- **Update Function**

```
RunService.Heartbeat:Connect(function(dT)
	-- Assign character air status
	character.inTheAir = HelperFunctions:CheckIfHumanoidIsInTheAir(hum)

	-- Check jump status and movement keys
	local noButtonsPressed = HelperFunctions:CheckValuesInTable(keysPressed,
	directionValues, false)

	local hasNotJumped = HelperFunctions:CheckValuesInTable(character.jumpStarted,
	jumpValues, false)

	-- Calculate move direction
	moveVector = AccelerationController:CalculateCorrectAcceleration(keysPressed,
	moveVector, character)

	local direction = humanoid.CFrame:VectorToWorldSpace(moveVector)

	-- Handle Direction Control
	AccelerationController:DirectionControl(character, moveVector, direction, dT, humanoid,
	noButtonsPressed)

	-- Deceleration
	AccelerationController:Decelerate(character, hasNotJumped, dT)

	-- Update velocity based on current jump status
	if hasNotJumped then
	--Apply normal gravity to player
		Gravity:EnforceNormalGravity(character, dT)
	else
		--- If we are jumping then calculate correct velocity
		character:CheckJumpForces(character, dT, humanoid)
	end

	-- Check if we have pressed any jumping key
	local notPressingJumpKey = HelperFunctions:CheckValuesInTable(jumpKeys,
	jumpKeyValues, false)

	-- If we have jumped then do the jump
	if not notPressingJumpKey then
		character:CheckJump(hum, humanoid, moveVector, jumpKeys, character)
	end

	-- Limit speed
	SpeedController:ControlSpeed(character, hasNotJumped)

	humanoid.Velocity = character.savedNormalVelocity + character.savedPlatformVelocity
end)
```