

# RCC\_WheelCollider

## Public Variables

```
// Car controller.
public RCC_CarControllerV3 CarController;

// WheelCollider.
public WheelCollider WheelCollider;

public Transform wheelModel;    // Wheel model for animating and aligning.

public WheelHit wheelHit;    // Wheelhit data.

public bool isGrounded = false;    // Is wheel grounded or not?

public int groundIndex = 0;    // Current ground index of wheelhit.

public bool alignWheel = true;    // Align the wheelmodel with wheelcollider position and rotation.

public bool drawSkid = true;    // Draw skidmarks.

// Locating correct position and rotation for the wheel.
public Vector3 wheelPosition = Vector3.zero;

public Quaternion wheelRotation = Quaternion.identity;

public bool canPower = false;    // Can this wheel apply power?
public float powerMultiplier = 1f;

public bool canSteer = false;    // Can this wheel apply steer?
public float steeringMultiplier = 1f;

public bool canBrake = false;    // Can this wheel apply brake?
public float brakingMultiplier = 1f;

public bool canHandbrake = false;    // Can this wheel apply handbrake?
public float handbrakeMultiplier = 1f;

public float wheelWidth = .275f;    // Width of the wheel.

public float wheelOffset = 0f;    // Offset by X axis.

public float camber = 0f;    // Camber angle.
public float caster = 0f;    // Caster angle.
public float toe = 0f;    // Toe angle.
```

```
//      Slips
public float wheelSlipAmountForward = 0f;    // Forward slip.
public float wheelSlipAmountSideways = 0f; // Sideways slip.
public float totalSlip = 0f;                  // Total amount of forward and sideways slips.

// List for all particle systems.
public List<ParticleSystem> allWheelParticles = new List<ParticleSystem>();

//      Tractions used for smooth drifting.
public float tractionHelpedSidewaysStiffness = 1f;

// Getting bump force.
public float bumpForce, oldForce, RotationValue = 0f;

public float deflateRadiusMultiplier = .8f;    // Deflated radius multiplier. Radius of the wheelcollider
will be multiplied by this value on deflate.

public float deflatedStiffnessMultiplier = .5f; // Deflated stiffness of the wheelcollider.
```

## Public Methods

```
/// Returns true if one of the wheel is slipping.
public bool IsSkidding() {}

/// Applies the motor torque.
public void ApplyMotorTorque(float torque) {}

/// Applies the steering.
public void ApplySteering(float steerInput, float angle) {}

/// Applies the brake torque.
public void ApplyBrakeTorque(float torque) {}

/// Deflates the wheel.
public void Deflate() {}

/// Inflates the wheel.
public void Inflate() {}
```