# Physic

## Collision

### Collision.relativeVelocity

* Vận tốc của hai vật va chạm
* Code:

if (collision.relativeVelocity.magnitude > 2)

{

    // do something

}

# Library

## Cách import 1 thư viện bên ngoài vào trong Unity

### Ví dụ cần import thư viện iTextPdf thì:

**Bước 1:** Create a new project on visual studio

**Bước 2:** Add iTextPdf qua: References -> Right click -> Manage NuGet Packages. Chọn tab browse -> Tìm iTextPdf -> Install

**Bước 3:** Configure the project to build as a Class library

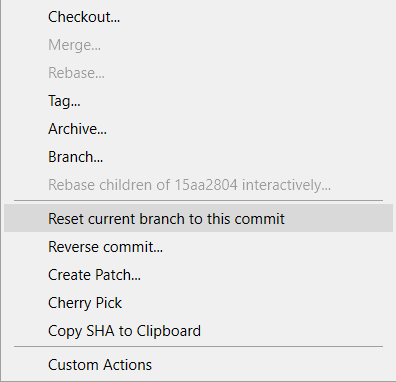
**Bước 4:** Build

**Bước 5:** Move all the .dll files from your output directory into the Unity project's Assets/Plugins

# API

# GIT

## Reset commit from a commit



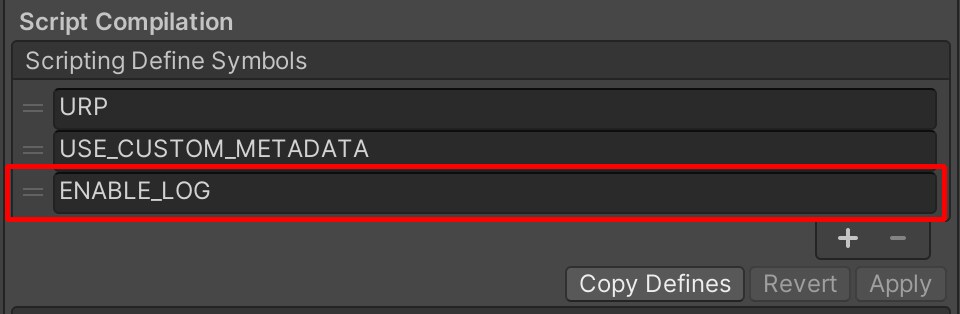
# Unity CS References

<https://github.com/Unity-Technologies/UnityCsReference/blob/master/Editor/Mono/Inspector/ShadowCascadeSplitGUI.cs>

# Optimize Performance

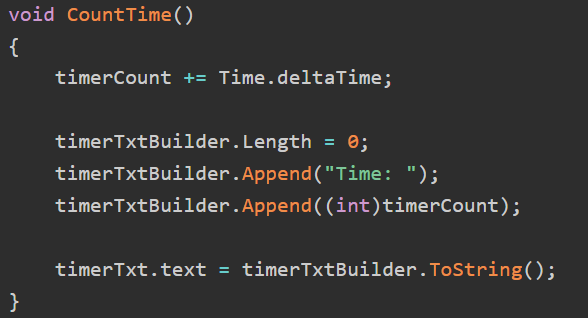
* Profiling
* Understand the Unity PlayerLoop
* ENABLE\_LOG

Đặt câu lệnh log trong #if ENABLE\_LOG



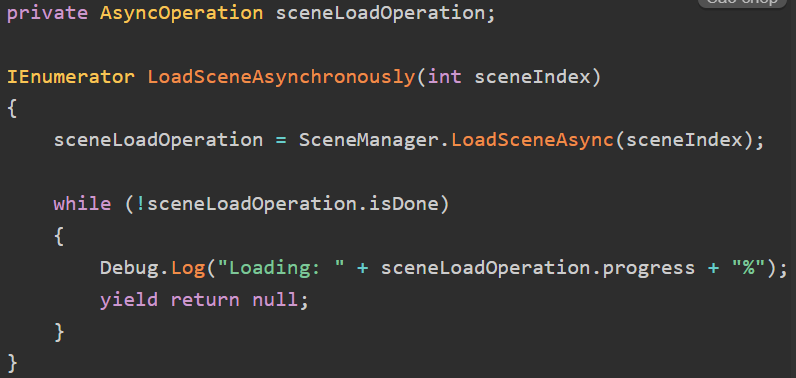
* Avoid adding components at runtime
* Cache GameObjects and components: GameObject.Find, GameObject.GetComponent and Camera.main calling them in Start, avoid in Update
* Use Object pools
* Use ScriptableObjects
* Checking collided use CompareTag
* Use StringBuilders



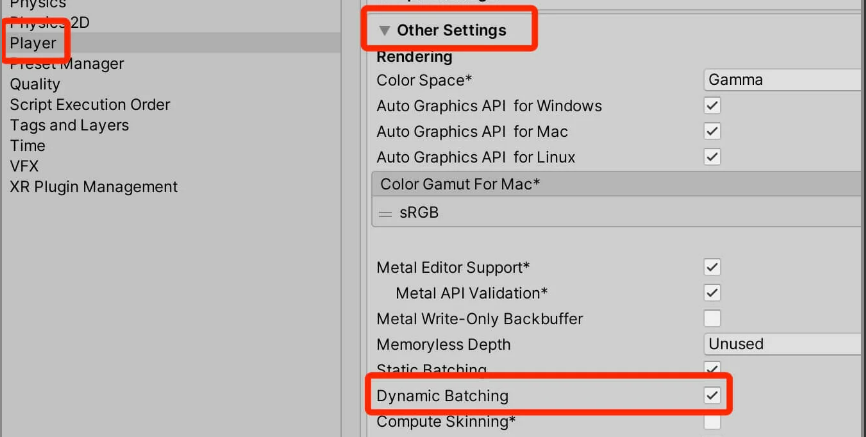


* Avoid using raycast
* Use sqrMagnitude thay cho Vector3.magnitude hoặc Vector3.Distance()
* Create Prefab bên ngoài scene
* Use LoadSceneAsync

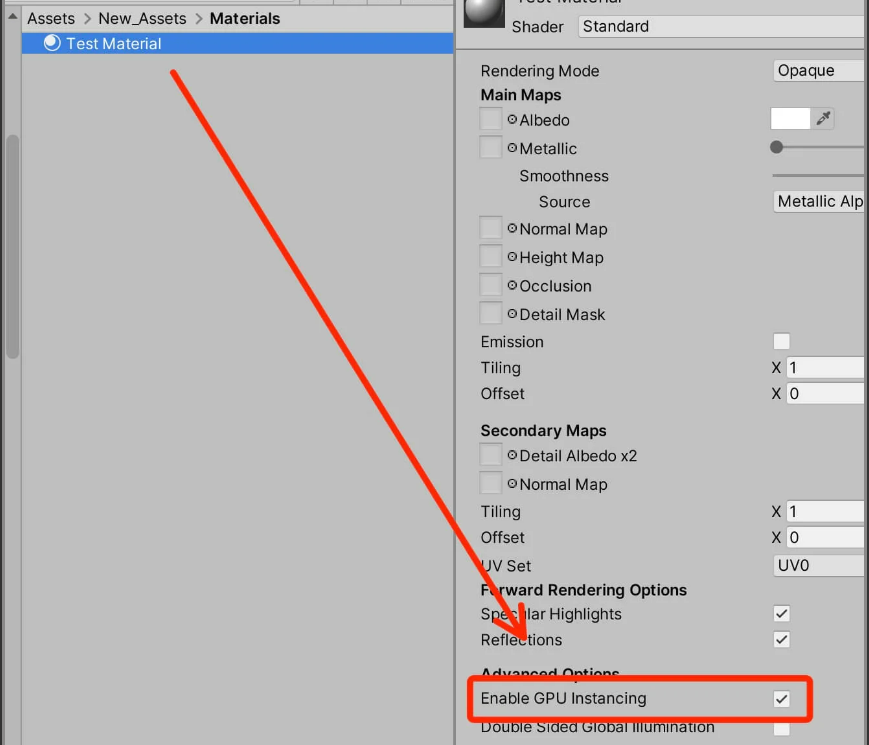
Hiển thị màn hình tải thời gian thực cho người dung



* Use for instead foreach
* IL2CPP : ngôn ngữ trung cấp cho C++, Mono là triển khai mã nguồn mở của .Net. Biên dịch IL2CPP lâu hơn nhưng khi biên dịch ra mã máy trên thiết bị người dùng cuối nhanh hơn.
* Gắn 1 collider trên gameobject là Static nếu nó không di chuyển. Nên sử dụng Discrete thay vì Continous trong Collision Detection
* Sử dụng Layer để tránh va chạm
* Avoid use polygon, sphere… collider
* Phân nhóm đối tượng động và tĩnh, tránh việc phải render lại các đối tượng
* Turn on dynamic batching

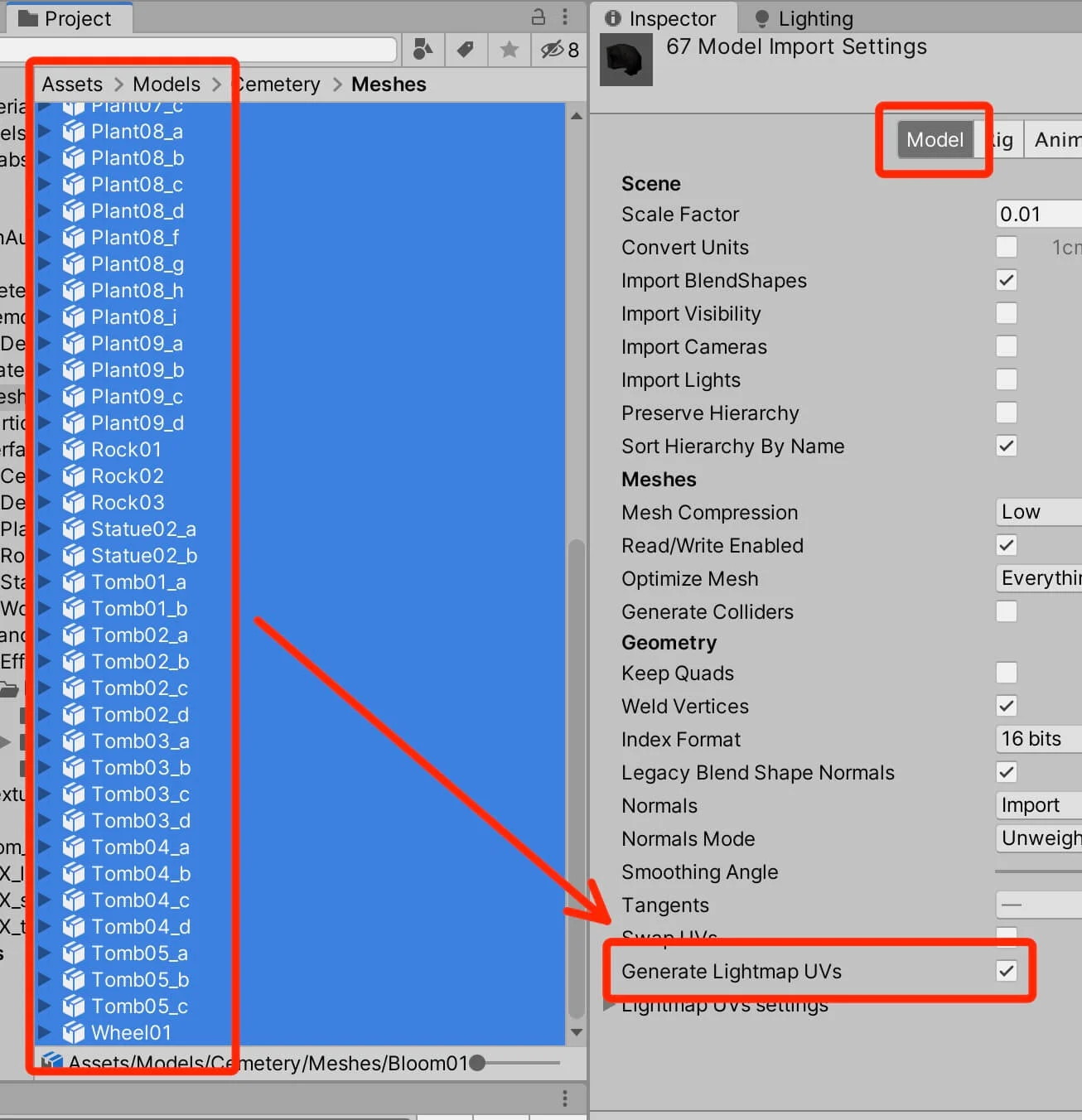


* Reducing draw calls with GPU instancing

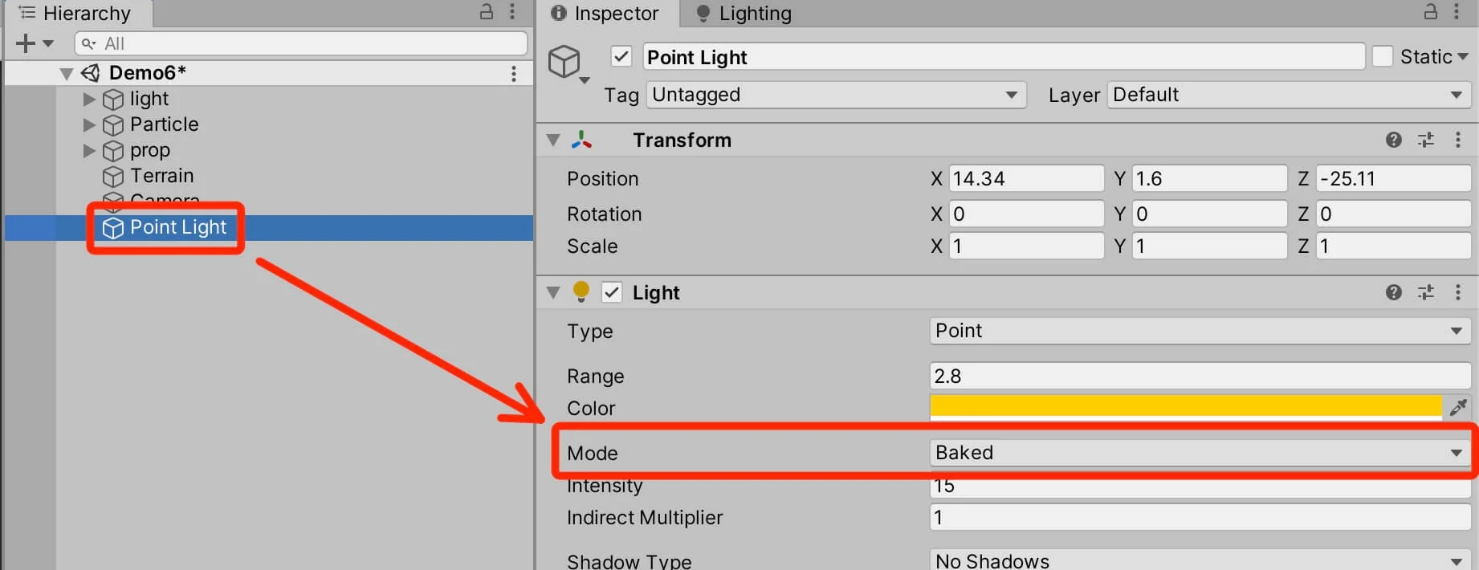


* Tối ưu light

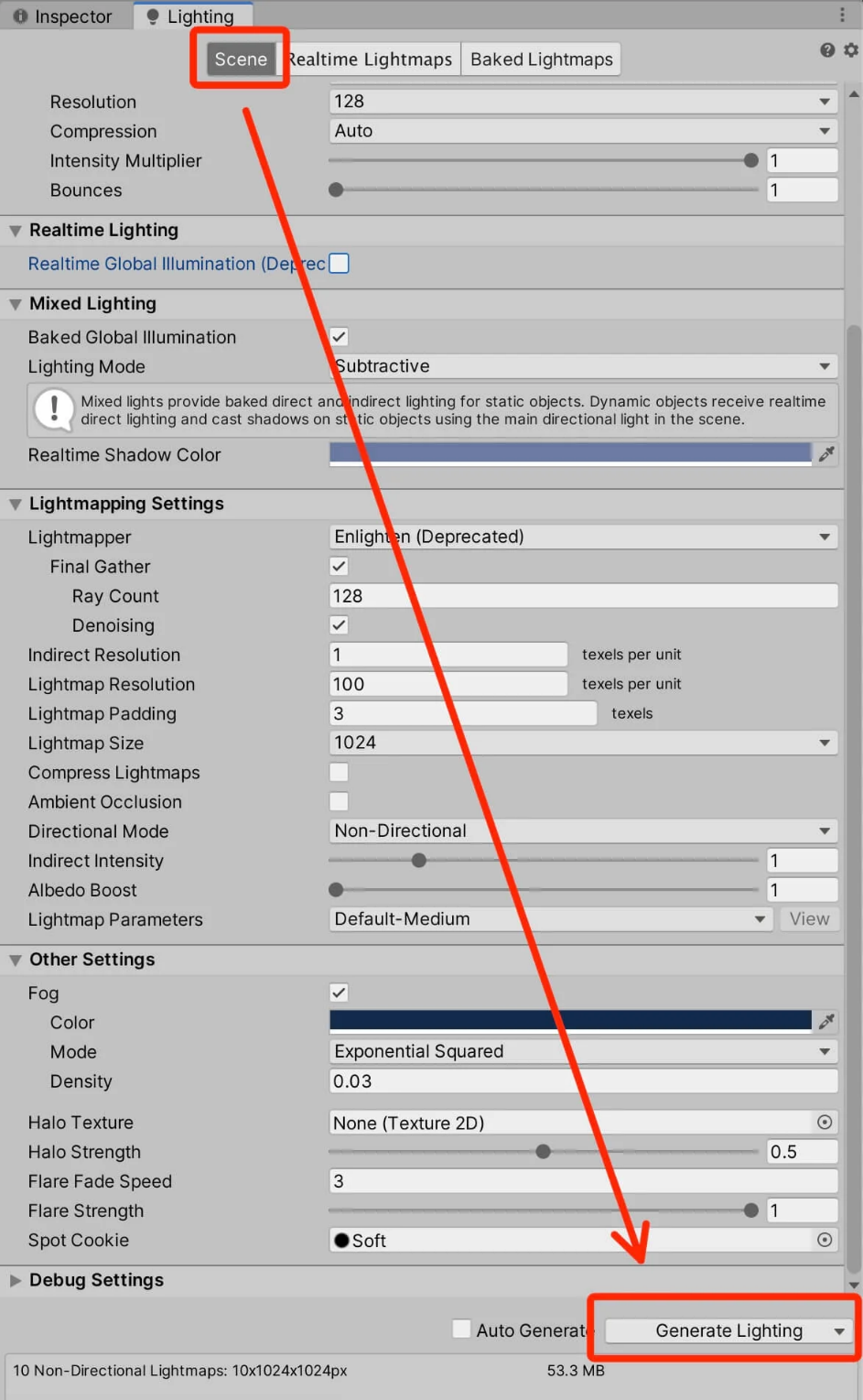
When you bake lights, Unity performs calculations for baked lights in the scene and saves the results as lighting data.



Next, you need to select the light you want to bake in the scene, and in the Inspector tab change the Mode to Baked:

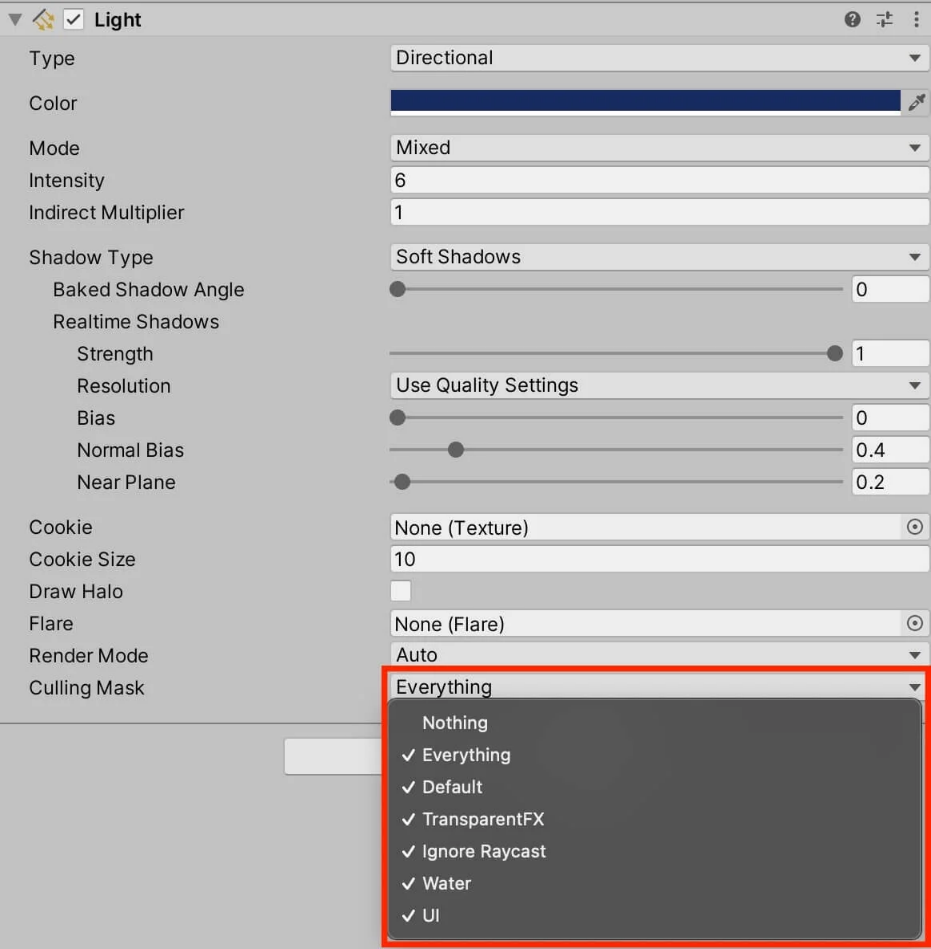


In the Lighting tab, click on the Scene tab and then scroll all the way to the bottom and press the Generate Lighting Button:



* Even if we turn off the Point Light object, we will still see the light in the game.

Chúng ta cũng có thể tận dụng thuộc tính Culling Mask cho mọi thành phần Light. Mặt nạ lọc hoạt động giống như các lớp va chạm, nó xác định các lớp nào bị ảnh hưởng bởi thành phần ánh sáng:



* Optimize sprite, sound
* Disable Raycast Target For Non-Interactive UI Elements



1. Mathf.InverseLerp(float a, float b, float value)

Mathf.Lerp() và Mathf.InverseLerp() là 2 phương thức trái ngược nhau.

* Mathf.Lerp(float a, float b, float t): trả về giá trị từ a đến b tại thời điểm t.
* Mathf.InverseLerp(float a, float b, float value): trả về thời điểm t tại giá trị value.

Ví dụ: Dưới đây là một cách để tạo hiệu ứng khi di chuyển 1 gameobject. Có thể tham khảo chi tiết tại đây:

<https://assetstore.unity.com/packages/tools/gui/simple-scroll-snap-140884>

