

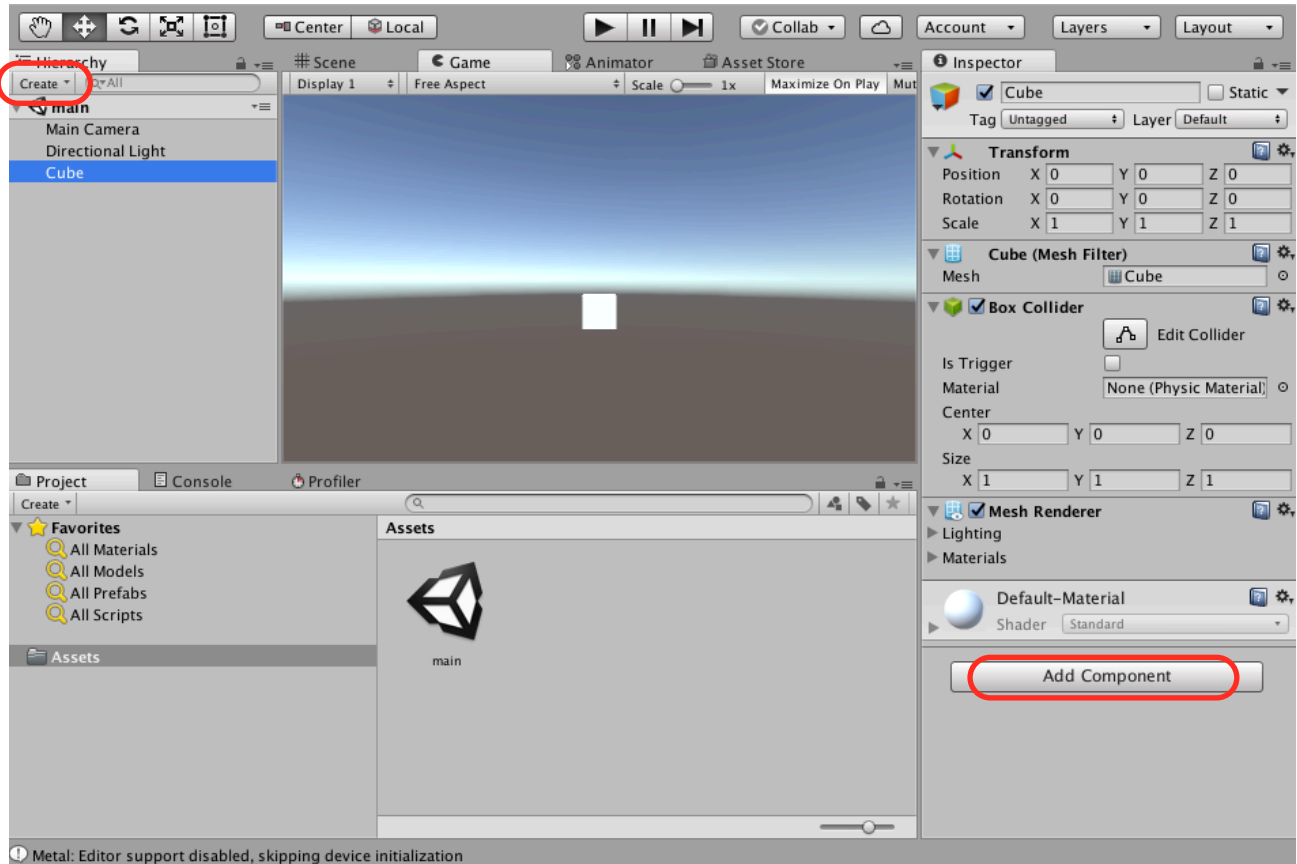
# スクリプト初めの一歩

---

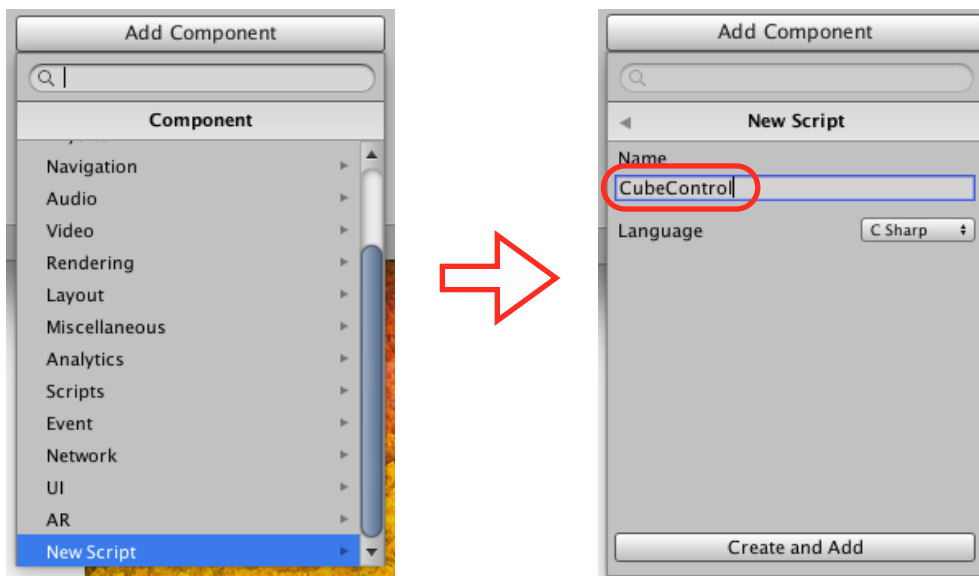
```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class CubeControl : MonoBehaviour {
6
7     // Use this for initialization
8     void Start () {
9
10    }
11
12    // Update is called once per frame
13    void Update () {
14
15    }
16 }
```

## Cubeにスクリプトをつける。

- ①ヒエラルキーのCreateからCubeを作ります。
- ②インスペクターのAdd Componentをクリックします。



- ③New Scriptをクリックします。
- ④名前を「CubeControl」として、Create and Addをクリックします。



## スクリプトを開く。

Assetsの中のCubeControlをダブルクリックして開きます。  
赤枠で示すように、ファイルに付けた名前がクラス名になります。  
※ファイル名を変更したらクラス名も変更しないといけません。

```

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

```

## Start関数とUpdate関数

```

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

```

Start関数は、実行時に、1回だけ呼ばれます。  
Update関数は、実行すると、繰り返し呼ばれます。

## ヒント

スクラッチでいうと、下の様なブロックが最初から置いてある状態です。



## Start関数とUpdate関数の処理を書く場所

```

1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class CubeControl : MonoBehaviour {
6
7     // Use this for initialization
8     void Start () {
9                  ← ここへ書く
10    }
11
12    // Update is called once per frame
13    void Update () {
14                 ← ここへ書く
15    }
16 }

```

中括弧{ }の中へ書きます。

スクラッチのブロックで考えると、上の図のようになります。

## 代入式

```

1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class CubeControl : MonoBehaviour {
6
7     int a;
8     int b; ← 変数の宣言 (変数を作る)
9     int c;
10
11    // Use this for initialization
12    void Start () {
13        a = 1;
14        b = 2;
15        c = a + b; ← 代入式
16    }
17
18    // Update is called once per frame
19    void Update () {
20
21    }
22 }

```

a = 1は、aに1を入れるという意味です。

## 変数のいろいろ

intは整数型です。

これに対し、小数点が付く数字を入れるには、float（小数）型にします。

```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class CubeControl : MonoBehaviour {
6
7     float a;
8     float b; ← floatは、小数型
9     float c;
10
11     // Use this for initialization
12     void Start () {
13         a = 1f;
14         b = 2.5f; ← 数字の後ろに
15         c = a + b;   f を付けます。
16     }
17
18     // Update is called once per frame
19     void Update () {
20
21     }
22 }
```

stringは、文字列型です。

```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class CubeControl : MonoBehaviour {
6
7     string a;
8     string b; ← stringは、文字列型
9     string c;
10
11     // Use this for initialization
12     void Start () {
13         a = "Usami";
14         b = "Osamu"; ← 文字列は、”(ダブル
15         c = a + b;   クォーテーション)で
16     }                囲みます。
17
18     // Update is called once per frame
19     void Update () {
20
21     }
22 }
```

## if文

if文では、条件式が成立したときに、中括弧の中の処理を実行します。

```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class CubeControl : MonoBehaviour {
6
7     int a;
8     int b;
9     int c;
10
11     // Use this for initialization
12     void Start () {
13         a = 1;
14         b = 1;
15         c = 0;
16     }
17
18     // Update is called once per frame
19     void Update () {
20
21         if (条件式) {
22
23         }
24
25     }
26 }
```

## ヒント

スクラッチでいうと下のようになっています。



## 条件式と処理を記述する。

```

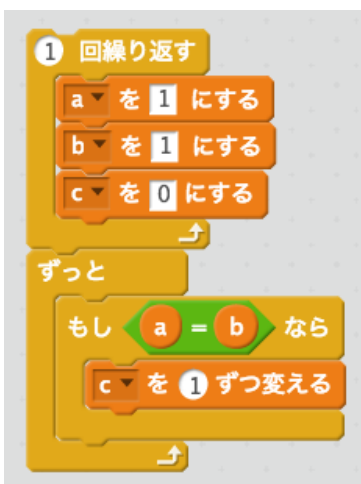
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class CubeControl : MonoBehaviour {
6
7     int a;
8     int b;
9     int c;
10
11     // Use this for initialization
12     void Start () {
13         a = 1;
14         b = 1;
15         c = 0;
16     }
17
18     // Update is called once per frame
19     void Update () {
20
21         if (a == b) {
22             c = c + 1;
23         }
24
25     }
26 }

```

変数aと変数bが等しいときに、cに1を足します。  
Update関数の中にあるので、変数cは、どんどん増えていきます。

## ヒント

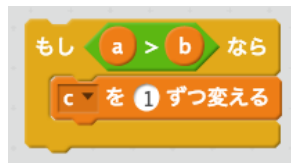
スクラッチでいうと下のようになっています。



## 条件式のいろいろ

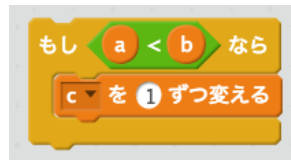
①aがbよりも大きい

```
if (a > b) {
    c = c + 1;
}
```



②aがbよりも小さい

```
if (a < b) {
    c = c + 1;
}
```



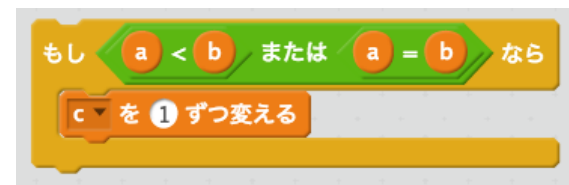
③aがb以上である

```
if (a >= b) {
    c = c + 1;
}
```



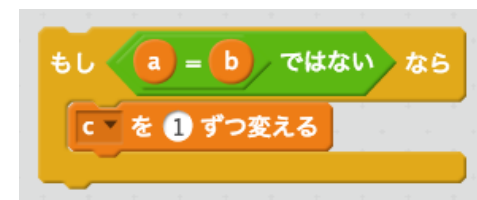
④aがb以下である

```
if (a <= b) {
    c = c + 1;
}
```



⑤aとbが等しくない

```
if (a != b) {
    c = c + 1;
}
```



※ $c = c + 1$ は、通常、 $c++$ で記述します。

## for文

for文は、中括弧の中の処理を繰り返します。

```
for (int i = 0; i < 5; i++) {
    c = c + 1;
}
```

※小括弧の中の書き方は覚えます。5の部分が繰り返し回数です。

## ヒント

スクラッチでいうと下のようになっています。

