

01_02_flying_with_the_motioncommander_teacher__

May 2, 2025

Part of the InnovatED STEM and DroneBlocks Land, Air, and Sea Robotics Curriculum
Licensed for educational use in schools only.

Redistribution, commercial use, or resale is strictly prohibited.

© 2025 InnovatED STEM & DroneBlocks. All rights reserved.

1 Motion Commands with CrazyflieSimulator

In this lesson, you'll explore how to fly a simulated or real drone using motion commands like forward, back, up, down, left, right, and rotate. Each command will be run in its own cell so you can learn their effect and parameters.

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

# Use simulation mode unless you're connected to a real drone
drone = CrazyflieSimulator(real=False)
```

1.1 Takeoff

Required before any motion command.

`takeoff(height, velocity)`

- `height`: target hover height (meters)
- `velocity`: ascent speed (m/s) — keep under 0.5

```
[ ]: drone.takeoff(0.4, 0.3)
time.sleep(2)
```

1.2 Up and Down

`up(distance, speed)` and `down(distance, speed)`

- Moves vertically up/down in meters
- `speed`: velocity in m/s (keep under 0.5)

```
[ ]: drone.up(0.2, 0.2)
time.sleep(1)
```

```
[ ]: drone.down(0.2, 0.2)
time.sleep(1)
```

1.3 Left and Right

`left(distance, speed)` and `right(distance, speed)`

- Moves left or right relative to the drone
- Distance in meters (stay under 0.5)

```
[ ]: drone.left(0.3, 0.2)
time.sleep(1)
```

```
[ ]: drone.right(0.3, 0.2)
time.sleep(1)
```

1.4 Forward and Backward

`forward(distance, speed)` and `backward(distance, speed)`

- Moves the drone along its front/back axis
- Speed and distance should be small for safety

```
[ ]: drone.forward(0.3, 0.2)
time.sleep(1)
```

```
[ ]: drone.backward(0.3, 0.2)
time.sleep(1)
```

1.5 Rotate (Yaw)

`rotate(yaw_rate, duration)`

- `yaw_rate`: degrees per second (positive = clockwise)
- `duration`: how long to turn

```
[ ]: drone.rotate(90, 1)
```

1.6 The `sleep()` Command

We use `time.sleep(seconds)` after each motion to give the drone time to complete its movement before issuing a new command.

Without `sleep()`, movements may overlap or be skipped.

1.7 Exercise 1: Try a Sequence of Motions with `sleep()`

```
[ ]: # Use small distances under 0.5m
# Add sleep between each command
# Example: drone.forward(0.2, 0.2)
```

1.8 Exercise 2: Try a Sequence of Motions without `sleep()`

```
[ ]: # Try running motion commands quickly without sleep()  
    # What happens? Does the drone complete each move?
```

1.9 Land and Close

Always land before finishing your session.

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