

A person with brown hair, wearing a brown sweater, is shown in profile, focused on a laptop screen. The background is dark blue with numerous white, glowing binary digits (0s and 1s) floating around, creating a digital atmosphere. A semi-transparent white banner is overlaid across the middle of the image, containing the title text.

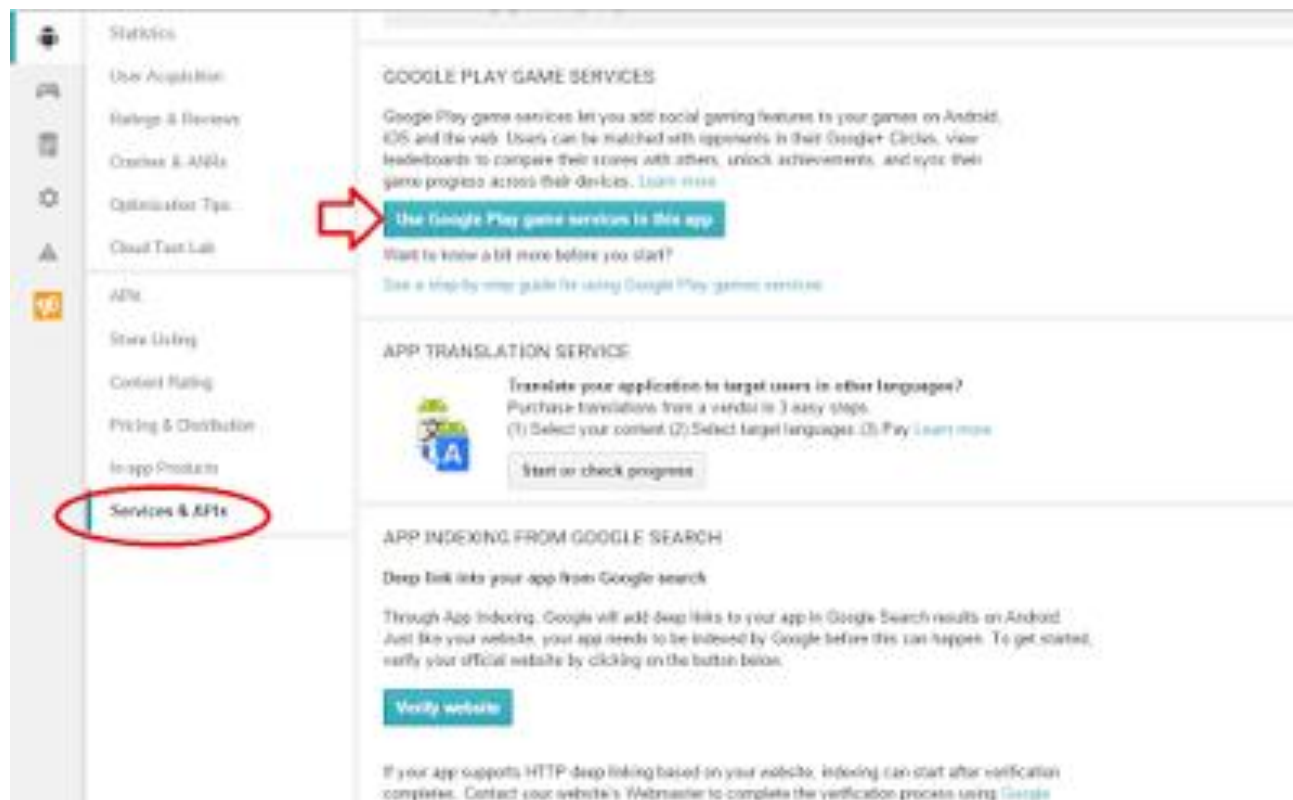
Bài 6: Leader board, archivment trong google play service

Mục tiêu

- Tạo Leaderboard
- Tạo Raycast

Tạo Leaderboard sử dụng Google Play Service

- Để tạo được leaderboard thì bạn phải cài đặt leaderboard ở trên google play developer.
- Vào google play developer, tạo leaderboard cho game của mình để lấy Id leaderboard.



Tạo Leaderboard sử dụng Google Play Service

- Tạo leaderboard: Vào Services & APIs:

Google Play Developer Console

LEADERBOARDS [Add new leaderboard](#) or [Continue to next step](#)

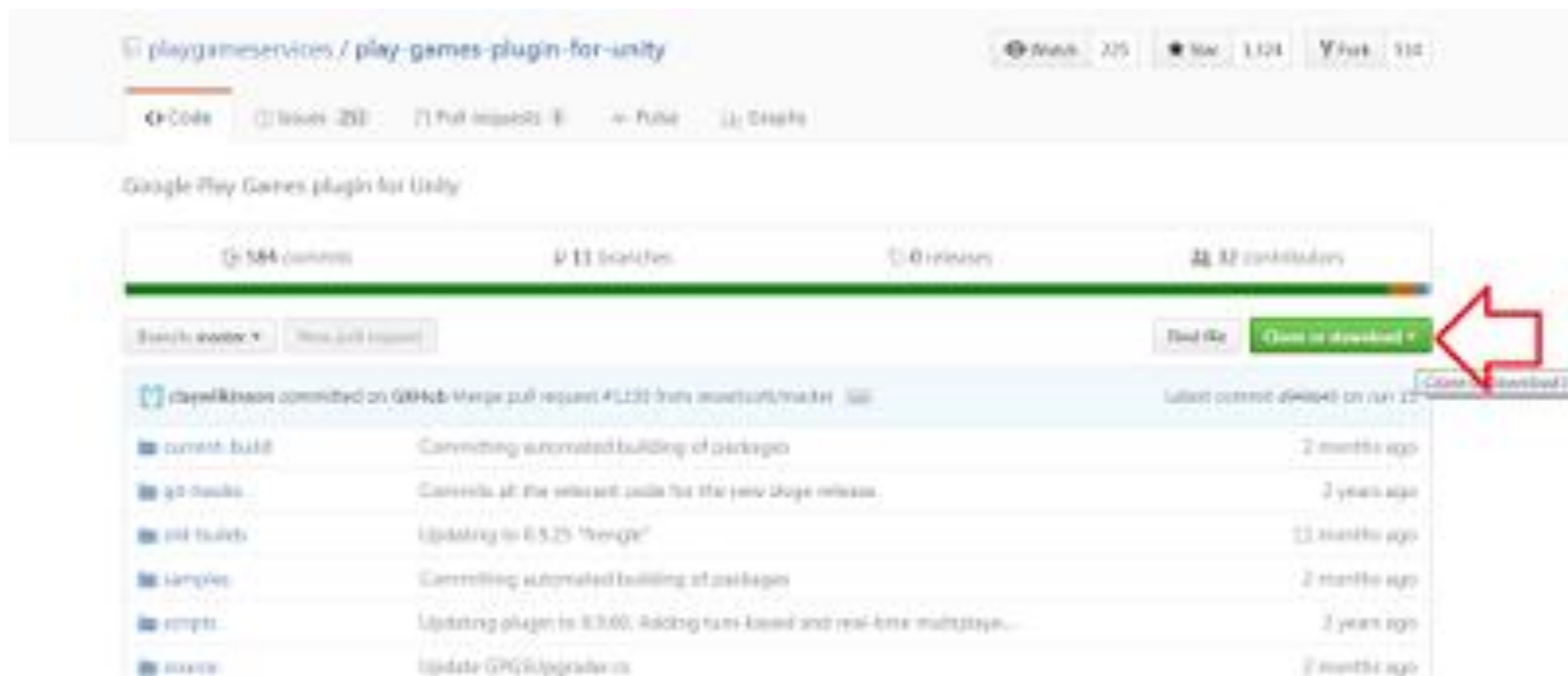
#	NAME	ID-G	Resource ID	STATUS
1	<input type="checkbox"/> High Score	XXXXXXXXXX	---	Ready to publish

get resources → [Get resources](#)

Learn all about implementing leaderboards in the [Developer documentation](#)

Tạo Leaderboard sử dụng Google Play Service

- Download plugin for unity:
 - Tiếp theo, bạn download package google play service cho unity ở đường link sau:
<https://github.com/playgameservices/play-games-plugin-for-unity>

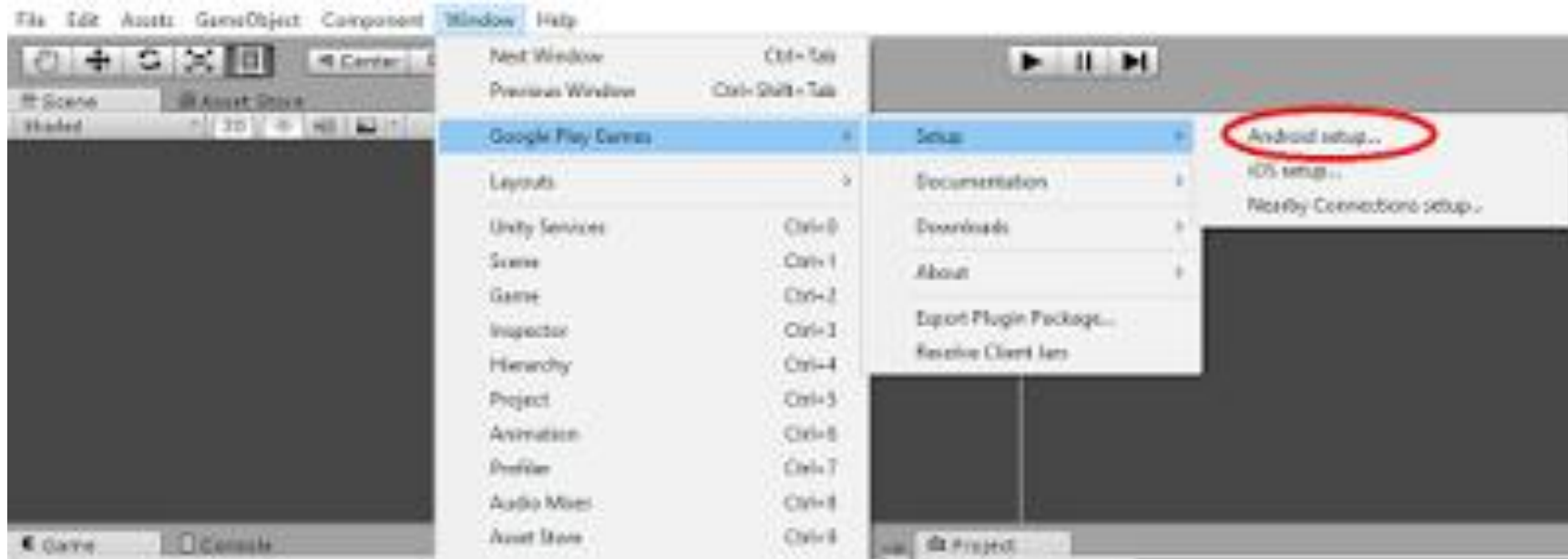


Tạo Leaderboard sử dụng Google Play Service

- Sau khi download và giải nén, package nằm ở thư mục current-build, nếu bạn muốn sử dụng các package cũ hơn thì có thể vào thư mục old-builds.
- Import package vào project Unity.
- Và cuối cùng là install google play service ở trong SDK Manager.
- Update 3 packages trong SDK manager:
 - + Android Support Library
 - + Google Play services
 - + Google Repository

Tạo Leaderboard sử dụng Google Play Service

- Cài đặt trong Unity:
 - Tiến hành cài đặt Google play service cho android trong Unity.
 - Vào Windows -> Google play Games -> Setup -> Android setup



Tạo Leaderboard sử dụng Google Play Service

- Thiết lập các thông số cài đặt:

Google Play Games - Android Configuration

To configure Google Play Games in this project, go to the Play Game console, then enter the information below and click on the Setup button.

[Open Play Games Console](#)

Constants class name
Enter the fully qualified name of the class to create containing the constants

Directory to save constants:

Constants class name:

Resources Definition
Paste in the Android Resources from the Play Console

Web App Client ID (Optional)
The web app client ID is needed to access the user's ID token and call other APIs on behalf of the user. It is not required for Game Services. Enter your oauth2 client ID below.
To obtain this ID, generate a web linked app in Developer Console. Example:
123456789012-abcdefghijklm.apps.googleusercontent.com

Client ID:

Tạo Leaderboard sử dụng Google Play Service

- Thiết lập các thông số cài đặt:
 - Constants class name: Cái này là đường dẫn và tên class được tự động tạo ra sau khi setup chứa id của leaderboard.
 - Resources Definition: Copy resources ở tài khoản develop thiết lập lúc này dán vào.
 - Web App client Id (Optional): Có thể điền vào nếu bạn có sử dụng 1 web back-end cho game không thì để trống.

Tạo Leaderboard sử dụng Google Play Service

- Trong quá trình làm test game để show được leaderboard thì bạn phải update file apk lên tài khoản dev.
- Trong phần setup của leaderboard:
 - + Linked apps: phải link app của bạn vào.
 - + Testing thêm email tài khoản muốn test vào phần Testing access.

Tạo Leaderboard sử dụng Google Play Service

- Một số hàm cần chú ý trong file code:
- - Hàm login vào google account: `LogIn();`
- - Hàm show leaderboard toàn bộ: `OnShowLeaderBoard();`
- - Hàm post điểm của mình lên leaderboard: `OnAddScoreToLeaderBoard();`
- - Hàm Logout: `OnLogOut();`

Tạo Leaderboard sử dụng Google Play Service

■ Code:

```
using UnityEngine;
using System.Collections;
using GooglePlayGames;
using UnityEngine.SocialPlatforms;
using UnityEngine.UI;

public class LeaderBoard : MonoBehaviour
{
    void Start(){
        PlayGamesPlatform.DebugLogEnabled = true;
        PlayGamesPlatform.Activate ();
    }

    //Login into your google+ account
    public void LogIn()
    {
        Social.localUser.Authenticate((bool success) =>
        {
            if (success)
            {
                Debug.Log("Login Success");
            }
        });
    }
}
```

Tạo Leaderboard sử dụng Google Play Service

```
■ OnAddScoreToLeaderBoard();  
    }  
    else  
    {  
        Debug.Log("Login failed");  
    }  
});  
}  
  
//show all available leaderboard  
public void OnShowLeaderBoard()  
{  
    Social.ShowLeaderboardUI();  
}
```

Tạo Leaderboard sử dụng Google Play Service

- *// Post your 5000 point to leader board*
public void OnAddScoreToLeaderBoard()
{
if (Social.localUser.authenticated) {
Social.ReportScore (5000, GPGSIds.leaderboard_high_score, (bool success) =>
{
if (success) {
((PlayGamesPlatform)Social.Active).ShowLeaderboardUI
(GPGSIds.leaderboard_high_score);
} else {
Debug.Log ("Add Score Fail");
}
});

Tạo Leaderboard sử dụng Google Play Service

```
} else {  
    LogIn();  
}  
}  
  
//On Logout of your Google+ Account  
public void OnLogOut()  
{  
    ((PlayGamesPlatform)Social.Active).SignOut();  
}  
}
```

Tạo Raycast trong Unity

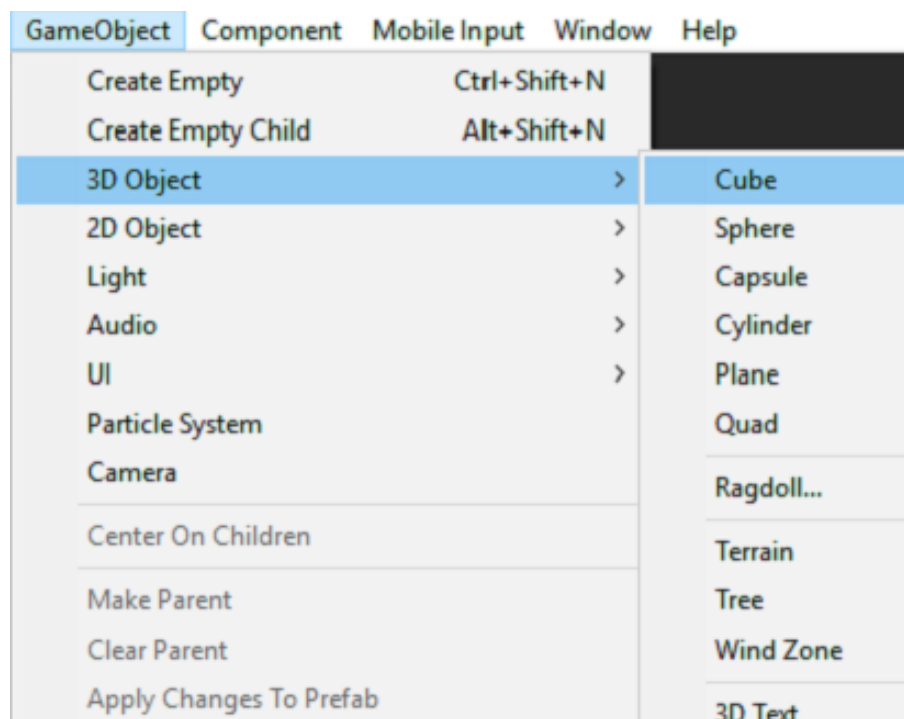
- Ví dụ về Raycast:
- Để bắt va chạm của viên đạn với vật thể trong game
- Để biết tay của bạn chạm vào vật nào trong game

Raycast là gì ?

- Raycasting thường được sử dụng trong những việc như xác định đường ngắm của người chơi hoặc AI, nơi một viên đạn sẽ đi qua, tạo ra tia laser và nhiều hơn nữa.
- Một raycast về cơ bản là một tia mà được gửi đi từ một vị trí trong không gian 3D hoặc 2D và di chuyển theo một hướng cụ thể.
- Unity đã xây dựng chức năng trong đó bạn có thể sử dụng để thực hiện Raycast trong trò chơi của bạn.

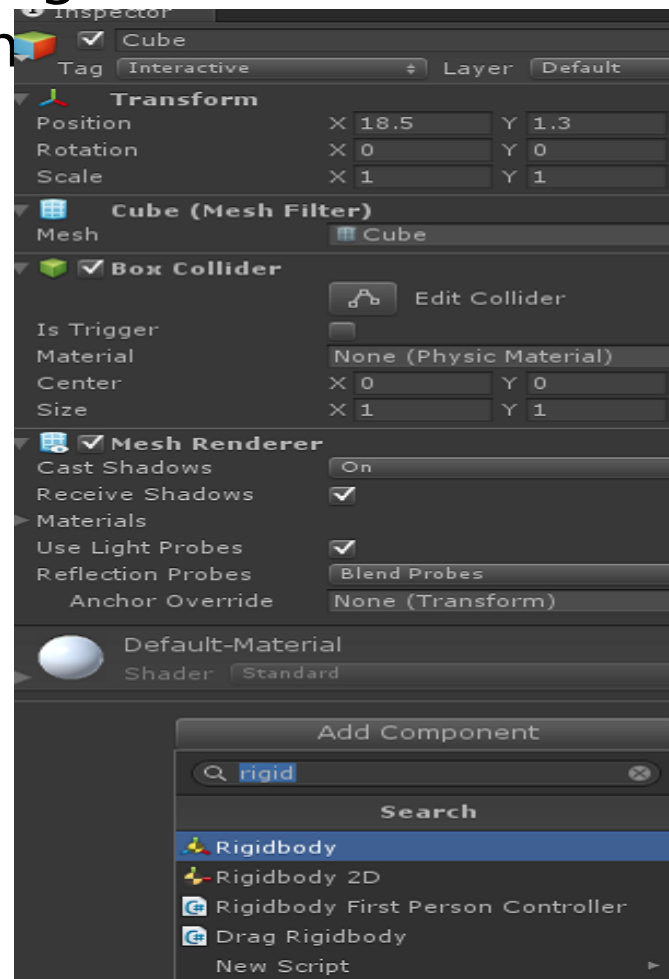
Cách dùng Raycast

- Bước 1: Import bộ asset tiêu chuẩn assets / characters package và kéo và thả một trong các bộ điều khiển prefab vào scene của bạn.
- Bước 2: Một khi bạn đã thiết lập scene cơ bản của bạn, chúng ta sẽ tạo ra một khối lập phương bằng cách vào GameObject → 3D Object → Cube, như dưới đây:



Cách dùng Raycast

- Bước 3: Chúng ta sẽ sử dụng một tag để xác định những gì chúng ta đang tìm kiếm. Để tạo ra một tag tùy chỉnh, chọn khối lập phương trong scene của bạn, và trong inspector, nhấp vào Tag và ch



Cách dùng Raycast

- Bước 3: Trong quản lý thẻ, nhấp vào biểu tượng + để tạo ra một tag mới. Hãy gọi tag này là "Interactive".
- Bước 4: Khi tag được tạo ra, nhấp chuột vào Cube lần nữa và chọn tag là Interactive
- Bước 5: Tiếp theo, nhấn vào khối hộp và thêm thành phần rigidbody bằng cách sử dụng cửa sổ Add Component

Cách dùng Raycast

- Bước 6: Bây giờ chúng ta thực hiện các raycast. Chọn main camera của bạn trong scene của bạn và thêm một thành phần C # Script mới cho nó gọi



Cách dùng Raycast

```
using UnityEngine;
using System.Collections;

public class LineOfSight : MonoBehaviour {

    private RaycastHit vision; //Used for detecting Raycast collision
    public float rayLength; //used for assigning a length to the raycast
    private bool isGrabbed; //used so we know whether or not we're holding an object
    private Rigidbody grabbedObject; //used to assign the object we're looking at to a variable we can use

    void Start ()
    {
        rayLength = 4.0f;
        isGrabbed = false;
    }

    void Update ()
    {
        //This will constantly draw the ray in our Scene View so we can see where the ray is going
        Debug.DrawRay(Camera.main.transform.position, Camera.main.transform.forward * rayLength, Color.red, 0.5f);

        //This statement is called when the Raycast is hitting a collider in the scene
        if(Physics.Raycast(Camera.main.transform.position, Camera.main.transform.forward, out vision, rayLength))
        {
            //determine if the object our raycast is hitting has the "Interactive" tag
            if(vision.collider.tag == "Interactive")
            {
                Debug.Log (vision.collider.name); //Output the name of the object our raycast is hitting to our console

                //If our user is looking at an interactive object, allow them to press E and pick it up but only if we aren't holding something
                if(Input.GetKeyDown (KeyCode.E) && !isGrabbed)
                {
                    grabbedObject = vision.rigidbody; //assign grabbedObject to the object we're looking at
                    grabbedObject.isKinematic = true; //set the rigidbody to kinematic so it ignores physics
                    grabbedObject.transform.SetParent (gameObject.transform); //set grabbedObject's parent to our camera
                    isGrabbed = true; //we are now grabbing an object so set this to true
                }
                //if we have an object already and user presses E, drop the object
                else if(isGrabbed && Input.GetKeyDown(KeyCode.E))
                {
                    grabbedObject.transform.parent = null; //set grabbedObject's parent back to nothing
                    grabbedObject.isKinematic = false; //allow grabbedObject to interact with physics again
                    isGrabbed = false; //we are no longer grabbing anything.
                }
            }
        }
    }
}
```

Cách dùng Raycast

- Khởi tạo biến trong Script:
 - Điều đầu tiên trong script của Raycast ta đang làm là khai báo 4 biến : RaycastHit, float, bool, vật rắn. Chúng ta sẽ sử dụng tất cả các biến này để quản lý logic. Trong hàm Start, chúng ta đăng ký một biến boolean và một biến float với giá trị mặc định. Điều này sẽ xảy ra khi game bắt đầu.

Cách dùng Raycast

- Trong hàm Update()

```
Debug.DrawRay(Camera.main.transform.position, Camera.main.transform.forward * rayLength, Color.red, 0.5f);
```

- Tất cả điều này là để vẽ một đường màu đỏ trong Scene khi game đang chạy
- Tham số đầu tiên xác định vị trí mà nó bắt đầu, tham số thứ hai là hướng di chuyển, tham số thứ ba xác định màu, và tham số thứ tư quy định cụ thể thời gian mà các đường thẳng được vẽ trên màn hình

Cách dùng Raycast

- Trong hàm Update()
- Dòng code sau sẽ gây ra Raycast, nếu Raycast chạm một cái gì đó.

```
if(Physics.Raycast(Camera.main.transform.position, Camera.main.transform.forward, out vision, rayLength))
```

- Tham số đầu tiên là vị trí raycast bắt đầu, ở đây chúng ta xác định là vị trí camera.
- Tham số thứ hai là hướng của raycast, ở đây chúng ta xác định là hướng về phía trước của camera
- Tham số thứ ba là tham số thông tin hit(điểm va chạm)
- Cuối cùng, tham số thứ tư là chiều dài của raycast, ở đây chúng ta đặt là rayLength

Cách dùng Raycast

- Trong hàm Update()

```
if(vision.collider.tag == "Interactive")
```

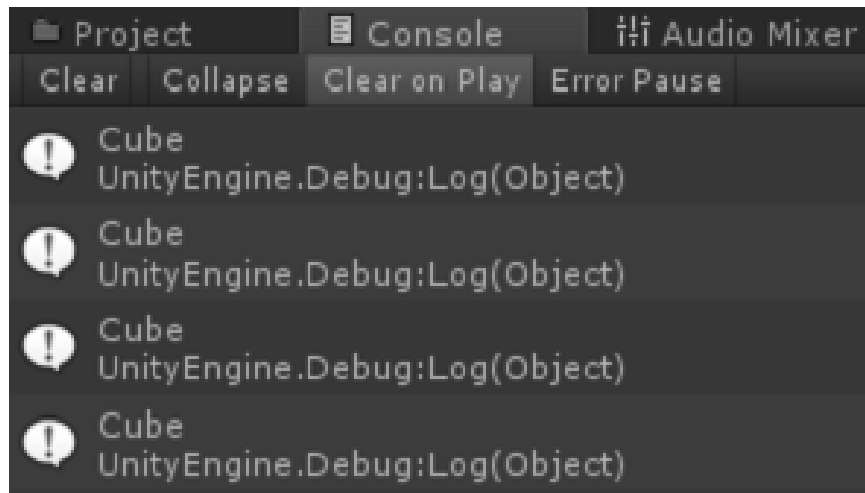
- Nếu đối tượng được gắn tag là "Interactive", mà chúng ta thiết lập trước đó, sau đó chúng ta sẽ thực hiện một công việc sau if.

Cách dùng Raycast

- Trong hàm Update()

```
Debug.Log (vision.collider.name);
```

- Dòng này chỉ đơn giản là xuất ra tên của các đối tượng, chúng ta có thể nhìn thấy nó trong cửa sổ console



Cách dùng Raycast

- Trong hàm Update()

```
if(Input.GetKeyDown(KeyCode.E) && !isGrabbed)

grabbedObject = vision.rigidbody;

grabbedObject.isKinematic = true;

grabbedObject.transform.SetParent(gameObject.transform);

isGrabbed = true;
```

- Nếu player đang tìm kiếm một đối tượng tương tác, điều kiện này sẽ cho phép người chơi nhấn phím E trên bàn phím
- Một khi player nhấn E và được phép nhận một đối tượng, chúng ta thực thi đoạn mã sau if

Cách dùng Raycast

- Trong hàm Update()

```
grabbedObject = vision.rigidbody;  
  
grabbedObject.isKinematic = true;  
  
grabbedObject.transform.SetParent (gameObject.transform);  
  
isGrabbed = true;
```

- Chúng ta gán cho grabbedObject một rigidbody để raycast có thể tương tác.
- Tiếp theo, chúng ta đặt grabbedObject là một đối tượng kinematic, điều này sẽ vô hiệu hóa các tương tác vật lý trên đối tượng

Cách dùng Raycast

- Trong hàm Update()

```
grabbedObject = vision.rigidbody;  
  
grabbedObject.isKinematic = true;  
  
grabbedObject.transform.SetParent (gameObject.transform);  
  
isGrabbed = true;
```

- Sau đó, chúng ta thiết lập các parent của grabbedObject là main camera. Điều này cho phép các đối tượng nắm lấy để di chuyển tương đối so với main camera
- Cuối cùng, chúng ta thiết lập isGrabbed là true để trò chơi của chúng ta biết rằng chúng ta đang sở hữu một đối tượng

Cách dùng Raycast

- Trong hàm Update()

```
else if(isGrabbed && Input.GetKeyDown(KeyCode.E))
```

- Nếu người chơi đang nắm giữ một đối tượng và nếu họ nhấn phím E.
- Nếu cả hai đều đúng, sau đó chúng ta thả các đối tượng bằng cách thực hiện các mã sau :

```
grabbedObject.transform.parent = null;
```

```
grabbedObject.isKinematic = false;
```

```
isGrabbed = false;
```

Cách dùng Raycast

- Trong hàm Update()

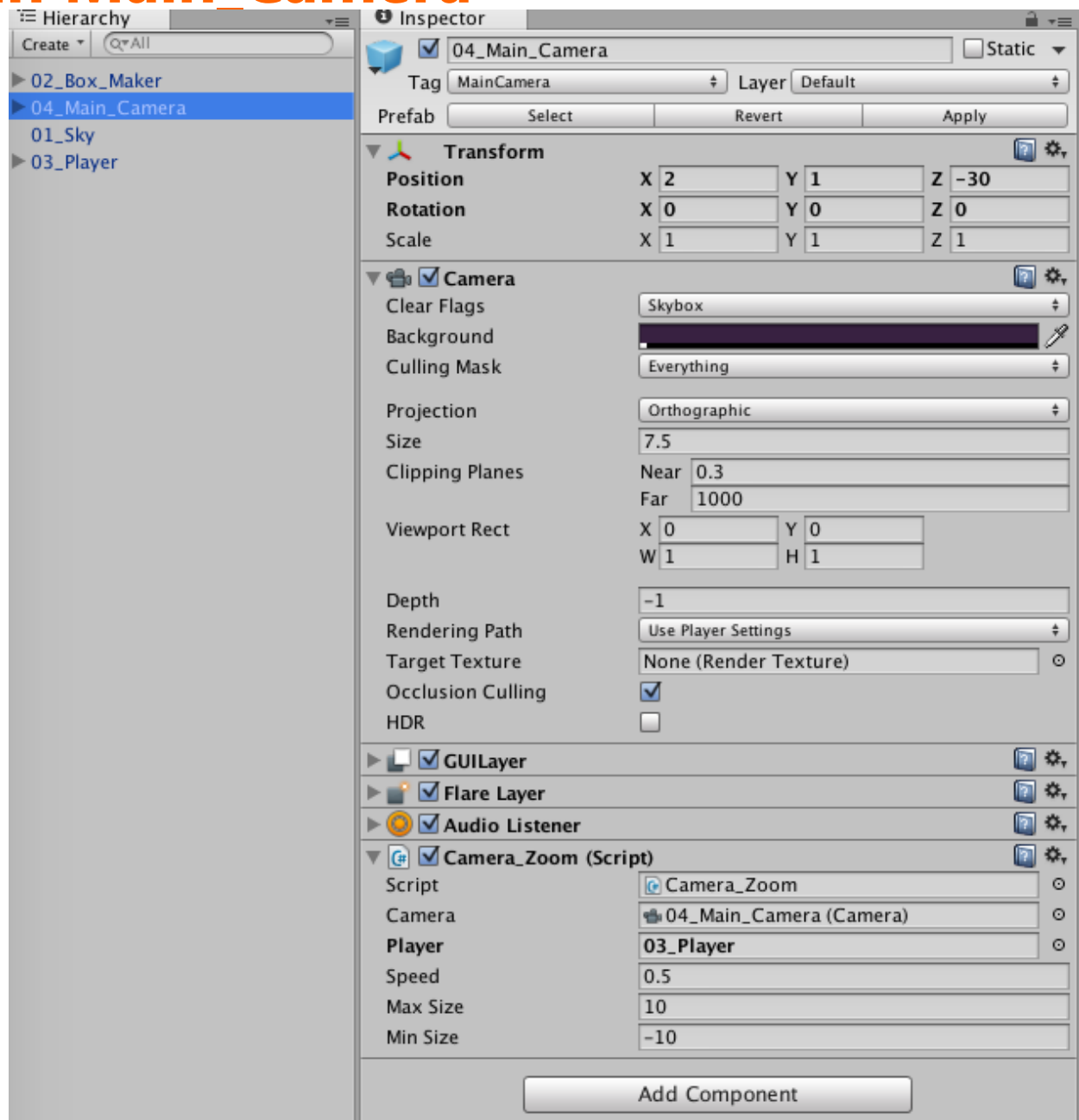
```
grabbedObject.transform.parent = null;  
  
grabbedObject.isKinematic = false;  
  
isGrabbed = false;
```

- Đầu tiên, chúng ta đặt cha của grabbedObject trở lại để nó không là con của camera.
- Sau đó, chúng ta set nó là non-kinematic để nó tương tác với vật lý.
- Cuối cùng, chúng ta thiết lập isGrabbed là false để chúng ta có thể nhận một đối tượng khác, nếu chúng ta muốn.



- Xây dựng scene Main_Camera
- Xây dựng Camera chạy theo Character

Cấu hình Main_Camera



Viết code cho Camera_Zoom.cs

- Khai báo các biến cần sử dụng

```
public Camera _camera;  
public GameObject _player;  
public float speed;  
public float MaxSize = 7.5f;  
public float MinSize = 5.5f;  
float CameraSize = 6f;
```

Viết code cho Camera_Zoom.cs

- Hàm start(): Lấy đối tượng được tạo trong cửa sổ Hierachy

```
void Start(){  
    //Lấy đối tượng được tạo trong cửa sổ Hierachy  
    _player = GameObject.Find("03_Player");  
}
```

Viết code cho Camera_Zoom.cs

- Hàm update(): Thay đổi camera theo vị trí của character

```
if (_player != null)
    CameraSize = 5f + _player.transform.position.y;

if (CameraSize >= MaxSize) {
    CameraSize = MaxSize;
}

if (CameraSize <= MinSize) {
    CameraSize = MinSize;
}
```

Viết code cho Camera_Zoom.cs

- Hàm update(): Cập nhật vị trí của camera

```
_camera.orthographicSize = Mathf.Lerp (_camera.orthographicSize, CameraSize, Time.deltaTime / speed);
```

Tổng kết

- Tạo Leaderboard
- Tạo Raycast



FPT POLYTECHNIC

THANK YOU!

www.poly.edu.vn