Title: Collision of any two objects in motion

EXPERIMENT NO: 9 **Name:** SHAKYA JAY JITENDRA

Date: 8-9-2025 **Reg. No:** 24BCG10123

Aim: To study the effect of collisions between two moving objects in Unity physics, for both head-on collisions and oblique (diagonal) collisions, and observe how they exchange velocity and change direction in Unity.

Description/Concept:

Two spheres with Rigidbody components are placed on a plane.

Both are given an initial velocity in the direction of their transform.forward (so motion depends on their rotation).

Collisions are detected using Unity's physics system.

Two cases are studied:

Head-on collision: Both objects move directly towards each other along the same line.

Diagonal collision: Both objects move at an angle such that they collide obliquely, changing direction after impact.

The collision response is governed by Newton's Laws of Motion and the principle of conservation of momentum.

Program/Coding:

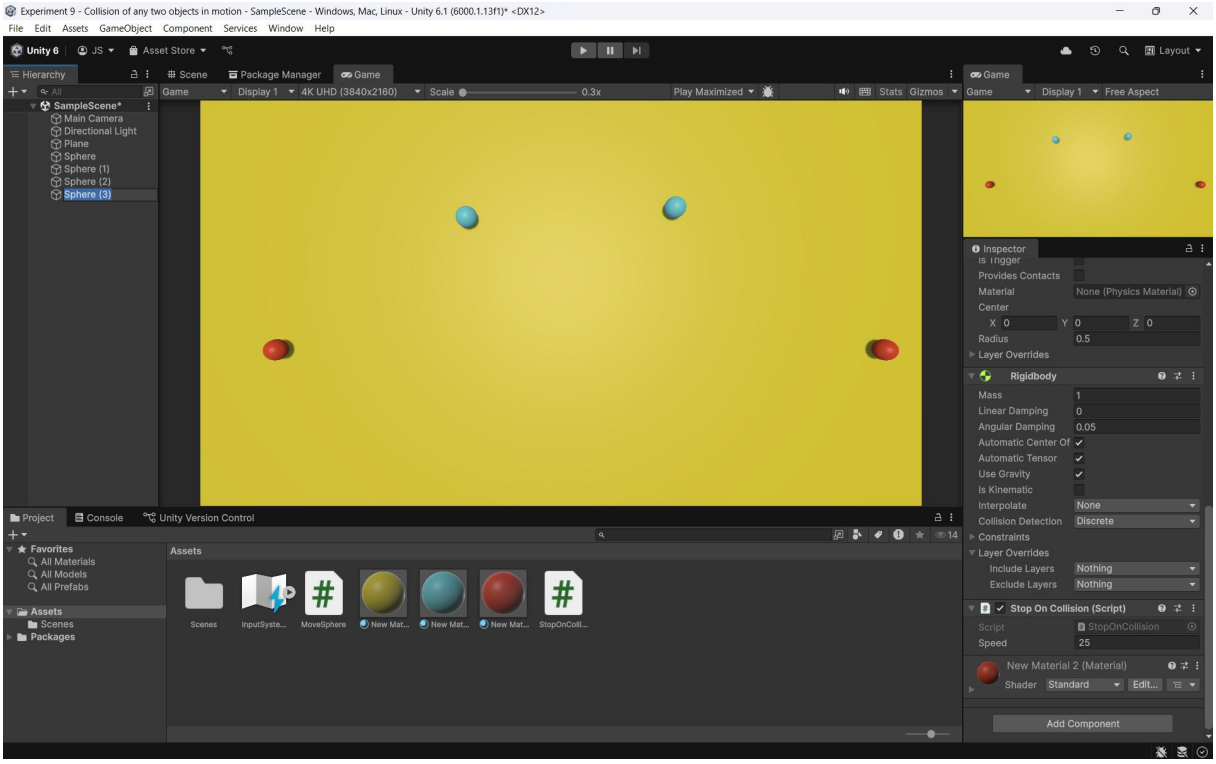
1)

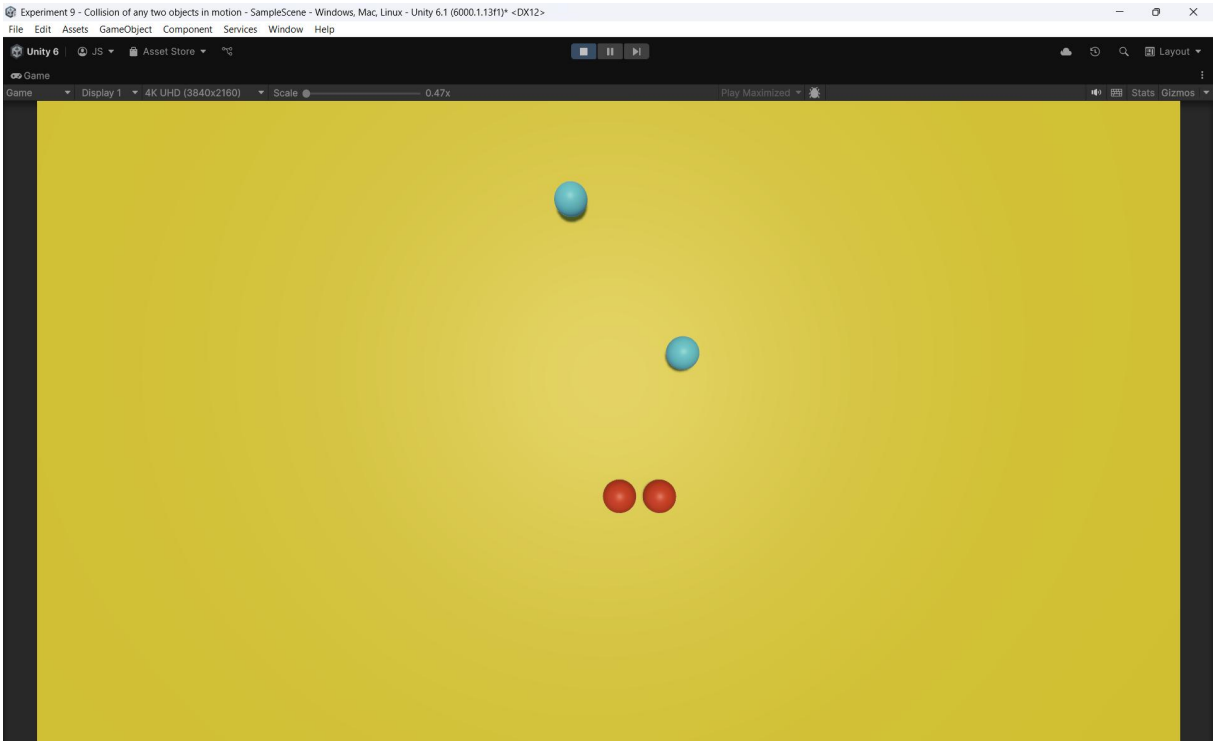
```
using UnityEngine;

public class StopOnCollision : MonoBehaviour
{
    public float speed = 5f;   // starting speed
    private Rigidbody rb;

    void Start()
    {
        rb = GetComponent<Rigidbody>();
        rb.linearVelocity = transform.forward * speed;  // move forward in facing
direction
    }

    void OnCollisionEnter(Collision collision)
    {
        // Ignore collision with Plane
        if (collision.gameObject.name == "Plane" ||
collision.gameObject.CompareTag("Ground"))
            return;
```

```
        // Stop this object
        rb.linearVelocity = Vector3.zero;
        rb.angularVelocity = Vector3.zero;

        // Stop the other object if it has a Rigidbody
        Rigidbody otherRb = collision.rigidbody;
        if (otherRb != null)
        {
            otherRb.linearVelocity = Vector3.zero;
            otherRb.angularVelocity = Vector3.zero;
        }

    }
}
```

2)

```
using UnityEngine;

public class MoveSphere : MonoBehaviour
{
    public float speed = 10f;    // Set in Inspector
    private Rigidbody rb;

    void Start()
    {
        rb = GetComponent<Rigidbody>();
        rb.linearVelocity = transform.forward * speed;  // Push sphere forward
    }
}
```





Output:

Head-On Collision (Case 1)

Objects approach along the same line.

On collision, they exchange velocities along that line.

If masses are equal → both reverse direction with equal speed.

If one is heavier → lighter object bounces back faster, heavier one slows slightly.

Diagonal/Oblique Collision (Case 2)

Objects collide at an angle.

After collision, their velocity vectors split into new directions.

Each object changes direction diagonally while continuing to move away.

The deflection angle depends on their masses and speeds.

Conclusion:

In head-on collisions, motion is constrained to one axis, and objects exchange velocity directly along that axis.

In diagonal collisions, momentum is conserved in both X and Z directions, leading to diagonal changes in trajectory.

Unity's physics engine correctly demonstrates momentum transfer and realistic redirection of objects in collisions.

Result:

Script was created for Collision of any two objects in motion were successfully implemented and tested in Unity Game Engine 6.1 Their application in game mechanics was understood through coding and console outputs.