

# C#

```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4 
```

大文字→スクリプトのファイル名と同じ

```
5 public class Test : MonoBehaviour
6 {
7     private float speed = 1.0f;
8     private int number = 1;
9     private string textChangeName = "Sample Text";
10 }
```

```
11 //Start is called before the first frame update
```

```
12 void Start( )
13 {
14     number = (int)speed;
15     Debug.Log(textChangeName);
16 }
```

```
17 //Update is called once per frame
```

```
18 void Update( )
19 {
20 }
21 }
22 }
```

```
23
24 void TextChangeName
25 {
26 }
27 }
```

```
28 }
```

変数と関数でできている

変数は空箱

型は箱の大きさ

権限 型 変数

```
private float speed = 1.0f;
       int number = 1;
       string testChangeName = "Sample Text";
```

× number = speed;

○ number = (int)speed;

testChangeName

変数の頭は小文字

TestChangeName

関数の頭は大文字

```
12 void Start( )
13 {
14     number = (int)speed;
15     Debug.Log(textChangeName);
16
17     number = ChangeSpeed(1);
18 }
19
20 //Update is called once per frame
21 void Update( )
22 {
23     返回值なし
24 }
25
26 void TextChangeName
27 {
28
29 }
30
31 private int ChangeSpeed (int plusNumber )
32 {
33     int l = 1;
34
35     l = l + plusNumber
36     return l;
37 }
38 }
```

引数

返回值:この関数に変数に返す値

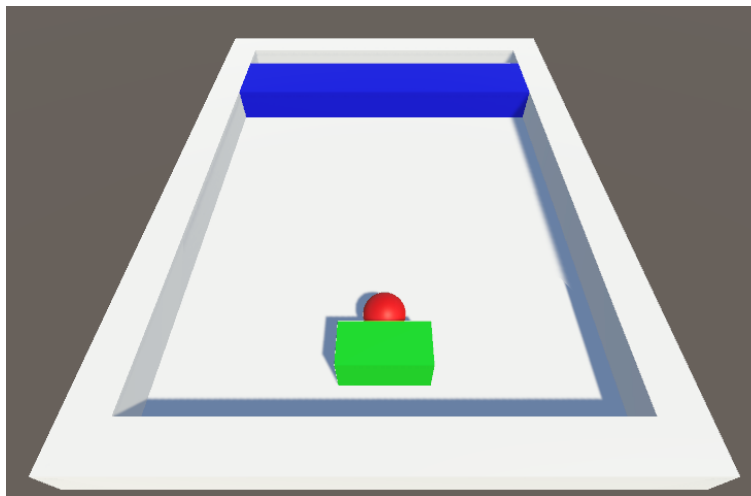
←intを書くと値をint型のデータを返す必要がある  
返す変数の型

←

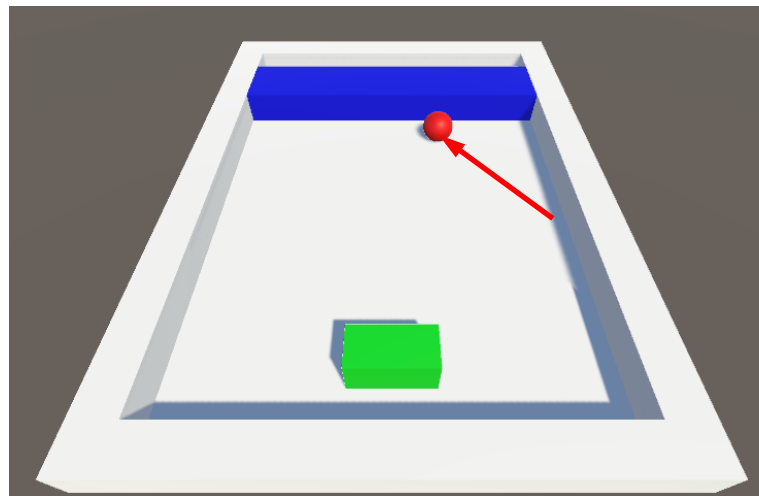
←returnで関数を終了して返す

スクリプトを「保存」することでunityに反映する

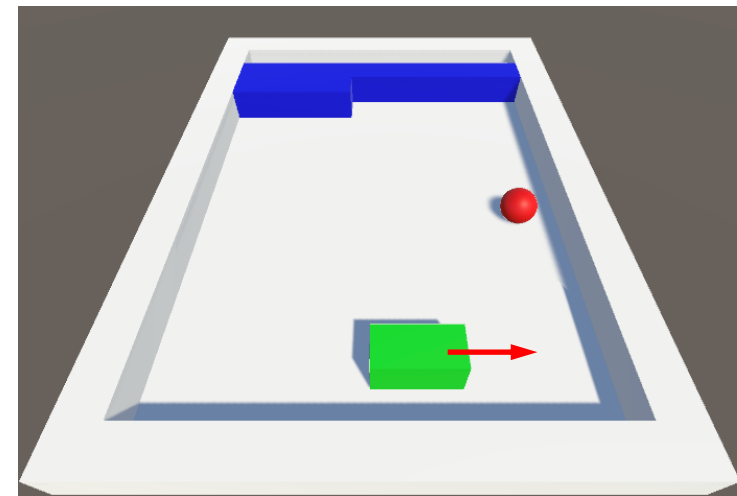
Ball



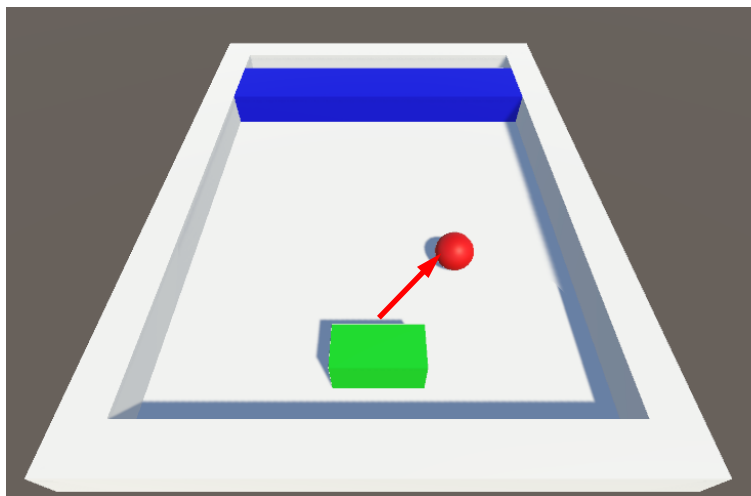
Block



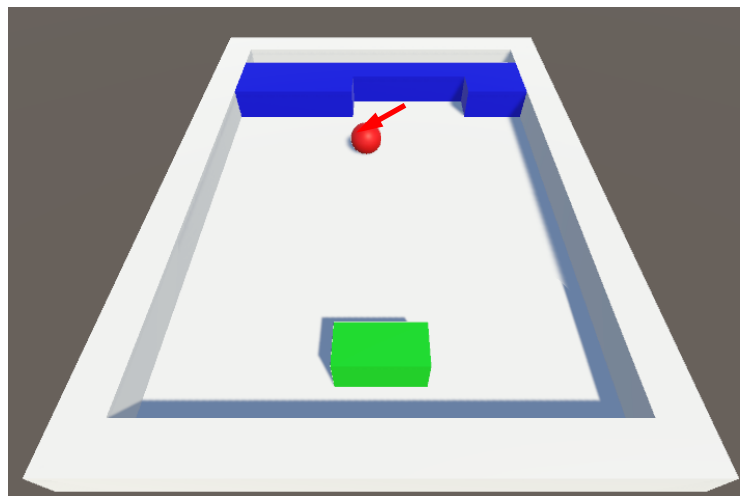
Player



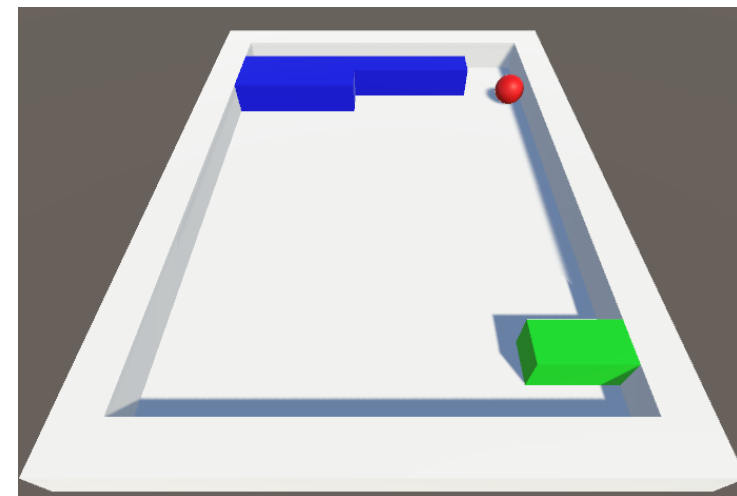
右斜め前に力を加える



ボールが当たったらブロックを消す



左右の矢印キーで動かす



## C#ファイル作成した際のデフォルト

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
```

編集画面で編集可能

```
public class Test : MonoBehaviour
{
```

C#のファイル名 ← 変数

```
    // Start is called before the first frame
```

```
update
```

```
void Start()
{
}
```

関数

← 起動直後1回だけ、{ }に挟まれたコードを実施

```
    // Update is called once per frame
```

```
void Update()
{
}
```

関数

← フレーム毎に{ }に挟まれたコードを実施

```
this.transform.position += Vector3.left * speed * Time.deltaTime;
```

playerの位置にスピードとフレーム間の時間を掛け分左に位置が移動する

## Playerのスク립ト

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
```

アクセスに制限がない

Playerの移動スピード

実数型 変数

```
public float speed = 1.0f;
```

speedという変数の箱に実数1.0を代入

```
public float speed = 1.0f;
```

```
    // Start is called before the first frame update
```

```
void Start()
{
```

```
}
```

if(Input.GetKey(KeyCode.LeftArrow))  
左矢印のキーコードが押されたならば

```
    // Update is called once per frame
```

```
void Update()
{
```

```
    if(Input.GetKey(KeyCode.LeftArrow))
```

```
    {
```

```
        if(this.transform.position.x > -4)
```

```
            this.transform.position += Vector3.left * speed * Time.deltaTime;
```

```
        }
```

```
    }
```

if(this.transform.position.x > -4)  
playerの位置xが-4より大きければ  
壁より内側にあれば

```
    if(Input.GetKey(KeyCode.RightArrow))
```

```
    {
```

```
        if(this.transform.position.x < 4)
```

```
            this.transform.position += Vector3.right * speed * Time.deltaTime;
```

```
        }
```

```
    }
```

左矢印キー

右矢印キー

# Ballのスク립ト

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
```

```
public class Ball : MonoBehaviour
{
    public float speed = 1.0f;
    private Rigidbody myRigid;
    // Start is called before the first frame update
    void Start()
    {
        myRigid = this.GetComponent<Rigidbody>();
        myRigid.AddForce((transform.forward + transform.right) * speed, ForceMode.VelocityChange);
    }
    // Update is called once per frame
    void Update()
    {
    }
}
```

Ballの中だけで使える

Ball

Ballのコンポーネント(例えばtransformなど)内のRigidbodyの情報をとってきて、myRigidに代入する

力のモード

力を加える

ベクトル(前方向と右方向つまり右斜め前方向に1のスピードの力)

# Blockのスク립ト

```
using System.Collections;  
using System.Collections.Generic;  
using UnityEngine;
```

```
public class Block : MonoBehaviour  
{  
    // Start is called before the first frame update  
    void Start()  
    {  
  
    }  
  
    // Update is called once per frame  
    void Update()  
    {  
  
    }  
    private void OnCollisionEnter(Collision collision)  
    {  
        Destroy(this.gameObject);  
    }  
}
```

衝突した時に以下のコードを実行する

衝突した時にBlockを消す