 1.3 ==========
198     // 1. Add a function to update score point
199     // (Hint: You can use number.toString() function to convert number to string)
200     updateScore(number) {
201         this.score += number;
202         this.scorePoints.getComponent(cc.Label).string = this.score.toString();
203     }
204     // ==============================
```

Add Node to the Component

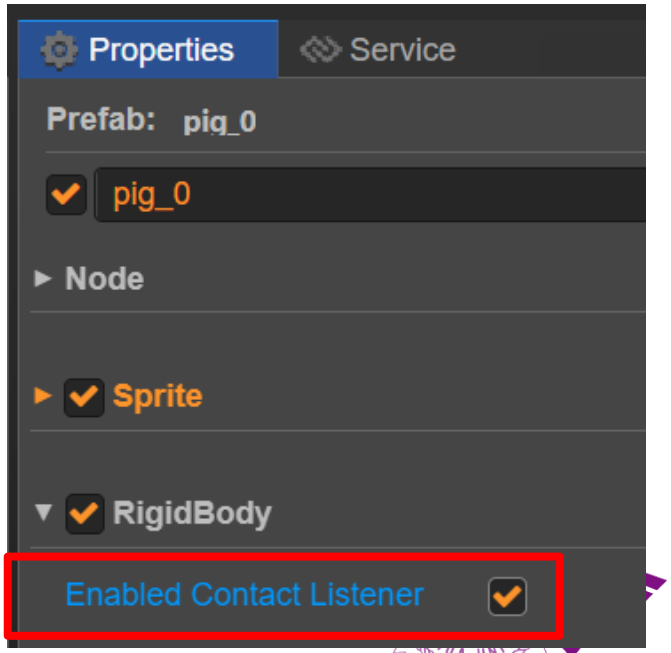
- Step1. Drag ScorePoints node into Score Points of bird node's bird component



Your
turn!
C++
PASCAL

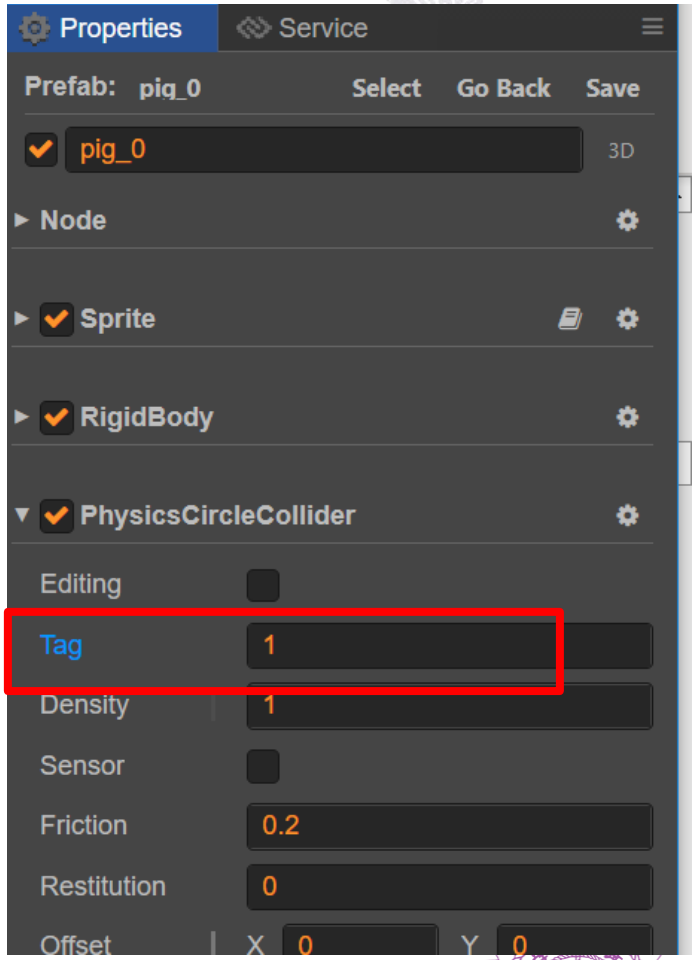
Contact with Pigs

- Step1. Find the pig prefab
  - Double click pig_bg and pig_sm
  - assets/Prefabs/pig_bg and pig_sm
- Step2. Enabled Contact Listener
  - Go to property tab
  - Find RigidBody Component
  - Check Enabled Contact Listener



Set up Collider

- Step1. Change the tag
  - Find PhysicsCircleCollider Component
  - Set the Tag to 1



Set up Contact function

- Step1. In bird.ts, write onBeginContact() (TODO 2.1.1)
  - assets/Scripts/bird.ts
- Step2. Use tag to figure out which object we're contact with. (TODO 2.1.2)
- Step3. Instantiate smoke prefab (TODO 2.1.2-1)
- Step4. Use this.scheduleOnce to destroy smoke prefab after 1.5s(TODO 2.1.2-2)

Set up Contact function

- Step4. Update the score (TODO 2.1.2-3)
- **Contact** objects include collision points and the normal vector, which are acquired by getWorldManifold()

```
210     onBeginContact(contact, self, other) {
211         if (other.tag == 1) { // enemy tag
212             console.log("BeginContact")
213             console.log(contact.getWorldManifold().points);
214 
215             var smoke = cc.instantiate(this.smokePrefabs);
216             smoke.setPosition(contact.getWorldManifold().points[0]);
217 
218             cc.find("Canvas/Environment").addChild(smoke);
219             this.scheduleOnce(function () {
220                 smoke.destroy();
221             }, 1.5);
222 
223             this.updateScore(30);
224         }
225         // ==============================
```

Cocos-docs_contact-getWorlManifold()

Set up Contact function

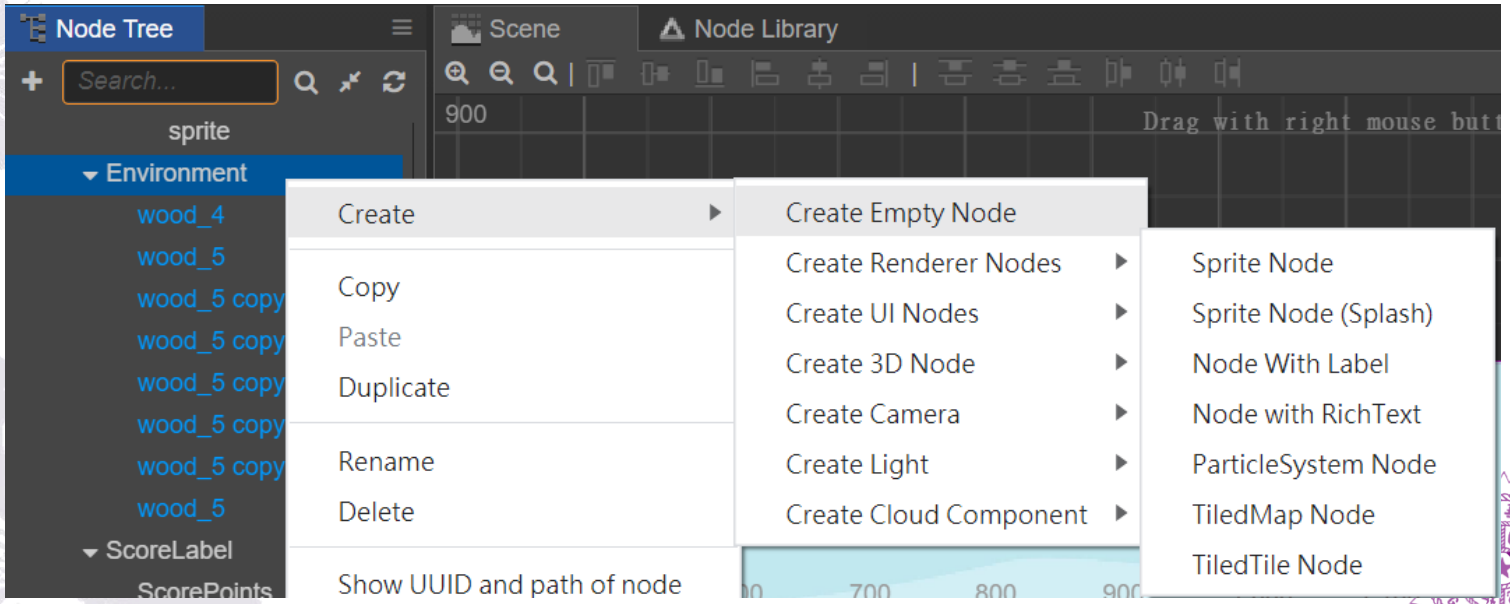
- Step5. Write onEndContact() (TODO 2.2)
- Step6. Change the tag of pigs in order not to instantiate smoke any more

```
243     // ========== TODO 2.2 ==========
244     // 1. Set up onEndContact.
245     // 2. Change the tag of pigs' in order not to instantiate smoke any more.
246     onEndContact(contact, self, other) {
247         if (other.tag == 1) other.tag = 0;
248     }
249     // ==============================
```

Your
turn!
C++
PASCAL

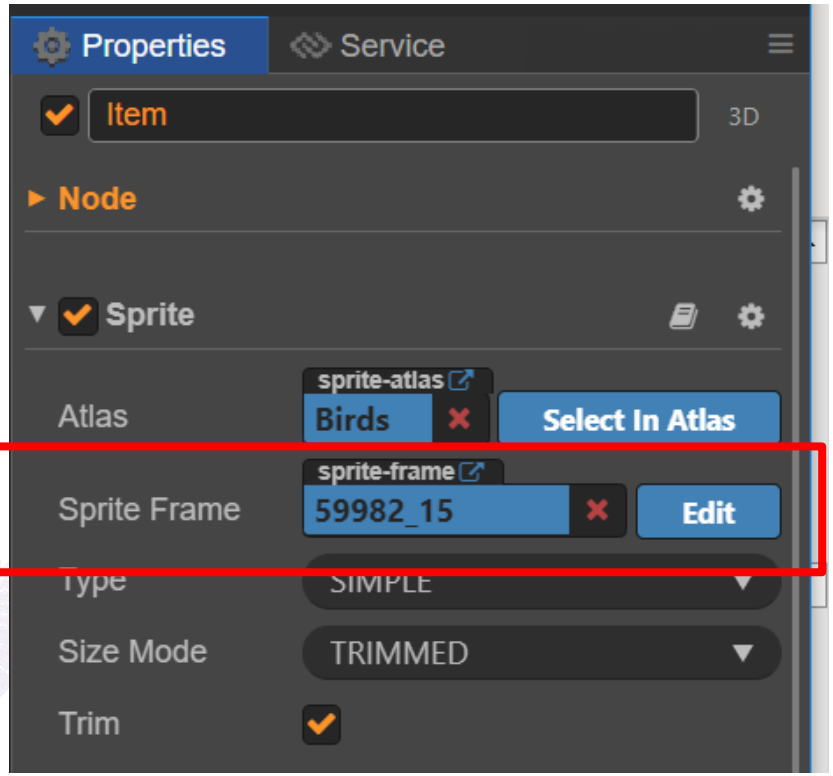
Create a Game Item Node

- Step1. In main scene, create a Sprite Node
  - Right Click Environment
  - Create/Create Renderer Nodes/Sprite Node
- Step2. Rename the node to “Item”



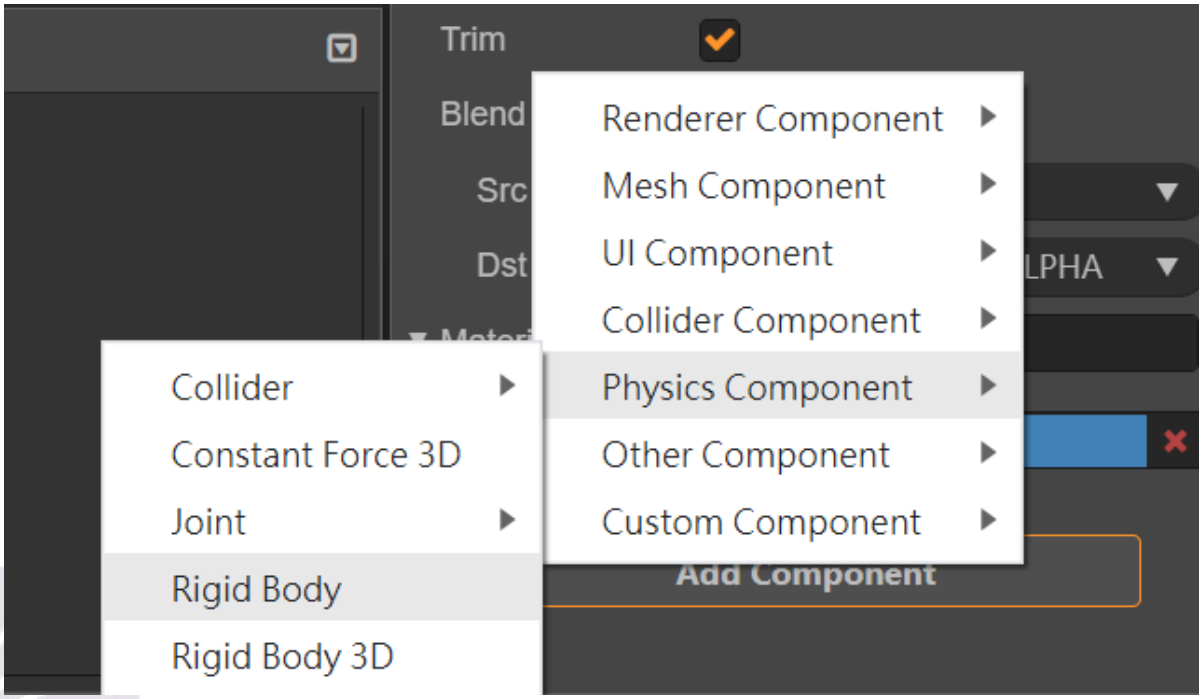
Create a Game Item Node

- Step3. Find a good picture for the Sprite Frame
  - assets/Images



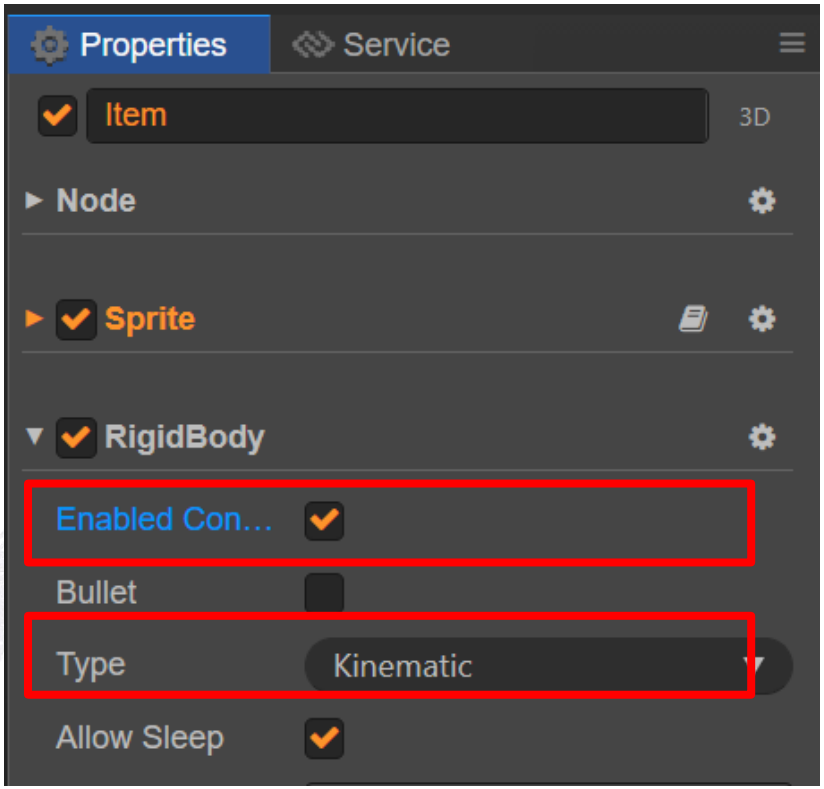
Set up Rigid Body

- Step1. Add Rigid Body Component to Item node
  - Add Component/Physics Component/Rigid Body



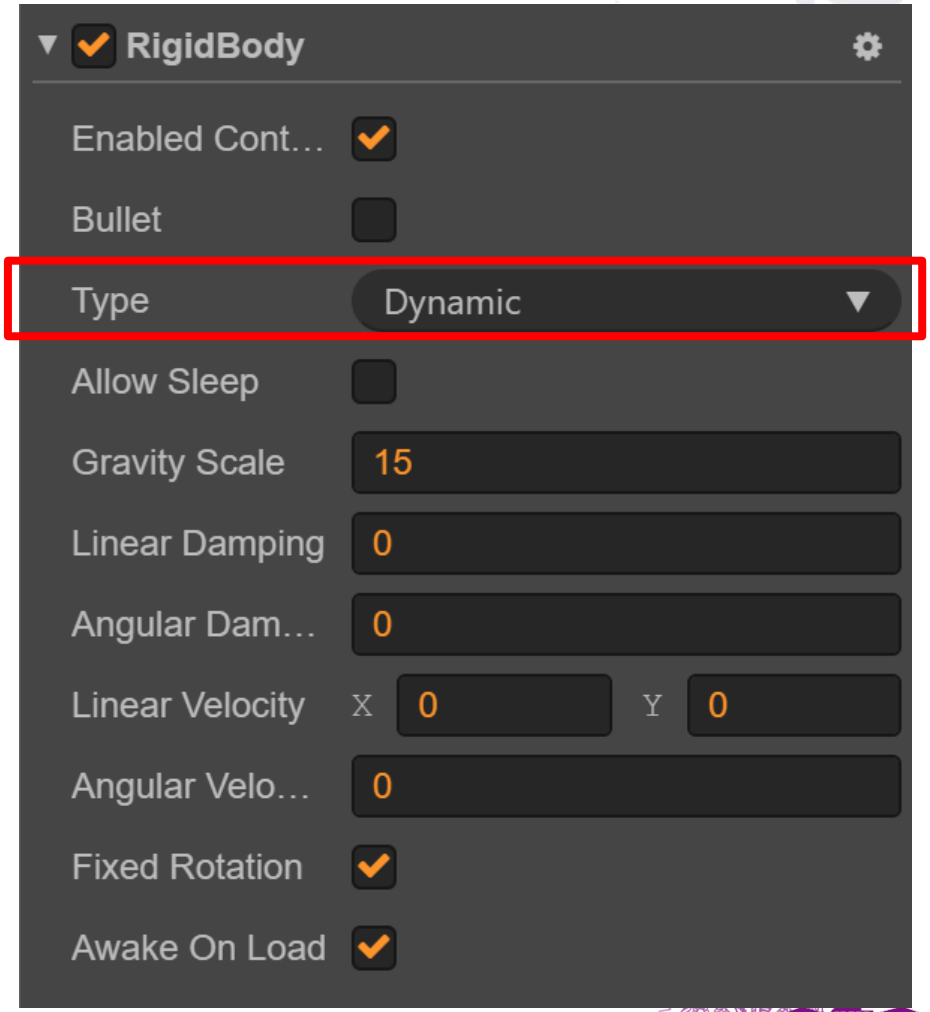
Set up Rigid Body

- Step2. Enabled Contact Listener
- Step3. Change the Type to Kinematic



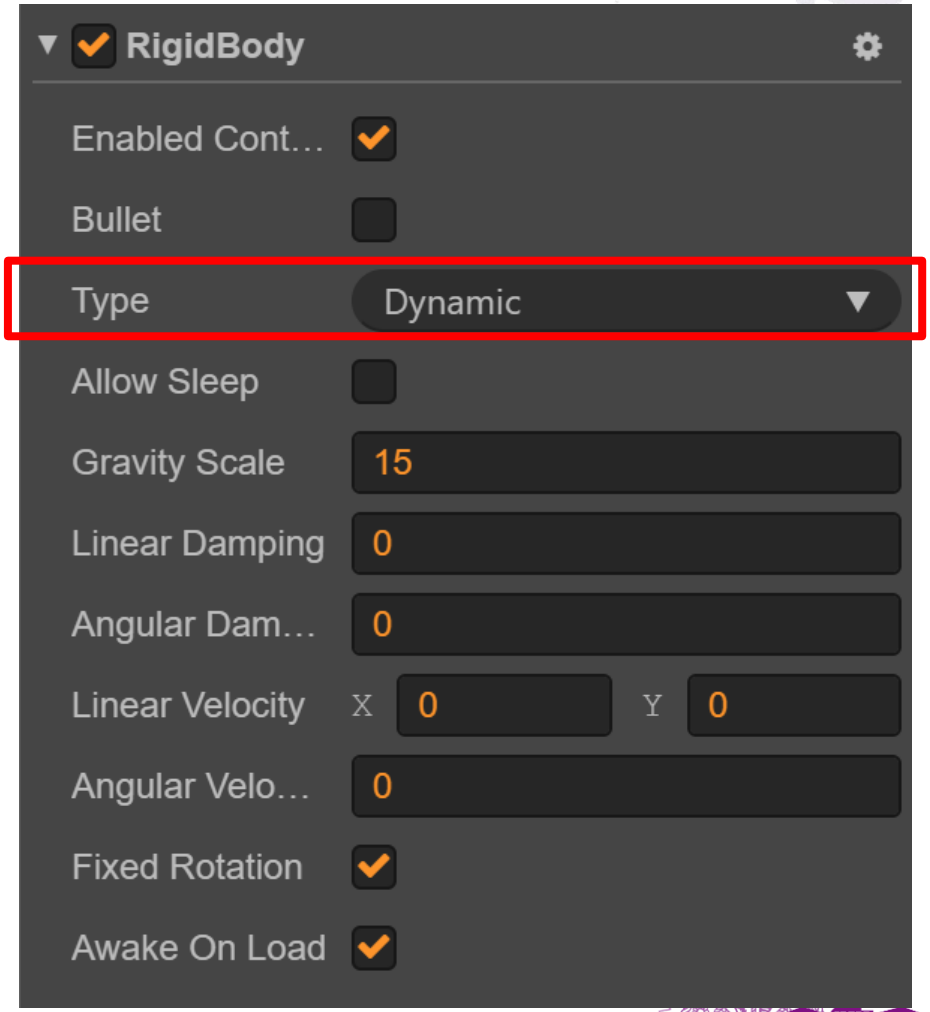
Rigid Body: Types

- **Static:** zero mass, zero velocity, that is not affected by gravity or force, but can set its position to move.
- **Dynamic:** with mass, its velocity can be set, will be affected by gravity.



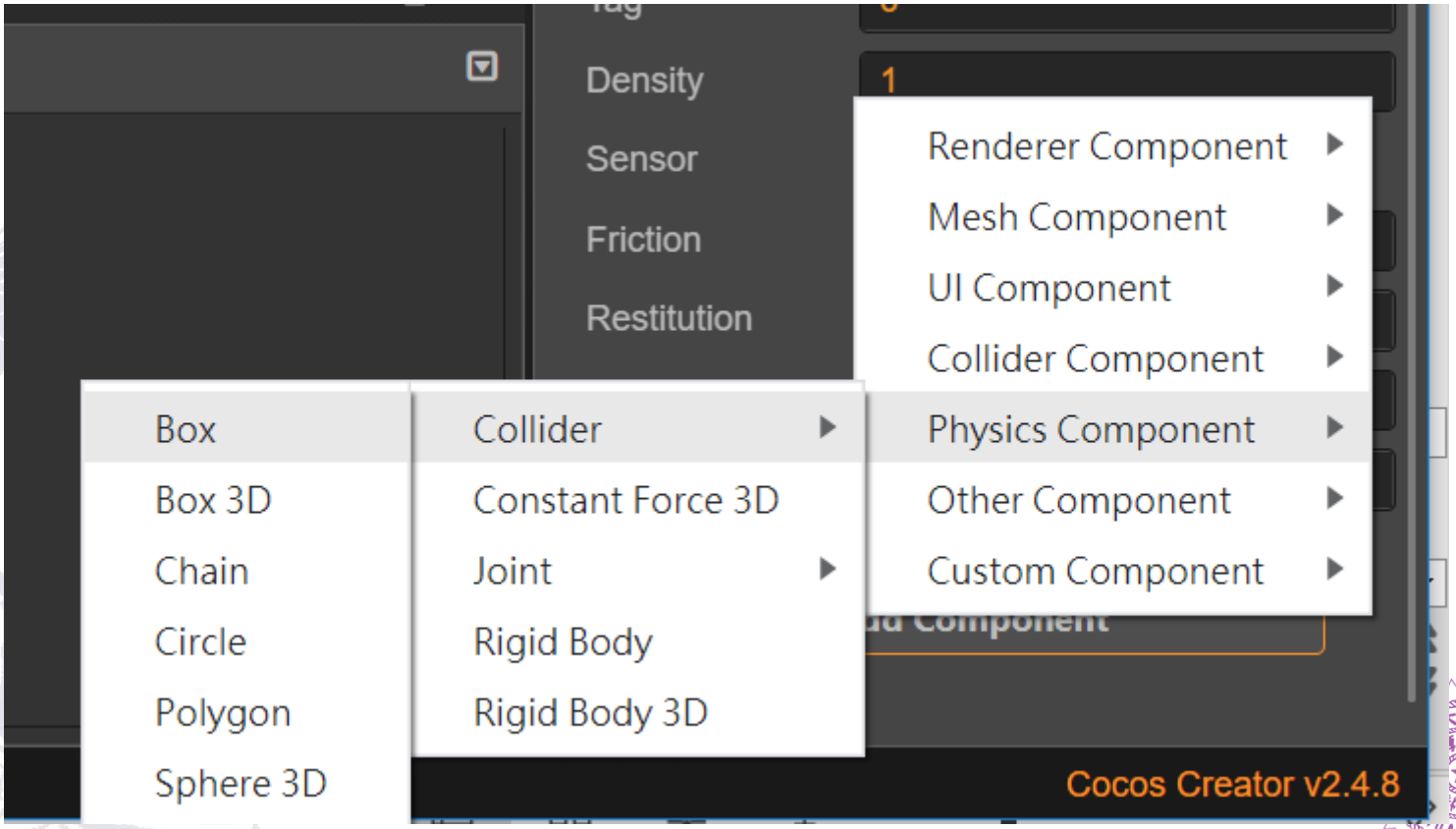
Rigid Body: Types

- **Kinematic:** zero mass, its velocity can be set, will not be affected by gravity, but can move by setting the velocity.
- **Animated:** derived from Kinematic type, mainly used for rigidbody and animation in combination (We won't cover this part).



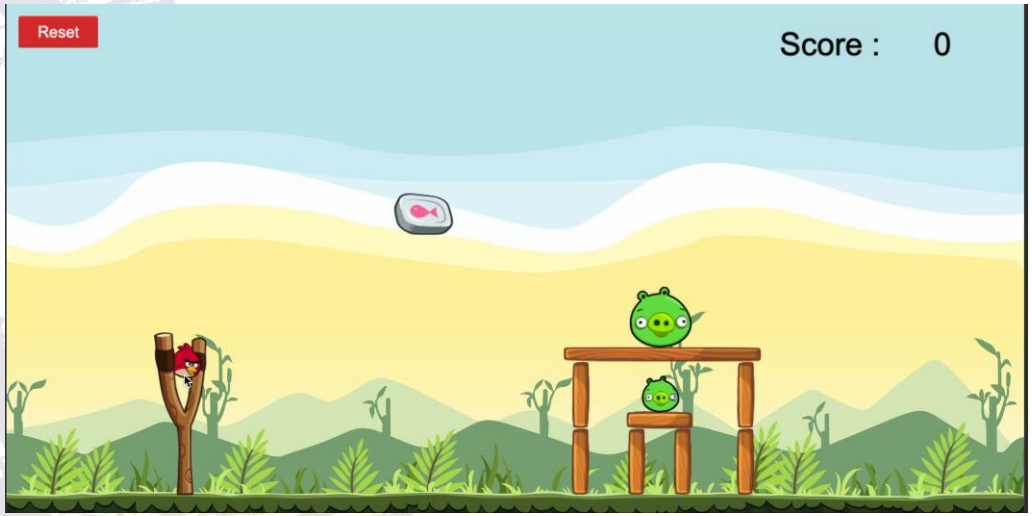
Set up Collider

- Step1. Add PhysicsBoxCollider to Item node
  - Add Component/Physics Component/Collider/Box

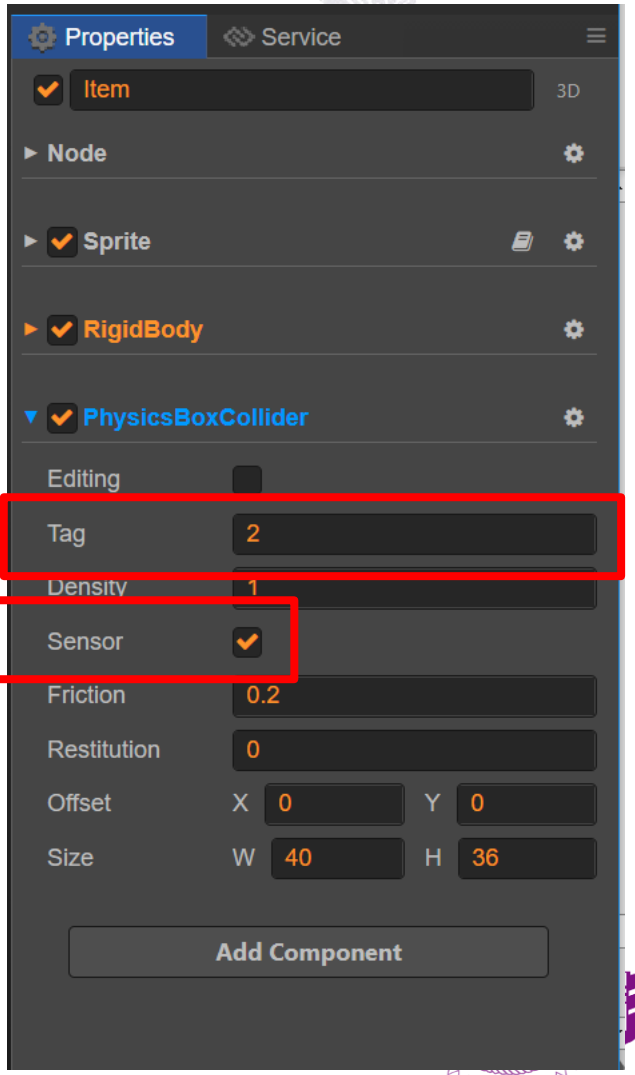


Set up Rigid Body

- Step2. Enabled Sensor to trigger (enable collision callback, but no physical collision effect will occur)
- Step3. Set the Tag to 2



Collision effect w/o sensor enabled



Set up Contact function

- Step1. In bird.ts, find the onBeginContact()
  - assets/Scripts/bird.ts
- Step2. Use tag to figure out item(TODO 3)
- Step3. Destroy the item (TODO 3.1)
- Step4. Add 10 score points (TODO 3.2)

```
227         // =========== TODO 3 ===========
228         // 3. if contact with item.
229         //    3-1. Destroy the item.
230         //    3-2. Add 10 points.
231         else if (other.tag == 2) { // game item tag
232
233             console.log("Trigger");
234
235             other.node.destroy();
236             this.updateScore(10);
237         }
238         // ============================
```

Your
turn!
C++
PASCAL