

게임엔진이해와실습

디지털콘텐츠공학과 20183349 최진
아

F l a p p y B i r d

Index

title1 | 게임 소개

title2 | 기존 게임 구성

title3 | 추가한 기능

title4 | 결과물

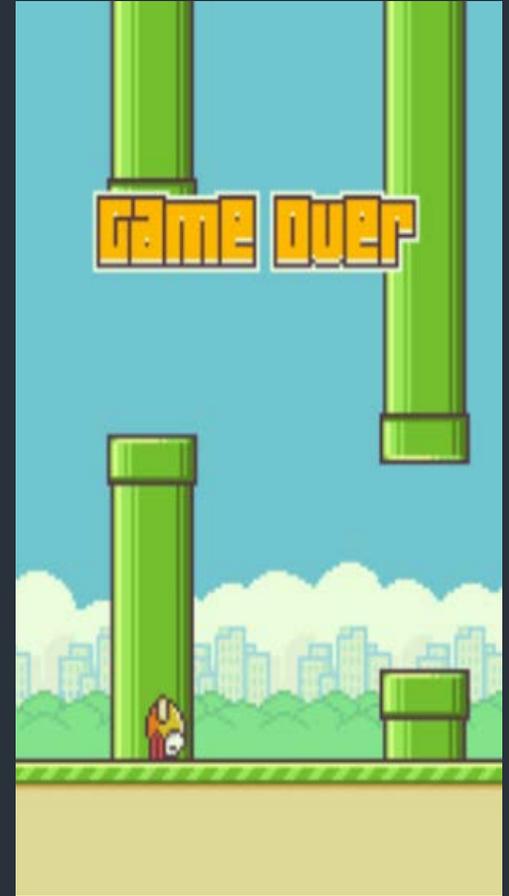
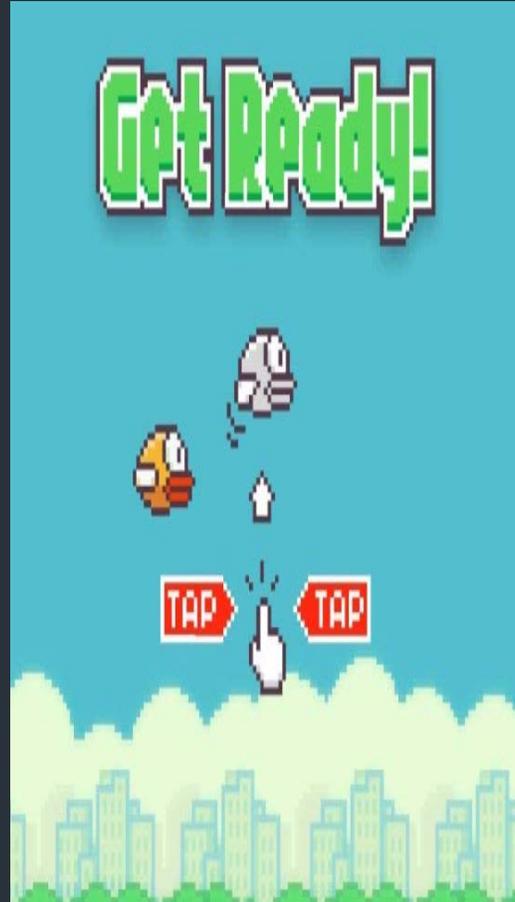
" FLAPPY BIRD "

- 새가 날아 다니며 대나무 사이 사이를 피하며 스코어 점수를 올리는 게임.



<첫 화면>

<게임 오버 화면>



<오브젝트에 닿지않았을 때>



<오브젝트에 닿았을 때>

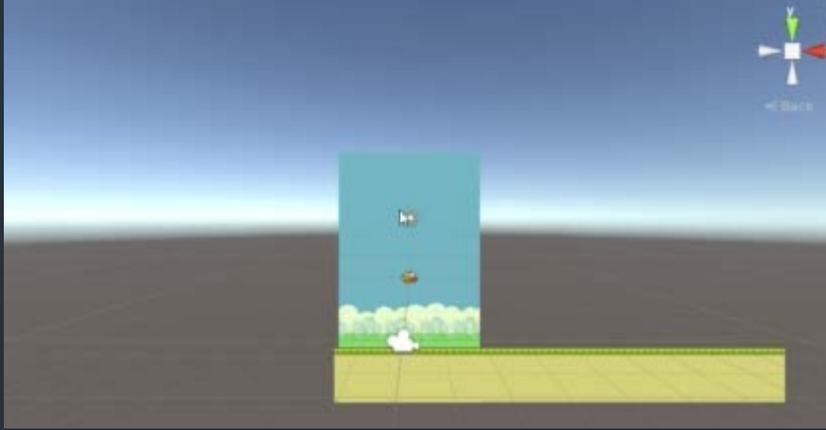


PREFABS을 사용하여 하나로 묶었다.

<맵 배경 3개 추가 코드 >

<배경의 길이 조정 영상.>

<배경의 길이 조정>



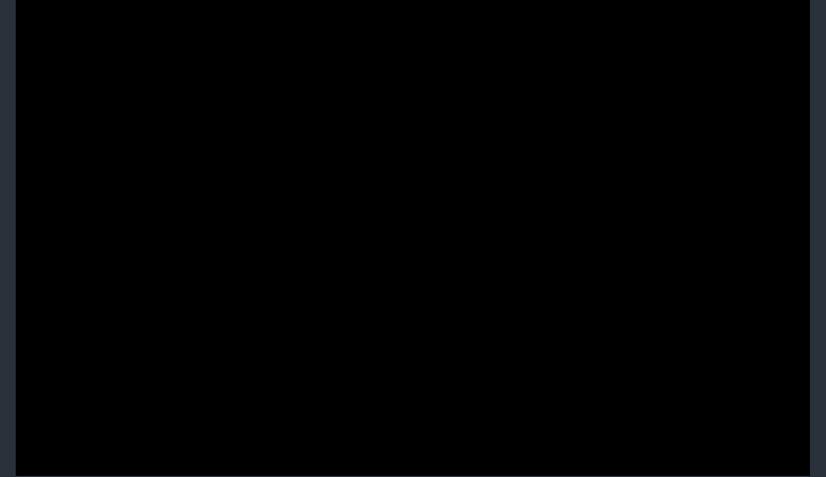
```
using System.Collections.Generic;
using UnityEngine;

public class GroundMove : MonoBehaviour
{
    float _distance = 7.67f;
    int _count = 2;
    int _index = 2;

    public GameObject[] grounds;
    // Start is called before the first frame update
    void Start()
    {
    }

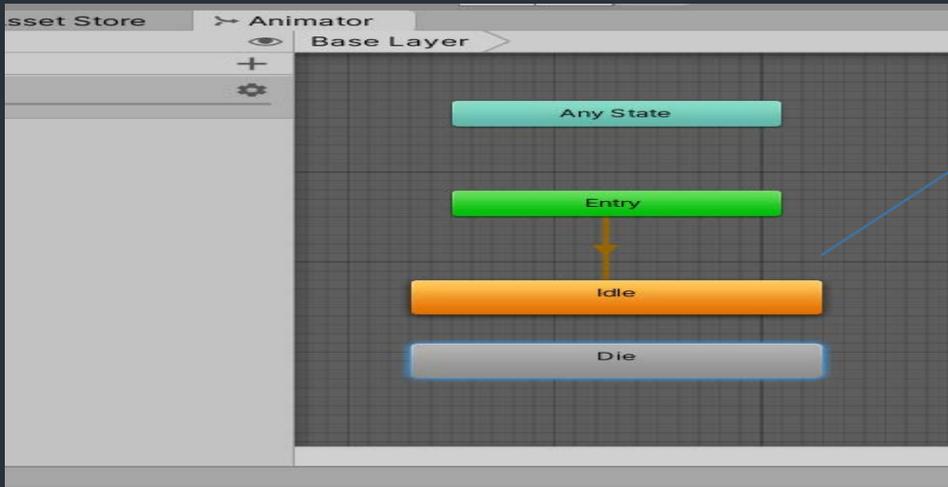
    // Update is called once per frame
    void Update()
    {
        if (Time.timeScale == 0)
            return;

        gameObject.transform.localPosition += new Vector3(-0.05f, 0, 0);
        _count = 2 + (int)(-gameObject.transform.localPosition.x / 7.67f);
        if (_index != _count)
        {
            grounds[_index - 2].transform.localPosition = new Vector3(_distance * _count, -5, 0);
            _index = _count;
        }
    }
}
```

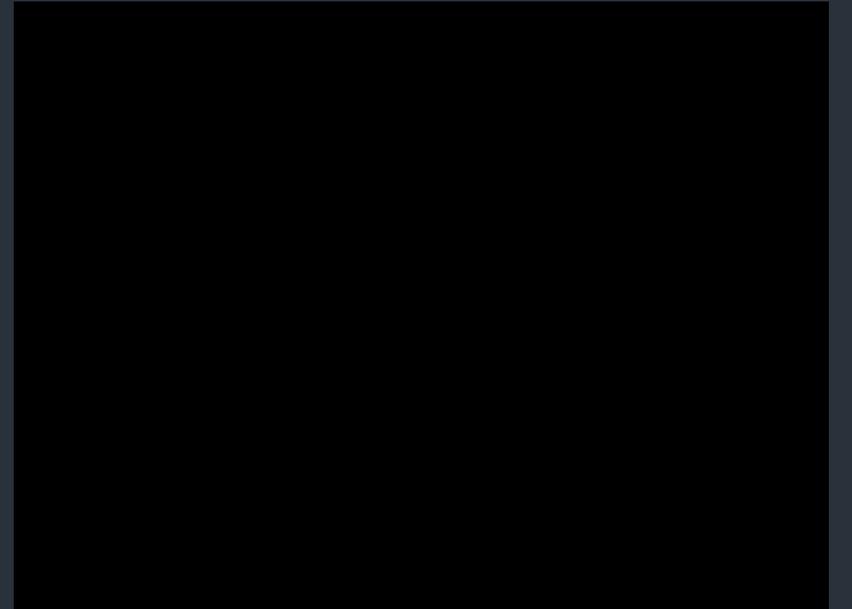


<애니메이션 기능 영상>

<애니메이션 기능을 추가하여 다양한 모션을 추가함>

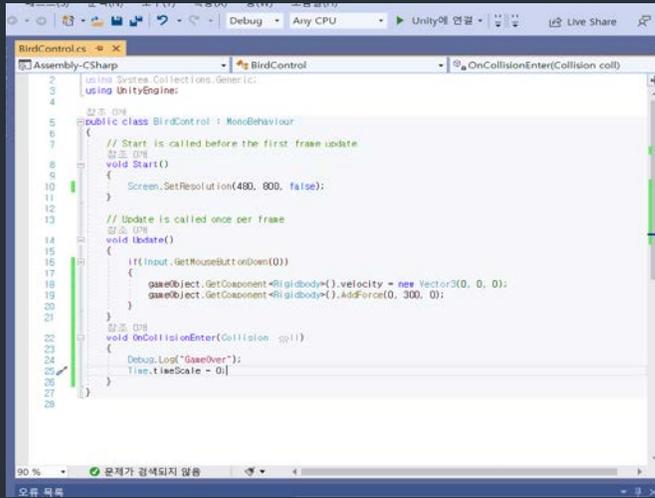


캐릭터가 날개짓을 하거나 물체에 부딪히면 모습이 변함.



title2 | 기존 게임 구성

<오브젝트 땅에 닿으면 gameover>



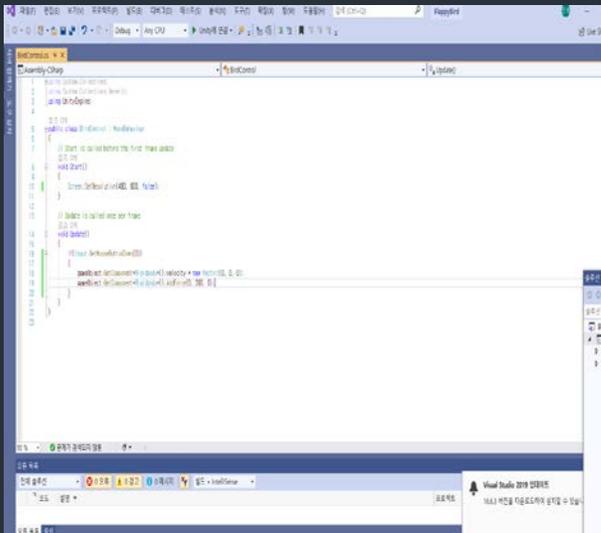
```
using UnityEngine;

public class BirdControl : MonoBehaviour
{
    // Start is called before the first frame update
    void Start()
    {
        Screen.SetResolution(480, 800, false);
    }

    // Update is called once per frame
    void Update()
    {
        if (Input.GetMouseButton(0))
        {
            gameObject.GetComponent<Rigidbody>().velocity = new Vector3(0, 0, 0);
            gameObject.GetComponent<Rigidbody>().AddForce(0, 300, 0);
        }
    }

    void OnCollisionEnter(Collision col)
    {
        Debug.Log("GameOver");
        Time.timeScale = 0;
    }
}
```

<점프하기>

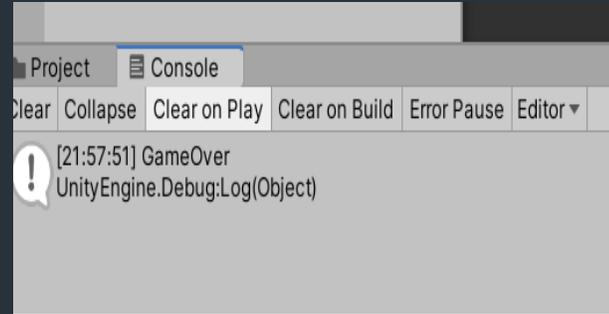


```
using UnityEngine;

public class BirdControl : MonoBehaviour
{
    // Start is called before the first frame update
    void Start()
    {
        Screen.SetResolution(480, 800, false);
    }

    // Update is called once per frame
    void Update()
    {
        if (Input.GetMouseButton(0))
        {
            gameObject.GetComponent<Rigidbody>().velocity = new Vector3(0, 0, 0);
            gameObject.GetComponent<Rigidbody>().AddForce(0, 300, 0);
        }
    }
}
```

<콘솔 창 gameover>



<클릭시 점프>



<캐릭터가 물체에 닿았을때 게임오버 영상>



<새가 점프하는 영상>



title2 | 기존 게임 구성

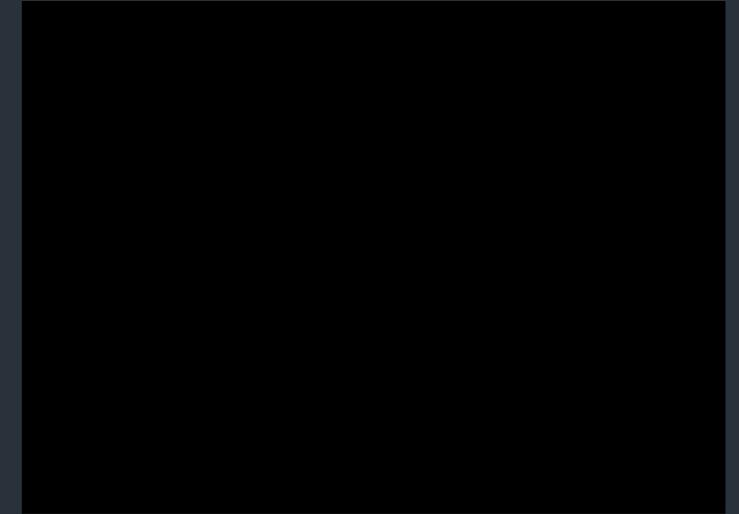
<다시 게임을 시작하는 영상>

<다시 게임을 시작하는 코드>

```
gameObject.GetComponent<Rigidbody>().velocity = new Vector3(0, 0, 0);  
gameObject.GetComponent<Rigidbody>().AddForce(0, 300, 0);  
}  
if(Input.GetKeyDown(KeyCode.R))  
{  
    Time.timeScale = 1;  
    Application.LoadLevel("Game");  
}
```

<게임 오버 콘솔 창>

```
Project Console  
Clear Collapse Clear on Play Clear on Build Error Pa  
[22:05:39] GameOver  
UnityEngine.Debug:Log(Object)  
[22:05:43] GameOver  
UnityEngine.Debug:Log(Object)  
[22:05:45] GameOver  
UnityEngine.Debug:Log(Object)
```



<대나무 맵 일정한 길이로 나오는 코드>

```
public class ColumnMaker : MonoBehaviour  
{  
    public GameObject Column;  
    private float nowTime;  
    private float makeTime = 2f;  
    // Start is called before the first frame update  
    void Start()  
    {  
        nowTime = Time.time;  
    }  
    // Update is called once per frame  
    void Update()  
    {  
        if(Time.time - nowTime > makeTime)  
        {  
            nowTime = Time.time;  
            GameObject temp = Instantiate(Column);  
            temp.transform.parent = gameObject.transform;  
            temp.transform.localPosition = new Vector3(-gameObject.transform.localPosition.x + 5, 0, 0);  
            temp.transform.localScale = new Vector3(1, 1, 1);  
        }  
    }  
}
```

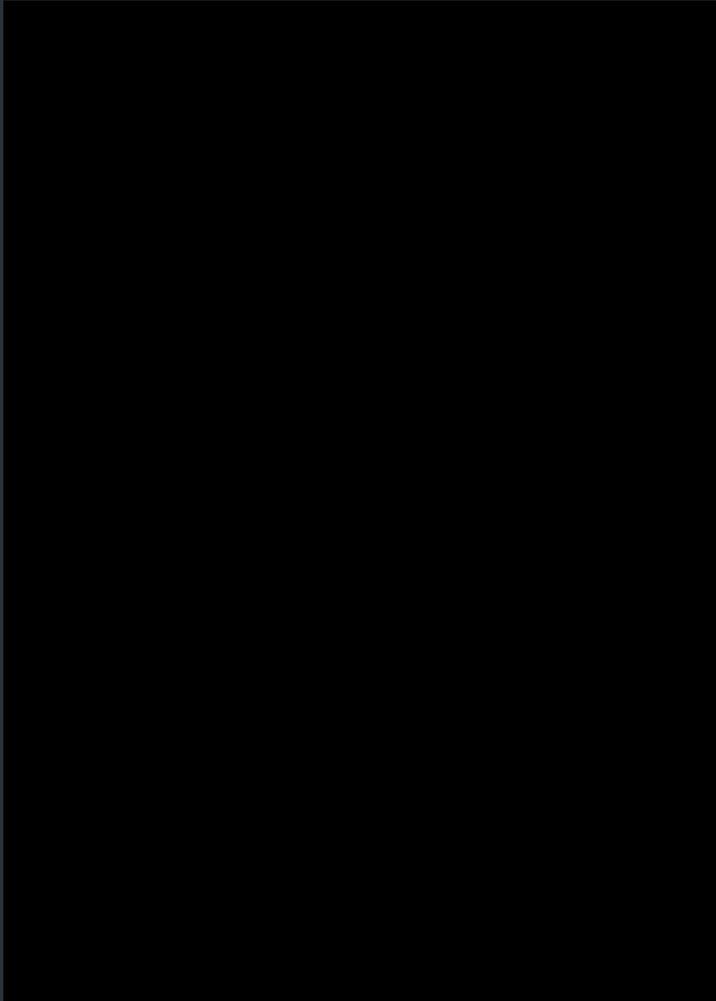
<대나무 맵 랜덤한 길이로 나오는 코드>

```
public class ColumnMaker : MonoBehaviour  
{  
    public GameObject Column;  
    private float nowTime;  
    private float makeTime = 2f;  
    // Start is called before the first frame update  
    void Start()  
    {  
        nowTime = Time.time;  
    }  
    // Update is called once per frame  
    void Update()  
    {  
        if(Time.time - nowTime > makeTime)  
        {  
            nowTime = Time.time;  
            GameObject temp = Instantiate(Column);  
            temp.transform.parent = gameObject.transform;  
            float randomY = Random.Range(-6.15f, -1.5f);  
            temp.transform.localPosition = new Vector3(-gameObject.transform.localPosition.x + 5, randomY, 0);  
            temp.transform.localScale = new Vector3(1, 1, 1);  
        }  
    }  
}
```

<대나무 맵 랜덤 길이 영상>



직접 만들어 본 기본 "FLIPPAY BIRD"



- 플레이를 할 때 화면 게임시작 버튼을 눌러 시작하게 하고 싶다.
- 플레이를 할 때 사용자가 지루하지 않도록 Audio Sound 추가하기
- 흥미롭게 코인을 모으는 게임을 개발 하고 싶다.



01

Audio Sounds

게임을 할 때 질리고 흥미가 없어지기 쉽다.
이 게임에 어울리는 음악을 가지고 게임을 즐긴다면 한층 더 재밌어 질 것이다.

02

타겟팅/코인

캐릭터(새) 가 대나무 숲을 통과할 때 캐릭터가 코인을 끌어 당기는 듯 한 느낌을 주고싶다. - 타겟팅 사용하기!

03

시작화면 꾸미기

게임을 시작하면 화면없이 바로 실행이 되기 때문에 완성도가 떨어진다. 사용자들 이 관심을 가질만 하고 그 게임에 어울리는 시작화면이 필요하다.

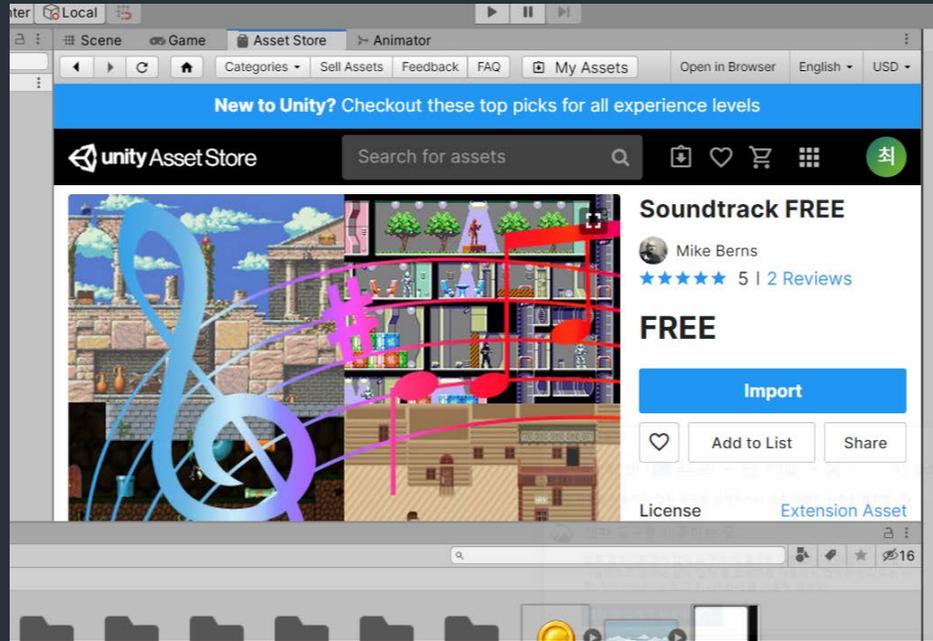
04

게임 설명창 만들기

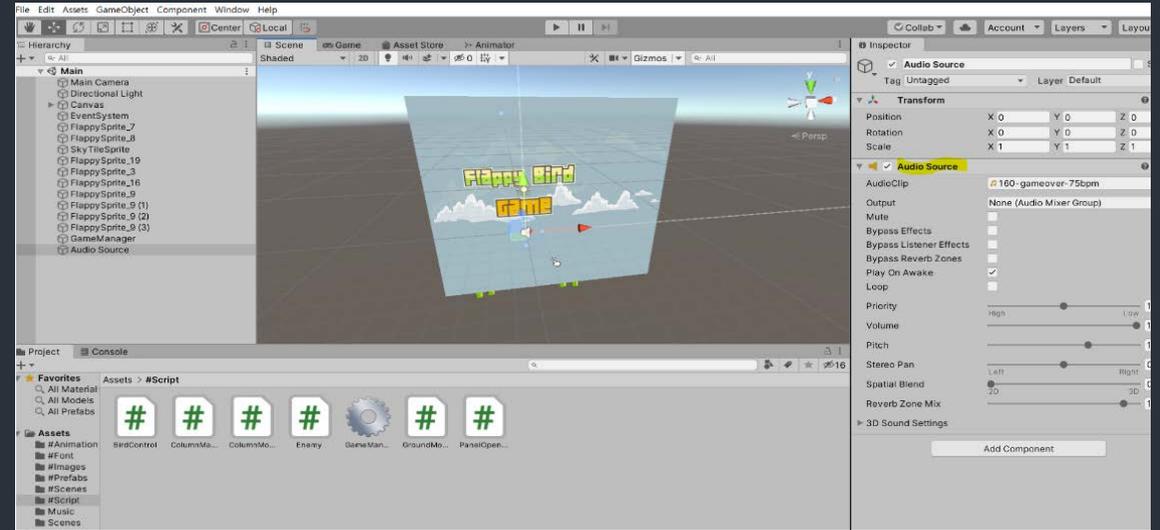
게임을 어떻게 플레이 하는지 , 사용자의 연령대가 다양하기 때문에 게임을 간단 하게 소개해 줄 버튼과 창이 필요하다고 생각하여 게임 설명창을 제작함.

1.Audio Sounds

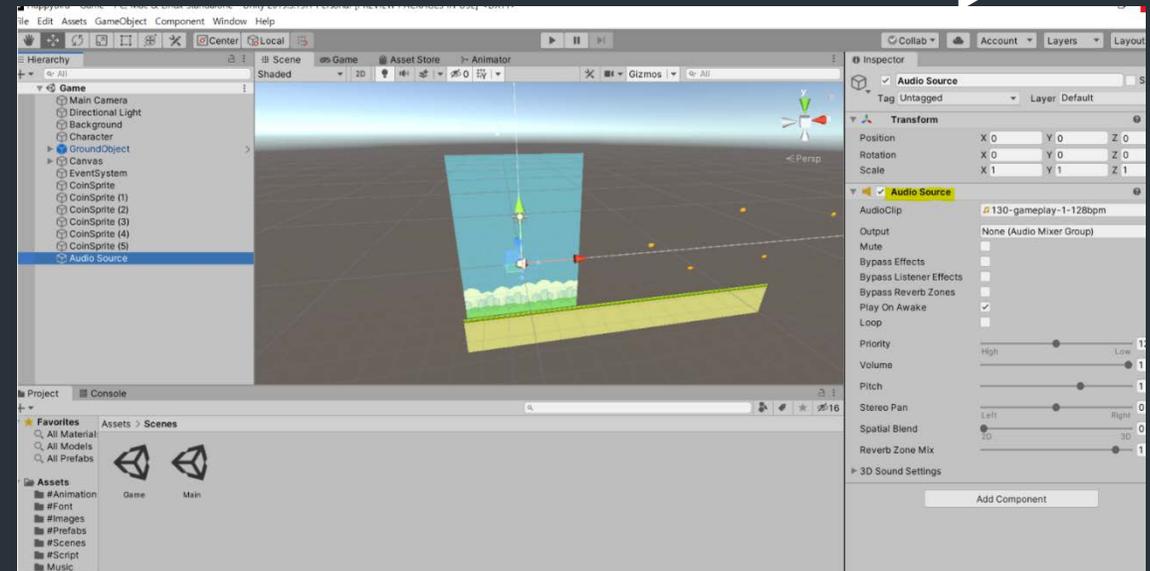
Soundtrack free 다운



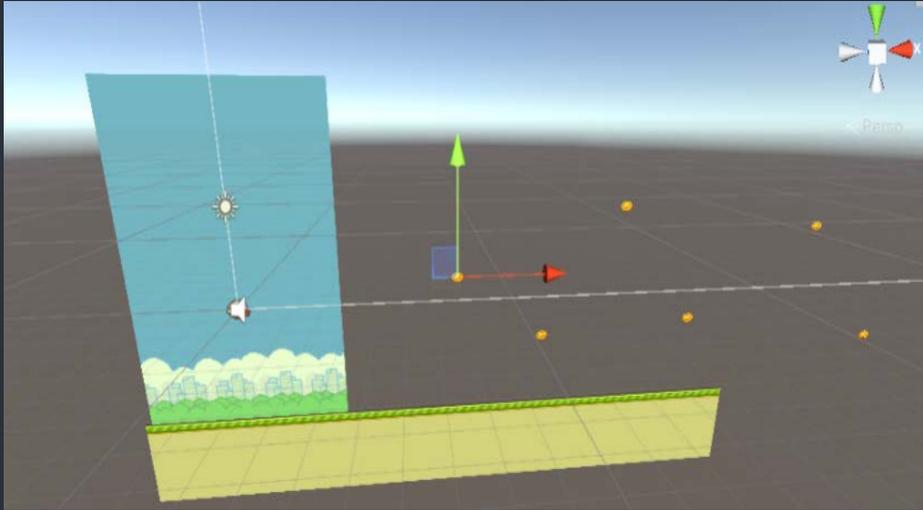
1. 시작화면 - Audio Source 삽입



2. 게임 화면 - Audio Source 삽입



2.타켓팅 / 코인



> 새가 일정한 거리를 갈 때마다 코인이 새에게 움직이게된다.
거리를 더 많이 갈 수록 코인을 더많이 얻게된다.

새에게 달라 붙는 코인을
볼 수 있다.
>타켓팅 기능 효과



타켓팅 코인 코드

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

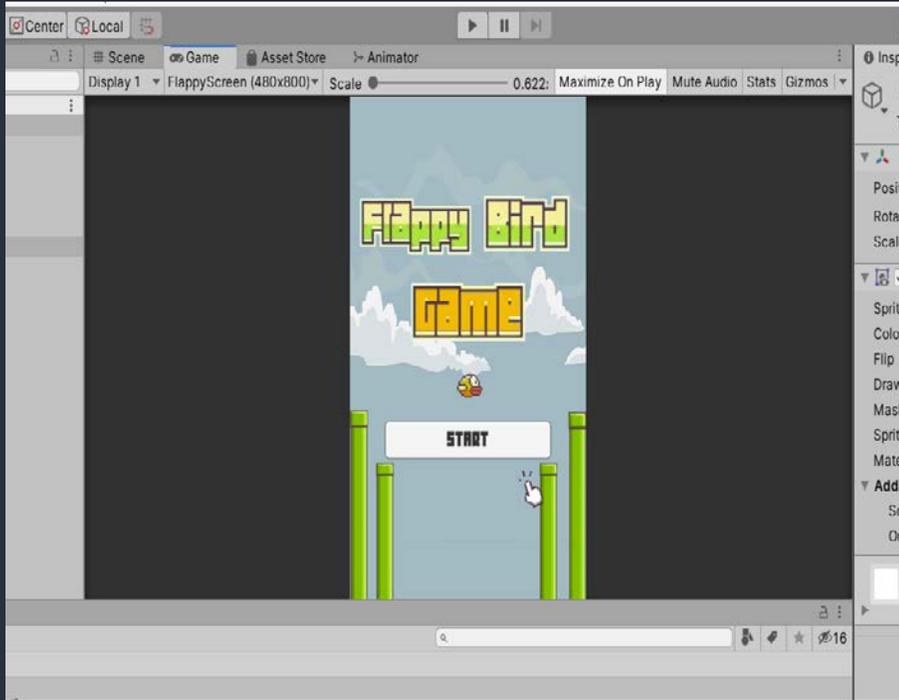
public class Enemy : MonoBehaviour
{
    // Start is called before the first frame update

    public Transform Target;
    void Start()
    {

    }

    // Update is called once per frame
    void Update()
    {
        Vector3 Direction = Target.position -
        GetComponent<Transform>().position;
        float Distance = Mathf.Sqrt((Direction.x * Direction.x) +
        (Direction.y * Direction.y));
        Direction.x /= Distance;
        Direction.y /= Distance;
        GetComponent<Transform>().position += Direction *
        Time.deltaTime;
    }
}
```

3. 시작 화면



- Flappy Bird Game의 시작화면을 디자인 해보았다.
- 새로운 씬을 만들어 start버튼을 누르면 게임 메인씬으로 가도록 코드를 짜보았다.
 - Build Setting 에서 씬 각각 추가.

타켓팅 코인 코드

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

public class GameManager : MonoBehaviour
{
    // Start is called before the first frame update
    void Start()
    {

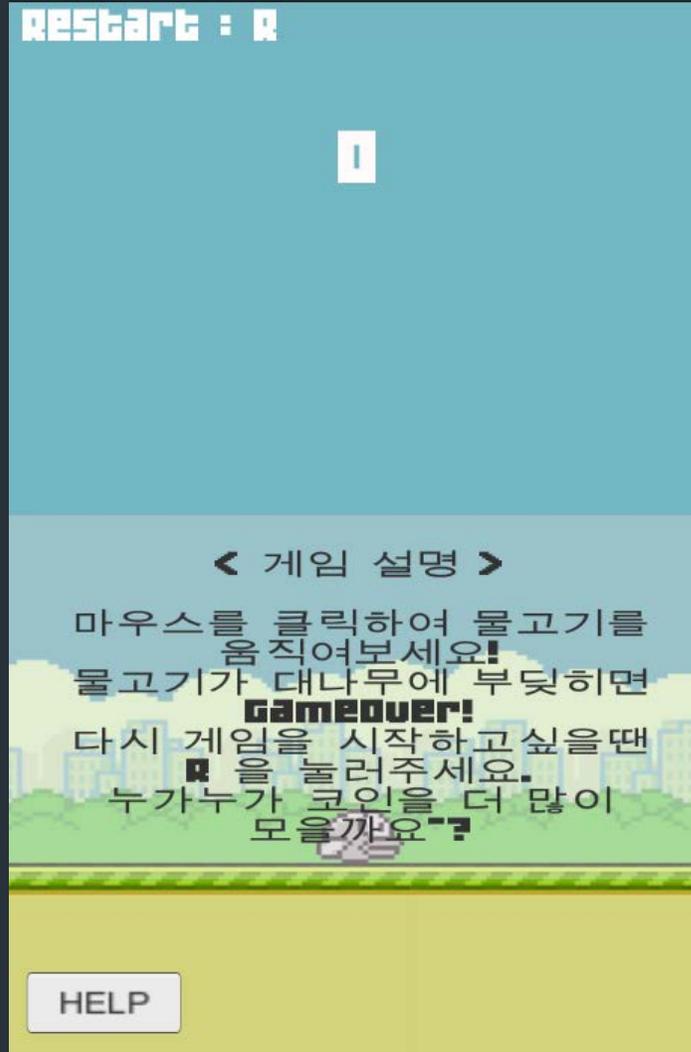
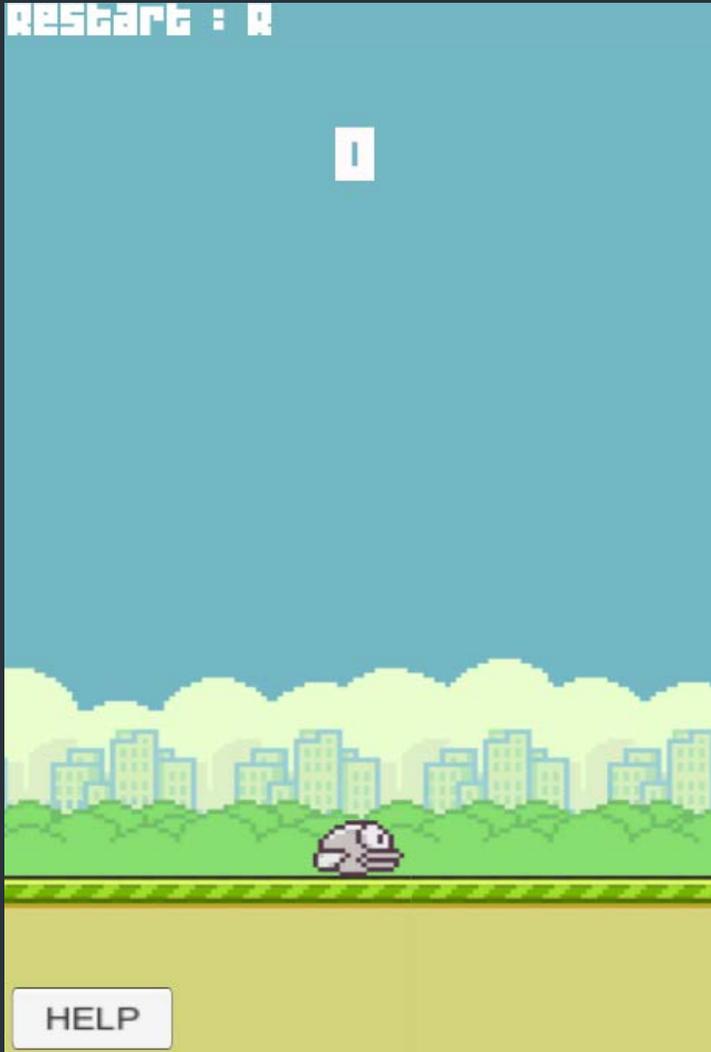
    }

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

    }

    public void LoadGame()
    {
        SceneManager.LoadScene("Game"코);
    }
}
```

4. 게임 설명창



▶게임 설명창
게임은 다양한 연령대의 사람들이 즐기기에
문에
아이들, 어른들 모두 이 게임을 이해하고 플레이하는데
있어서 게임을 설명해줄 창이 필요하다고 생각하였다.

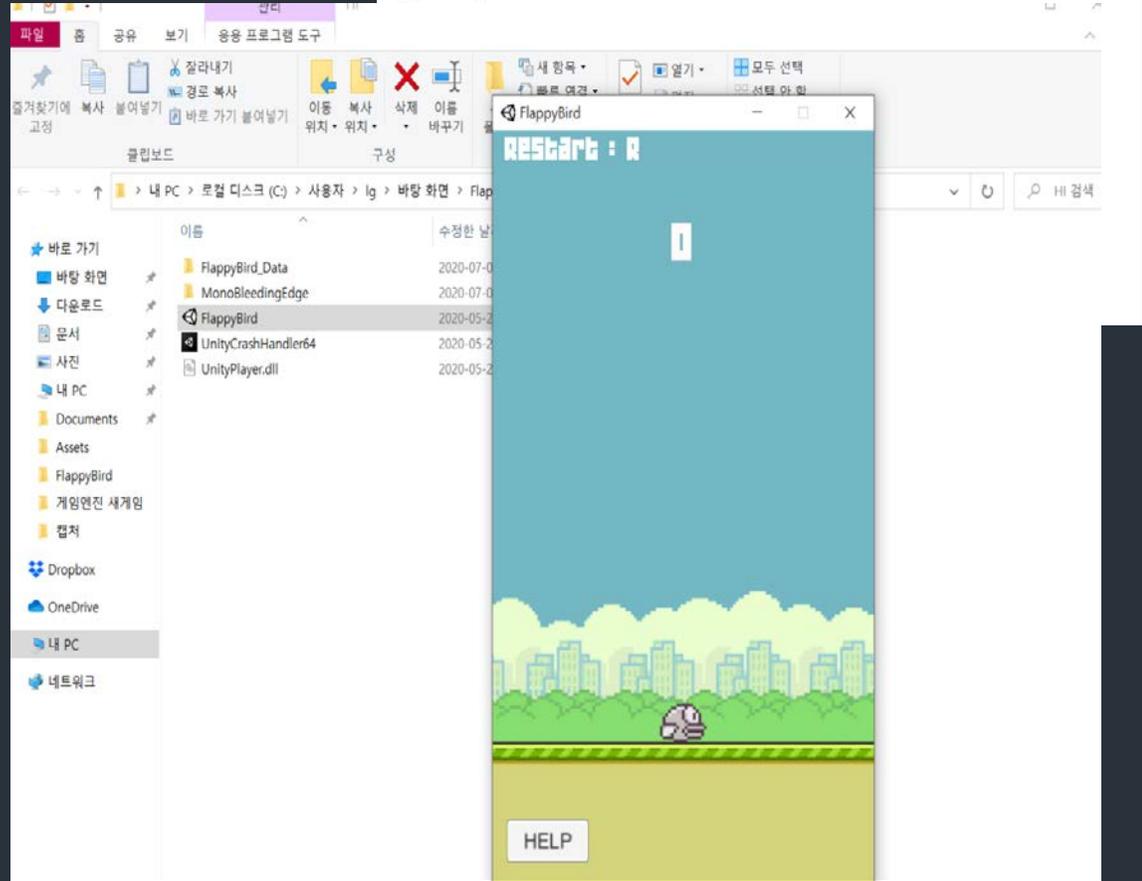
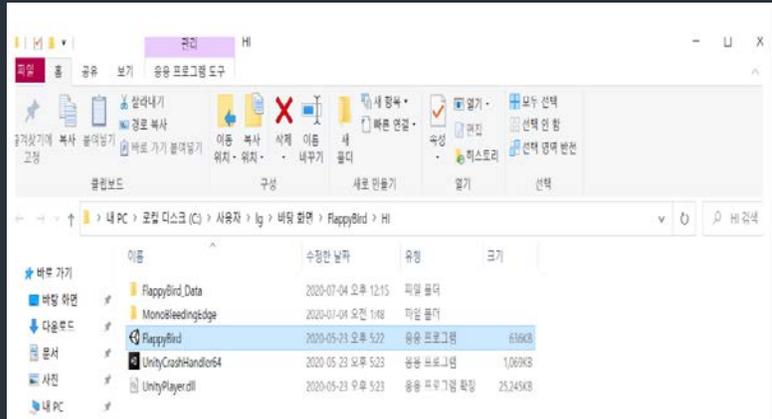
▶HELP를 버튼을 클릭하게 되면 <게임 설명> 창이 뜨게 되고 사용자들에게 이해하기 쉽게 친절히 설명이 되어있다.

이 HELP 설명창은 클릭을 하면 설명창이 뜨고 다시 클릭을 하게 되면 창이 사라지게끔 만들어보았다.

= CANVAS , PANAL , BUTTON사용

최종 결과물 Flappy B

Pc 버전 사용 가능



감사합니다.