02_02_ifelse_simulator_teacher_

May 2, 2025

1 Conditional Logic with `if/else` Statements

In this notebook, you'll learn how to make your drone behave differently based on its current state using `if`, `else`, and `elif`.

We'll use sensor readings and state values like height, position, yaw, velocity, and status from the simulator.

[ ]:
```
#  Setup and Drone Initialization
import time
from crazyflie_sim import CrazyflieSimulator

drone = CrazyflieSimulator(real=False)
```

1.1 Conditional Checks with Drone State

You can access information from the drone using: - `drone.get_height()` → meters - `drone.get_position()` → (x, y) - `drone.get_yaw()` → degrees - `drone.get_velocity()` → (vx, vy, vz) - `drone.get_status()` → e.g. `'airborne'`, `'landed'` - `drone.get_log()` → command history list

[ ]:
```
print("Height:", drone.get_height())
print("Position:", drone.get_position())
print("Yaw:", drone.get_yaw())
print("Velocity:", drone.get_velocity())
print("Status:", drone.get_status())
print("Command Log:", drone.get_log())
```

1.2 Example 1: Check if drone is on the ground

[ ]:
```
if drone.get_status() == 'landed':
    print(" Drone is currently on the ground.")
else:
    print(" Drone is in the air!")
```

1.3 Example 2: Take off only if not already flying

[ ]:
```
if drone.get_status() == 'landed':
    drone.takeoff(1.0, 0.3)
    time.sleep(2)
else:
    print("Already flying.")
```

1.4 Example 3: Turn based on yaw value

[ ]:
```
yaw = drone.get_yaw()
if yaw < 180:
    print("Turning right to balance orientation.")
    drone.rotate(90, 1)
else:
    print("Turning left to balance orientation.")
    drone.rotate(-90, 1)
```

1.5 Exercise 1: Check if drone is below 1m and go up if needed

[ ]:
```
# if drone.get_height() < 1.0:
#     drone.up( , )
# else:
#     print("Altitude is sufficient.")
```

1.6 Exercise 2: If yaw > 180, rotate left to correct

[ ]:
```
# yaw = drone.get_yaw()
# if yaw > 180:
#     drone.rotate( , )
```

1.7 Exercise 3: Print different messages based on velocity

[ ]:
```
# vx, vy, vz = drone.get_velocity()
# if abs(vx) > 0.2 or abs(vy) > 0.2:
#     print(" Drone is moving fast!")
# else:
#     print(" Drone is moving slowly.")
```

1.8 Land and close the drone

[ ]:
```
drone.land(0.3)
drone.close()
```