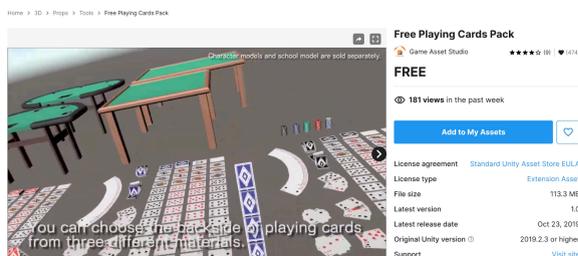
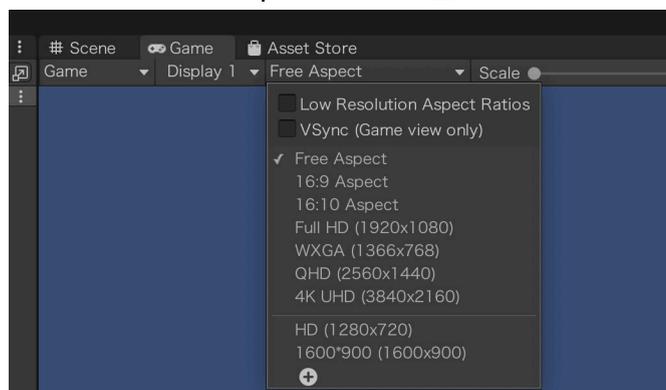


UD01 神経衰弱1(動画解説有り)

1. 【よむ】スクラッチでも作った神経衰弱をUnityで作っていきます。いきなり52枚のトランプを使うのは大変なので、まずは6枚で作って行きます。
2. 新規でプロジェクトを作ります。今回は2Dです。「2Dコア」で「SinkeiSuijyaku」というプロジェクトを新規作成してください。
3. 無料のトランプアセットをインポートします。
 - a. 「Free Playing Cards Pack」をダウンロード & インポート



4. 画面サイズを1600*900に設定します。
5. GameタブのFreeAspect → + を押して、1600*900を作成し、それを選択する。



6. Mainカメラのインスペクターの Sizeを450にします。(縦の解像度(=ピクセル数) ÷ 2)
7. トランプをSpriteにするための準備を行います。
 - a. Assets → Playing Cards → Image → PlayingCards内のイメージを全選択(CTRL+A)する
 - b. TextureType をDefaultから、「Sprite(2D and UI)」にする
 - c. Pixels Per Unit を 100 から1にする。(1ピクセル移動するためには1を指定すればよくなる)
 - d. しばらく待って、ダイアログが表示されるので、「Apply」を押す。
8. クラブの1-3とハートの1-3をヒエラルキーにD&D
9. カードが大きすぎるので、スケールを0.01位にする
10. 重なっているのを適当に移動させよう。

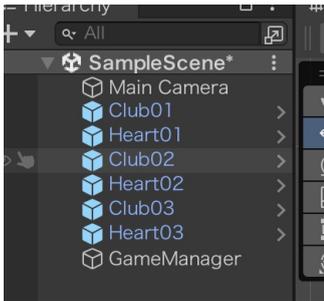


11. クリックしたら消えるようにします。(ペアとか関係なく、とりあえず無条件に消える)

- a. GameManagerという空のゲームオブジェクトを作って「Create Empty」、GameManagerというスクリプトを作り、そこに、この消すためのコードを書こう。
- b. 遊テックのホームページの「ノート」→「クリックされたゲームオブジェクトを取得する」を参考にしよう。

12. 6枚のカードをプレハブ化しよう。Assets直下にPrefabsというフォルダを作ってその中に入れる。

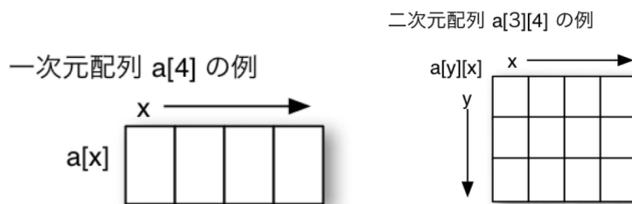
13. プレハブ化したので、いつでもクローンを作れるので、ヒエラルキーから元々あったゲームオブジェクトを削除する。青いの↓



14. 【よむ】今回は、「配列」と「リスト」について学びます。どちらもScratchのリストをイメージしてもらって、「変数が集まったもの」という理解で大丈夫です。今回の神経衰弱では、配列とリストの両方を使います。

リストと配列の違いについて。

- a. 配列: 宣言時に個数を決めると後から変更できないが、2次元以上の配列が作成できる。



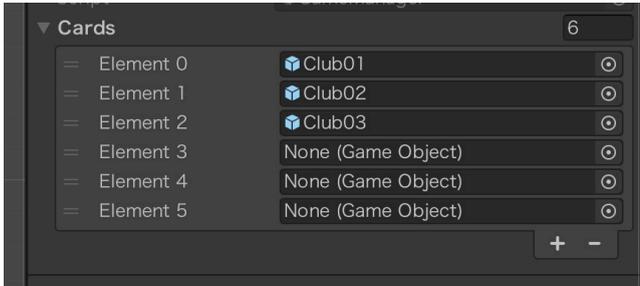
- b. リスト: 1次元しか扱えないが、後から個数を増やしたり減らしたりできる。
- c. 使い分け: 二次元以上の場合、配列一択です。一次元の老婆は、個数が分かっている場合は配列を(高速動作)、個数が分かっていない場合はリスト(要素数の変更可能)を使います。
 - i. 配置用→配列
 - ii. 選択カード→リスト(Scratchで神経衰弱を作った時と同じ使い方をします)

15. GameManagerのスク립トに追記する。

- a. 配列を作ります。publicを付けてインスペクターからD&Dして代入出来るようにする。

```
public class GameManager : MonoBehaviour
{
    public GameObject[] cards = new GameObject[6]; //カードを入れる配列の宣言
}
```

16. D&Dしてプレハブのカードを配列に入れていく。順番は適当で良い



17. スクリプトで配置してみよう。

```
public class GameManager : MonoBehaviour
{
    public GameObject[] cards = new GameObject[6]; //カードを入れる配列の宣言

    // Start is called before the first frame update
    void Start()
    {
        Instantiate(cards[0], new Vector3(-400, 200, 0), cards[0].transform.rotation); //0番目を配置
    }
}
```

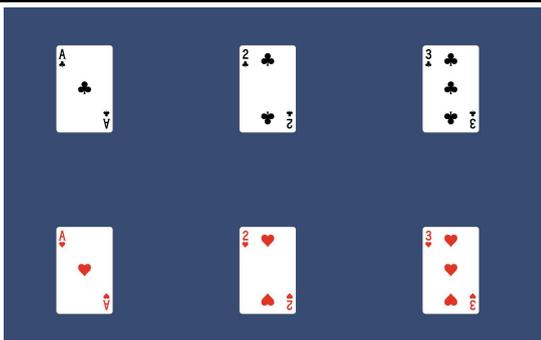
Instantiateの行をコピーして、残り5個も配置しよう。横3 * 縦2にしよう。

座標(X,Y)は以下にしよう。中央が(0,0)です。

(-400,200) (0,200) (400,200)

(-400,-200) (0,-200)(400,-200)

```
Instantiate(cards[0], new Vector3(-400, 200, 0), cards[0].transform.rotation); //0番目を配置
Instantiate(cards[1], new Vector3(0, 200, 0), cards[1].transform.rotation); //1番目を配置
Instantiate(cards[2], new Vector3(400, 200, 0), cards[2].transform.rotation); //2番目を配置
Instantiate(cards[3], new Vector3(-400, -200, 0), cards[3].transform.rotation); //3番目を配置
Instantiate(cards[4], new Vector3(0, -200, 0), cards[4].transform.rotation); //4番目を配置
Instantiate(cards[5], new Vector3(400, -200, 0), cards[5].transform.rotation); //5番目を配置
```



18. For文で書き換え

- Instantiateを6行書いたものをFor文を使った2重ループに書き換えてみよう。
- Instantiateの6行をコメント化する。
- 次のコードを加える。

```
int i = 0; //cards配列のindex
for (int yi = 0; yi < 2; yi++) //yi行目
{
    for (int xi = 0; xi < 3; xi++) //xi列目
    {
        Instantiate(cards[i], new Vector3(-400 + xi * 400, 200 - yi * 400,
0), cards[i].transform.rotation);
        i++;
    }
}
```

```
void Start()
{
    int i = 0;
    for (int yi = 0; yi < 2; yi++) //yi行目
    {
        for (int xi = 0; xi < 3; xi++) //xi列目
        {
            Instantiate(cards[i], new Vector3(-400 + xi * 400, 200 - yi * 400, 0), cards[i].transform.rotation);
            i++;
        }
    }
}
```

19. ランダムに並べる。

- 毎回同じ並びだとつまらないので、ランダムに並ぶようにしよう。ホームページのノートの「Unity C# で配列のランダムソート」を参考にしてコードを追加してください。

20. 以上でカードを並べて、クリックしたらそのカードが消えるところまで作ることが出来ました。

準備運動完了！

次回からは、ペアがそろった場合だけカードが消えるようにしていきましょう。

【ここまでのコード】

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using System;//追加
using System.Linq;//追加

public class GameManager : MonoBehaviour
{
    public GameObject[] cards = new GameObject[6];//カードを入れる配列の宣言

    // Start is called before the first frame update
    void Start()
    {
        //シャッフル
        cards = cards.OrderBy(x => Guid.NewGuid()).ToArray();

        int i = 0;//配列cardsのindex
        for (int yi = 0; yi < 2; yi++)//yi行目
        {
            for (int xi = 0; xi < 3; xi++)//xi列目
            {
                Instantiate(cards[i], new Vector3(-400 + xi * 400, 200 - yi * 400, 0),
cards[i].transform.rotation);
                i++;
            }
        }
    }

    // Update is called once per frame
    void Update()
    {
    }
}
```

```

1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4  using System;//追加
5  using System.Linq;//追加
6
7  public class GameManager : MonoBehaviour
8  {
9      public GameObject[] cards = new GameObject[6];//カードを入れる配列の宣言
10
11     // Start is called before the first frame update
12     void Start()
13     {
14         //シャッフル
15         cards = cards.OrderBy(x => Guid.NewGuid()).ToArray();
16     |
17     int i = 0;//配列cardsのindex
18     for (int yi = 0; yi < 2; yi++)//yi行目
19     {
20         for (int xi = 0; xi < 3; xi++)//xi列目
21         {
22             Instantiate(cards[i], new Vector3(-400 + xi * 400, 200 - yi * 400, 0), cards[i].transform.rotation);
23             i++;
24         }
25     }
26 }
27
28 // Update is called once per frame
29 void Update()
30 {
31 }
32 }
33 }
34

```