**CAR CODE CONTROL**

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

public class Car : MonoBehaviour

{

public void GetInput()

{

m\_horizontalInput = Input.GetAxis("Horizontal");

m\_verticalInput = Input.GetAxis("Vertical");

}

private void Steer()

{

m\_steeringAngle = maxSteerAngle \* m\_horizontalInput;

frontDriverW.steerAngle = m\_steeringAngle;

frontPassengerW.steerAngle = m\_steeringAngle;

}

private void Accelerate()

{

frontDriverW.motorTorque = m\_verticalInput \* motorForce;

frontPassengerW.motorTorque = m\_verticalInput \* motorForce;

}

private void UpdateWheelPoses()

{

UpdateWheelpose(frontDriverW, frontDriverT);

UpdateWheelpose(frontPassengerW, frontPassengerT);

UpdateWheelpose(rearDriverW, rearDriverT);

UpdateWheelpose(rearPassengerW, rearPassengerT);

}

private void UpdateWheelpose(WheelCollider \_collider, Transform \_transform)

{

Vector3 \_pos = \_transform.position;

Quaternion \_quat = \_transform.rotation;

\_collider.GetWorldPose(out \_pos, out \_quat);

\_transform.position = \_pos;

\_transform.rotation = \_quat;

}

private void FixedUpdate()

{

GetInput();

Steer();

Accelerate();

UpdateWheelPoses();

if (Input.GetKeyDown(KeyCode.A))

{

Debug.Log("ASDADASDASD");

User.gameObject.SetActive(true);

User.transform.position = new Vector3(gameObject.transform.position.x + 3, gameObject.transform.position.y + 0.7f, gameObject.transform.position.z + 3) ;

gameObject.GetComponent<Car>().enabled = false;

}

}

private float m\_horizontalInput;

private float m\_verticalInput;

private float m\_steeringAngle;

public WheelCollider frontDriverW, frontPassengerW;

public WheelCollider rearDriverW, rearPassengerW;

public Transform frontDriverT, frontPassengerT;

public Transform rearDriverT, rearPassengerT;

public float maxSteerAngle = 30;

public float motorForce = 50;

public GameObject User;

}using System.Collections;

using System.Collections.Generic;

using UnityEngine;

public class Car : MonoBehaviour

{

public void GetInput()

{

m\_horizontalInput = Input.GetAxis("Horizontal");

m\_verticalInput = Input.GetAxis("Vertical");

}

private void Steer()

{

m\_steeringAngle = maxSteerAngle \* m\_horizontalInput;

frontDriverW.steerAngle = m\_steeringAngle;

frontPassengerW.steerAngle = m\_steeringAngle;

}

private void Accelerate()

{

frontDriverW.motorTorque = m\_verticalInput \* motorForce;

frontPassengerW.motorTorque = m\_verticalInput \* motorForce;

}

private void UpdateWheelPoses()

{

UpdateWheelpose(frontDriverW, frontDriverT);

UpdateWheelpose(frontPassengerW, frontPassengerT);

UpdateWheelpose(rearDriverW, rearDriverT);

UpdateWheelpose(rearPassengerW, rearPassengerT);

}

private void UpdateWheelpose(WheelCollider \_collider, Transform \_transform)

{

Vector3 \_pos = \_transform.position;

Quaternion \_quat = \_transform.rotation;

\_collider.GetWorldPose(out \_pos, out \_quat);

\_transform.position = \_pos;

\_transform.rotation = \_quat;

}

private void FixedUpdate()

{

GetInput();

Steer();

Accelerate();

UpdateWheelPoses();

// This condition connects the car with the character

if (Input.GetKeyDown(KeyCode.A)) //

{

Debug.Log("ASDADASDASD");

User.gameObject.SetActive(true);

User.transform.position = new Vector3(gameObject.transform.position.x + 3, gameObject.transform.position.y + 0.7f, gameObject.transform.position.z + 3) ;

gameObject.GetComponent<Car>().enabled = false;

}

}

private float m\_horizontalInput;

private float m\_verticalInput;

private float m\_steeringAngle;

public WheelCollider frontDriverW, frontPassengerW;

public WheelCollider rearDriverW, rearPassengerW;

public Transform frontDriverT, frontPassengerT;

public Transform rearDriverT, rearPassengerT;

public float maxSteerAngle = 30;

public float motorForce = 50;

public GameObject User;

}