Two PX4 x500_depth Drones in One Gazebo Gz World – Quick Steps

1. Prerequisites (done once)
   - Ubuntu 22.04 with PX4-Autopilot cloned.
   - Gazebo Gz (Harmonic) installed via PX4 setup script.
   - ROS 2 Humble (optional, for camera feed) and ros_gz_bridge installed.

   Build PX4 SITL at least once:
   ```
   cd ~/PX4-Autopilot
   make px4_sitl
   ```

2. Start Drone 1 (rear drone, instance 1)
   Open Terminal 1:
   ```
   cd ~/PX4-Autopilot
   PX4_SYS_AUTOSTART=4002 PX4_GZ_MODEL=x500_depth \
     ./build/px4_sitl_default/bin/px4 -i 1
   ```

   Wait for the pxh> prompt, then in that shell:
   ```
   param set NAV_DLL_ACT 0
   ```

   This starts Gazebo Gz with the default world and spawns model x500_depth_1.

3. Start Drone 2 (front drone, instance 2)
   Open Terminal 2:
   ```
   cd ~/PX4-Autopilot
   PX4_SYS_AUTOSTART=4002 PX4_GZ_MODEL=x500_depth \
     PX4_GZ_MODEL_POSE="3,0" \
     ./build/px4_sitl_default/bin/px4 -i 2
   ```

   In the second pxh> shell:
   ```
   param set NAV_DLL_ACT 0
   ```

   Now the world contains two drones: x500_depth_1 at the origin and
   x500_depth_2 about 3 m in front along the +X axis.

4. Arm and Take Off (per drone)
   In each pxh> shell (Terminal 1 and Terminal 2):
   ```
   commander arm
   commander takeoff
   ```

   Make sure the Gazebo simulation is unpaused so both vehicles lift off.

5. Camera Topics (optional, for ROS 2)
   Each drone publishes its RGB camera over Gz topics, for example:
   ```
   /world/default/model/x500_depth_1/link/camera_link/sensor/IMX214/image
   /world/default/model/x500_depth_2/link/camera_link/sensor/IMX214/image
   ```

   To bridge the rear drone camera to ROS 2:
   ```
   ros2 run ros_gz_bridge parameter_bridge \
     "/world/default/model/x500_depth_1/link/camera_link/sensor/IMX214/image@"\
   ```

"sensor_msgs/msg/Image@gz.msgs.Image" \
"/world/default/model/x500_depth_1/link/camera_link/sensor/IMX214/camera_info@"\
"sensor_msgs/msg/CameraInfo@gz.msgs.CameraInfo"

Then visualize in rqt_image_view or RViz2.

Summary:
- Use separate terminals and PX4 instance IDs (-i 1, -i 2) to run two vehicles.
- PX4_GZ_MODEL=x500_depth selects the drone model; PX4_GZ_MODEL_POSE sets spawn offset.
- Set NAV_DLL_ACT=0 in SITL to bypass the "no GCS" preflight check.
- Arm and take off both drones via commander, and optionally bridge cameras into ROS 2.