1. Script & File Structure Keywords

Keyword	Syntax	Description
extends	extends Node3D	Used to inherit from a node or class.
class_name	class_name VehicleController	Gives your script a global class name
@tool	@tool	Runs the script in editor as well

2. Variable Declaration Keywords

var	var speed = 10	Declares a variable
var: Type	var speed: float = 10	Typed Variable
const	const MAX_SPEED := 120	Constant Value
static	static var counter := 0	Shared between all instances

3. Export and Editor Keywords

export	@export var speed: float = 10.0	Exposes variable to Inspector
export_range	@export_range(0, 100) var volume := 50	Limit values range in inspector
export_enum	@export_enum("Easy", "Medium", "Hard") var difficulty	Dropdown enum in Inspector
onready	@onready var mesh: MeshInstance3D = $Mesh	Assigns value after node is ready

4. Function Keywords

func	func calculate() -> void: pass	Defines a function
return	return result	Returns a value from a function
Pass	Func placeholder(): pass	Empty placeholder statement

5. Build-in Lifecycle Callbacks

_ready()	func _ready() -> void: initialize()	Called once when the node is ready
_process(delta)	func _process(delta: float) -> void: update_state(delta)	Called every rendered frame
_physics_process	func _physics_process(delta: float) -> void: apply_logic(delta)	Called at a fixed timestep
_input(event)	func _input(event: InputEvent) -> void: pass	Handles input event

6. Conditional Control Flow

if / elif / else	if value > 0:
	handle_positive()
	elif value < 0:
	handle_negative()
	else:
	handle_zero()
		Executes code based on conditions

match	match status: 0: handle_idle() 1: handle_active() _: handle_default()	Pattern-based branching

7. Loops

for	For i in range(5): print(i)	Iterates over a sequence
while	while condition: process()	Repeats while a condition is true
break	break	Exits a loops immediately
continue	continue	Skips the current loop iteration

8. Logical Operators

and	if a > 0 and b > 0: pass	Logical AND
or	if is_valid or is_cached: pass	Logical OR
not	if not is_ready: pass	Logical negation

9. Object-Oriented Keywords

self	self.value = 10	Refers to the current instance
super	super _ready()	Calls a method from the base class
is	if object is Node: pass	Checks an object's type
as	var node:= object as Node	Safely casts an object to a type

10. Signals

signal	signal updated(value)	Declares a custom signal
emit_signal	emit_signal("updated", 6)	Emits a signal

11. Asynchronous Flow

await	await get_tree() process_frame	Pauses execution until a signal completes

12. Error Handling and Debugging

assert	assert(value >= 0)	Stops execution if the condition is false
push_error	push_error("Invalid config")	Displays and error in the editor
push_warning	push_warning("Deprecated usage")	Displays a warning in the editor

13. Special Values

null	var reference = null	Represents no value
true/fales	var enabled = true	Boolean literals