

# **Scaling from One to Billions**

## **How to change the world as a software engineer**

**Daniel Roy Greenfeld**

# **Scaling from One to Billions**

## **How to change the world as a software engineer**

**Daniel Roy Greenfeld**

# Olá Portugal!

2022  
**djangocion.eu**  
porto

# What this talk is about

- What we do at an individual level scales up
- What happens when what we do scales up?

# What this talk is about

- What we do at individual level scales up
- What happens when what we do scales up?

Let's go!

Way back in

20004

i had long hair



# Got interview at NASA





**No way I was going to work for NASA**

**No computer science degree  
Only 7 years professional experience**

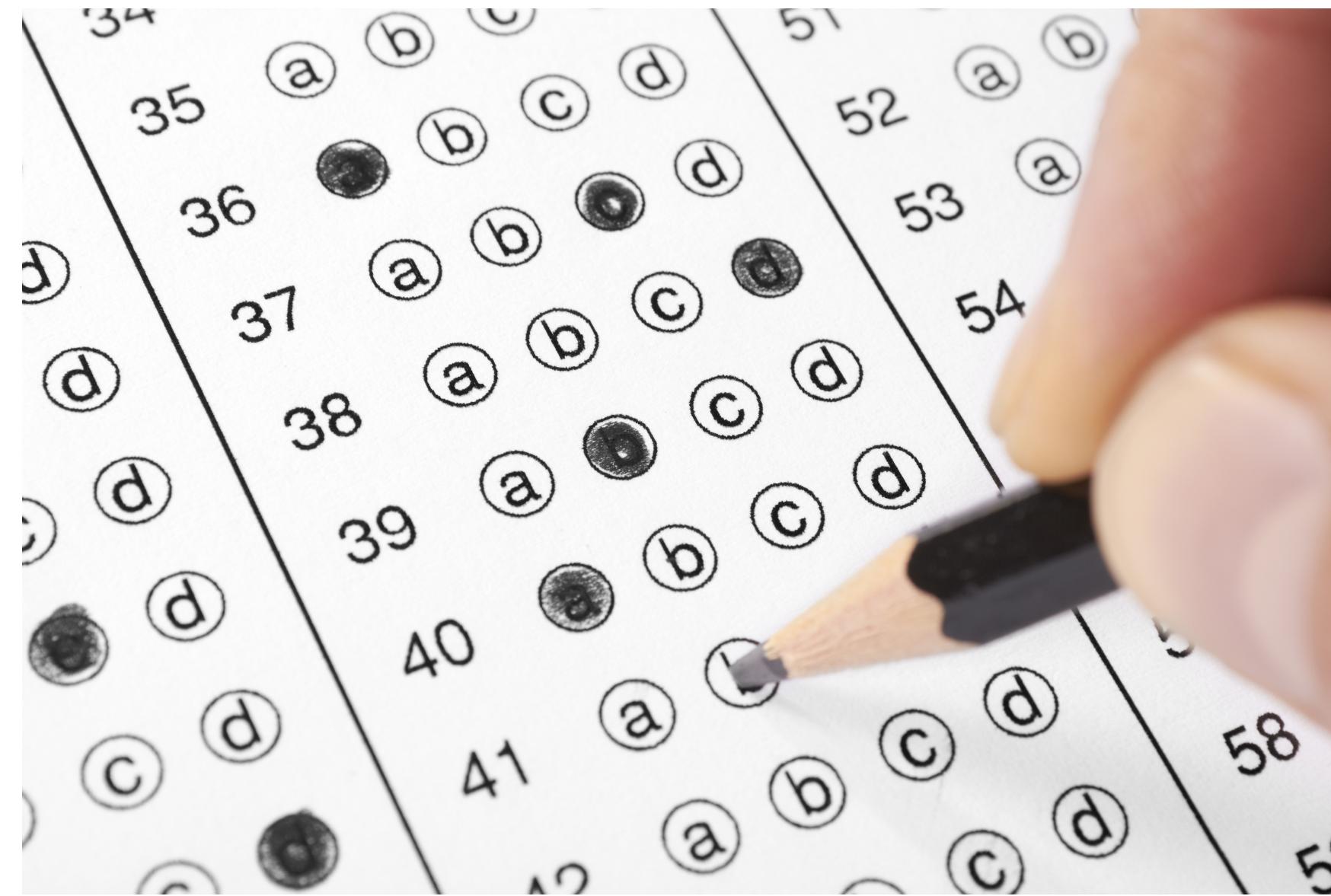
Pydanny

# Treat interview as a practice session



Still from Jackie Chan's 1984 movie "Meals of Wheels"

# Test new approaches to typical questions



**Have fun!**

# How it went

# "What's your biggest weakness?"

- NASA Interviewer

**"My biggest weakness is that I'm emotionally tied to my work so if I'm not doing well I take it personally. Then I work longer and harder to improve."**

- My normal response

This time:

"I'm stupid and lazy"

# NASA Interviewer

WTF?!

# NASA Interviewer

"You going to explain that?"

I'm stupid

# I'm stupid

- Can't figure things out
- Can't remember things
- Too stupid not to ask stupid questions

# Let's go over each item

I'm stupid

Can't figure things out

I'm stupid

Can't figure things out

Always look for libraries first

# Can't figure things out

# OMG argparse

```
import argparse

# sub-command functions
def foo(args):
    print(args.x * args.y)

def bar(args):
    print('(%s)' % args.z)

# create the top-level parser
parser = argparse.ArgumentParser()
subparsers = parser.add_subparsers()

# create the parser for the "foo" command
parser_foo = subparsers.add_parser('foo')
parser_foo.add_argument('-x', type=int, default=1)
parser_foo.add_argument('y', type=float)
parser_foo.set_defaults(func=foo)

# create the parser for the "bar" command
parser_bar = subparsers.add_parser('bar')
parser_bar.add_argument('z')
parser_bar.set_defaults(func=bar)

parser.parse_args()
```

# Can't figure things out

## Yay Typer!

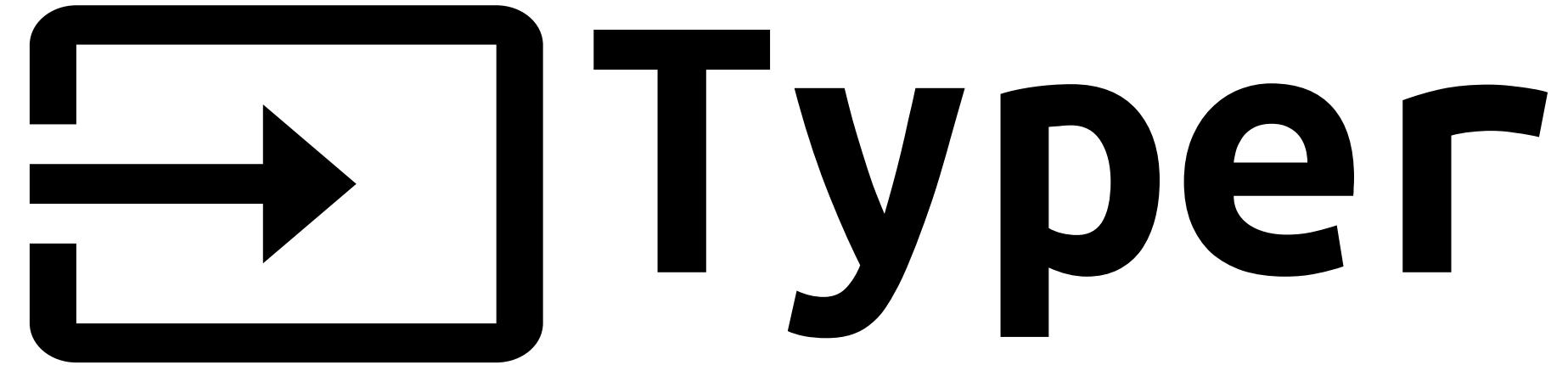
```
import typer

app = typer.Typer()

@app.command()
def foo(y: float, x: int = 1):
    print(x * y)

@app.command()
def bar(z: str):
    print(f'(({z}))')

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



[typer.tiangolo.com](https://typer.tiangolo.com)

I'm stupid

Too stupid not to ask stupid questions

*hint:*

There are no stupid questions

*hint:*

The only bad question is the one  
you do not ask

*important professional advice:*

Don't try to impress people by not  
asking questions

Ashamed to ask questions?

No one cares

No one remembers

Too stupid not to ask stupid questions

Obey the Thirty Minute Rule

[daniel.feldroy.com/30](http://daniel.feldroy.com/30)



# Obey the Thirty Minute Rule

Don't waste more than 30 minutes on a problem without asking questions

# Obey the Thirty Minute Rule

Change to 60 minutes if you like

# Important: Don't be a jerk and not let other people ask questions



I'm stupid

Can't remember things

Can't remember things

Docstrings are awesome

Can't remember things

Typehints are awesome too, use  
them to reduce docstrings

# Remember this?

Typehints means we don't have to write so many docs

```
import typer

app = typer.Typer()

@app.command()
def foo(y: float, x: int = 1):
    print(x * y)

@app.command()
def bar(z: str):
    print(f'(({z}))')

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

Can't remember things

When following a talk or tutorial,

Write down even the slide bullets!

For most people writing down notes  
enhances learning

I'm stupid

Can't remember things

Me back in the day:

- [pydanny-event-notes.readthedocs.io/](https://pydanny-event-notes.readthedocs.io/)
- [daniel.feldroy.com/posts/2011-12-story-of-live-noting](https://daniel.feldroy.com/posts/2011-12-story-of-live-noting)

I'm stupid

Can't remember things

Documentation makes

me look good!

Documentation makes me look good!

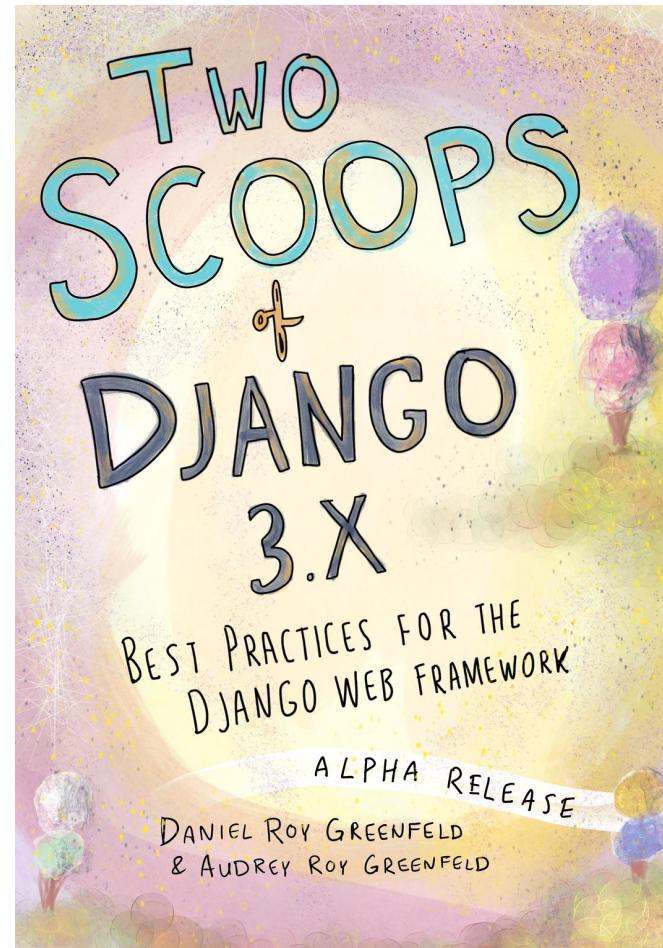
I can pull out old tech details that  
others don't have

Documentation makes me look good!

Can be used later on other projects

# Documentation makes me look good!

## Great for book content



Documentation can make you look good too:

[docs.djangoproject.com](https://docs.djangoproject.com)

[docs.python-requests.org](https://docs.python-requests.org)

[fastapi.tiangolo.com](https://fastapi.tiangolo.com)

**django**



 **FastAPI**

I'm lazy

# I'm lazy

- Don't wanna do anything twice
- Don't wanna debug code when I had it working before
- Don't wanna look hard for docs

# Here we go on laziness!

I'm lazy

Don't wanna do anything twice

If I write the same code twice I stick  
in a function

I'm lazy

Don't wanna do anything twice

I use typehints so I write less docs

I'm lazy

Don't wanna do anything twice

I stick the function into a utils  
module

I'm lazy

Don't wanna do anything twice

I put the function on GitHub so I  
don't lose it

I'm lazy

Don't wanna do anything twice

I put the project on PyPI so I can  
install it easily

I'm lazy

Don't wanna do anything twice

Isn't this the foundation of open  
source?

I'm lazy

Don't wanna debug code that worked before

- Manually testing code by watching it run is hard
- ...and boring
- ...and hence is error prone
- ...meaning you have to do more work

I'm lazy

Don't wanna debug code when I had it working before

Write down how to test lest you forget

## Running the test suite

To run all of the Django Packages tests:

```
docker-compose -f dev.yml run django python manage.py test
```

To run tests for a particular Django Packages app, for example the feeds app:

```
docker-compose -f dev.yml run django python manage.py test feeds
```

I'm lazy

Don't wanna look hard for docs

Every project should have a README that covers:

- Installation
- Running the code
- Testing the code

# Did I get the job at NASA?

?



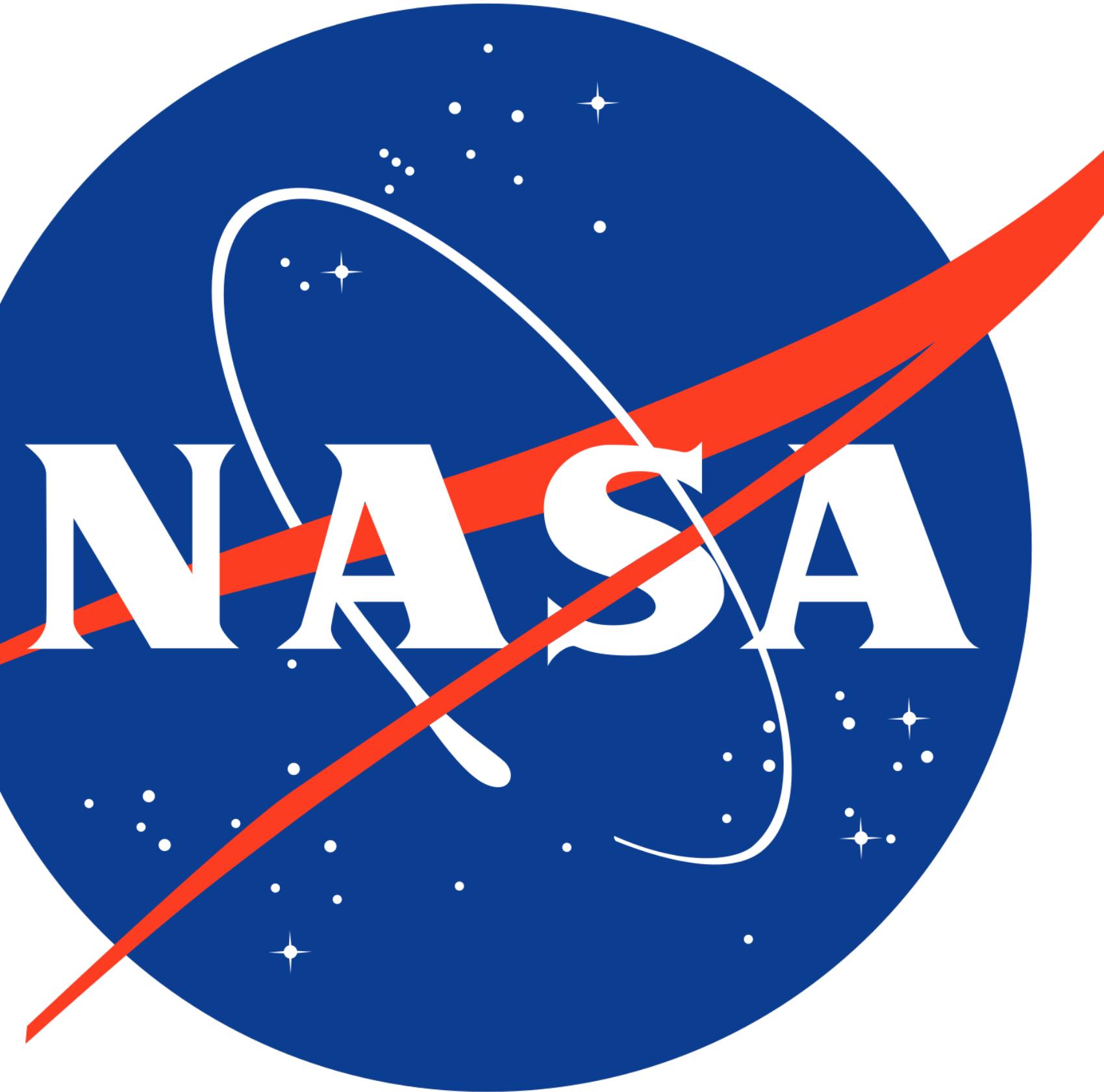
In hindsight I was lucky

Thank you  
Sharon Campbell

One of my favorite managers ever

# Sharon Campbell

If you're watching this, please contact me,  
Octopus Energy  could use your talents



## Worked at NASA 2005-2010

- Got started in Python (Thanks Chris Shenton)
- Met my wife
- Open sourced my first code:
  - What became django-crispy-forms
- **Learned about climate change**
- Coded a lot



# Worked at NASA

Got educated about global climate change

# Doing what I could as an individual

- Recycling
- Composting
- Reduce my carbon load

# Doing what I could as an individual

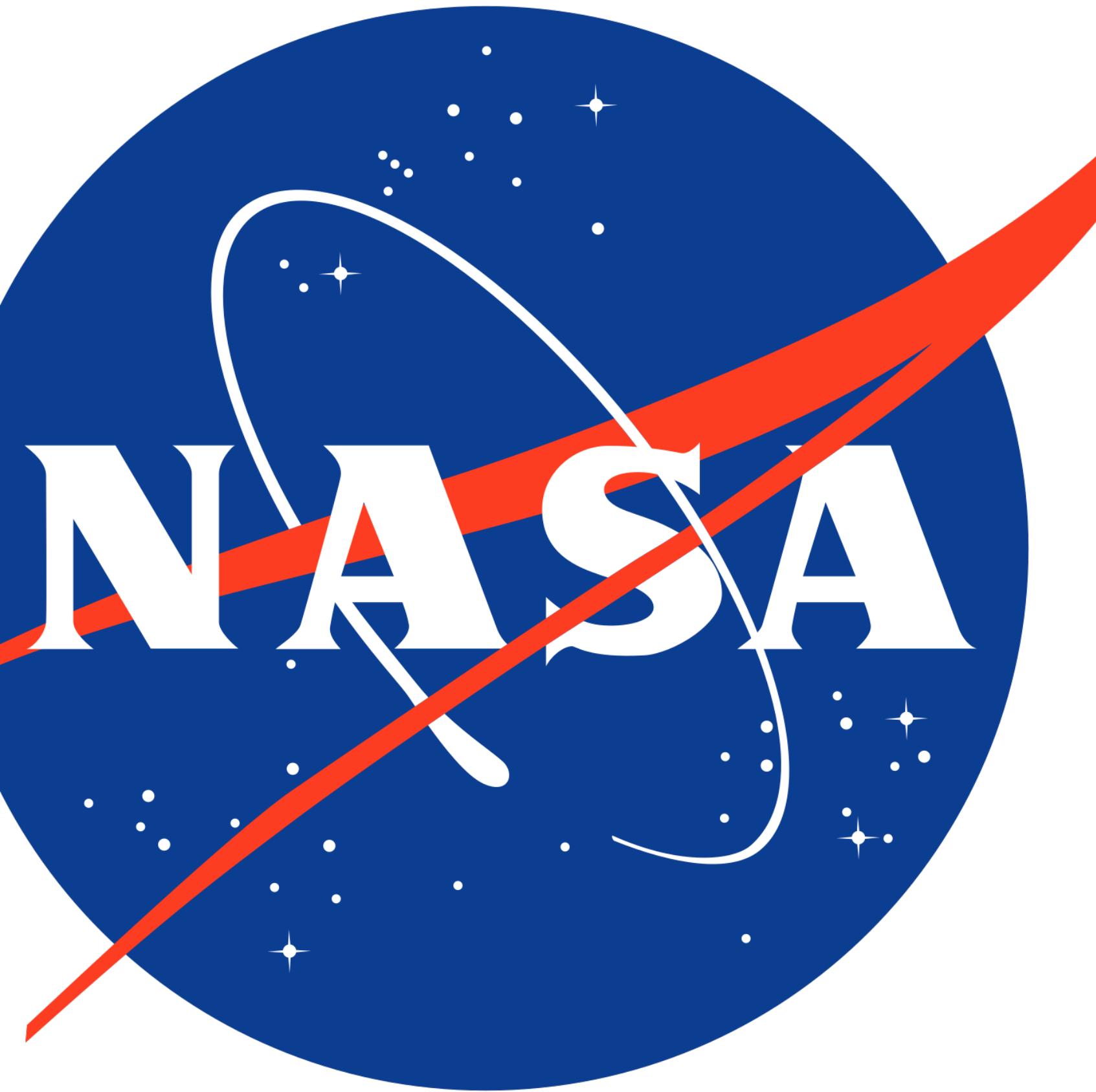
- Recycle?
- Compost?
- Reduce my carbon load?

Does any of that make a difference?

# Individual effort vs.

- **Gigantic fossil fuel interests?**
- **Politicians across the planet?**
- **Those who deny the 99% scientific consensus?**

What could I do as an individual?  
Let's return to this later...



## Worked at NASA 2005-2010

- Got started in Python (Thanks Chris Shenton)
- Met my wife
- Open sourced my first code:
  - What became django-crispy-forms
- Learned about climate change
- **Coded a lot**

# Coded a Lot

# Code, Code, Code

[daniel.feldroy.com/posts/code-code-code](https://daniel.feldroy.com/posts/code-code-code)

If you want to get good at anything  
**Practice**

Practice what you learn from this event

Use it

# Always be typing

# Pairing is great

# But if you're trying to improve?

# Always be the person typing

# Taking a class?

# Always be typing

# No exercises in the class?

**Then you are watching  
a spectator sport**

You will learn as a spectator  
just not as fast  
and...

Mastery can't be  
found as a spectator

**"You can't become a tech CEO by  
watching the social network"**

You can't win at  
sports by watching TV

# To master anything...

# To master coding...

... you have to practice

# Practice makes perfect

# Always be typing

# Code, Code, Code

[daniel.feldroy.com/posts/code-code-code](https://daniel.feldroy.com/posts/code-code-code)

# Diving into open source 2010 onwards

Powered by my own stupid laziness and desire to practice  
*(plus the rush when others liked my work)*

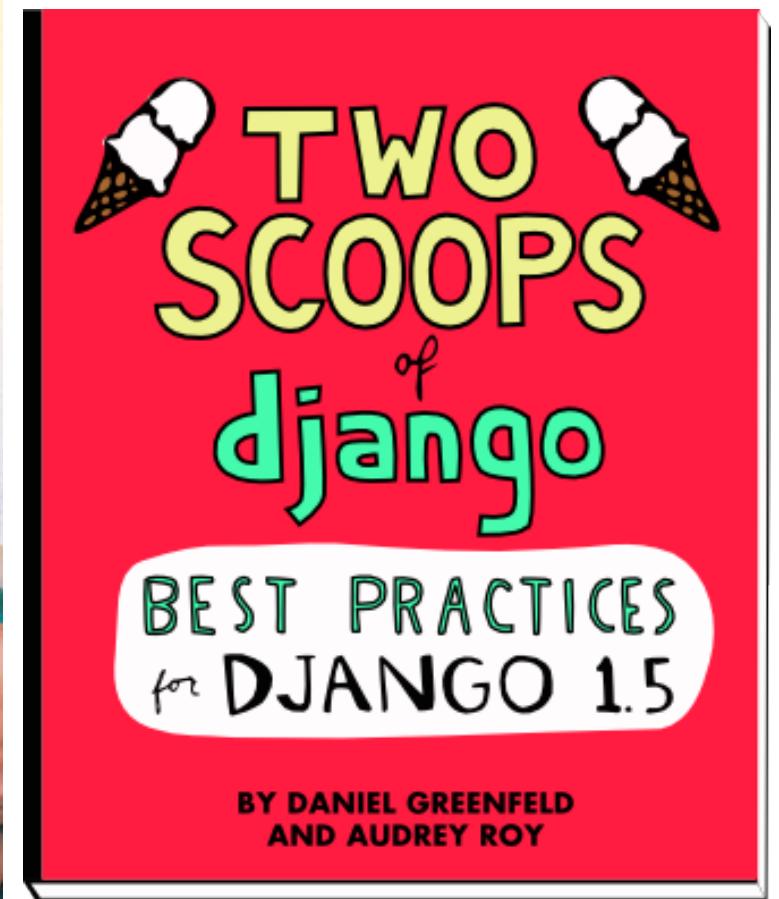
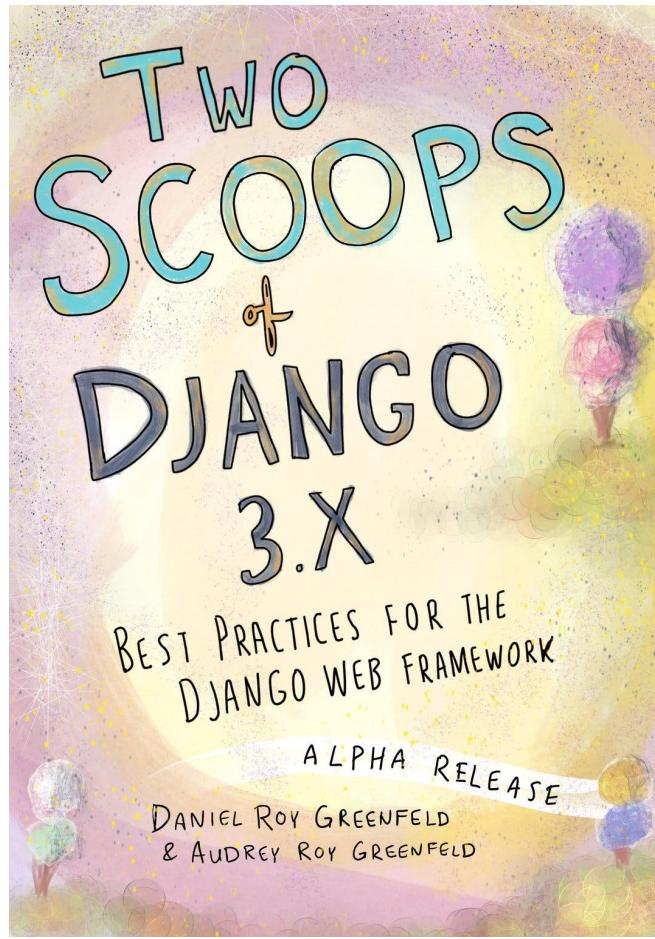
# Open source

A sampling powered by my own stupid laziness and desire to practice

- django-uni-form => django-crispy-forms
- Django Packages
- Django CBV docs refactor
- Cookiecutter
- Cookiecutter Django
- cached-property

# 2013 onwards:

# Sharing Django tribal knowledge



Sharing  
code and knowledge  
is awesome!

# My work used for good

- People using it to land jobs
- Building dream projects
- Various community service efforts
  - Health care firms
  - Charities and non-profits
  - Catching terrorist bombers (Boston Marathon 2013)

But what about when  
my work is used for

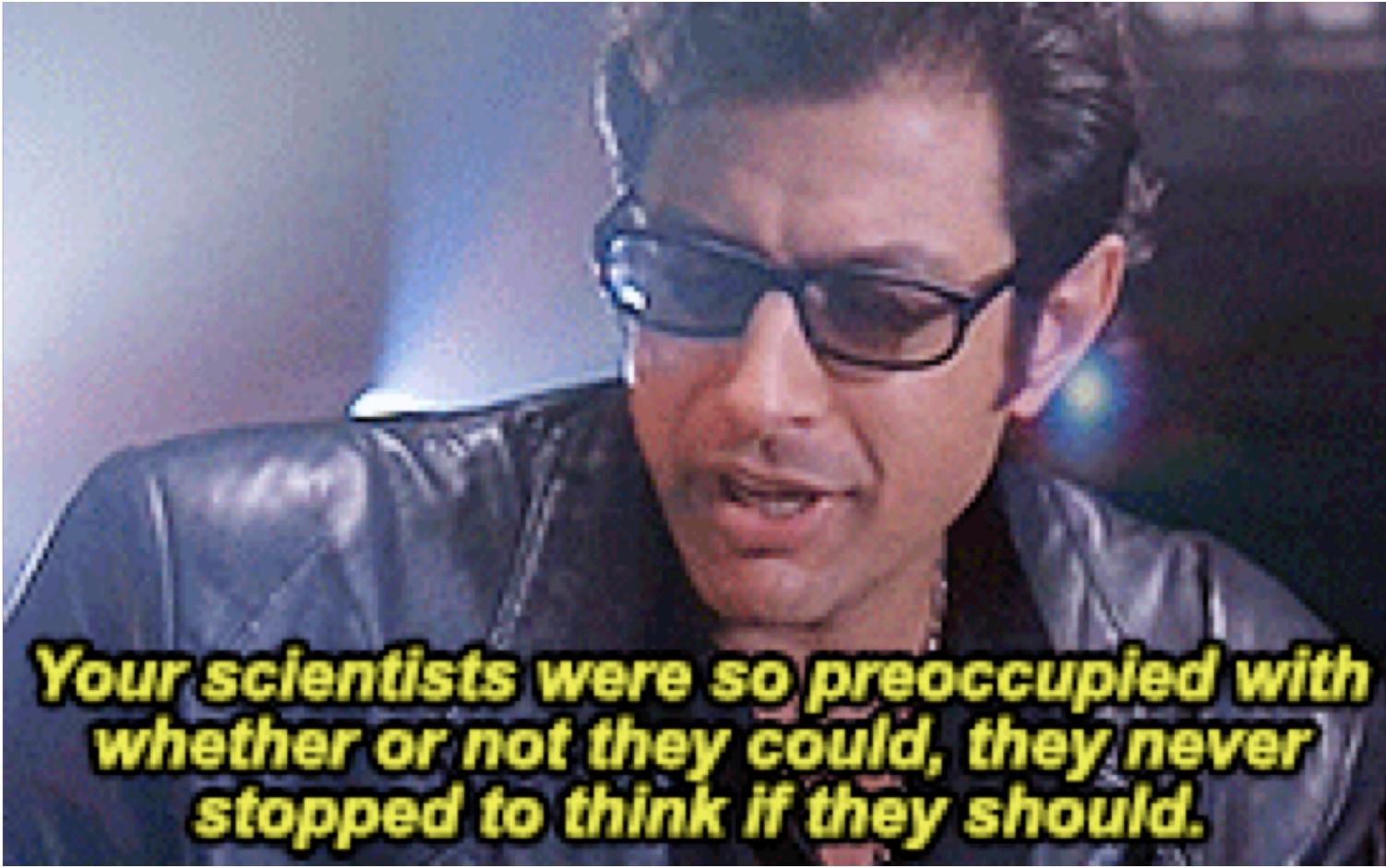
evil?

# But what if my work was used for evil?

- **Logistics for nations who oppress their people**
- **Commercial ventures to spy on private citizens**
- **Site promoting flat earth / no moon landings**
- **App to sell wildlife illegally**
- **Lots more**



# Ethics in Code



**Your scientists were so preoccupied with whether or not they could, they never stopped to think if they should.**

# Questions we should ask of every employer/client

# Who is helped?

# Who is hurt?

# Does this project:

- Promote falsehood over truth?
- Subvert election processes?
- Intrude on individual sovereignty?
- Spew unnecessary carbon into the atmosphere?

In other words, does this hurt our future?

# Does this hurt our future?

Climate change is real

- A series of incrementally hotter years
- In 2022 Europe and South Asia suffered a devastating heat wave
- The South Western USA is under severe water restrictions

This isn't Southern California, rather Auburge, France 2022.



It doesn't have to be this way...

...and here's why

**PROGRAMMING**  
*gives you*  
**SUPER  
POWERS**

A superhero cape is depicted as a series of yellow five-pointed stars of varying sizes, some with black outlines, arranged in a fan-like shape. A single red heart is positioned at the top right corner of the star cluster.

# Programming Gives Us Super Powers

- To improve the lives of ourselves and our families
- To build new and innovative businesses
- To share ideas and knowledge around the world

# Programming Gives Us Super Powers

- To improve the lives of ourselves and our families
- To build new and innovative businesses
- To share ideas and knowledge around the world
- **To save the planet**

**My impact on climate as an individual is tiny**

My climate impact as a software developer  
working on decarbonization and electrification

**Gigantic**



This is going to be my legacy

# Join me in the good fight against climate change

Octopus Energy is hiring

*We have a booth at DjangoCon!*



# Octopus doesn't work for you?

- **Wrong country**
- **Wrong skills**
- **Don't like eight legged mascots**

That's Okay

**PROGRAMMING**  
*gives you*

**SUPER  
POWERS**

A superhero cape is depicted as a series of yellow five-pointed stars of varying sizes, some with black outlines, arranged in a fan-like shape. A single red heart is also integrated into the design.

# Work for the competition

Doesn't matter which green firm "wins", we all benefit

A Aspiration

 Atmos Bank

 Heart Aerospace

# About Me

Daniel Roy Greenfeld

Director of Engineering USA

Author, Coder, Leader

Husband of Audrey Roy Greenfeld

Father

Superhero saving the planet



# Questions



# Links

- <https://daniel.feldroy.com/posts/thirty-minute-rule>
- <https://daniel.feldroy.com/posts/code-code-code>
- <pydanny-event-notes.readthedocs.io/>
- <daniel.feldroy.com/posts/2011-12-story-of-live-noting>
- <typer.tiangolo.com/>
- <fastapi.tiangolo.com/>
- <octopus.energy/careers/join-us/>