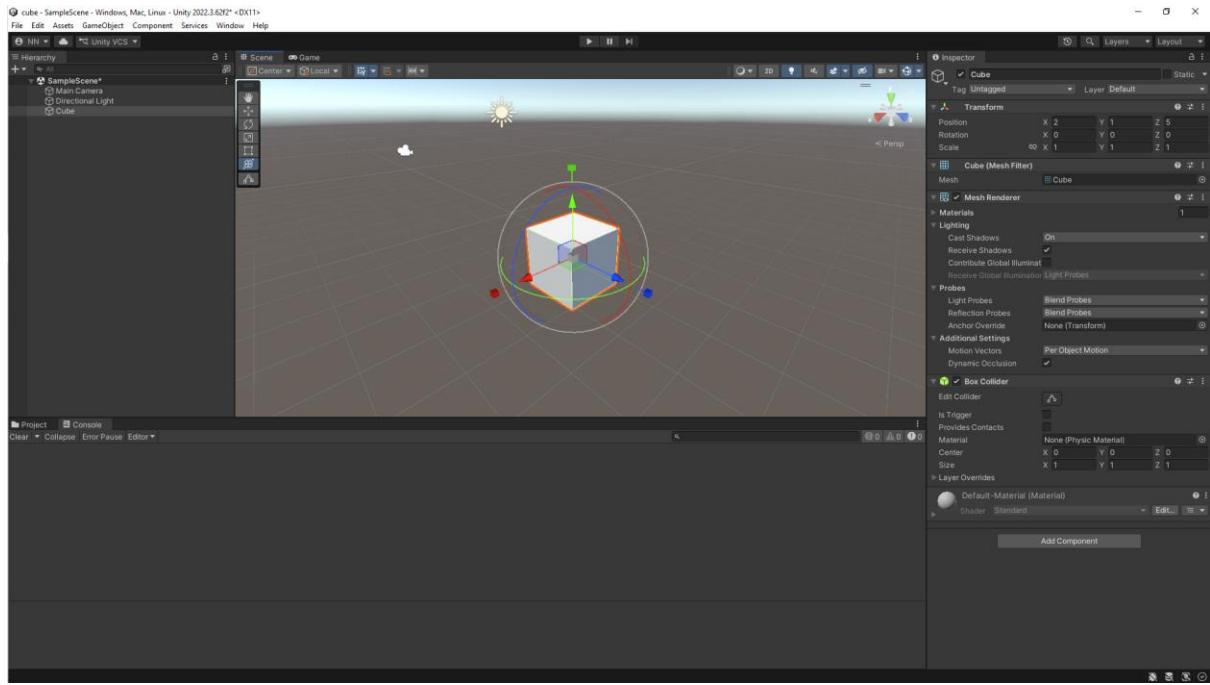


Phần A:

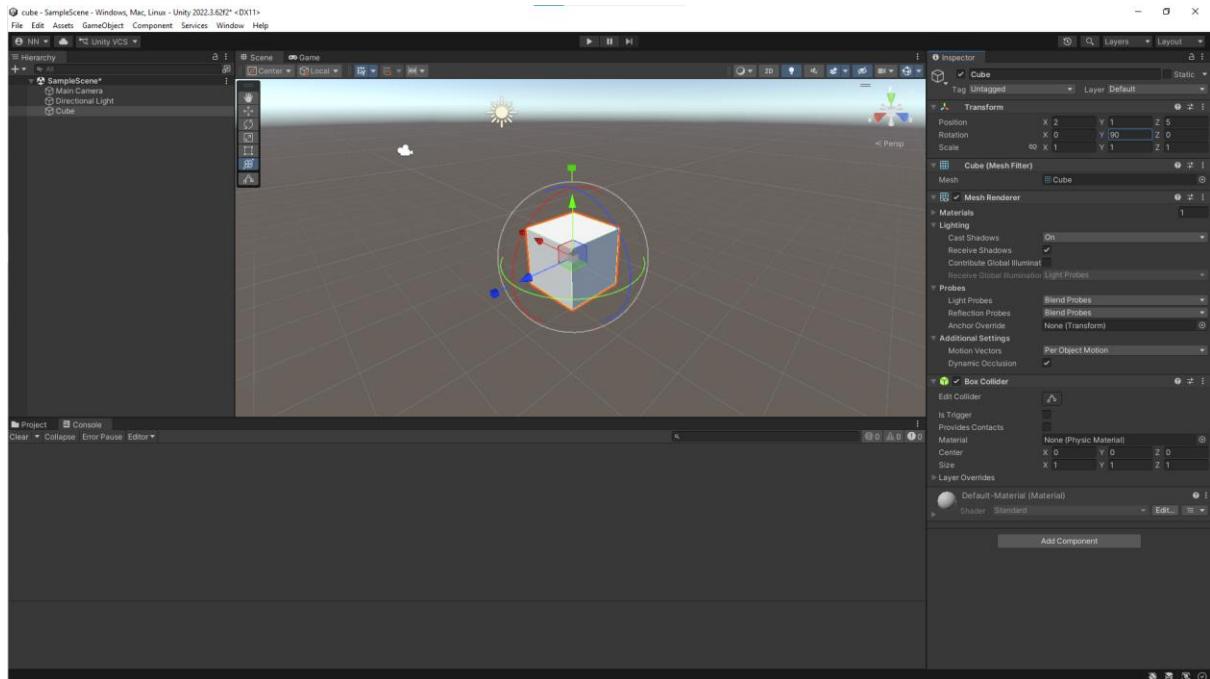


-Trục Y hướng lên trên

-Trục X hướng về phía Camera

Phần B:

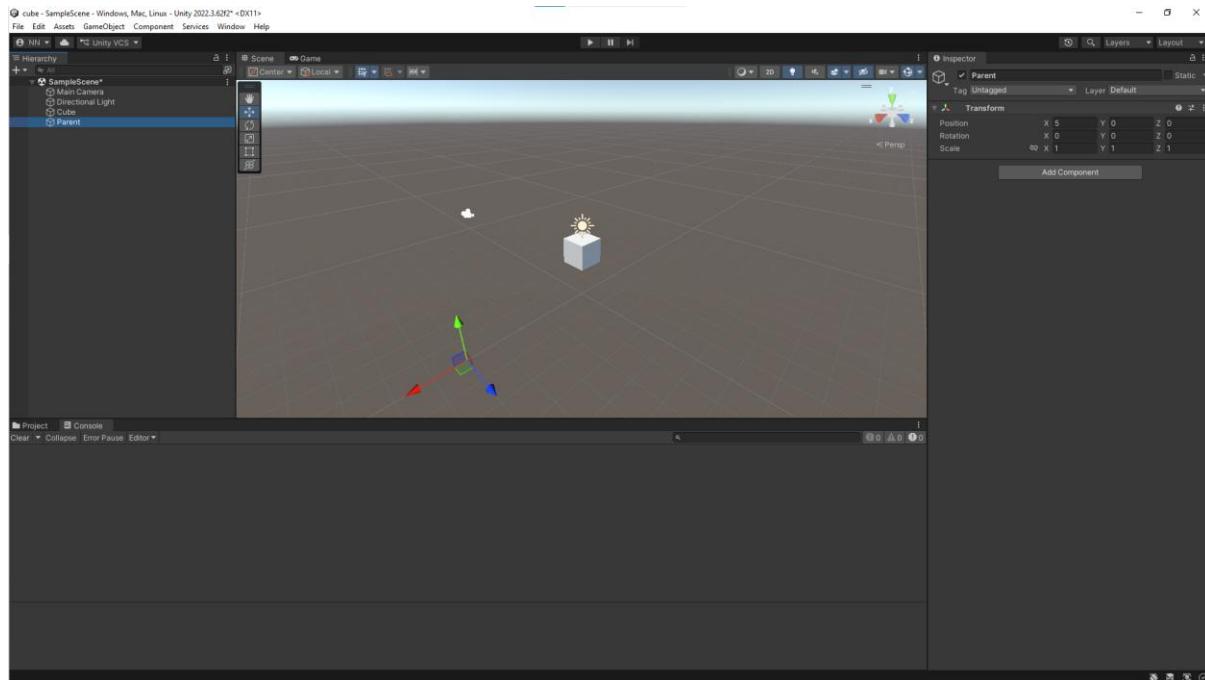
Rotation Y=90



-Cube quay từ phải sang trái theo chiều kim đồng hồ lấy trục y làm trục xoay

- Điều này cho thấy Unity sử dụng hệ tọa độ tay trái (Left-Handed Coordinate System), trong đó góc quay dương quanh trục Y tạo ra chuyển động theo chiều kim đồng hồ, và trục Z hướng về phía Camera.

Phần C:



-World position cube : (5,2,0)

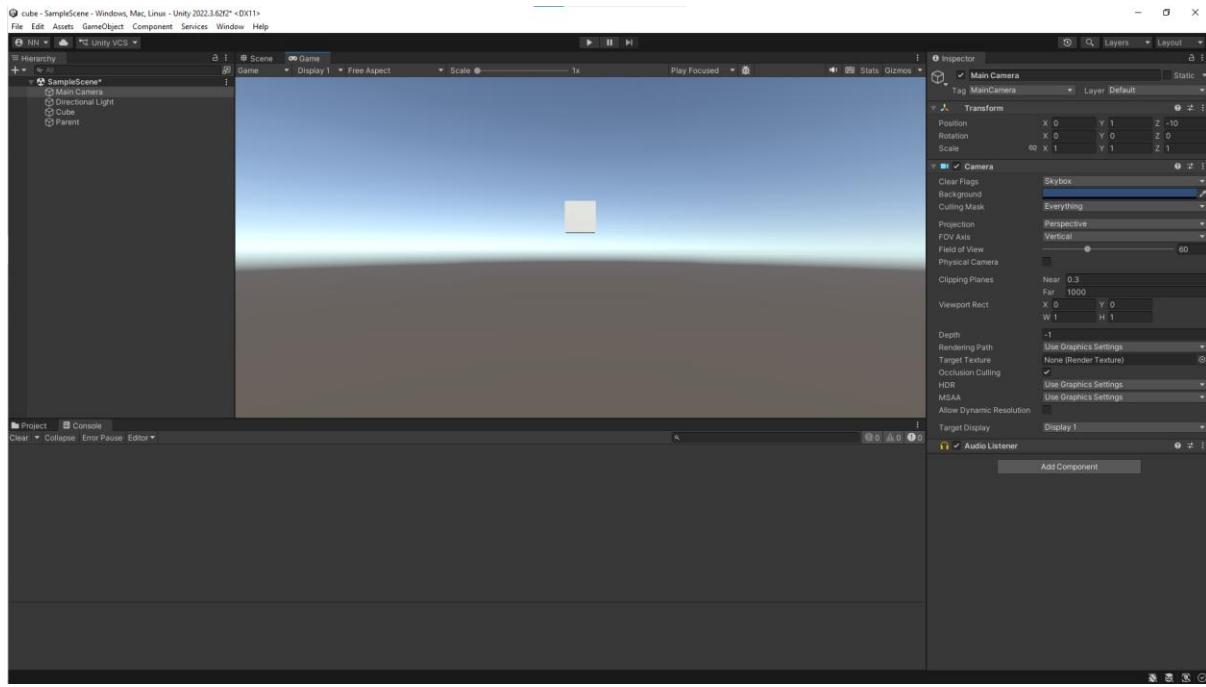
- World Position = Parent Position + Local Position của Cube $\rightarrow (5,0,0) + (0,2,0) = (5,2,0)$

-Local position của world không thay đổi

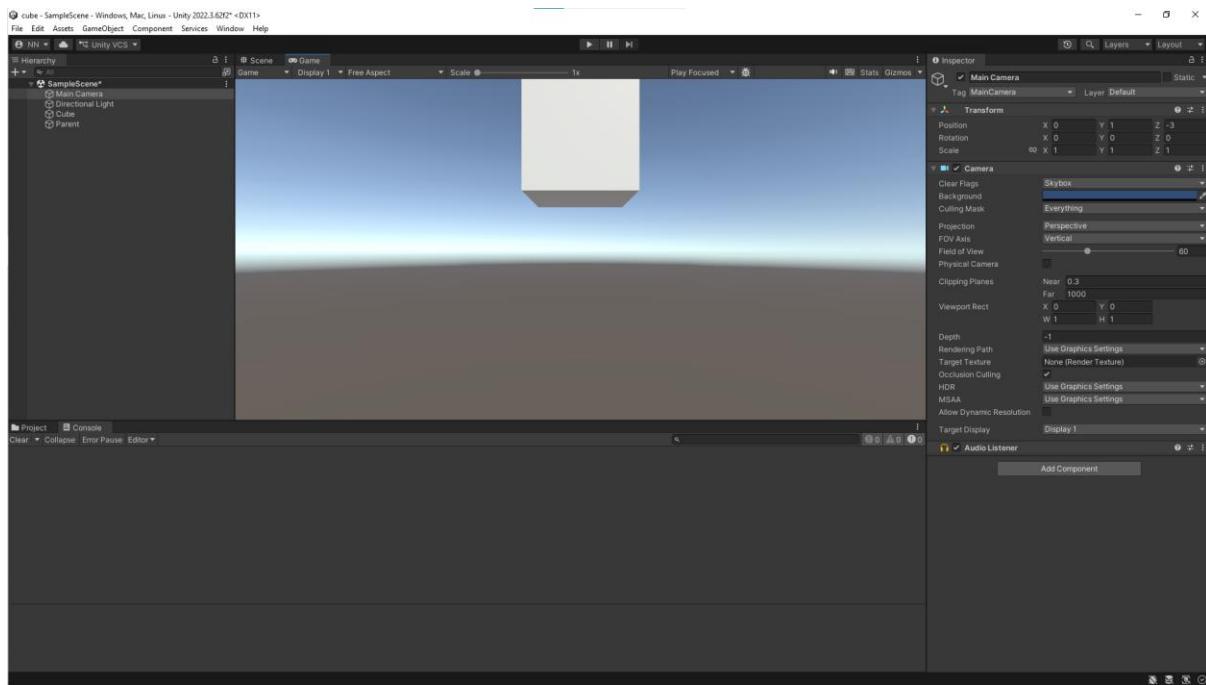
-World position của cube thay đổi dựa theo vị trí của Parent

Phần D:

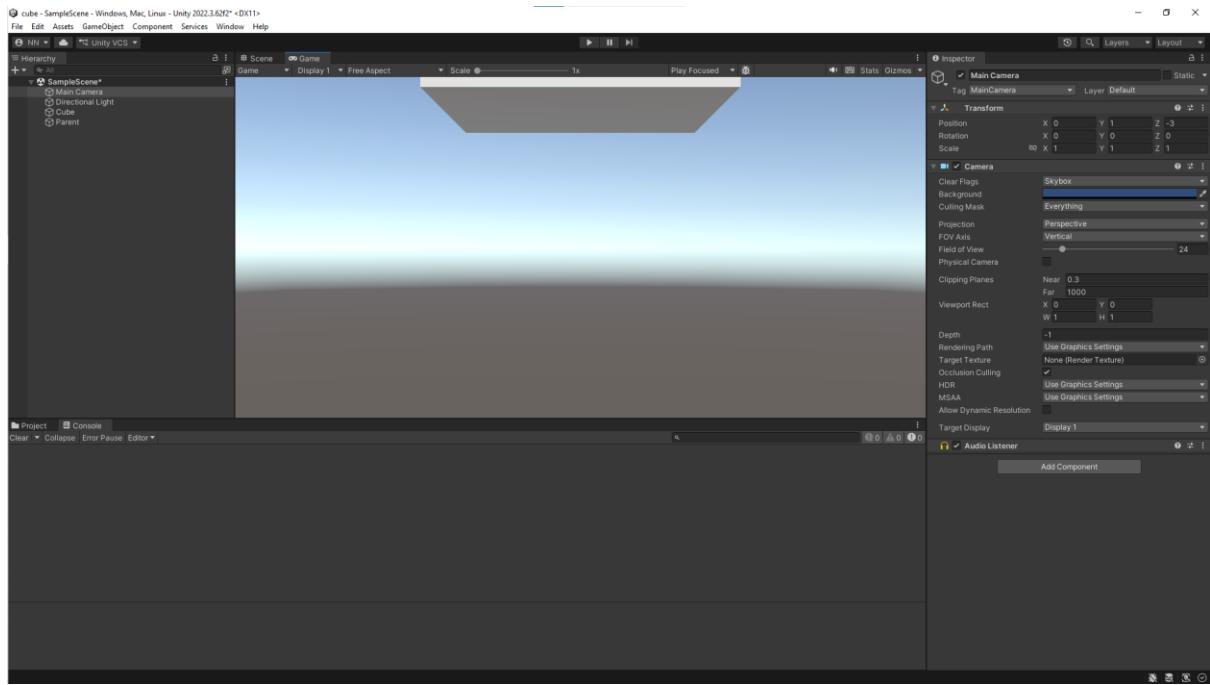
-Main Camera: Z = -10



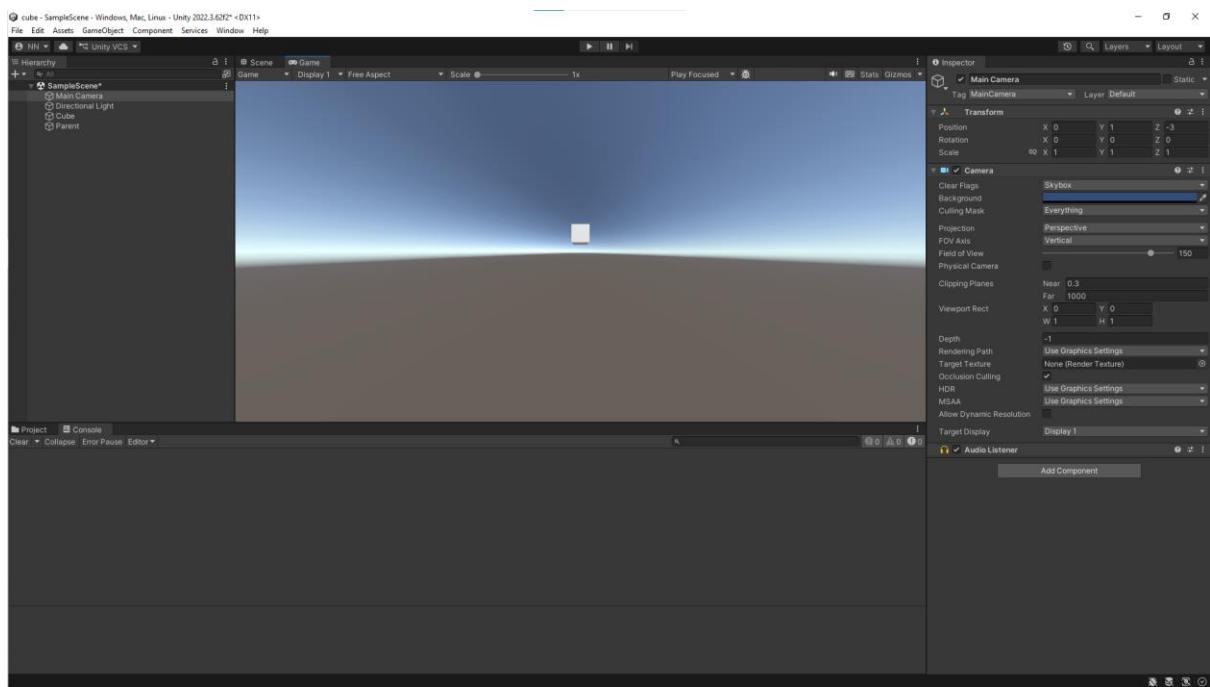
-Main Camera: Z= -3



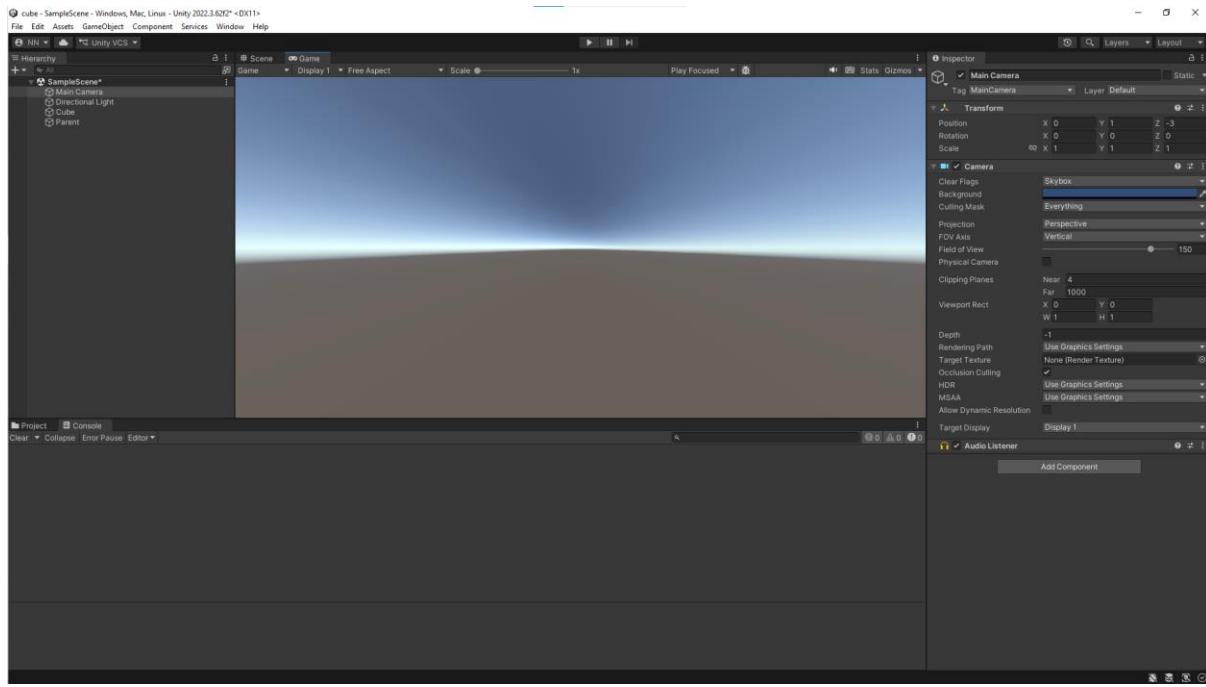
-Field of View = 24



-Field of View : 150



-Near Clip Plane : 4

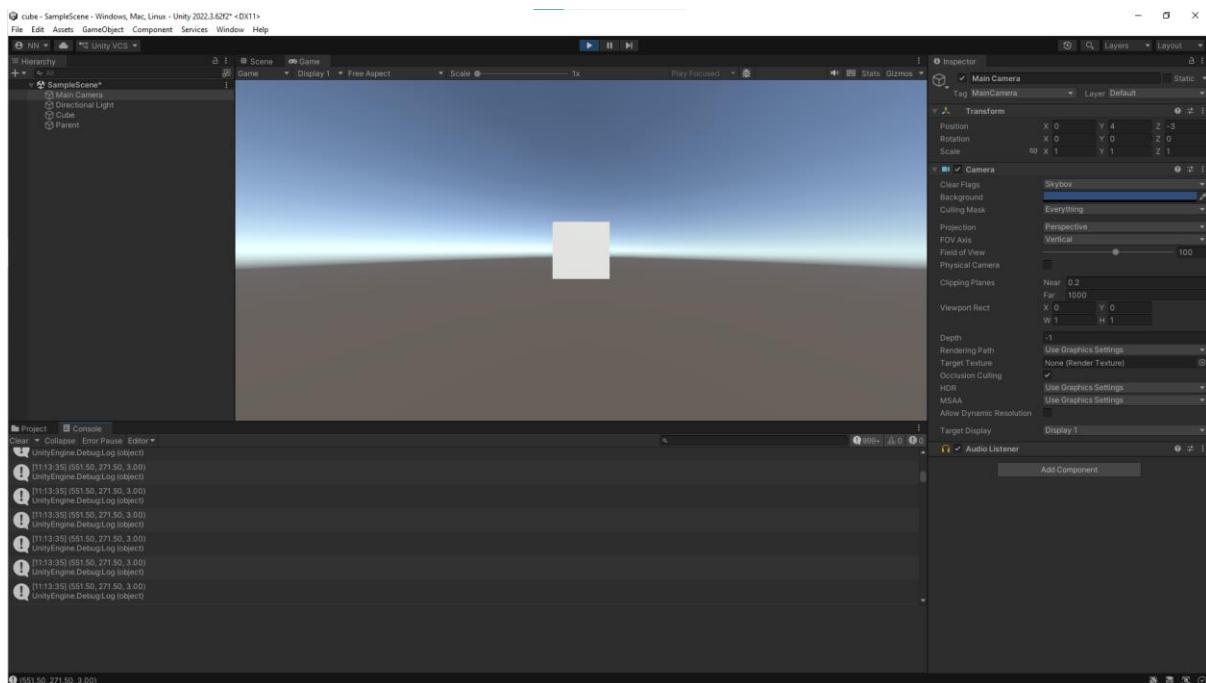


- Field of View ảnh hưởng đến kích thước hiển thị của Object do Camera sử dụng phép chiếu phối cảnh. Khi Field of View thay đổi, góc nhìn của Camera thay đổi, làm object trông to hoặc nhỏ hơn dù không đổi vị trí.

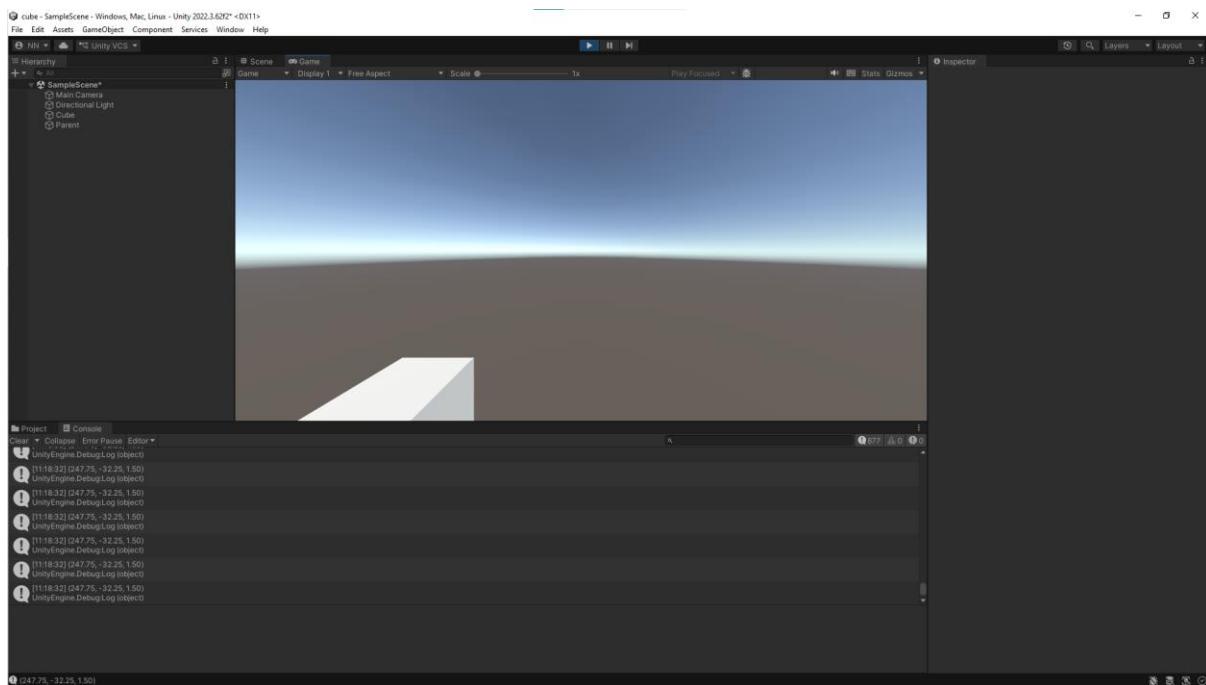
- Near Clip Plane dùng để cắt xén vùng nhìn của Camera. Khi Object không trong phạm vi cắt xén này, Object sẽ không được hiển thị hoặc biến mất khỏi màn hình.

Phần E:

-Hình Cube giữa màn hình :



-Hình Cube ở góc trái màn hình:



-World Space: mô tả thế giới 3D trong Unity

-Screen Space: hiển thị vị trí 2D trên màn hình theo pixel, phụ thuộc vào Camera và độ phân giải

Kết luận

Sau khi hoàn thành bài tập này, em đã hiểu rõ hơn về cơ chế hệ tọa độ trong không gian 3D cũng như quá trình hiển thị từ World Space lên Screen Space thông qua Camera. Việc nắm vững các khái niệm này là nền tảng quan trọng để làm việc hiệu quả với các đối tượng 3D trong Unity. Bài tập cũng giúp em hiểu rõ hơn về tính năng Left-Handed Coordinate System, đặc biệt là cách vật thể xoay và định hướng trong không gian. Việc quan sát sự thay đổi của Cube khi xoay quanh trục Y giúp em liên hệ trực tiếp giữa lý thuyết và thực tế, từ đó tránh được những nhầm lẫn khi lập trình hoặc thiết kế scene. Thao tác với tính năng Camera, Field of View và Clipping Plane giúp em hiểu vì sao cùng một vật thể nhưng có thể trông to, nhỏ hoặc thậm chí biến mất khỏi màn hình.