

# CODING

CO SPACES MAKER

How to ▾ Coding ▾ Marketplace Get vouchers | Gallery Login [Sign up](#)

The CoSpaces Edu edition designed by teachers and students.

[About](#)  
[Blockly](#)  
[JavaScript & TypeScript](#)  
[Learn more](#)

GET COSPACES NEWS

enter your email here →

Coding  
Ambassadors

About  
Forum  
Press  
Terms of Use  
Privacy Policy

2017 © Delightex GmbH

[f](#) [t](#) [p](#) [i](#) [y](#) [b](#)

[Contact us](#)

<https://cospaces.io/maker/coding.html>

# CODING介紹

**CO SPACES MAKER**

How to ▾ Coding ▾ Marketplace Get vouchers | Gallery Login [Sign up](#)

## CODING WITH COSPACES MAKER

ay Mode Script Build Mode Script

**Let's code!**

Make your virtual worlds come to life by animating and programming them. CoSpaces Maker gives you a choice of coding languages - from Blockly for

# 兩種CODING語言

## CHOOSE YOUR FAVORITE CODING LANGUAGE



[Learn more](#)



[Learn more](#)

# BLOCKLY

## What is Blockly?

Blockly is a visual programming language, similar to Scratch. You just add, adapt and combine code blocks via drag and drop. It's easy to use, even for kids.



## Using Blockly in CoSpaces Maker

Start creating Blockly scripts with the in-built code editor. Any code you create can be played in VR with the CoSpaces Maker app.



# JAVASCRIPT

## PROGRAM YOUR OWN VIRTUAL WORLD WITH JAVASCRIPT & TYPESCRIPT

With the help of a script you can make simple interactions or even create games. All projects you create are automatically playable in gyroscope and VR mode for mobile apps.

### API Documentation

CoSpaces Maker currently offers JavaScript and TypeScript as scripting languages. The API documentation can be found below. Every method usually contains a code sample and a demo project.

[Go to the API](#)



# JAVASCRIPT API

Filter

**Space**

finishPlayMode()  
goTo()  
log()  
showSceneNavigation()

**Scene**

clear()  
createFunction()  
createItem()  
createTextBillboard()  
createTimeVaryingFunction()  
currentTime()  
getItem()  
getProperty()  
loadSound()  
onPropertyChanged()  
renderShadows()  
schedule()  
scheduleRepeating()  
setProperty()

**Sound**

currentPosition()

Description:

Create a specific item at a certain position in the space.

**.createItem(modelId, xPos, yPos, zPos)**

**Returns**  
*Type:* object  
Object representing the item

**modelId**  
*Type:* string  
Predefined model id of the item you want to create.

3D low poly objects:

LP\_Wom, LP\_Man, LP\_Mouse, LP\_Snake, LP\_Rabbit, LP\_Butterfly, LP\_BlackBird, LP\_WhiteBird, LP\_Swan, LP\_Cat, LP\_Dog, LP\_Lion, LP\_Horse, LP\_Camel, LP\_Elephant, LP\_SkateBoard, LP\_Bike, LP\_Car, LP\_Bus, LP\_Truck, LP\_Train, LP\_Plane, LP\_Rocket, LP\_Boat2, LP\_ShipsWheel, LP\_SafetyWheel, LP\_Compass, LP\_Fence, LP\_Bridge, LP\_Stone, LP\_Stone2, LP\_Flowers, LP\_FlowersBed, Grass, LP\_Tree3, Tree3, Tree, LP\_Pine, Pine, LP\_Cactus, LP\_Palm, LP\_Cloud1, LP\_Cloud2, LP\_Cloud3, LP\_House, LP\_Building1, LP\_Building2, LP\_Building3, LP\_Building4, LP\_Chair, LP\_Table, LP\_Clock, LP\_VRGlasses, LP\_BackPack, LP\_Axe, LP\_TreasureBox, LP\_Balloons, LP\_Balance, LP\_Heart, LP\_Lightning, LP\_Star, LP\_Sphere, LP\_Egg, LP\_Flag

Flexible objects:

Capsule, Cone, ConeFrustum, Cuboid, EllipticCylinder, Ellipsoid, Hemiellipsoid, Frustum, Torus, SemiTorus

**xPos**  
*Type:* number  
x coordinate

**yPos**  
*Type:* number  
y coordinate

**zPos**  
*Type:* number  
z coordinate

# JAVASCRIPT API功能函數

Filter

**Space**

finishPlayMode()  
goTo()  
log()  
showSceneNavigation()

**Scene**

clear()  
createFunction()  
**createItem()**  
createTextBillboard()  
createTimeVaryingFunction()  
currentTime()  
getItem()  
getProperty()  
loadSound()  
onPropertyChanged()  
renderShadows()  
schedule()  
scheduleRepeating()  
setProperty()

**Sound**

currentPosition()

Description:  
Create a specific item at a certain position in the space.

**.createItem(modelId, xPos, yPos, zPos)**

**Returns**  
*Type:* object  
Object representing the item

**modelId**  
*Type:* string  
Predefined model id of the item you want to create.

3D low poly objects:  
LP\_Wom, LP\_Man, LP\_Mouse, LP\_Snake, LP\_Rabbit, LP\_Butterfly, LP\_BlackBird, LP\_WhiteBird, LP\_Swan, LP\_Cat, LP\_Dog, LP\_Lion, LP\_Horse, LP\_Camel, LP\_Elephant, LP\_SkateBoard, LP\_Bike, LP\_Car, LP\_Bus, LP\_Truck, LP\_Train, LP\_Plane, LP\_Rocket, LP\_Boat2, LP\_ShipsWheel, LP\_SafetyWheel, LP\_Compass, LP\_Fence, LP\_Bridge, LP\_Stone, LP\_Stone2, LP\_Flowers, LP\_FlowersBed, Grass, LP\_Tree3, Tree3, Tree, LP\_Pine, Pine, LP\_Cactus, LP\_Palm, LP\_Cloud1, LP\_Cloud2, LP\_Cloud3, LP\_House, LP\_Building1, LP\_Building2, LP\_Building3, LP\_Building4, LP\_Chair, LP\_Table, LP\_Clock, LP\_VRGlasses, LP\_BackPack, LP\_Axe, LP\_TreasureBox, LP\_Balloons, LP\_Balance, LP\_Heart, LP\_Lightning, LP\_Star, LP\_Sphere, LP\_Egg, LP\_Flag

Flexible objects:  
Capsule, Cone, ConeFrustum, Cuboid, EllipticCylinder, Ellipsoid, Hemiellipsoid, Frustum, Torus, SemiTorus

**xPos**  
*Type:* number  
x coordinate

**yPos**  
*Type:* number  
y coordinate

**zPos**  
*Type:* number  
z coordinate

# JAVASCRIPT API 程式範例

Filter

**Space**

- finishPlayMode()
- goTo()
- log()
- showSceneNavigation()

**Scene**

- clear()
- createFunction()
- createItem()**
- createTextBillboard()
- createTimeVaryingFunction()
- currentTime()
- getItem()
- getProperty()
- loadSound()
- onPropertyChanged()
- renderShadows()
- schedule()
- scheduleRepeating()
- setProperty()

**Sound**

<https://cospac.es/F5CM>

LP\_Building3, LP\_Building4, LP\_Chair, LP\_Table, LP\_Clock, LP\_VRGlasses, LP\_BackPack, LP\_Axe, LP\_TreasureBox, LP\_Balloons, LP\_Balance, LP\_Heart, LP\_Lightning, LP\_Star, LP\_Sphere, LP\_Egg, LP\_Flag

Flexible objects:  
Capsule, Cone, ConeFrustum, Cuboid, EllipticCylinder, Ellipsoid, Hemiellipsoid, Frustum, Torus, SemiTorus

**xPos**  
Type: number  
x coordinate

**yPos**  
Type: number  
y coordinate

**zPos**  
Type: number  
z coordinate

Examples  TypeScript  JavaScript

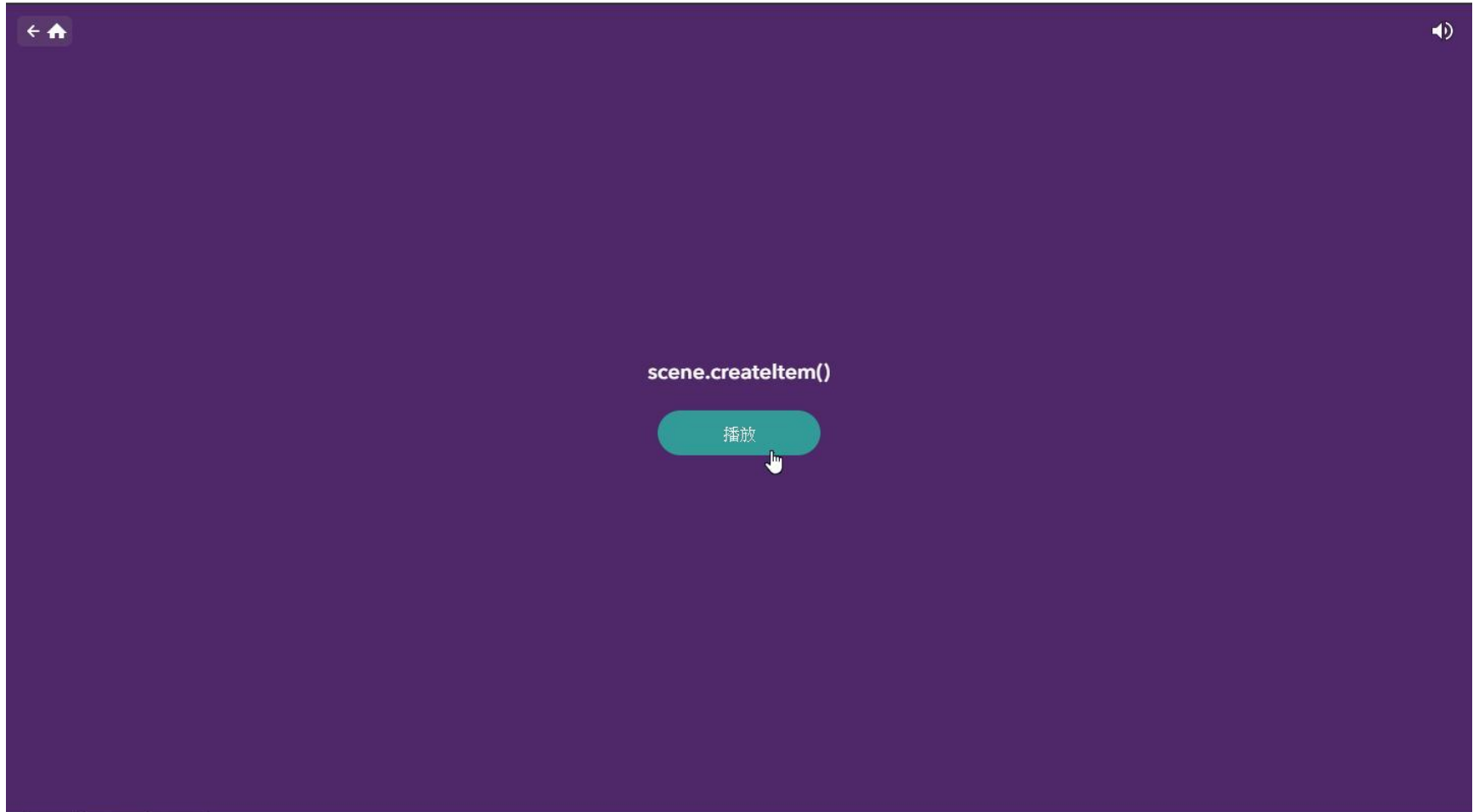
We can programmatically create items with the `createItem()` method. The method takes 4 arguments. The first argument is a predefined id of the model you want to create. Then we need to give it the coordinates of where we want to create the item in the 3D space. Here we are creating 3 items with different positions.

```
1 var woman = Scene.createItem('LP_Wom', 0, 0, 0);
2 var house = Scene.createItem('LP_House', 2, -5, 0);
3 var cloud = Scene.createItem('LP_Cloud3', -4, -3, 7);
```

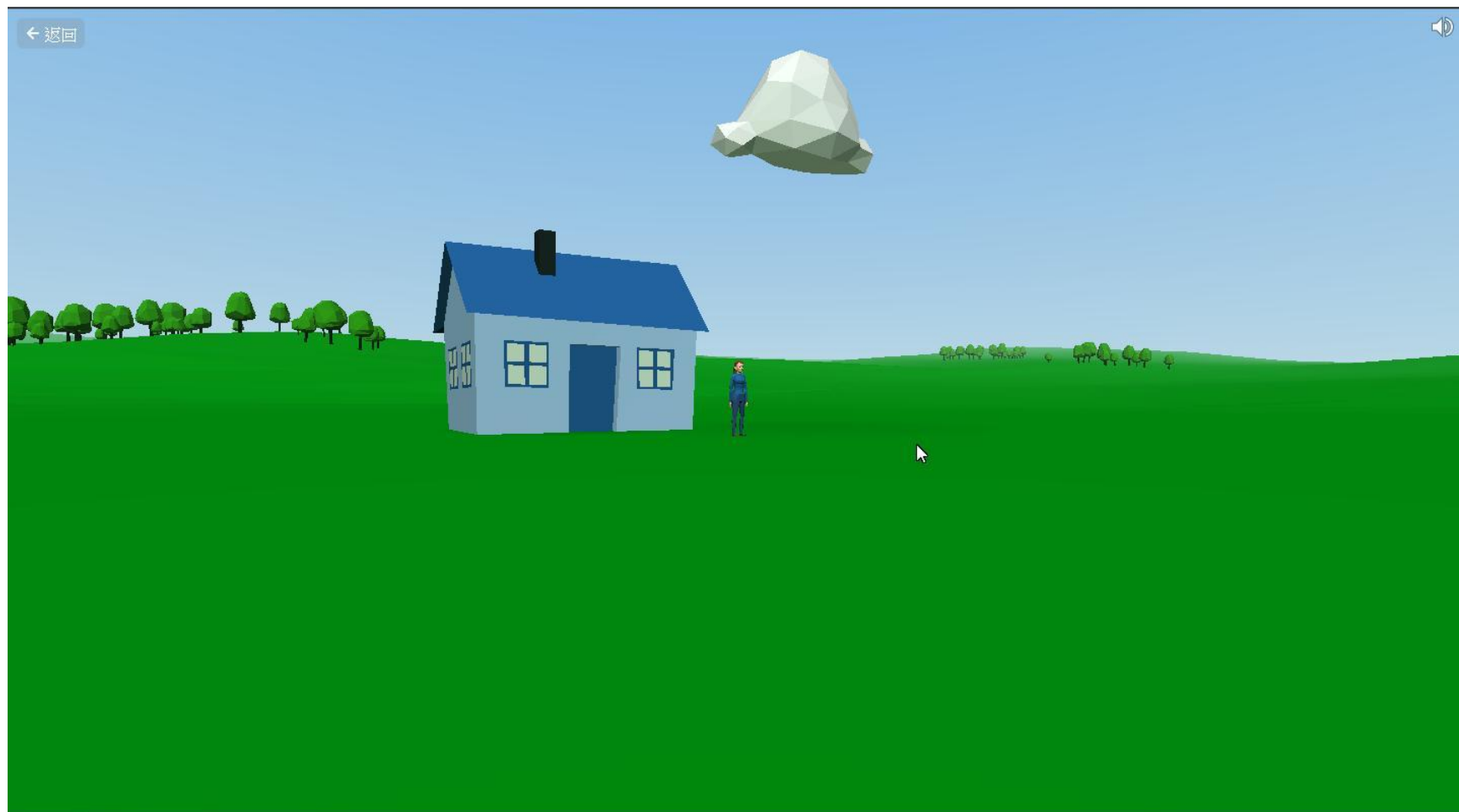
[See the example in action >](#)



# 播放範例



# 以程式建立模型物件



The End ~ Have Fun.