

# Intro to Flask

Brian Cherinka  
.Astronomy 8 - June 18, 2016  
Oxford, UK

# What is Flask

- Flask is a Python Micro-Framework!
- Complete Separation of Data, Back-end Code, and UI/UX
- In Model-Controller-View (MCV) framework
  - Controllers - Your server side code
  - Views - Your front end html template code
  - Model - Your data source (e.g. a database)
- Note: Flask can be very confusing ...
  - Views == Templates
  - Controllers == Views

# Definitions

- Definitions
  - Jinja2 - Html templating system
  - Routes
  - wsgi / uwsgi - Web Server Gateway Interface
  - nginx - server host (like Apache)

# Github

- Go to [github.com/havok2063/introtoflask](https://github.com/havok2063/introtoflask)
- Clone or Download button, and copy link
- `git clone github.com/havok2063/introtoflask .`

# Let's

```
#!/usr/bin/env python2  
# encoding: utf-8
```

```
from __future__ import print_function, division, absolute_import  
from flask import Flask
```

```
app = Flask(__name__)
```

```
@app.route('/')  
def main():  
    return 'This is my brand new app'
```

```
if __name__ == '__main__':  
    app.run()
```



# Simple Example

- Go to `introtoflask/simpleexample`
- `python app.py -d -p 8000`
- `-d` tells Flask to run the app in debug mode

# g and session

- Flask allows for variables to be passed around within special objects
- g - stores variables within a single request
  - `from flask import g`
  - Storing: `g.ra = 234.2344`
  - Retrieving: `ra = g.get('ra', None)`
- session - stores variables across an entire browser session
  - `from flask import session [as current_session]`
  - Storing: `session['ra'] = 234.2344`
  - Retrieving: `ra = session['ra']` or `session.get('ra', None)`
  - session behaves like a Python dictionary

# What is Flask



# Hosting Your App

- Host your locally running app with ngrok
- Host a production version with Nginx or Apache
- Host on a cloud-based platform (e.g Heroku)

# ngrok

- Allows for secure tunnels to localhost
- <https://ngrok.com/>
- `ngrok http [port number]`

# References