About you, me, us.

- Me: embedded geek -> fun with drones
- You: want to code with drones
- Basic python knowledge required
- This is a lightning tutorial
- Questions?

Why do this?

- Sad mavlink on drones-discuss
- Need a *simple* API that handles common requests
- On LAN or coprocessor
- Let the API also work on Android, WAN, etc...

DroneAPI overview

- Small surface area (attributes, observers and changing state)
- Connect to the API provider to find a vehicle
- The vehicle object is a model of vehicle state
- Read vehicle attributes (v.location, v.mode, v.parameters ["MAX_THRUST"] etc…)
- Call v.flush() after writes
- Use the observer pattern for notification of state changes.

Lesson 0: Install the SDK

- github: diydrones/droneapi-python - "Embedded Linux Talk"
- Connect to a vehicle or SITL

```
mavproxy.py --master=localhost:14550
module load droneapi.module.api
```

Lesson 1: Go to a location

```
api start simple_goto.py

api              = local_connect()
vehicle          = api.get_vehicles()[0]
vehicle.mode     = VehicleMode("GUIDED")
origin           = Location(-34.36, 149.16, 30,
is_relative=True)

vehicle.commands.goto(origin)
vehicle.flush()
```

Lesson 2: A GCS in 50 lines

- A GCS?
- Observers
- Tk for a GUI

```
root = Tk()
root.wm_title("microGCS - the worlds crummiest GCS")
...
def updateGUI(label, value):
    label['text'] = value
...
v.add_attribute_observer('attitude', lambda attr: updateGUI(attitudeLabel, v.
attitude))
```

Doing this for real

- Find a friend - diydrones.com
- Find or build a drone
- Safety
  - Fly semi-manually first
  - No automatic arming
  - The 'mode switch' is your friend
  - Use a GCS
- Flight tutorial links included in the github

Thanks

- Looking for a fun hobby?
- Looking for a fun job?
- Want to see a balloon pop? (Watch this)
- Want to do this on Android/Java? (See Fredia)
- Contact me: github.com/geeksville
- Questions?