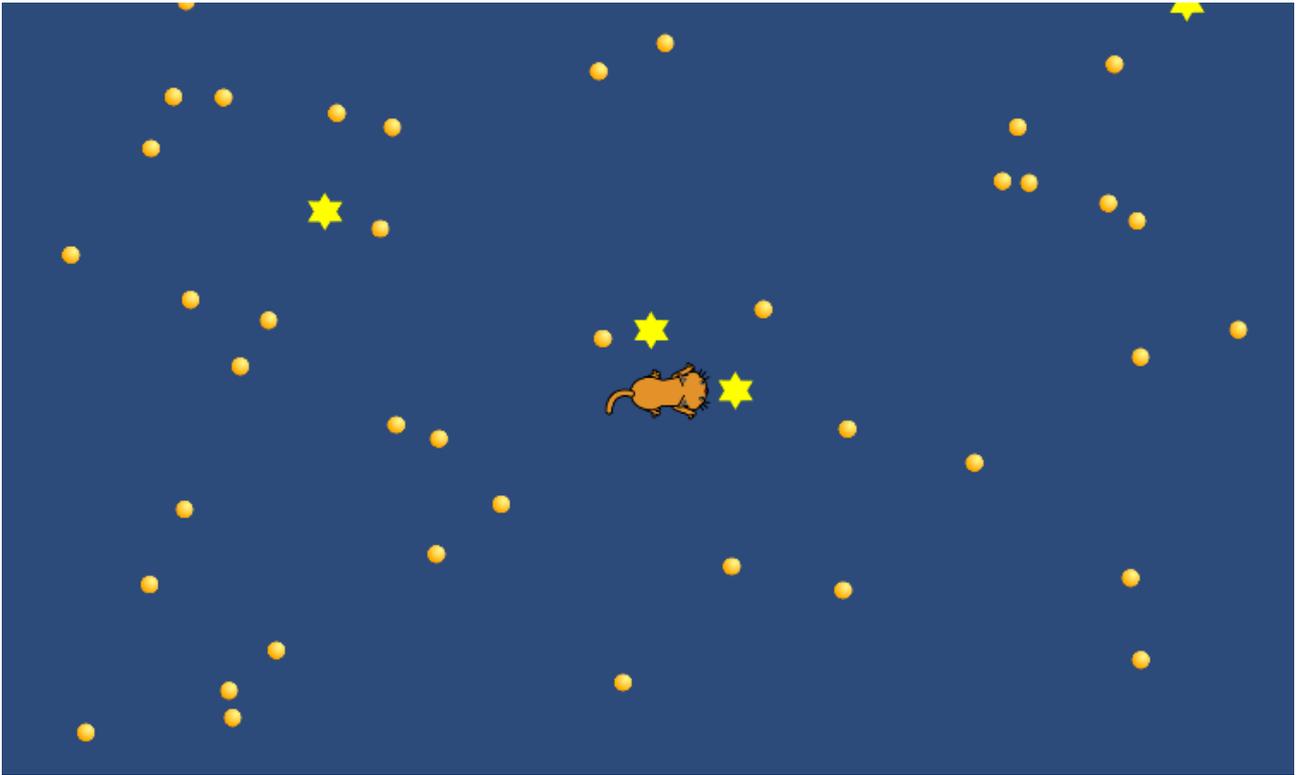


イト スターズ ユニティーばん  
**Eat Stars(Unity版)**

---



---

ZERONE studio

---

## プロジェクトを作る。

2Dでプロジェクトを作ります。

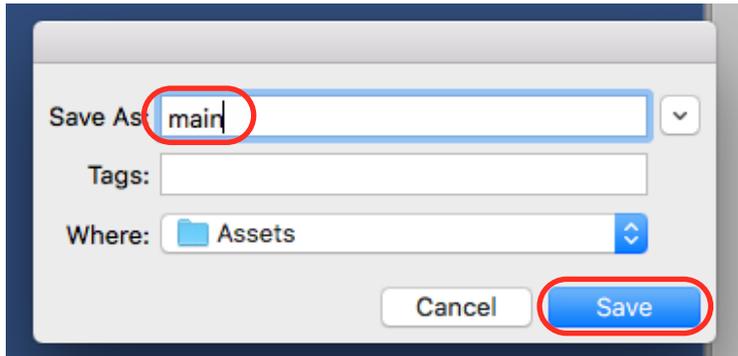
プロジェクト名は、なんでも構いません。

## セーブする。

コマンドキーを押しながら「S」を押します。

最初は、ウィンドウが出てきますので、mainという名前でセーブします。

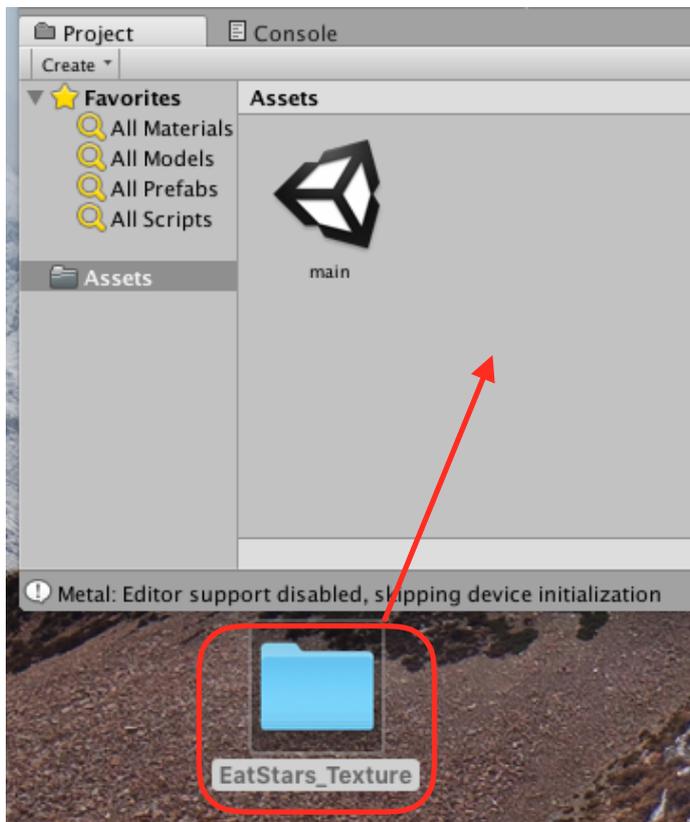
その後は、コマンドキー+S（以下「cmd+S」と書く）で上書きセーブします。



## 素材を入れる。

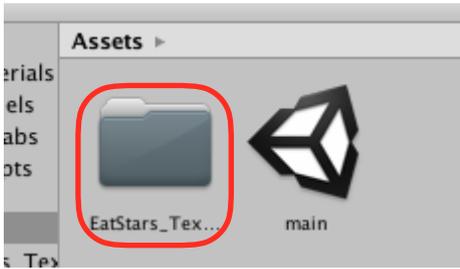
イトスターズテクスチャ

EatStars\_Textureを、Assetsの直下に、DDします。

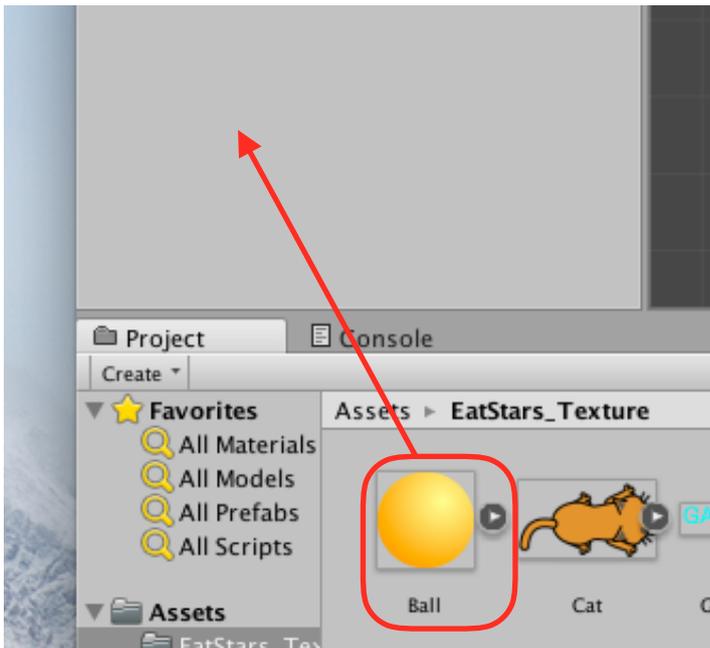


## Ballを出す。

①赤枠で囲ったフォルダをダブルクリックします。

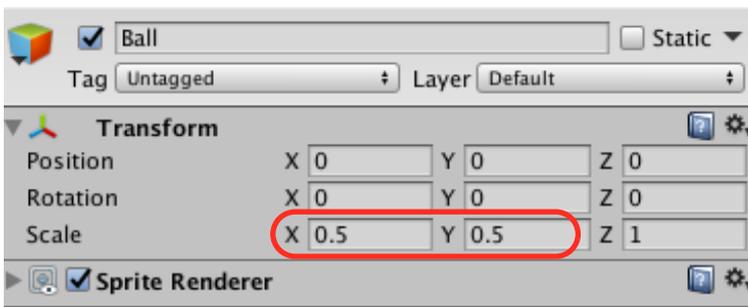


②BallをヒエラルキーにDDします。



③大きさを調整します。

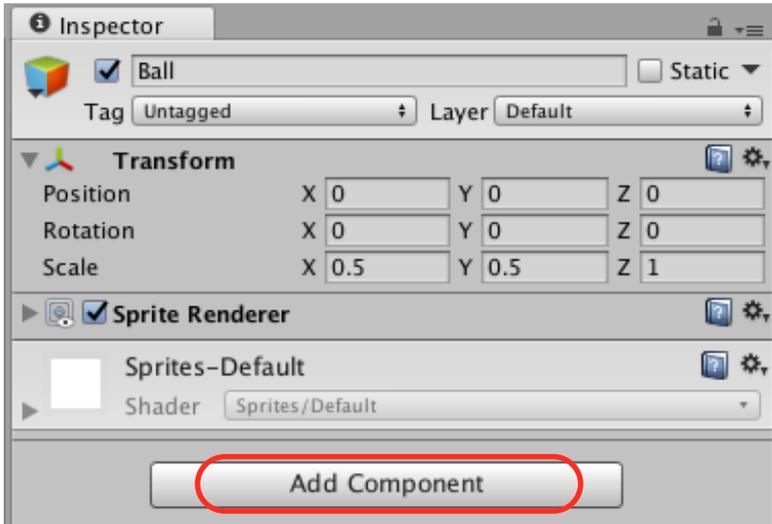
インスペクターで、ScaleのXとYを「0.5」にします。



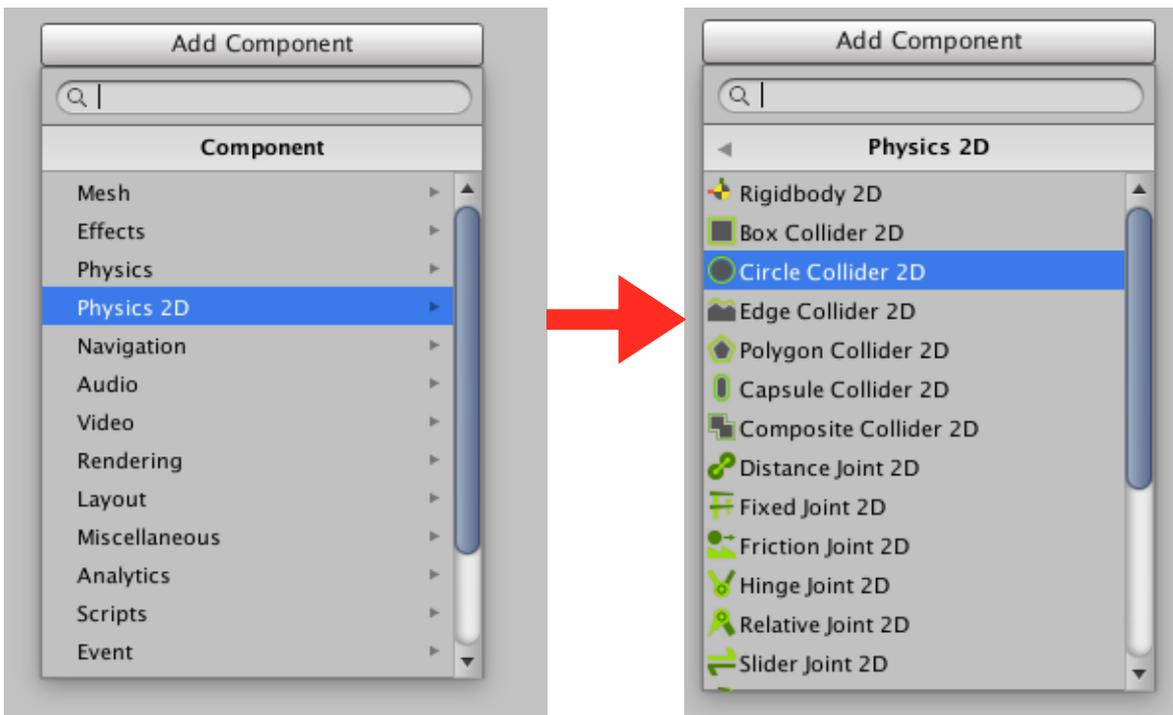
セーブして、実行してみましょう。

## Ballに当たり判定 (Collider) を付ける。

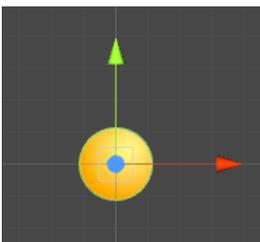
①ヒエラルキーでBallを選び、インスペクターのAdd Componentをクリックします。



②Physics 2D→Circle Collider 2Dの順にクリックします。

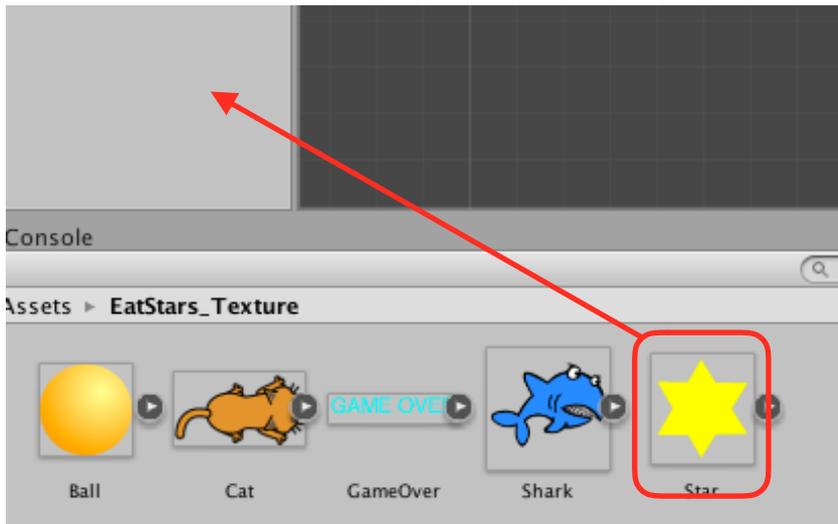


③Ballの周囲に緑色の枠線が現れれば、成功です。

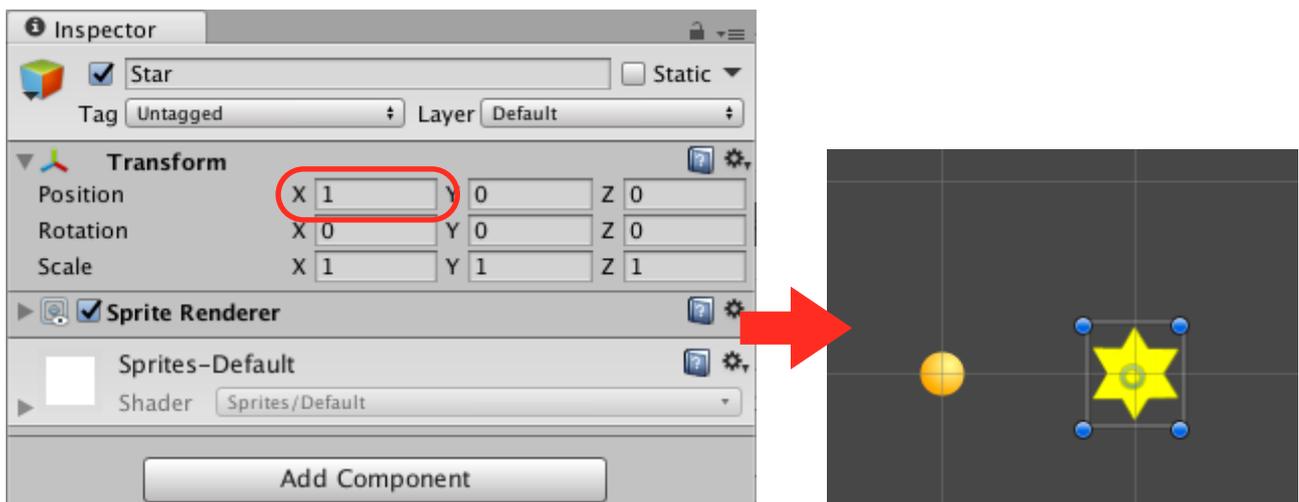


## Starを出す。

①StarをヒエラルキーにDDします。



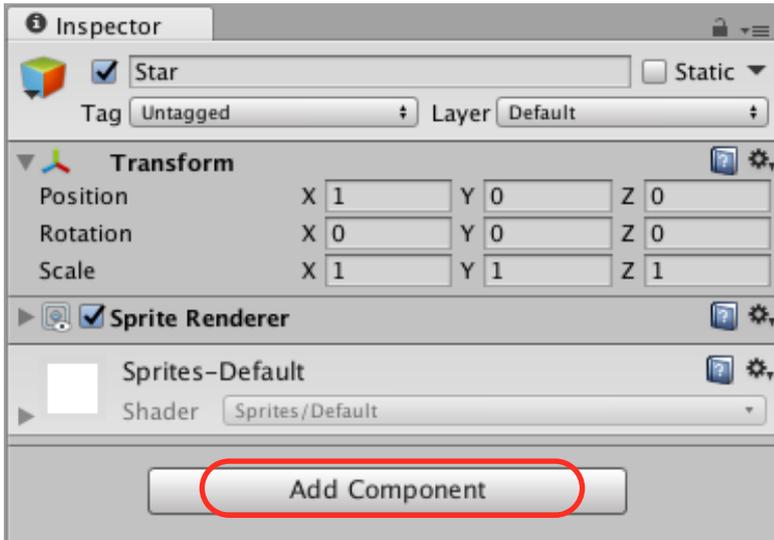
②Ballと重なっているので、横へずらします。



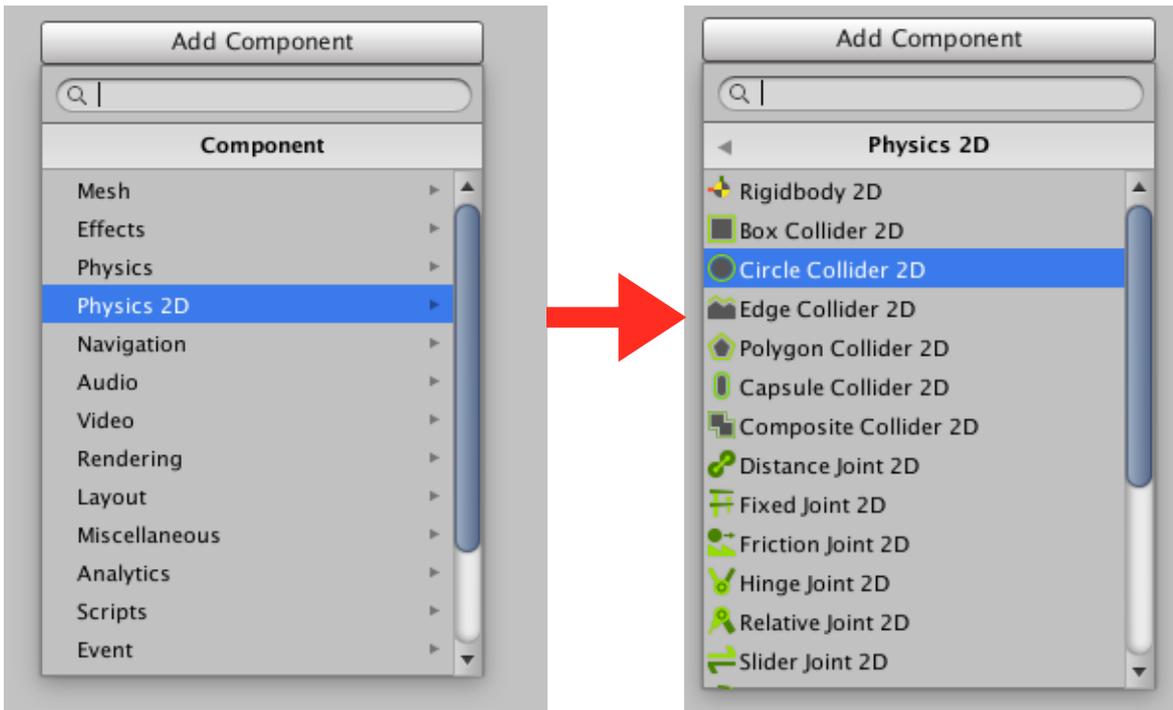
実行してみましょう。

## Starに当たり判定 (Collider) を付ける。

①ヒエラルキーでStarを選び、インスペクターのAdd Componentをクリックします。



②Physics 2D→Circle Collider 2Dの順にクリックします。



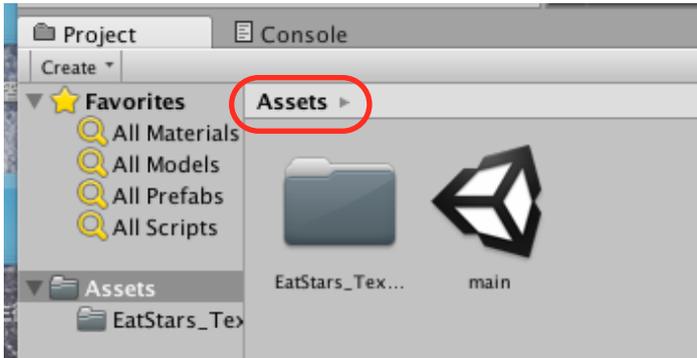
③Starの周囲に緑色の枠線が現れば、成功です。



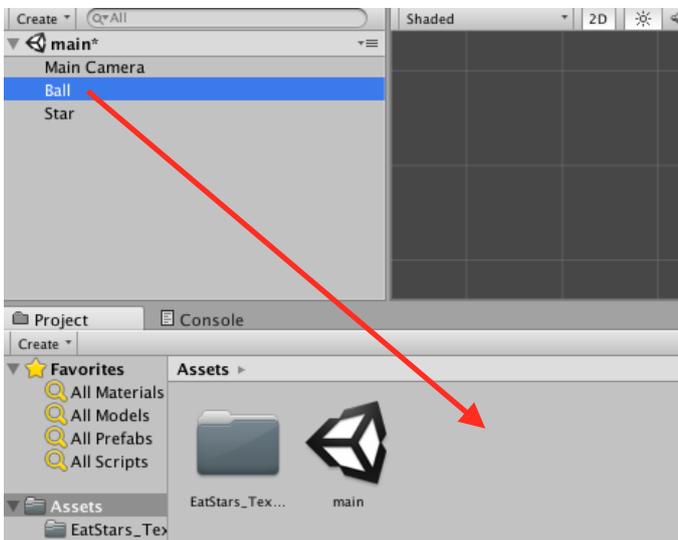
## BallとStarをプレハブ化する。

①Assetsをクリックします。

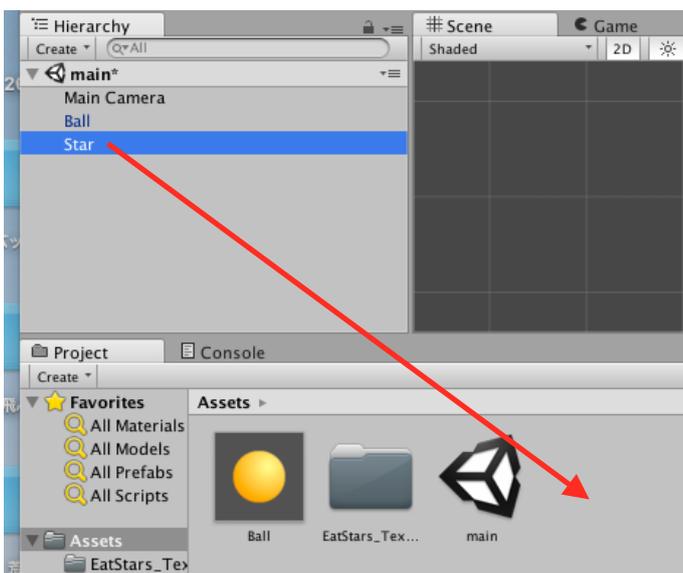
※Assets直下に、プレハブを作るためです。



②BallをAssetsへDDします。

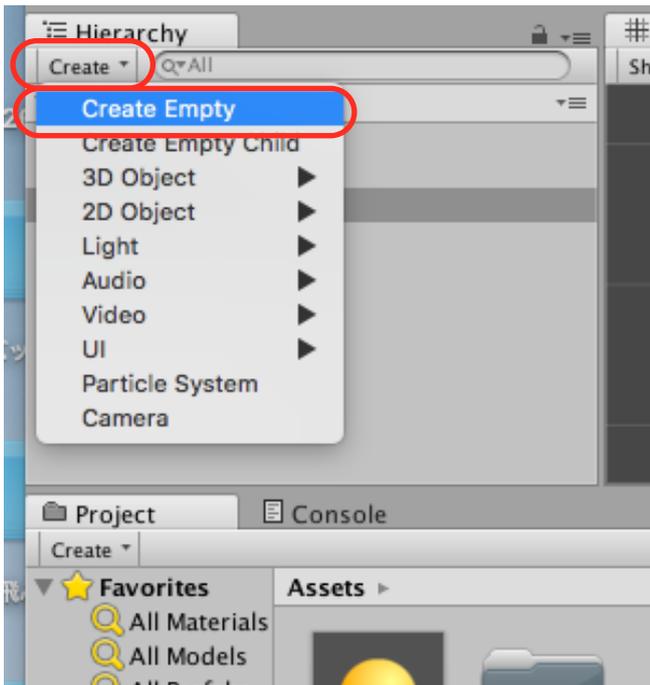


③StarをAssetsへDDします。

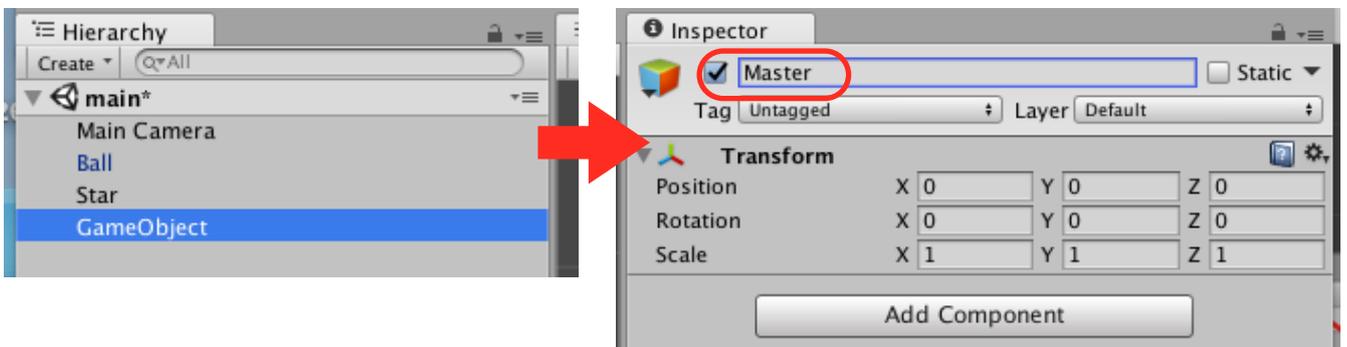


## Masterを作る。

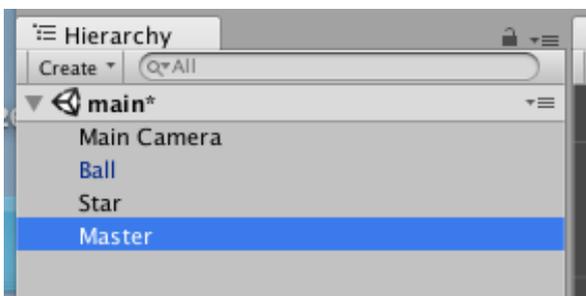
①ヒエラルキーのCreateをクリックし、Create Emptyをクリックします。



②ヒエラルキーのGameObjectを選択し、インスペクターで名前を「Master」にします。  
※名前変更の場合は、エンターキーを最後に押すことを忘れないようにしてください。

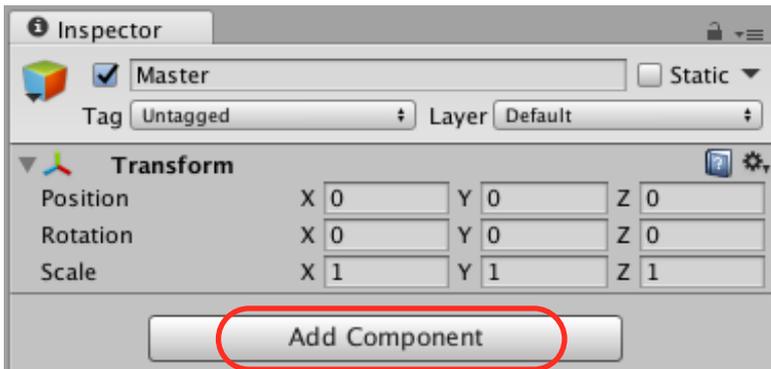


③ヒエラルキーのGameObjectがMasterに変わります。

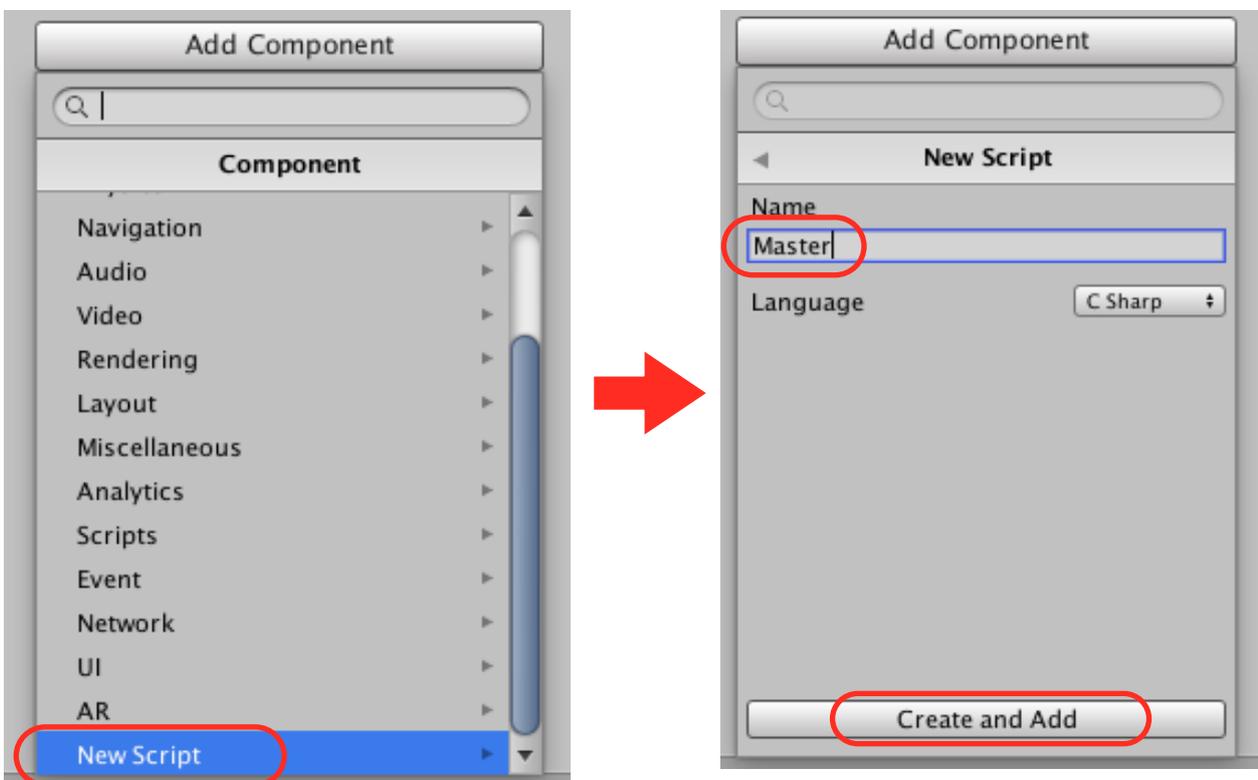


## Masterにスクリプトを付ける。

①ヒエラルキーでMasterを選び、インスペクターのAdd Componentをクリックします。

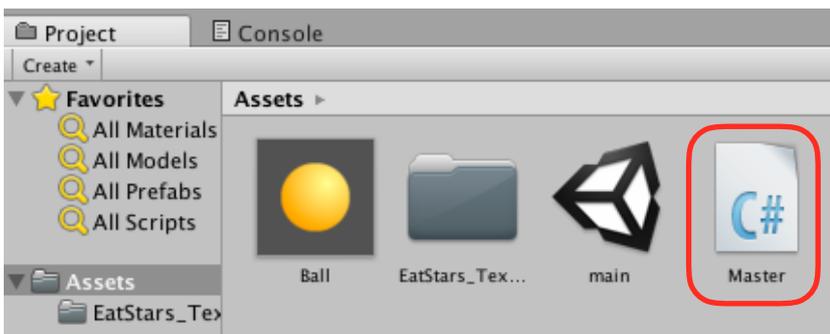


②New Scriptをクリックし、名前を「Master」として、Create and Addをクリックします。



③Assetsの中にC#のファイルが出来ています。これをダブルクリックします。

※MonoDevelopが開くまで、しばらく待ってください。



④赤枠の2行をタイピングします。

これで、プレハブを入れる箱（変数）を作ります。

タイピングしたら、Command+Sでセーブします。

```

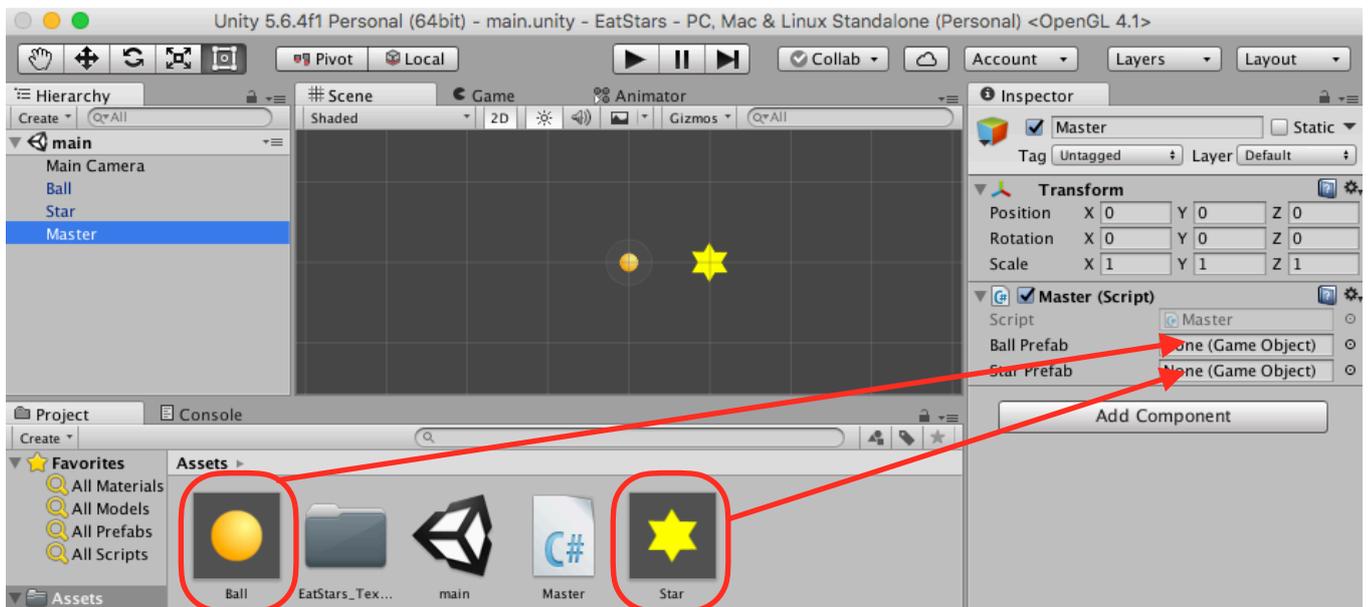
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class Master : MonoBehaviour {
6
7     public GameObject ballPrefab;
8     public GameObject starPrefab;
9
10    // Use this for initialization
11    void Start () {
12
13    }
14
15    // Update is called once per frame
16    void Update () {
17
18    }
19 }

```

⑤Unityの画面に戻って、ヒエラルキーでMasterを選択します。

インスペクターを見ると、Ball PrefabとStar Prefabの枠が出てきます。

⑥プレハブを枠内へDDします。



※これで、Masterスクリプトに、プレハブがセットされました。

## MasterスクリプトでBallを生成する。

①forを使って、40回繰り返す処理を作ります。

```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class Master : MonoBehaviour {
6
7     public GameObject ballPrefab;
8     public GameObject starPrefab;
9
10    // Use this for initialization
11    void Start () {
12        for (int i = 0; i < 40; i++) {
13
14        }
15    }
16
17    // Update is called once per frame
18    void Update () {
19
20    }
21 }
```

②x座標、y座標をランダムに決定するため、x座標を入れる変数「xPos」と、y座標を入れる変数「yPos」を作ります（宣言します）。

```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class Master : MonoBehaviour {
6
7     public GameObject ballPrefab;
8     public GameObject starPrefab;
9
10    float xPos;
11    float yPos;
12
13
14    // Use this for initialization
15    void Start () {
16        for (int i = 0; i < 40; i++) {
17
18        }
19    }
20
21    // Update is called once per frame
22    void Update () {
23
24    }
25 }
```

③乱数を生成して、xPos, yPosに代入します。

※乱数は繰り返し生成したいので、for文の処理の中に書きます。

```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class Master : MonoBehaviour {
6
7     public GameObject ballPrefab;
8     public GameObject starPrefab;
9
10    float xPos;
11    float yPos;
12
13    // Use this for initialization
14    void Start () {
15
16        for (int i = 0; i < 40; i++) {
17
18            xPos = Random.Range (-8f, 8f);
19            yPos = Random.Range (-5f, 5f);
20
21        }
22    }
23
24    // Update is called once per frame
25    void Update () {
26
27    }
28 }
```

④Ballの生成処理を記述します。

```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class Master : MonoBehaviour {
6
7     public GameObject ballPrefab;
8     public GameObject starPrefab;
9
10    float xPos;
11    float yPos;
12
13    // Use this for initialization
14    void Start () {
15
16        for (int i = 0; i < 40; i++) {
17
18            xPos = Random.Range (-8f, 8f);
19            yPos = Random.Range (-5f, 5f);
20
21            Instantiate (ballPrefab, new Vector2 (xPos, yPos), Quaternion.identity);
22
23        }
24    }
25
26    // Update is called once per frame
27    void Update () {
28
29    }
30 }
```

Command+Sでセーブしたら、Unityに戻って、実行してみましょう。

## MasterスクリプトでStarを生成する。

①青枠の部分をコピーして、赤枠部分を作ります。

※やり方がわからないときは、聞いてください。

```

1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class Master : MonoBehaviour {
6
7     public GameObject ballPrefab;
8     public GameObject starPrefab;
9
10    float xPos;
11    float yPos;
12
13    // Use this for initialization
14    void Start () {
15
16        for (int i = 0; i < 40; i++) {
17
18            xPos = Random.Range (-8f, 8f);
19            yPos = Random.Range (-5f, 5f);
20
21            Instantiate (ballPrefab, new Vector2 (xPos, yPos), Quaternion.identity);
22
23        }
24
25        for (int i = 0; i < 40; i++) {
26
27            xPos = Random.Range (-8f, 8f);
28            yPos = Random.Range (-5f, 5f);
29
30            Instantiate (ballPrefab, new Vector2 (xPos, yPos), Quaternion.identity);
31
32        }
33    }
34
35    // Update is called once per frame
36    void Update () {
37
38    }
39 }

```

②赤枠部分の一部を編集します。

```

for (int i = 0; i < 3; i++) {

    xPos = Random.Range (-8f, 8f);
    yPos = Random.Range (-5f, 5f);

    Instantiate (starPrefab, new Vector2 (xPos, yPos), Quaternion.identity);

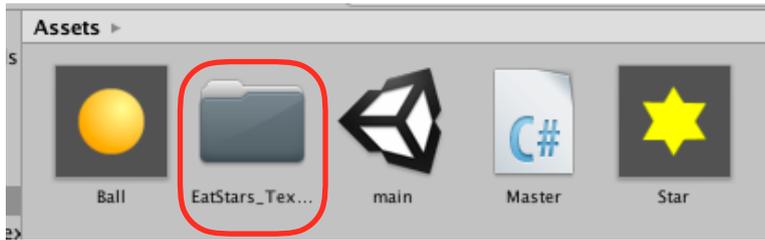
}

```

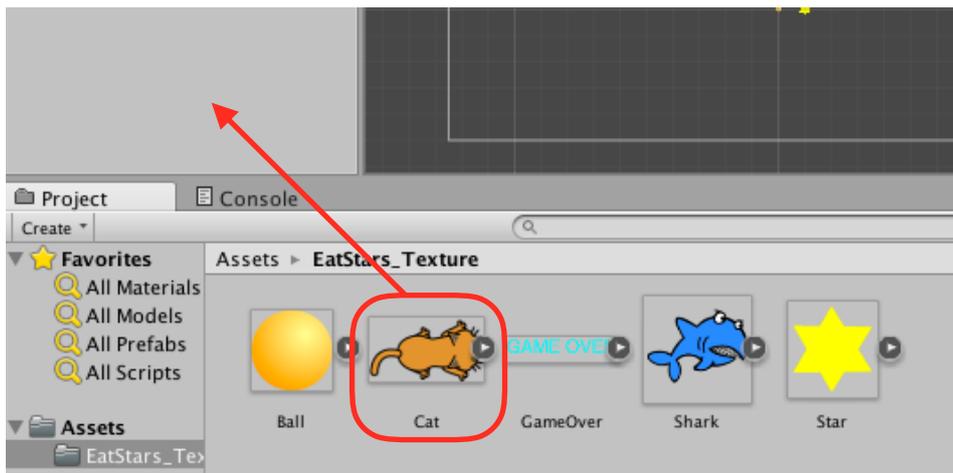
実行してみましょう。

## Catを作る。

①赤枠で囲ったフォルダをダブルクリックします。

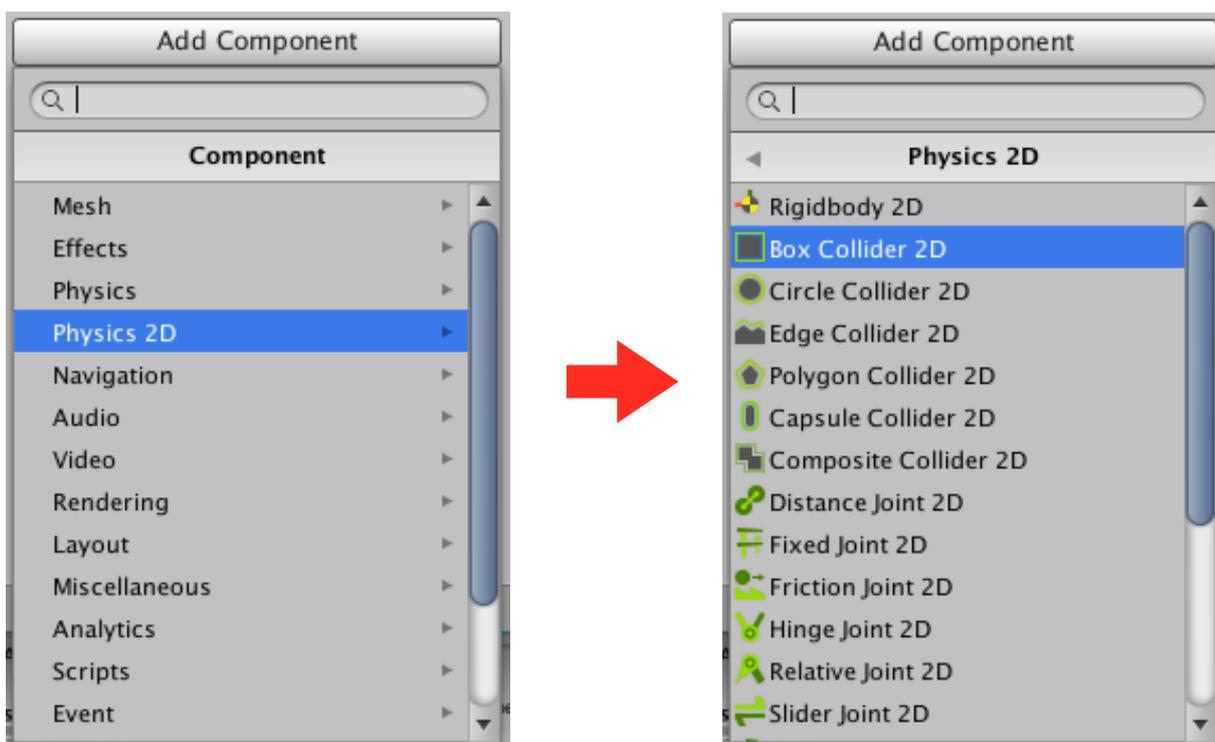


②CatをヒエラルキーへDDします。



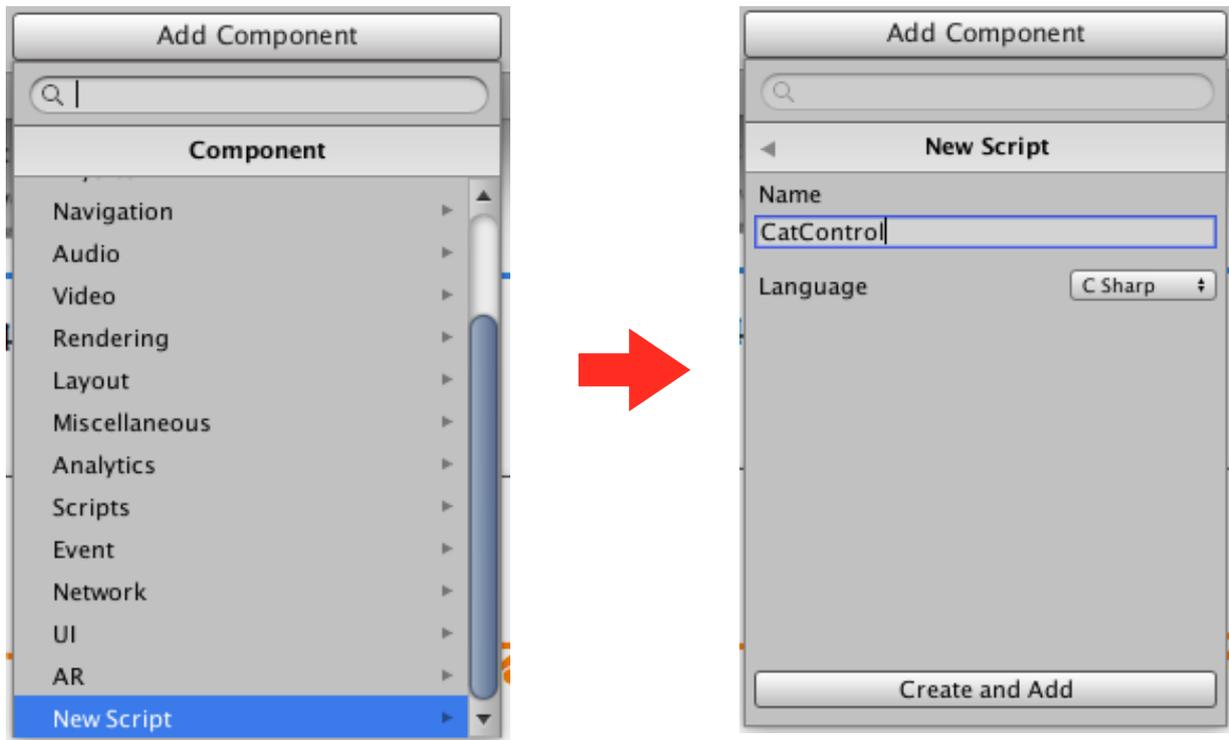
③Catに当たり判定（コライダー）をつけます。

Catを選択した状態で、インスペクターのAdd Component→Physics 2D→Box Collider 2Dの順にクリックします。

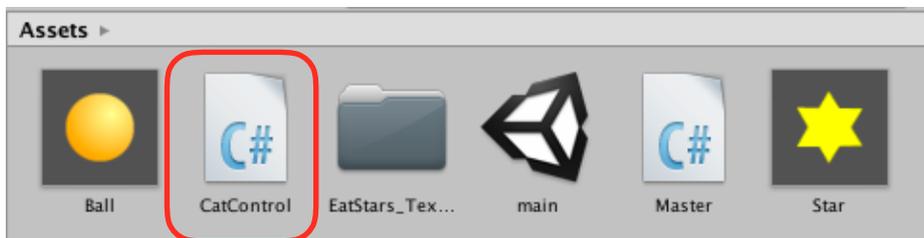


## Catにスクリプトを付ける。

①Catを選択した状態で、インスペクターのAdd Component→New Scriptの順にクリックし、名前を「CatControl」とします。



②Assetsの直下にある、CatControlをダブルクリックして開きます。



## CatControlスクリプトを編集する。

①変数を2つ宣言します。hMoveとvMoveです。

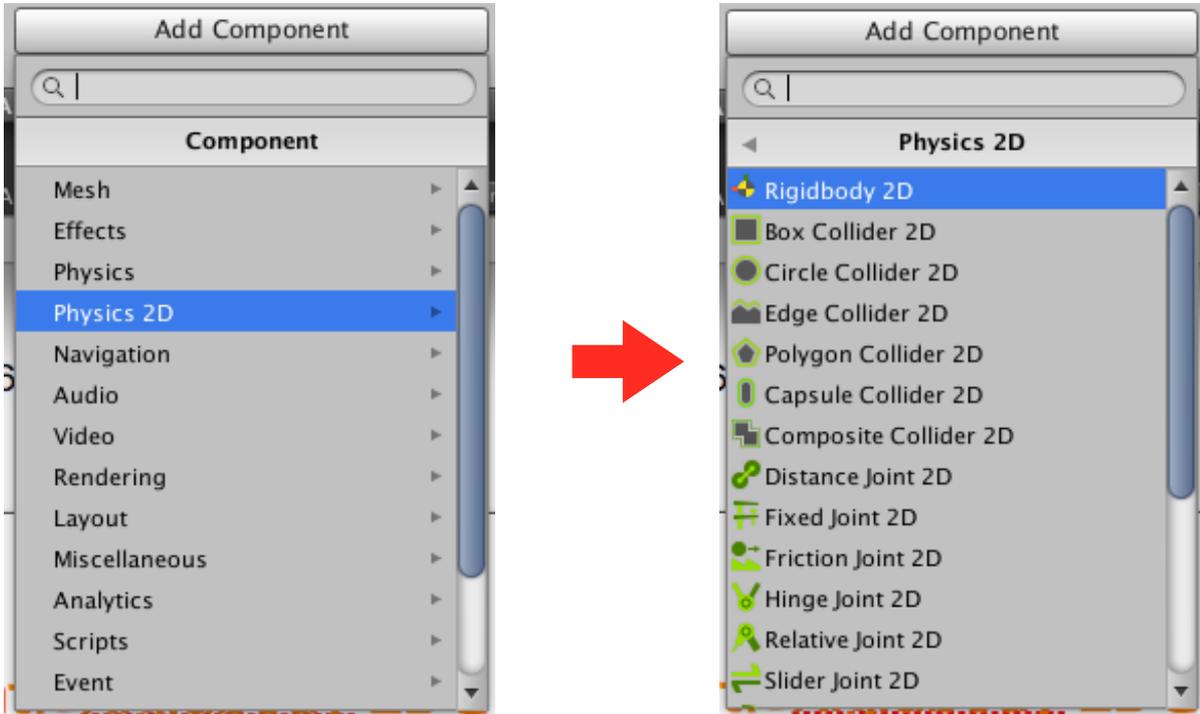
```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class CatControl : MonoBehaviour {
6
7     float hMove;
8     float vMove;
9
10    // Use this for initialization
11    void Start () {
12
13    }
14
15    // Update is called once per frame
16    void Update () {
17
18    }
19 }
```

②左右矢印キーの入力を検出します。

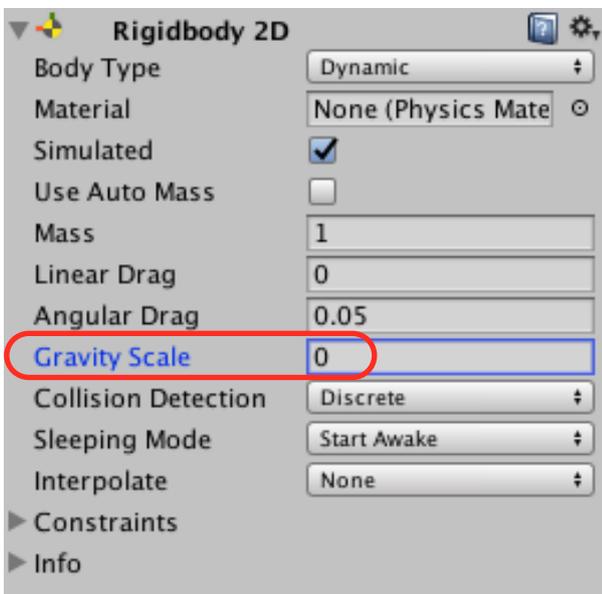
```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class CatControl : MonoBehaviour {
6
7     float hMove;
8     float vMove;
9
10    // Use this for initialization
11    void Start () {
12
13    }
14
15    // Update is called once per frame
16    void Update () {
17
18        hMove = Input.GetAxisRaw ("Horizontal");
19
20    }
21 }
```

## CatにRigidbody 2Dを付ける。

①Catを選択した状態で、Add Component→Physics 2D→Rigidbody 2Dの順にクリックします。



②インスペクターのRigidbody 2DのGravity Scale（重力の大きさ）を「0」にします。



実行してみましょう。

## CatControlスクリプトを編集する（2）。

①Rigidbody2D型の変数「rgb」を作ります。

②float型の変数「speed」を作ります。

```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class CatControl : MonoBehaviour {
6
7     float hMove;
8     float vMove;
9
10    Rigidbody2D rgb;
11    public float speed = 5f;
12
13    // Use this for initialization
14    void Start () {
15
16    }
17
18    // Update is called once per frame
19    void Update () {
20
21        hMove = Input.GetAxisRaw ("Horizontal");
22
23    }
24 }
```

③rgbに、CatのRigidbody 2Dを代入します。

```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class CatControl : MonoBehaviour {
6
7     float hMove;
8     float vMove;
9
10    Rigidbody2D rgb;
11    public float speed = 5f;
12
13    // Use this for initialization
14    void Start () {
15
16        rgb = GetComponent<Rigidbody2D> ();
17
18    }
19
20    // Update is called once per frame
21    void Update () {
22
23        hMove = Input.GetAxisRaw ("Horizontal");
24
25    }
26 }
```

④左右矢印キーの入力に対して、速度を与える。

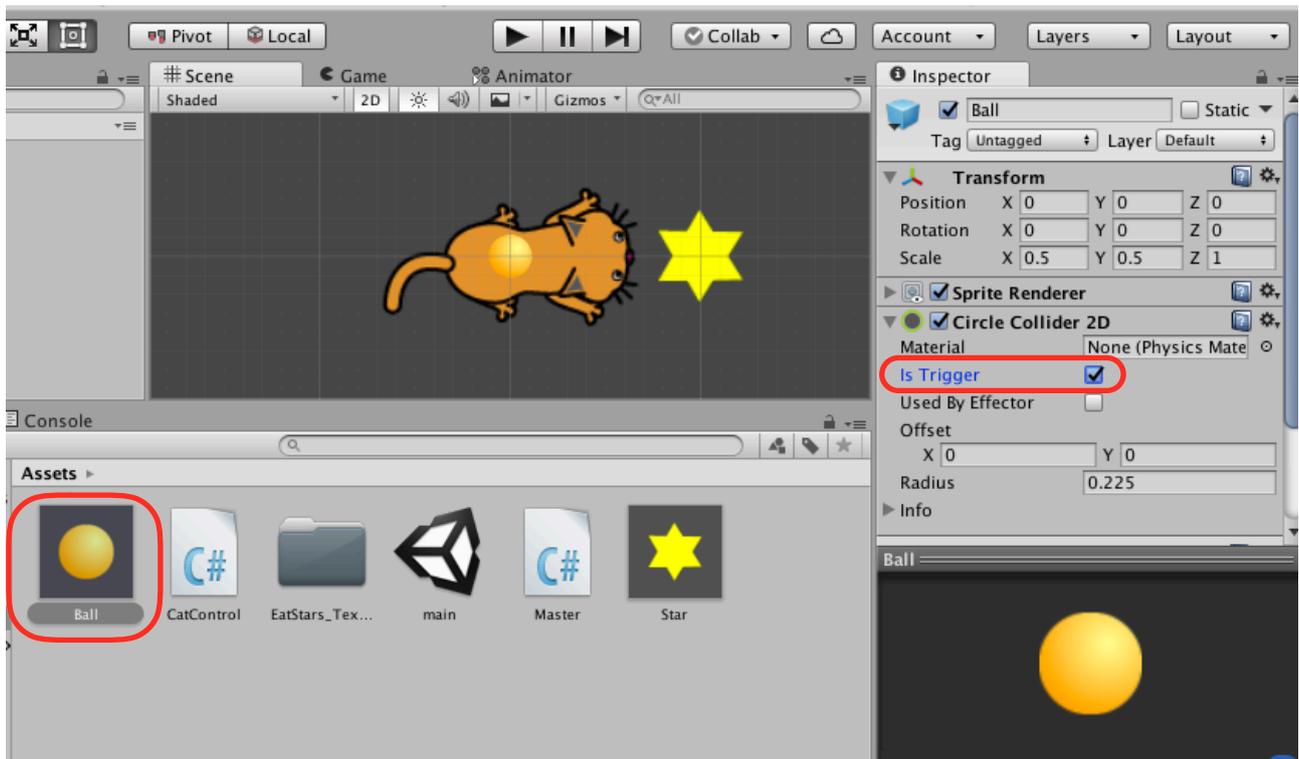
```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class CatControl : MonoBehaviour {
6
7     float hMove;
8     float vMove;
9
10    Rigidbody2D rgb;
11    public float speed = 5f;
12
13    // Use this for initialization
14    void Start () {
15
16        rgb = GetComponent<Rigidbody2D> ();
17
18    }
19
20    // Update is called once per frame
21    void Update () {
22
23        hMove = Input.GetAxisRaw ("Horizontal");
24
25        rgb.velocity = new Vector2 (hMove * speed, 0f);
26
27    }
28 }
```

では、実行してみましょう。

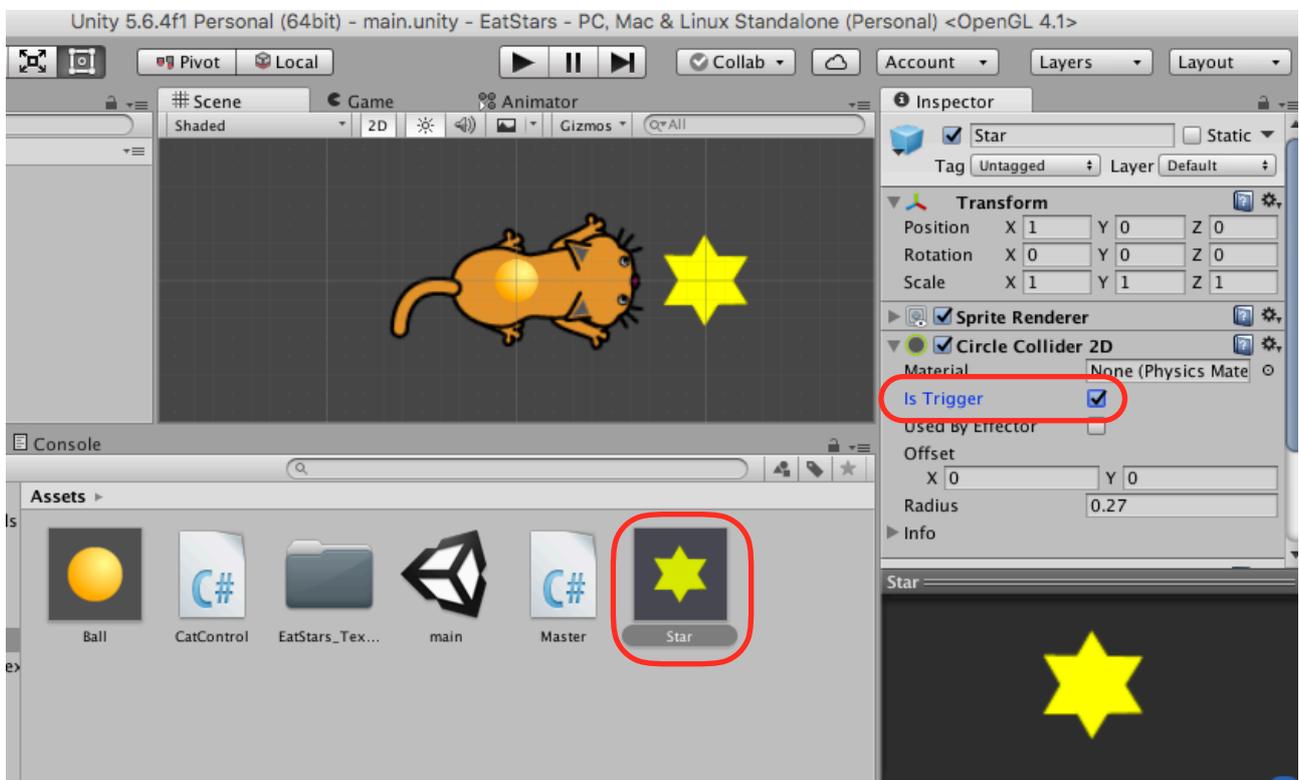
左右キーを押して、Catが動くかを確認してみます。

## 当たり判定が重なるようにする。

①Assets直下のBallプレハブを選択して、Is Triggerをチェックする。



②Assets直下のStarプレハブを選択して、Is Triggerをチェックする。

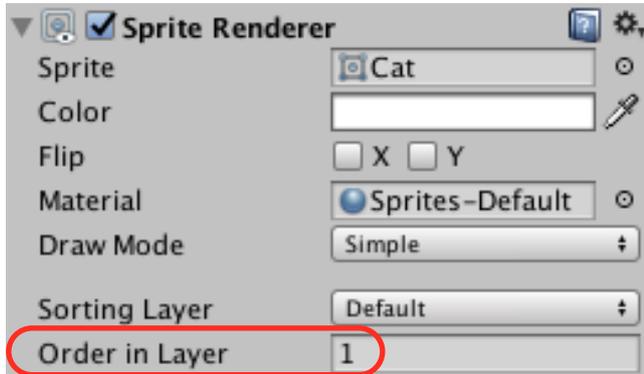


実行してみましょう。

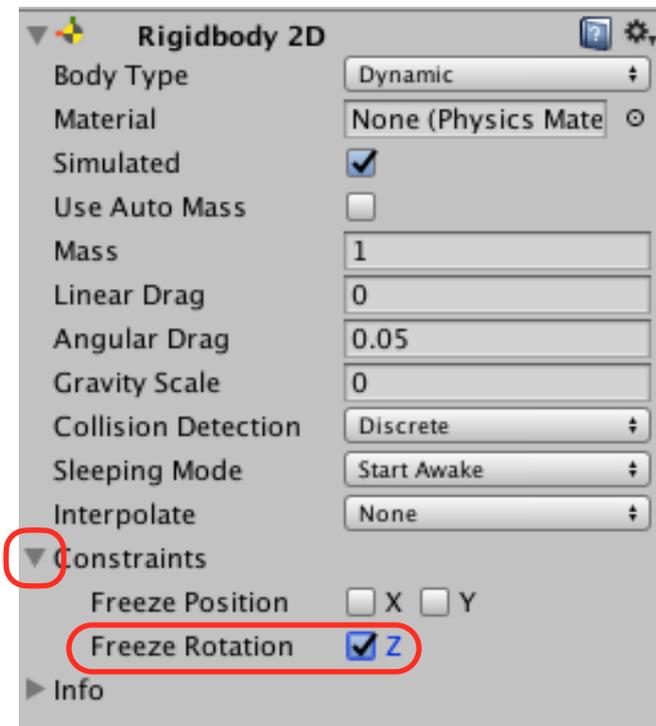
BallとStarをすり抜けるようになれば、成功です。

## Catの細かな調整をする。

①ヒエラルキーでCatを選択し、インスペクターのSprite RendererのOrder in Layerを「1」にする。



②ヒエラルキーでCatを選択し、インスペクターのRigidbody 2DのConstraintsの三角マークをクリックし、Freeze Rotation Zをチェックする。



実行してみましょう。  
左右に動きましたか？

## CatControlスクリプトを編集する（3）。

①赤線部分を追加・編集します。

```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class CatControl : MonoBehaviour {
6
7     float hMove;
8     float vMove;
9
10    Rigidbody2D rgb;
11    public float speed = 5f;
12
13    // Use this for initialization
14    void Start () {
15
16        rgb = GetComponent<Rigidbody2D> ();
17
18    }
19
20    // Update is called once per frame
21    void Update () {
22
23        hMove = Input.GetAxisRaw ("Horizontal");
24        vMove = Input.GetAxisRaw ("Vertical");
25
26        rgb.velocity = new Vector2 (hMove * speed, vMove * speed);
27
28    }
29 }
```

実行してみましょう。

上下にも動きましたか？

②赤枠部分を追加します。

```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class CatControl : MonoBehaviour {
6
7     float hMove;
8     float vMove;
9
10    Rigidbody2D rgb;
11    public float speed = 5f;
12
13    // Use this for initialization
14    void Start () {
15
16        rgb = GetComponent<Rigidbody2D> ();
17
18    }
19
20    // Update is called once per frame
21    void Update () {
22
23        hMove = Input.GetAxisRaw ("Horizontal");
24        vMove = Input.GetAxisRaw ("Vertical");
25
26        rgb.velocity = new Vector2 (hMove * speed, vMove * speed);
27
28        if (hMove > 0f) {
29            transform.eulerAngles = new Vector3 (0f, 0f, 0f);
30        } else if (hMove < 0f) {
31            transform.eulerAngles = new Vector3 (0f, 0f, 180f);
32        } else if (vMove > 0f) {
33            transform.eulerAngles = new Vector3 (0f, 0f, 90f);
34        } else if (vMove < 0f) {
35            transform.eulerAngles = new Vector3 (0f, 0f, 270f);
36        }
37    }
38 }
```

実行してみましょう。

Catの向きが変わりましたか？

## BallとStarが消えるようにする。

```

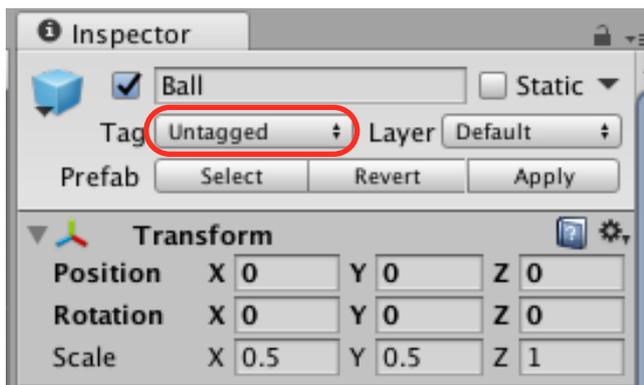
19
20 // Update is called once per frame
21 void Update () {
22
23     hMove = Input.GetAxisRaw ("Horizontal");
24     vMove = Input.GetAxisRaw ("Vertical");
25
26     rgb.velocity = new Vector2 (hMove * speed, vMove * speed);
27
28     if (hMove > 0f) {
29         transform.eulerAngles = new Vector3 (0f, 0f, 0f);
30     } else if (hMove < 0f) {
31         transform.eulerAngles = new Vector3 (0f, 0f, 180f);
32     } else if (vMove > 0f) {
33         transform.eulerAngles = new Vector3 (0f, 0f, 90f);
34     } else if (vMove < 0f) {
35         transform.eulerAngles = new Vector3 (0f, 0f, 270f);
36     }
37 }
38
39 void OnTriggerEnter2D (Collider2D col) {
40     Destroy (col.gameObject);
41 }
42 }

```

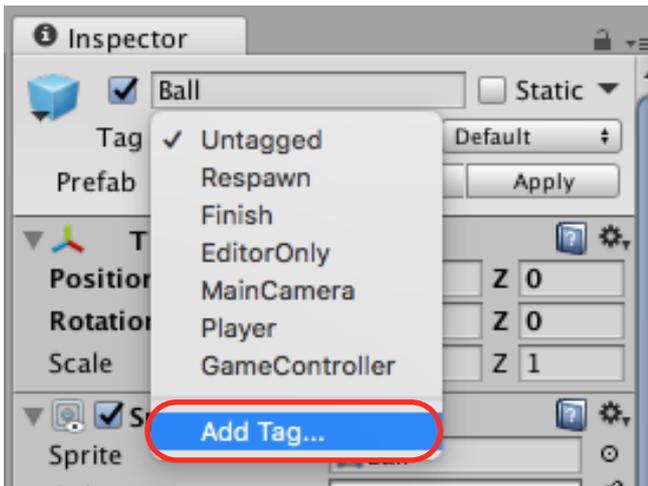
実行してみましょう。上手く消えましたか？

## BallとStarにTagを付ける。

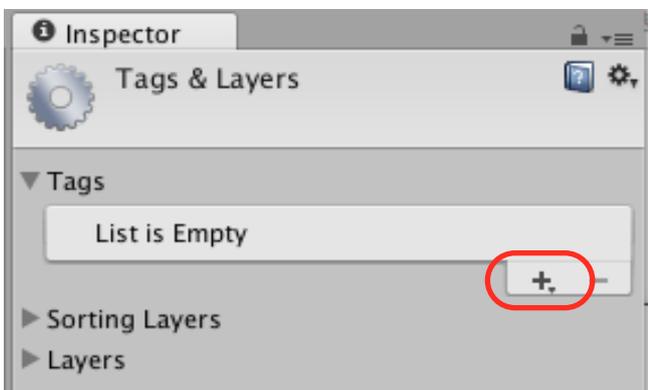
①Tag (名札) を作ります。適当なGameObjectを選択してUntaggedをクリックします。



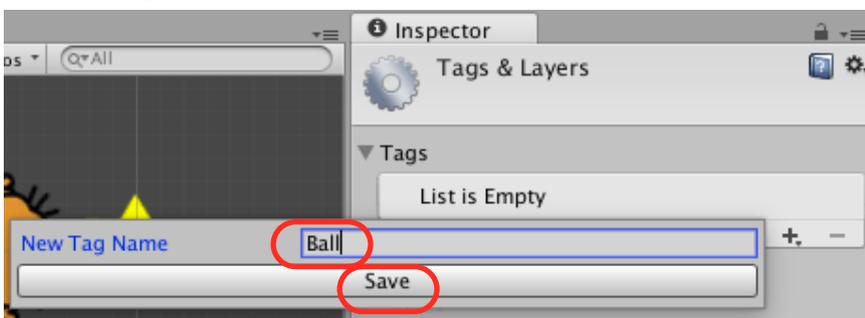
②Add Tagを選択します。



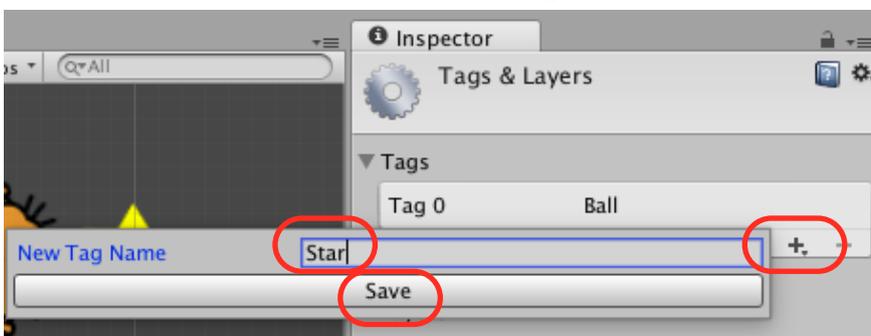
③+をクリックします。



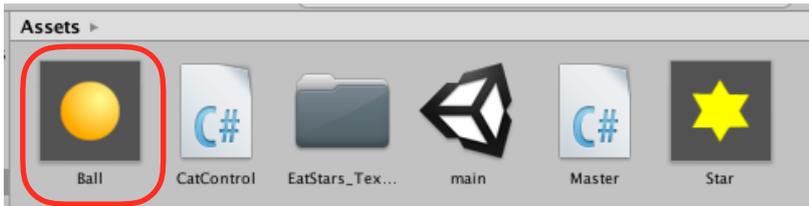
④New Tag Nameを「Ball」とし、Saveをクリックします。



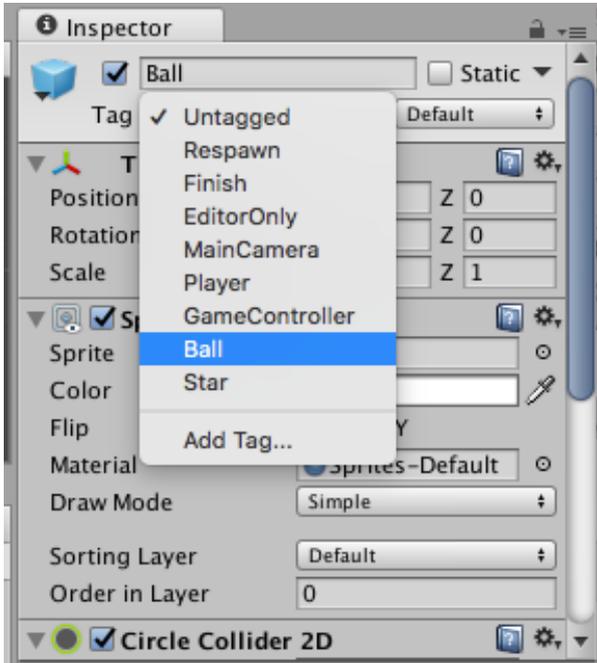
⑤同じように、+をクリックし、New Tag Nameを「Star」とし、Saveをクリックします。



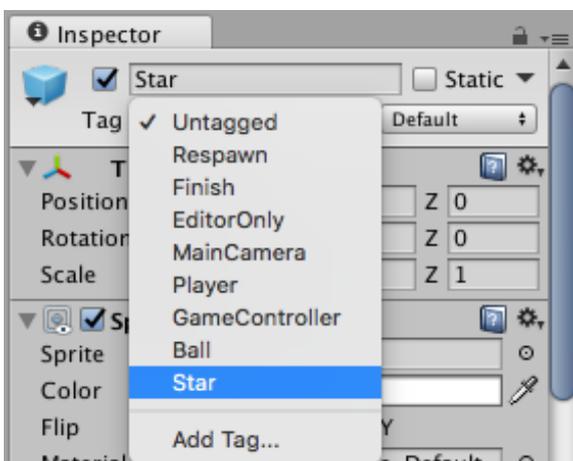
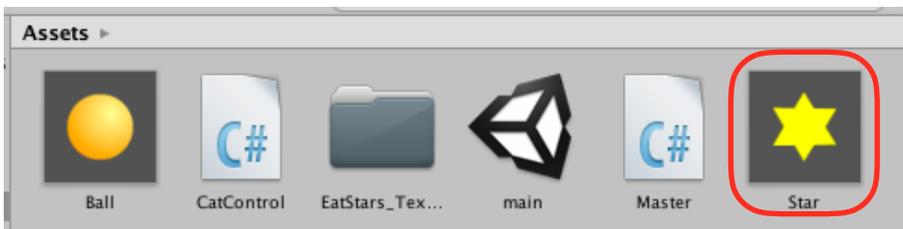
⑥プレハブ化したBallをクリックします。



⑦インスペクターで、Untaggedをクリックし、Ballを選択します。



⑧Starにも同様にTagを付けます。



## Catのスク립トを編集する。

OnTriggerEnter2Dの中身を下のよう書き換えます。

```
void OnTriggerEnter2D (Collider2D col) {  
    if (col.tag == "Ball") {  
        Destroy (col.gameObject);  
    }  
    if (col.tag == "Star") {  
        Destroy (col.gameObject);  
    }  
}
```

続く。。。