

Bài 2 : Khởi tạo dự án 2D, tạo Game Object và Sprite

Giảng viên:

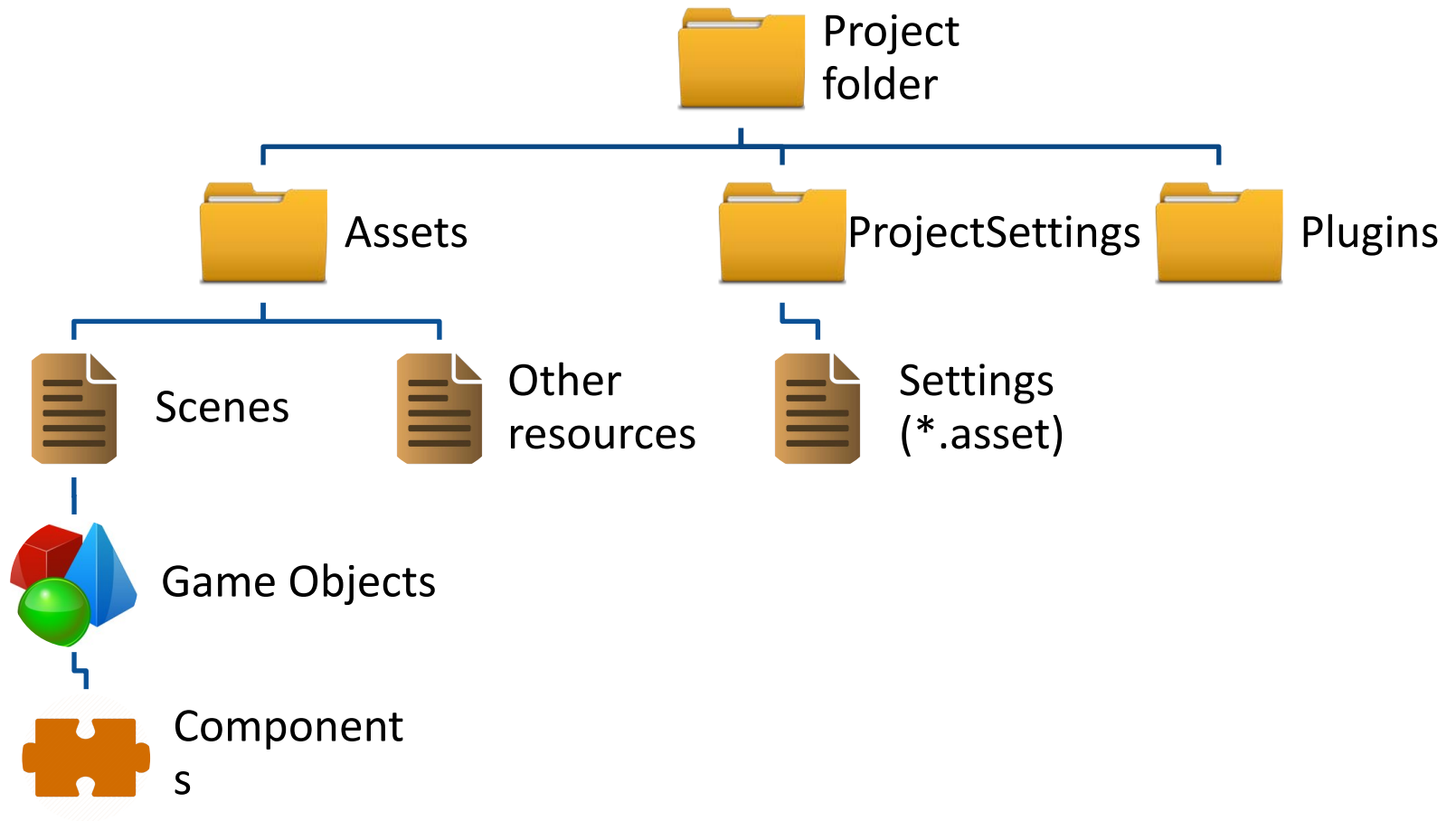
MỤC TIÊU

- **Cấu trúc Project**
- **A. Khởi tạo và cấu hình dự án Game 2D**
- **B. Tạo các đối tượng cơ bản**
 - **1. Game Object**
 - **2. Sprite**
 - 3. *Animation và điều khiển hành động nhân vật*
 - 4. *Prefab*
 - 5. *Script và điều khiển máy trạng thái*
 - 6. *Thành phần vật lý và xử lý va chạm*
 - 7. *Sử dụng Text,*
 - 8. *Sử dụng Particle System*
 - 9. *Chuyển đổi màn chơi*
 - 10. *Sound*
 - 11. *Design Pattern trong Game*

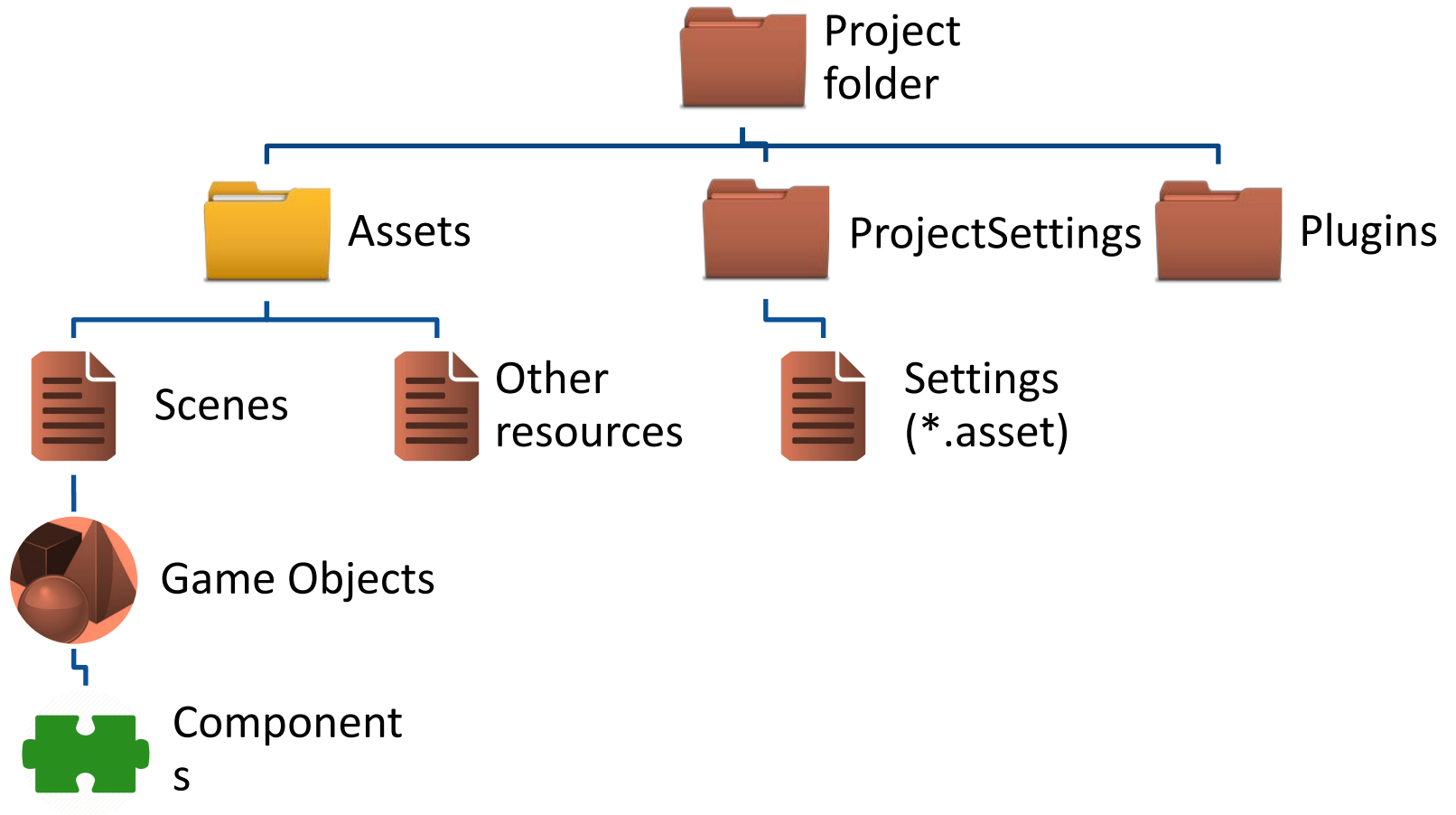
Nội dung

- **Cấu trúc Project**
- **A. Khởi tạo và cấu hình dự án Game 2D**
- **B. Tạo các đối tượng cơ bản**
 - **1. Game Object**
 - **2. Sprite**
 - 3. *Animation và điều khiển hành động nhân vật*
 - 4. *Prefab*
 - 5. *Script và điều khiển máy trạng thái*
 - 6. *Thành phần vật lý và xử lý va chạm*
 - 7. *Sử dụng Text,*
 - 8. *Sử dụng Particle System*
 - 9. *Chuyển đổi màn chơi*
 - 10. *Sound*
 - 11. *Design Pattern trong Game*

Cấu trúc Project



Cấu trúc Project



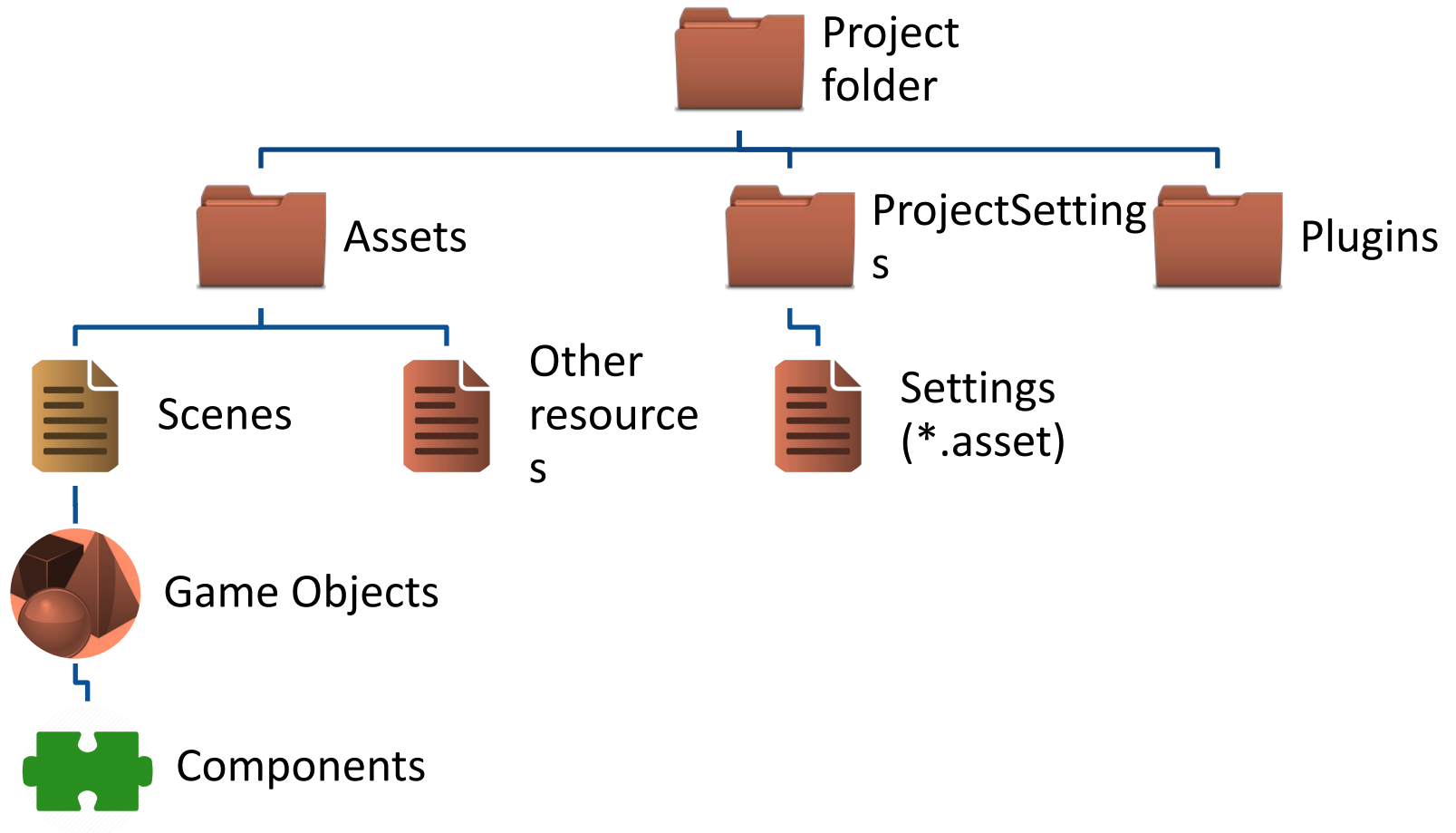
Cấu trúc Project- Asset

- Asset là những tài nguyên xây dựng nên một dự án Unity như:
 - Tập tin hình ảnh;
 - Mô hình 3D;
 - Các tập tin âm thanh.
- Các tập tin mà chúng ta dùng để tạo nên trò chơi là tài sản (Assets).
- Tất cả các tập tin, thư mục của các dự án Unity đều được lưu trữ trong một thư mục có tên "Assets".

Cấu trúc Project- Asset

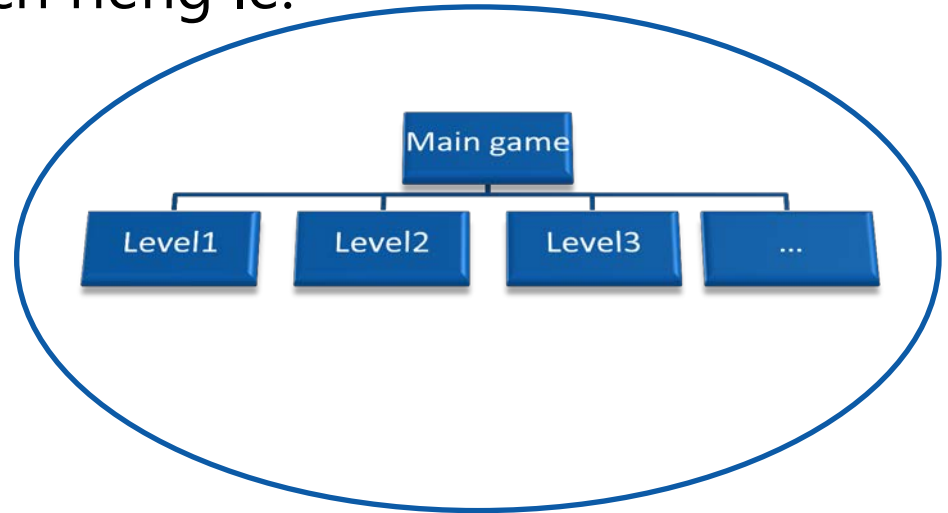


Cấu trúc Project- Scenes

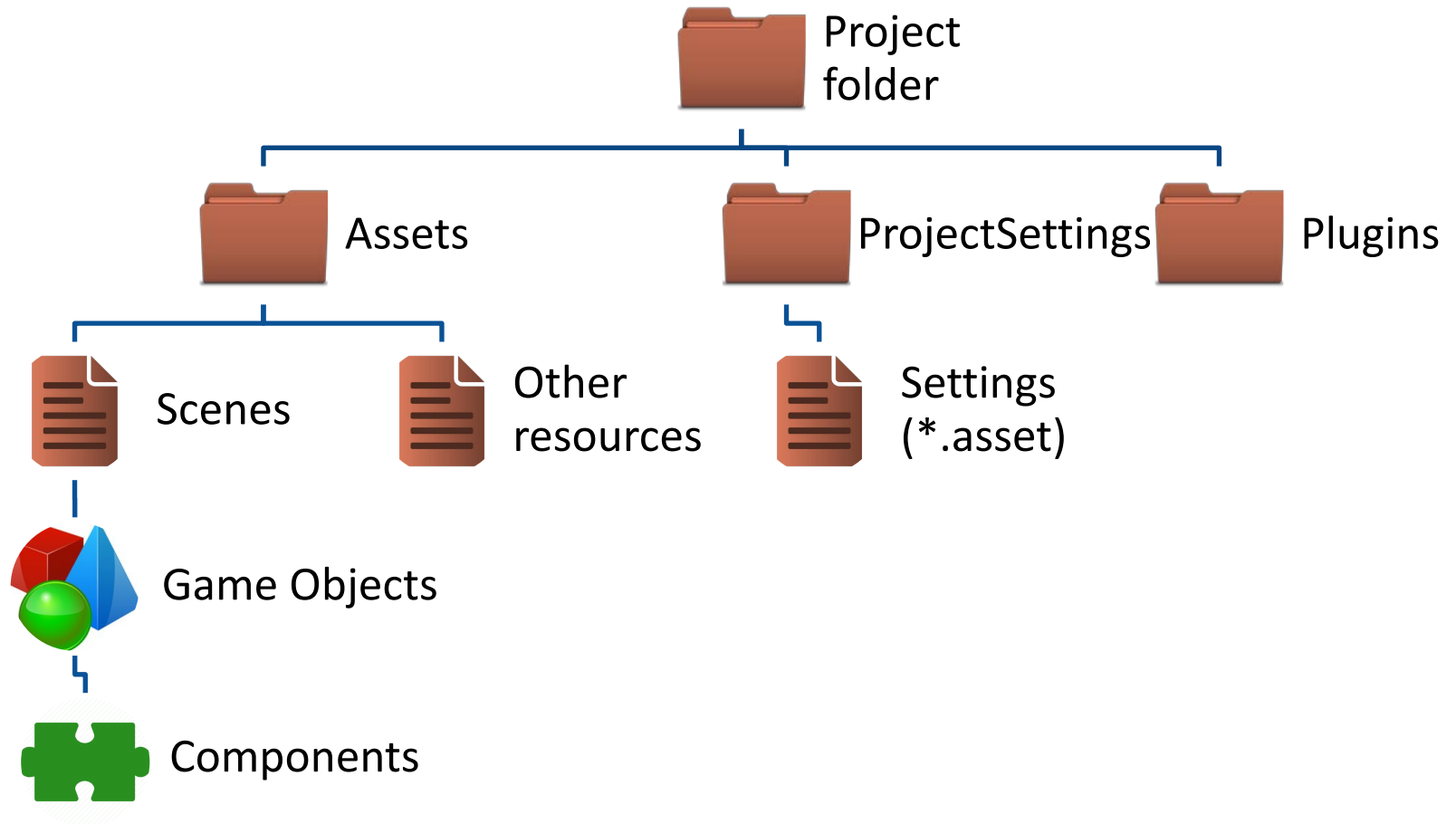


Cấu trúc Project- Scenes

- Trong Unity, scene- một cảnh (hay một phân đoạn) nghĩa là một màn chơi riêng biệt hoặc một khu vực hay thành phần có trong nội dung của trò chơi (ví dụ như Game menu).
- Bằng cách tạo nên nhiều Scene cho trò chơi, chúng ta có thể phân phối thời gian tải hoặc kiểm tra các thành phần khác nhau của trò chơi một cách riêng lẻ.
- Mỗi scene là một level
- Một game có thể có nhiều levels



Cấu trúc Project- GameObject



Cấu trúc Project- GameObject

- Một đối tượng cụ thể trong game gọi là một game object, có thể là nhân vật, đồ vật nào đó. Ví dụ: cây cối, xe cộ, nhà cửa, người...
- Khi Assets được sử dụng trong Scene, chúng trở thành Game Object – một thuật ngữ được sử dụng trong Unity (đặc biệt là trong mảng lập trình).
- Game Object chứa ít nhất một thành phần là Transform.
- Transform là thông tin về vị trí, góc xoay và tỉ lệ của đối tượng, tất cả được mô tả bởi bộ 3 số X, Y, Z trong hệ trục tọa độ. Thành phần này có thể được tùy biến lại trong quá trình lập trình nhằm thay đổi vị trí, góc quay và tỉ lệ của đối tượng (so với đối tượng gốc) qua các đoạn mã.

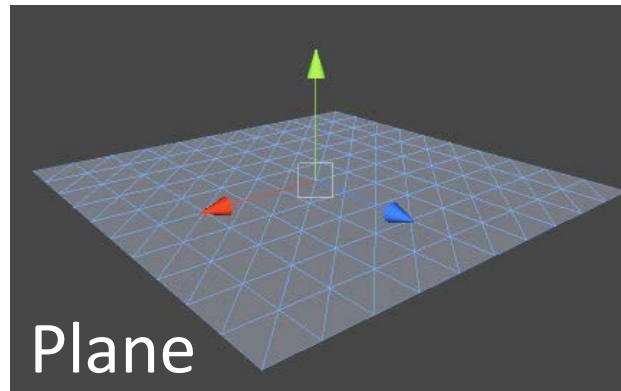
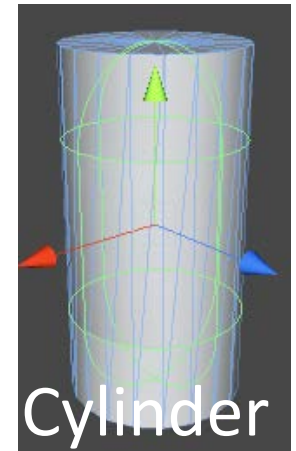
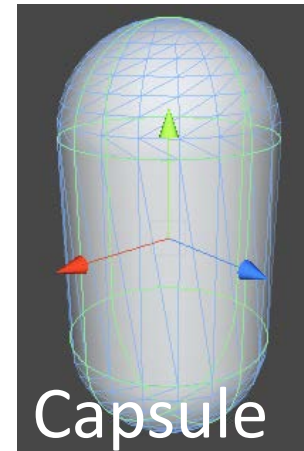
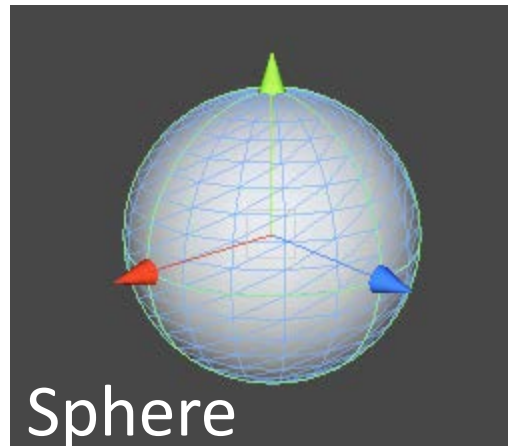
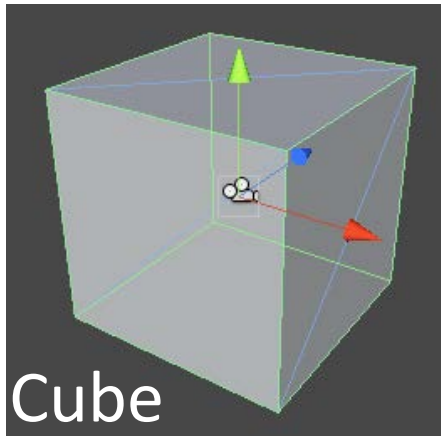
Cấu trúc Project- GameObject

- GameObject là một lớp cơ bản trong Unity
- Mỗi GameObject chứa nhiều thành phần(components)
- Mỗi Component chứa nhiều properties
- Có thể thay đổi giá trị property để điều khiển một object.
- Có thể thêm/sửa/xóa một component trong mỗi Gameobject.



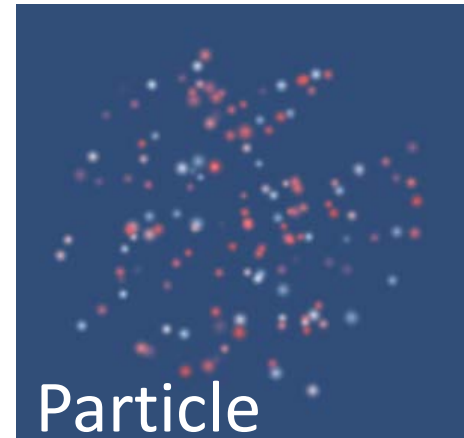
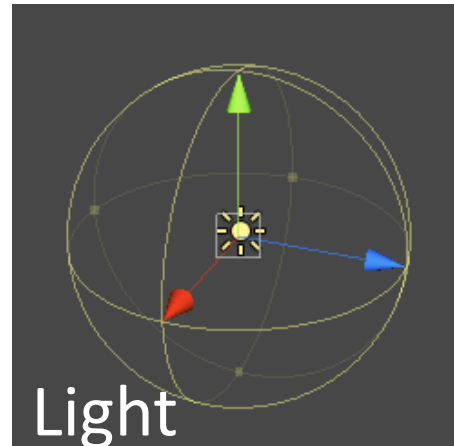
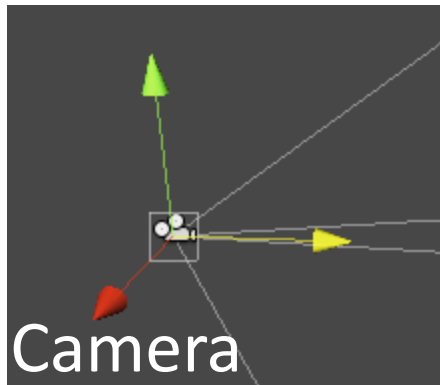
Cấu trúc Project- GameObject

- Ví dụ về GameObject:
 - Object cơ bản: Cube, Sphere, Capsule, Cylinder, Plane



Cấu trúc Project- GameObject

- Ví dụ về GameObject:
 - Object đặc biệt: Empty Object , Camera, Light , Particle

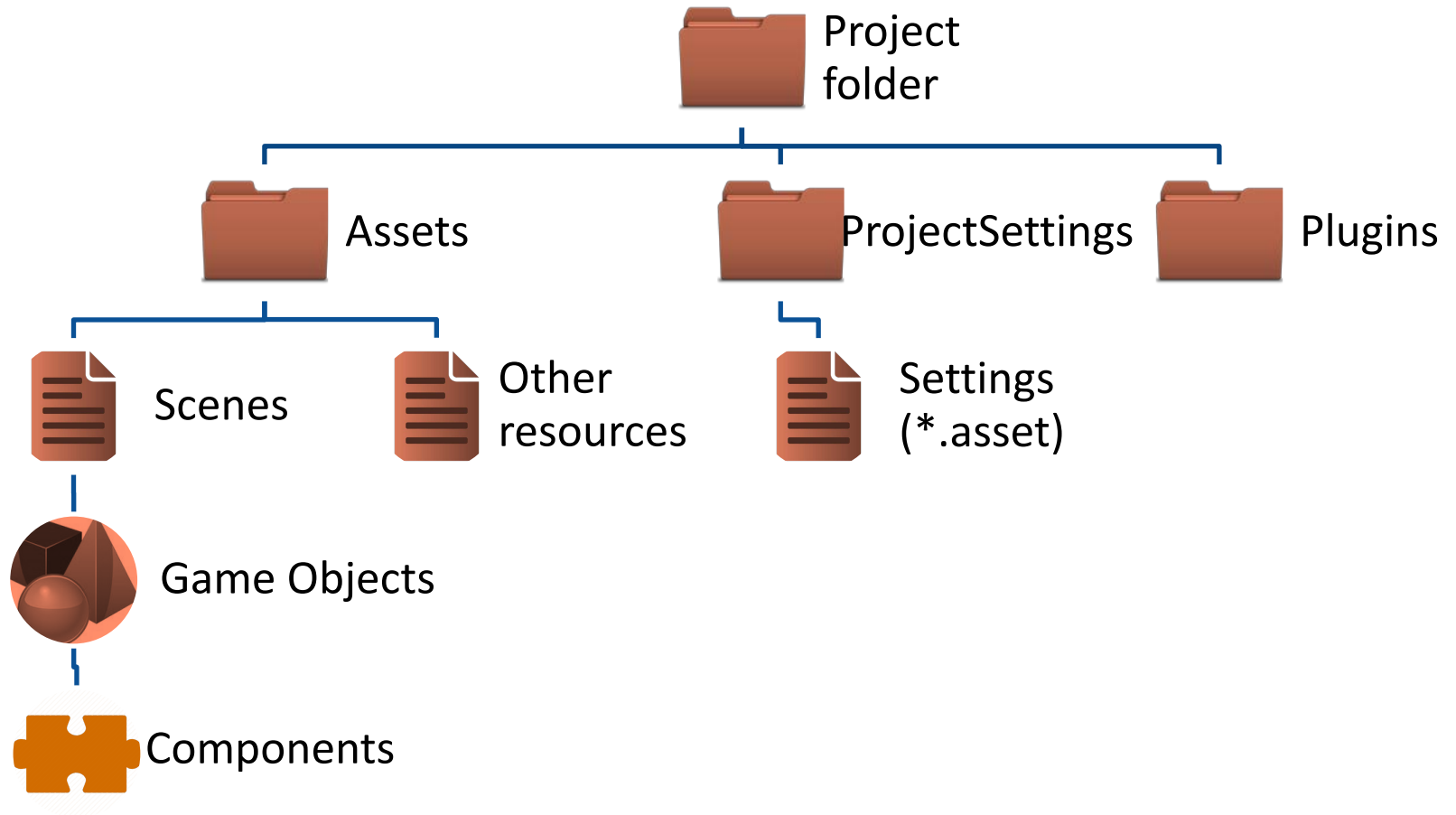


Cấu trúc Project- GameObject

- Ví dụ về GameObject:
 - Object mô hình: table, house, man, room,...



Cấu trúc Project- Components



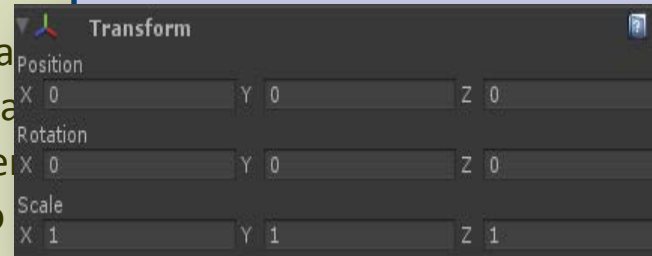
Cấu trúc Project- Components

- Một Game Object sẽ có nhiều thành phần cấu tạo nên nó như là hình ảnh (sprite render), tập hợp các hành động (animator), thành phần xử lý va chạm (collision), tính toán vật lý (physical), mã điều khiển (script), các thành phần khác... Mỗi thứ như vậy gọi là một **component** của Game Object.
- Bằng cách “gắn” (attach) chúng vào trong Game Object, chúng ta có thể ngay lập tức áp dụng tác động của chúng lên đối tượng.
- Những Components phổ biến trong quá trình phát triển trò chơi đều được Unity hỗ trợ sẵn.
- Để tạo nên các yếu tố tương tác trong trò chơi, chúng ta sẽ sử dụng Script (mã kịch bản), chúng cũng được xem như là một Components trong Unity.

Cấu trúc Project- Components

Game
object

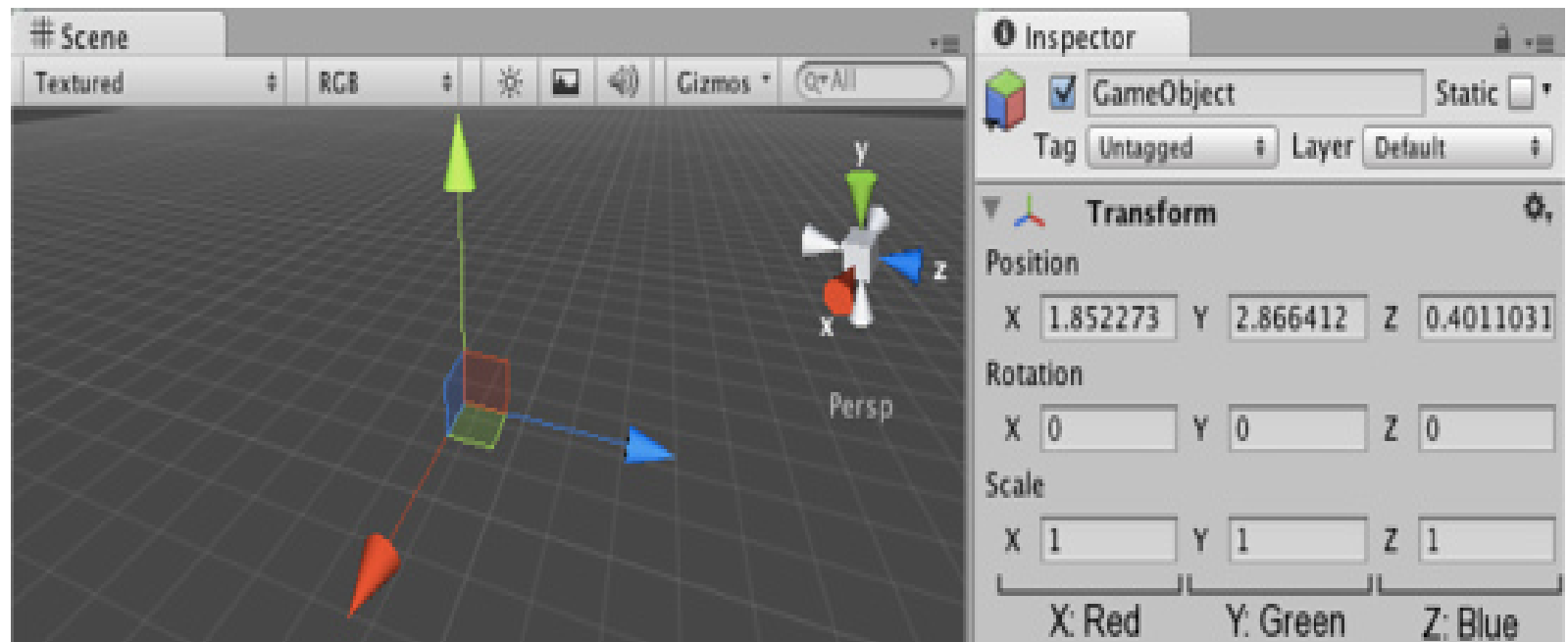
- transform
- rigidbody
- camera
- Light
- anima
- consta
- rende
- audio
- guiText
- networkView
- guiTexture
- collider
- hingeJoint
- particleEmitter
- layer
- active
- tag



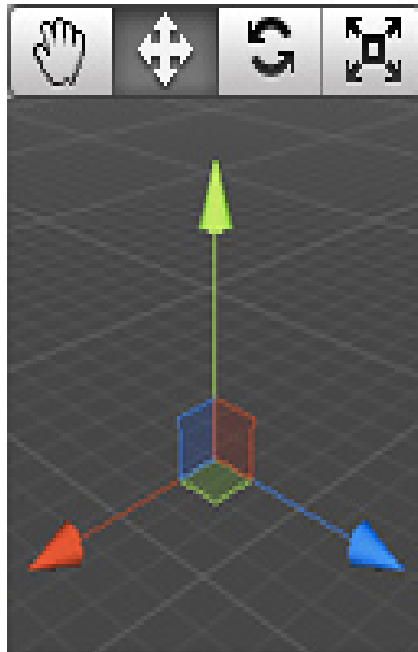
Component

Cấu trúc Project- Transform Components

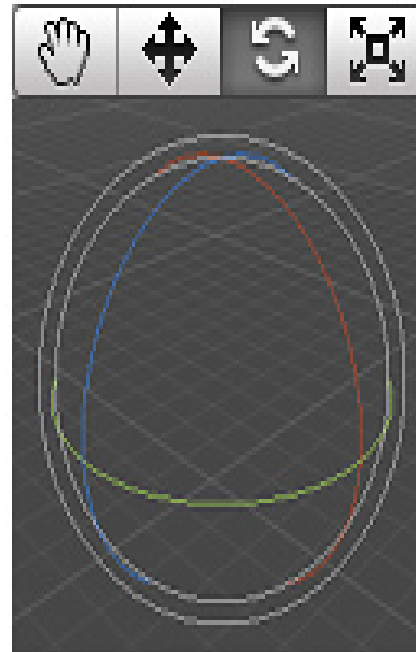
- Position: in X, Y and Z coordinates
- Rotation: around X, Y, and Z axes, measured in degrees
- Scale: along X, Y, and Z axes Value "1" the original size.



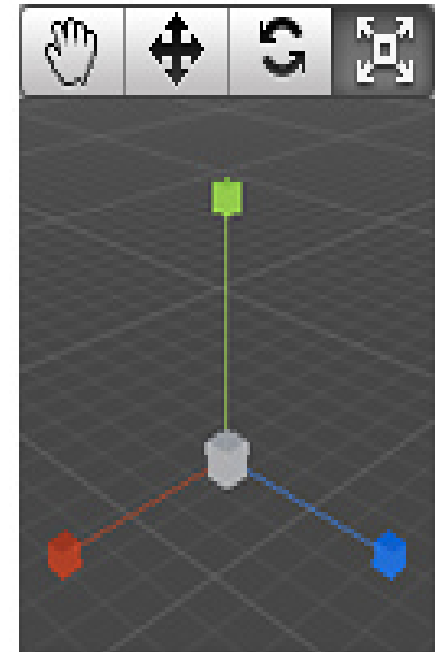
Cấu trúc Project- Object Transform



Translate (W)



Rotate (E)



Scale (R)

Cấu trúc Project- Script

- Script được Unity xem như một Component.
- Unity cung cấp cho chúng ta khả năng viết Script bằng cả 3 ngôn ngữ là JavaScript, C# và Boo (một dẫn xuất của ngôn ngữ Python).
- Unity đã xây dựng sẵn một tập hợp đa dạng các Class, Function, ... để có thể ứng dụng trong quá trình lập trình.
- Để viết Script, chúng ta sẽ làm việc với một trình biên tập Script độc lập của Unity, hoặc với chương trình Mono Developer được tích hợp và đồng bộ với Unity.
- Mono Developer là một IDE khá tốt để lập trình khi cung cấp đầy đủ các chức năng của một IDE hoàn chỉnh tương tự Visual Studio. Mã nguồn viết trên Mono Developer sẽ được cập nhật và lưu trữ trong dự án Unity.


Cấu trúc Project- Prefabs

- Là một khái niệm trong Unity dùng để sử dụng lại các đối tượng giống nhau có trong game bằng cách chỉ cần khởi tạo lại các giá trị vị trí, tỉ lệ biến dạng và góc quay từ một đối tượng ban đầu.
- Ví dụ: Các đối tượng là đồng tiền trong game Mario đều có xử lý giống nhau, nên ta chỉ việc tạo ra một đối tượng ban đầu, các đồng tiền còn lại sẽ sử dụng prefabs. Hoặc khi ta lát gạch cho một cái nền nhà, các viên gạch cũng được sử dụng là prefabs.



DEMO

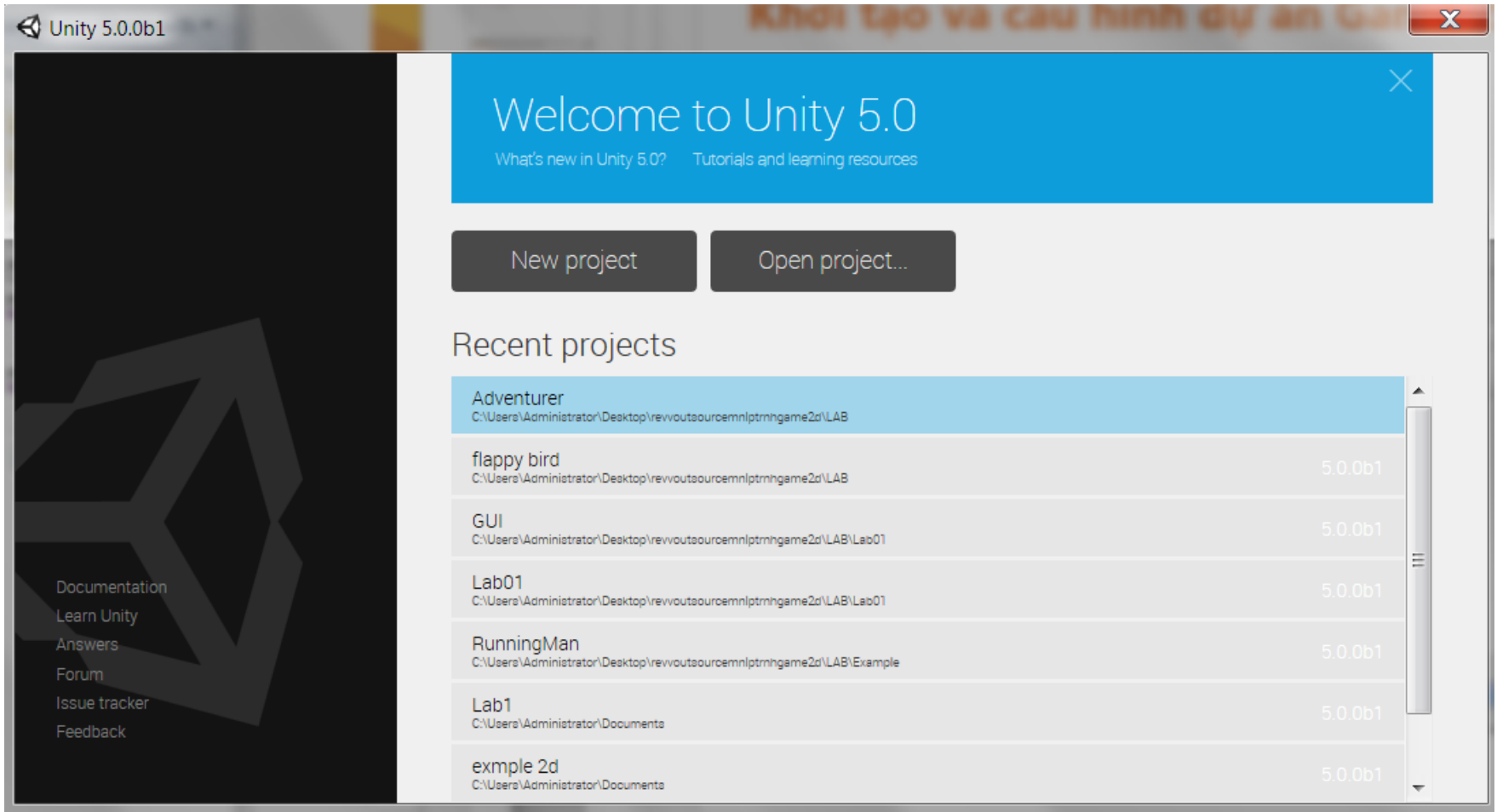
Cấu trúc
Project

A large white hand cursor icon, resembling a computer mouse pointer, is positioned below the word 'DEMO'. The index finger of the hand is pointing upwards towards the letter 'O' in 'DEMO'.

Khởi tạo và cấu hình dự án Game 2D

Tạo dự án

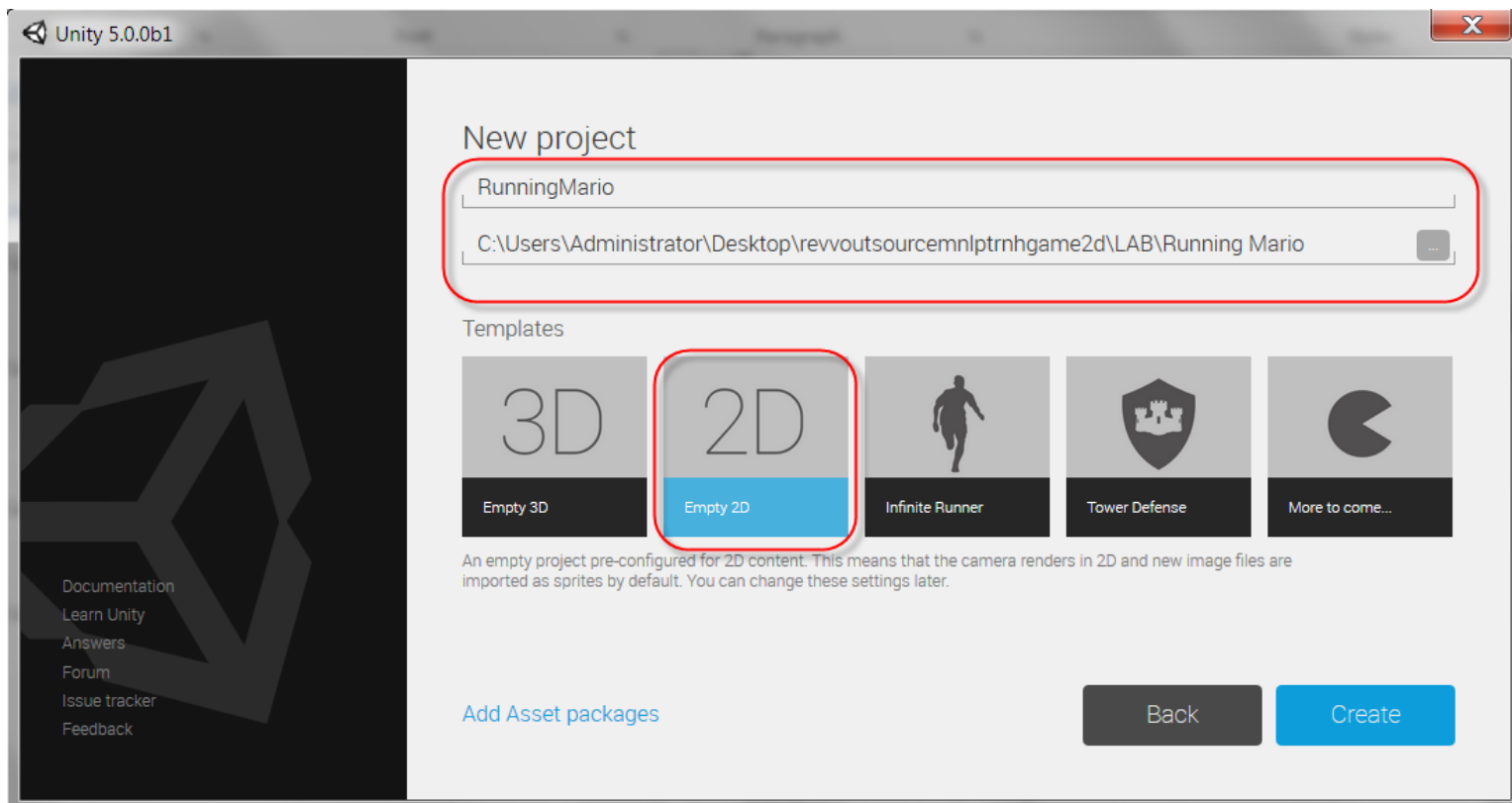
- Bật Unity lên, vào menu File/New Project



Khởi tạo và cấu hình dự án Game 2D

Tạo dự án

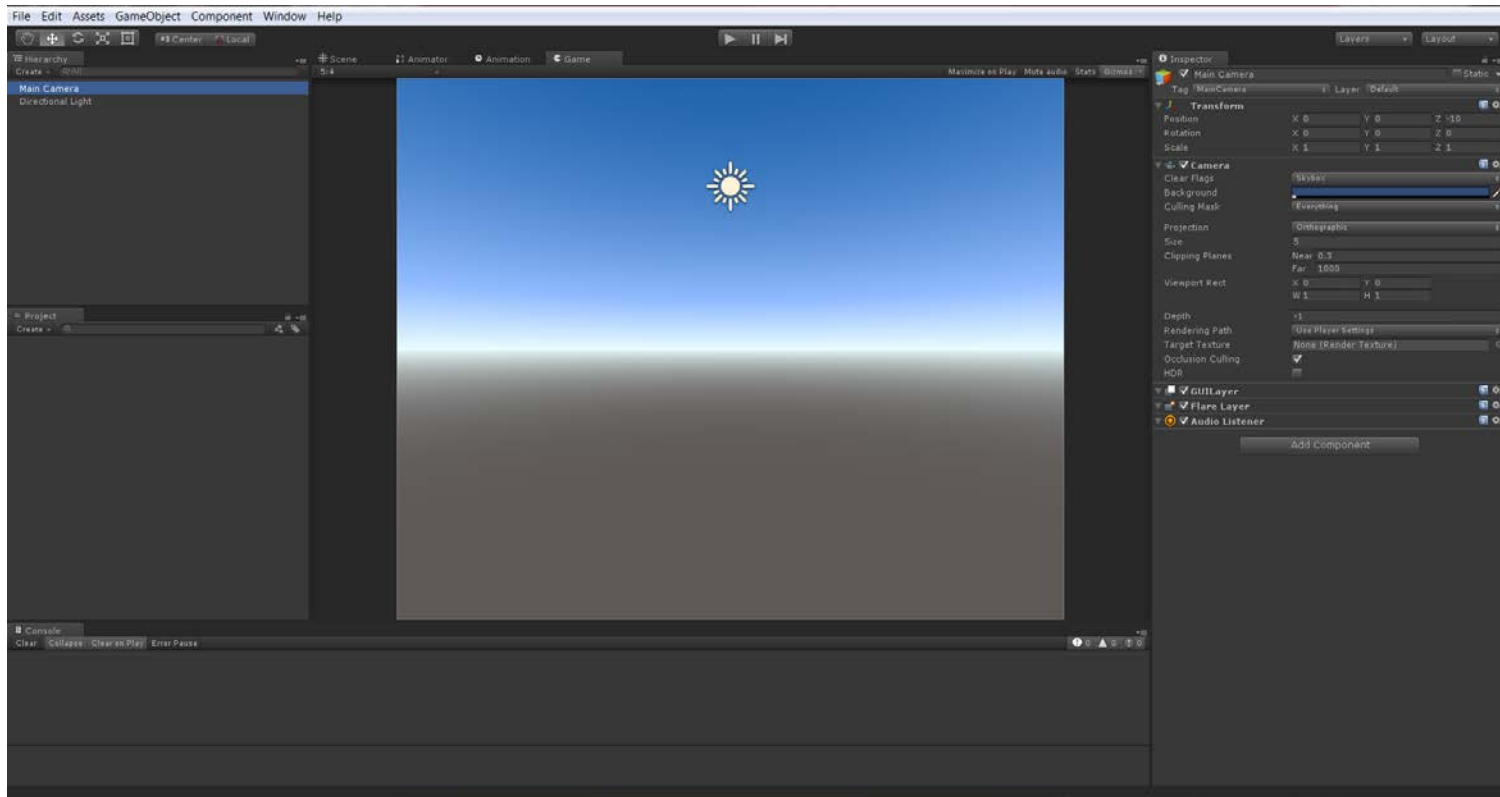
- Chọn loại project là 2D sau đó gõ tên project xong nhấn Create Project.



Khởi tạo và cấu hình dự án Game 2D

Tạo dự án

- Chọn loại project là 2D sau đó gõ tên project xong nhấn Create Project. Kết quả như sau:



Khởi tạo và cấu hình dự án Game 2D

Cấu hình dự án 2D

- Sau khi tạo dự án xong, ở Hierarchy sẽ có một đối tượng là Main Camera. Chúng ta cần điều chỉnh kiểu hiển thị Scene là 2D
- Ở cửa sổ Hierarchy chọn Main Camera. Ở cửa sổ Inspector sẽ hiển thị các thông số của camera, ta sẽ chọn lại giá trị Projection hay phép chiếu là phép chiếu vuông góc thay cho phép chiếu phối cảnh.



Khởi tạo và cấu hình dự án Game 2D

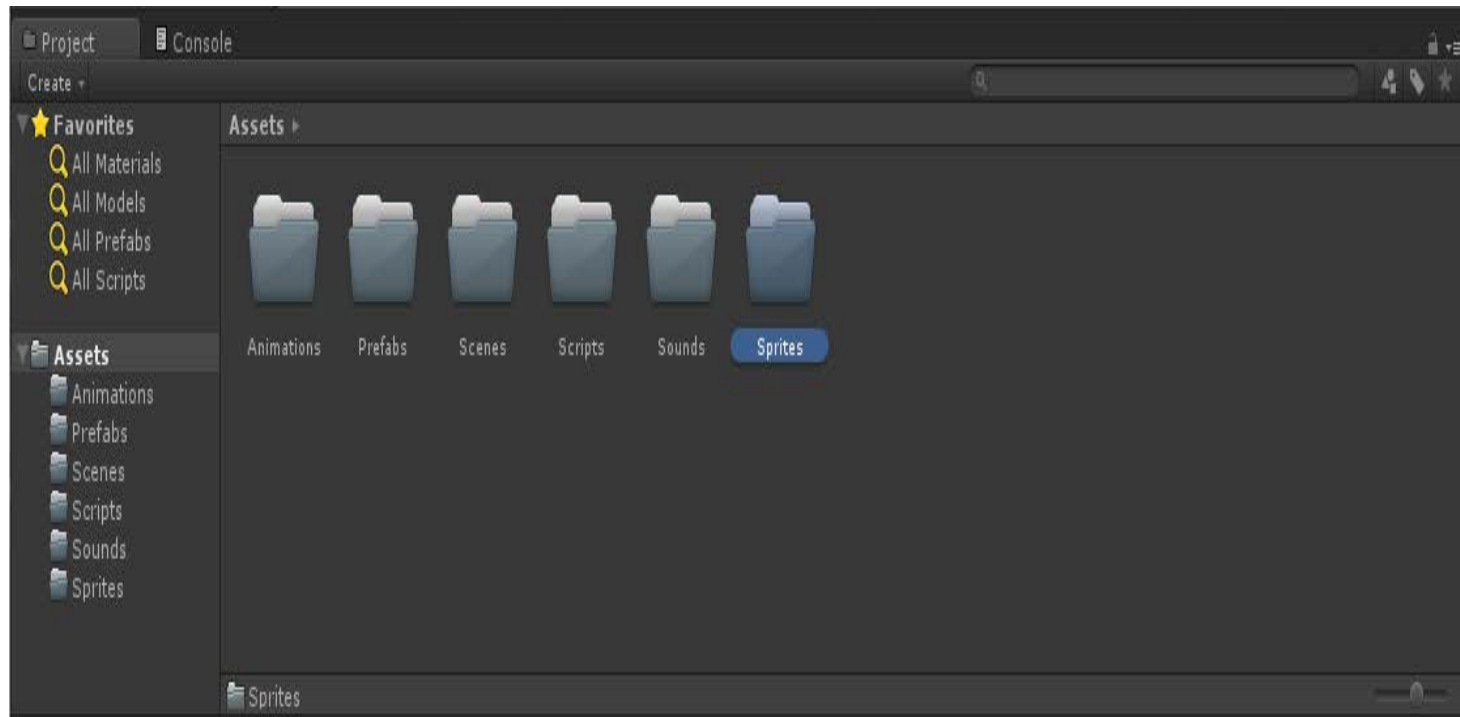
Cấu hình dự án 2D

- Tiếp theo, ở cửa sổ Project, ta sẽ tạo sẵn các thư mục để chứa tất cả các tài nguyên có sử dụng trong game sau này lần lượt là: Animations, Prefabs, Scripts, Sprites, Sounds, Scenes.
- R-Click vào thư mục Assets ở cửa sổ Project, ta chọn Create, chọn Folder.

Khởi tạo và cấu hình dự án Game 2D

Cấu hình dự án 2D

- Kết quả như hình sau:





DEMO

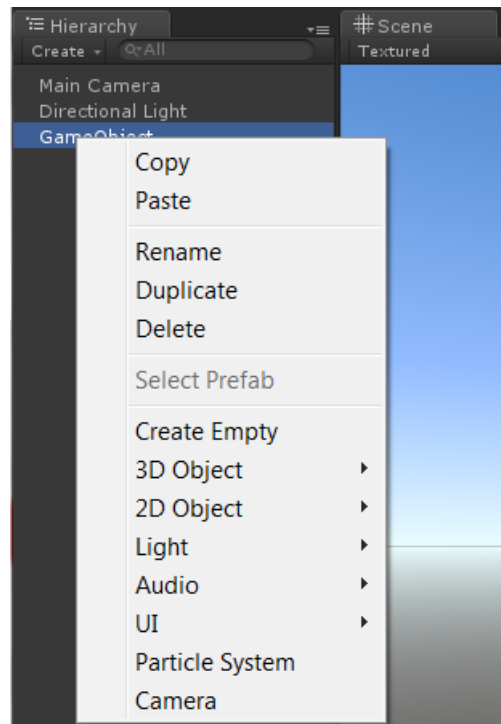
Khởi tạo dự
án Game 2D



Game Object

Tạo một Empty Object

- Ở menu ta chọn GameObject, chọn Create Empty.
- Ta có thể đổi tên, sao chép hay xoá các đối tượng thông qua cửa sổ này.



Game Object

Tạo một Empty Object

- Empty game object là một đối tượng đơn giản nhất, khi mới tạo ra.
- Empty game object chỉ chứa các thông số biến đổi (transform) ngoài ra nó sẽ không chứa bất kỳ một component nào cả.
- Có thể thêm các component hoặc để nhóm các đối tượng khác lại với nhau thành một nhóm, hoặc sử dụng các empty object cho các mục đích khác (sẽ ứng dụng sau)...

Game Object

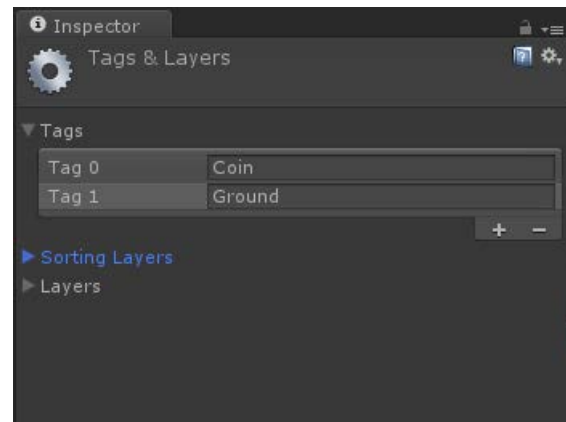
Tạo một Parent object và Child object

- **Parent object:** là một đối tượng chứa các đối tượng con khác, gắn liền với nhau.
- **Child object:** là một đối tượng nằm trong một đối tượng khác, gắn liền với nhau.
- Để tạo các đối tượng parent và child, ta chỉ việc kéo thả một đối tượng có sẵn vào trong đối tượng đã có trong cửa sổ Hierarchy. Khi thay đổi các đối tượng con thì chỉ có tác động trên đối tượng con đó. Khi thay đổi đối tượng cha thì các đối tượng con sẽ thay đổi theo.

Game Object

Tạo một Tag

- Tag là một thuộc tính của của một game object. Ta sử dụng thuộc tính này mục đích xác định và phân biệt các đối tượng với nhau, khi xử lý sự kiện hay bất kể vấn đề gì cần.
- Chọn vào đối tượng muốn tag, sau đó vào Inspector để thêm tag.



Thêm danh sách các tag của người dùng



DEMO

Tạo Game
Object

A white hand cursor icon, resembling a mouse pointer, is positioned below the word 'DEMO'. The index finger is pointing upwards towards the letter 'O'.

Sprite(lý thuyết)

- Sprite là các đối tượng đồ hoạ 2D sử dụng cho các nhân vật , đạo cụ, đạn, và các thành phần khác của một trò chơi 2D.
- Các đồ hoạ lấy từ hình ảnh bitmap – Texture2D . Lớp Sprite chủ yếu xác định các thành phần của hình ảnh nên sử dụng cho một hình ảnh cụ thể. Thông tin này sau đó có thể được sử dụng bởi một thành phần là SpriteRenderer trên các GameObject để thực thi đồ hoạ.
- Trong Unity 2D có hai loại Sprite là single Sprite và multiple Sprite .
- Ta nên chọn kiểu của sprite là single nếu ta muốn sử dụng trọn vẹn nội dung của bức ảnh cho một đối tượng game .
- Ta nên chọn kiểu của sprite là multiple khi file ảnh chứa nhiều hình ảnh và mỗi hình ảnh dùng cho một đối tượng .
- Ví dụ như một file ảnh chứa các bộ phận tay, chân, đầu ... của một nhân vật game hoặc file ảnh chứa các ảnh của một animation – hình ảnh động thì thường chọn là multiple sprite .Việc sử dụng multiple vừa để tiết kiệm dung lượng vừa để dễ quản lý .

Sprite

Ví dụ về 2 loại sprite là Single sprite và Multiple sprite.

- Single sprite



- Multiple sprite



Sprite

Cách tạo một Single sprite

- Ở cửa sổ **Project**, chọn thư mục **Assets**, R-Click vào thư mục **Sprites**, chọn **Import New Assets**, sau đó tìm đến một hình ảnh nào đó.
- Ở cửa sổ Inspector, chọn Texture Type là Sprite, Sprite Model là single, sau đó nhấn Apply.



Sprite

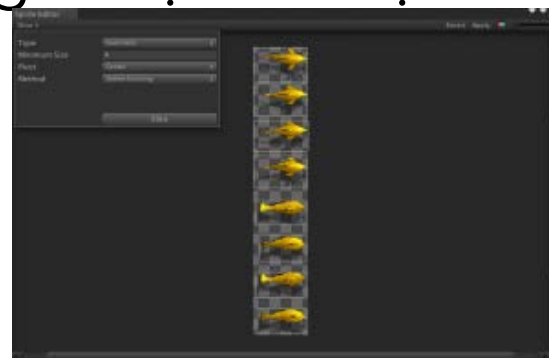
Sử dụng Multiple Sprite

- Từ phiên bản 4.3 Unity có thêm công cụ Sprite Editor để sử dụng trong việc cắt ảnh từ multiple sprite .
- Để thay đổi kiểu của một bức ảnh thành multiple sprite ta làm như sau : click chọn bức ảnh cần chuyển trong mục Inspector ta chọn Texture Type là Sprite và Sprite Mode là Multiple .

Sprite

Sử dụng Multiple Sprite

- Để cắt ảnh ta click vào nút Sprite Editor , trong Sprite Editor ta có thể click vào nút Slice để cắt ảnh theo hai kiểu :
 - Automatic : Unity editor sẽ tự động tìm những hình ảnh khác nhau hoặc ta có thể dùng chuột để chọn vùng ảnh mà ta cần cắt.



- Grid : cắt ảnh theo dạng lưới là những ô hình chữ nhật với kích thước tùy chọn.



Sprite

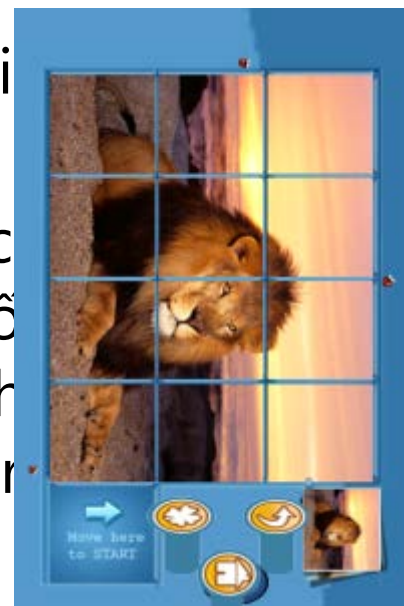
Sử dụng Multiple Sprite

- Sau khi đã chọn được các ảnh cần cắt ta ấn Apply để lưu lại.
- Sau khi cắt ảnh thì các ảnh đã cắt ra được sử dụng cho một đối tượng trong game giống như một single sprite.



Sprite

- Vận dụng để làm game Xếp hình : game mô tả một trò chơi hồi bé chúng ta hay chơi đó là ghép các mảnh ghép đang xáo trộn của một bức hình thành hình hoàn chỉnh .
- Nếu không sử dụng mutiple sprite thì với ghép ta sẽ phải sử dụng một file ảnh .
- Nếu game có khoảng 20 bức ảnh với 3 cỡ từ dễ tới khó khi là 3×4 , 4×5 , 5×6 thì số ảnh sẽ là 880 file ảnh cho mỗi một mảnh sẽ rất tốn bộ nhớ và khó khăn trong lập trình .
- Trong khi đó nếu ta dùng multiple sprite theo kiểu Grid thì sẽ dễ dàng hơn nhiều và cũng tiết kiệm bộ nhớ vì ta chỉ cần sử dụng 20 file ảnh.



Sprite

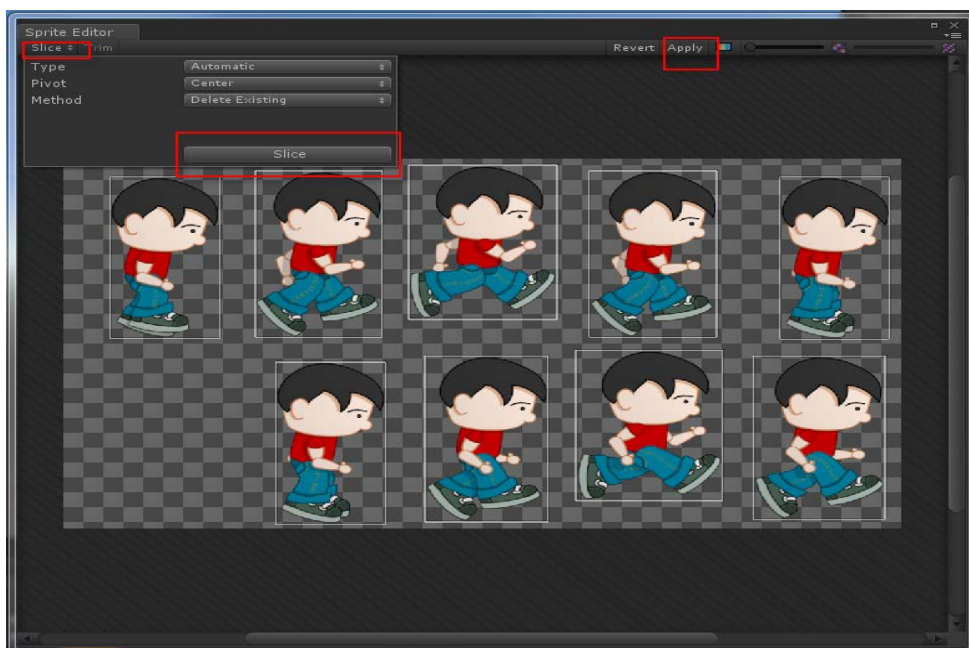
Tạo Multiple sprite

- Import New Assets, chọn đến một tập tin ảnh chứa nhiều Sprite như ở hình ví dụ trên, sau đó ở cửa sổ Inspector chọn Texture Type là Sprite, Sprite Model là Multiple. Xong nhấn Apply.
- Tiếp theo ta cần xác định các vùng bao, để xác định các sprite con thuộc Multiple sprite vừa import vào đó bằng cách ở bảng Inspector chọn Sprite Editor.

Sprite

Tạo Multiple sprite

- Sprite Editor → Type:Automatic → Slice, Unity sẽ tự động xác định vùng bao nhỏ nhất cho các sprite cho chúng ta.
- Hoặc chúng ta có thể tự dùng chuột để xác định các vùng bao này.
- Nhấn nút Apply

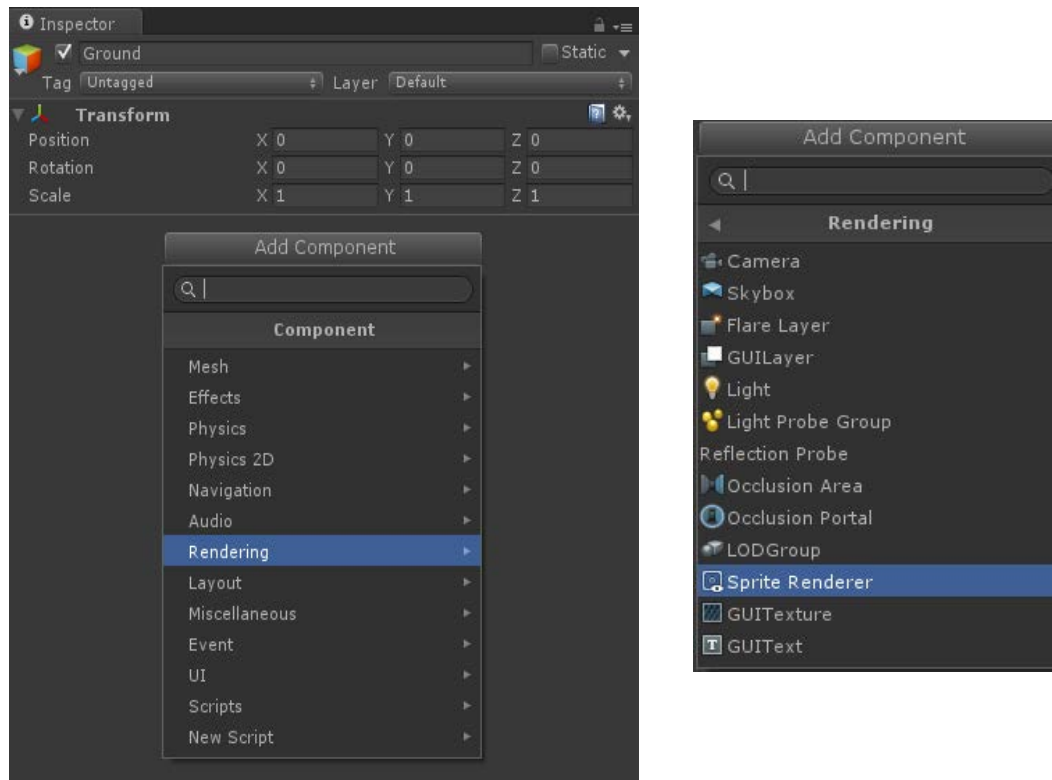


Sprite editor

Sprite

Thêm Sprite render cho Empty GameObject

- Tạo một Empty GameObject, đặt tên là Ground. Sau đó chọn đối tượng này, ở cửa sổ Inspector ta chọn **Add Component** sau đó chọn **Rendering**, chọn **Sprite Render**.



Sprite

Thêm Sprite render cho Empty GameObject

- Tiếp theo, ở mục Sprite Render ở cửa sổ Inspector, ta chọn Sprite để vẽ (Nút khoan tròn đỏ)
- **Chú ý:** bạn có thể kéo thả trực tiếp từ thư mục sprite vào cửa sổ Scenes.



Tạo multiple
sprite

Kết luận

- Cấu trúc Project
- Khởi tạo và cấu hình dự án Game 2D
- Cách tạo các đối tượng cơ bản
 - 1. *Tạo Game Object*
 - 2. *Tạo Sprite*

Chuẩn bị bài sau



FPT POLYTECHNIC

THANK YOU!

www.poly.edu.vn