



← Notes

▲ Programming - The Django Way

20

Django

Python

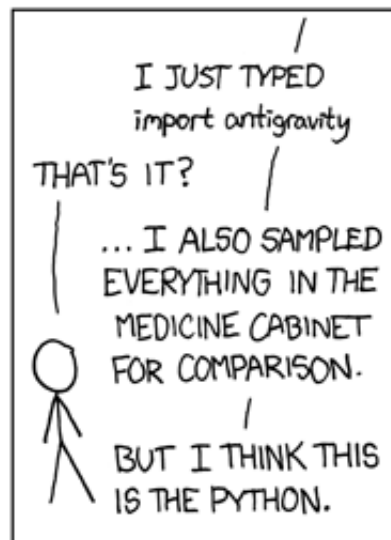
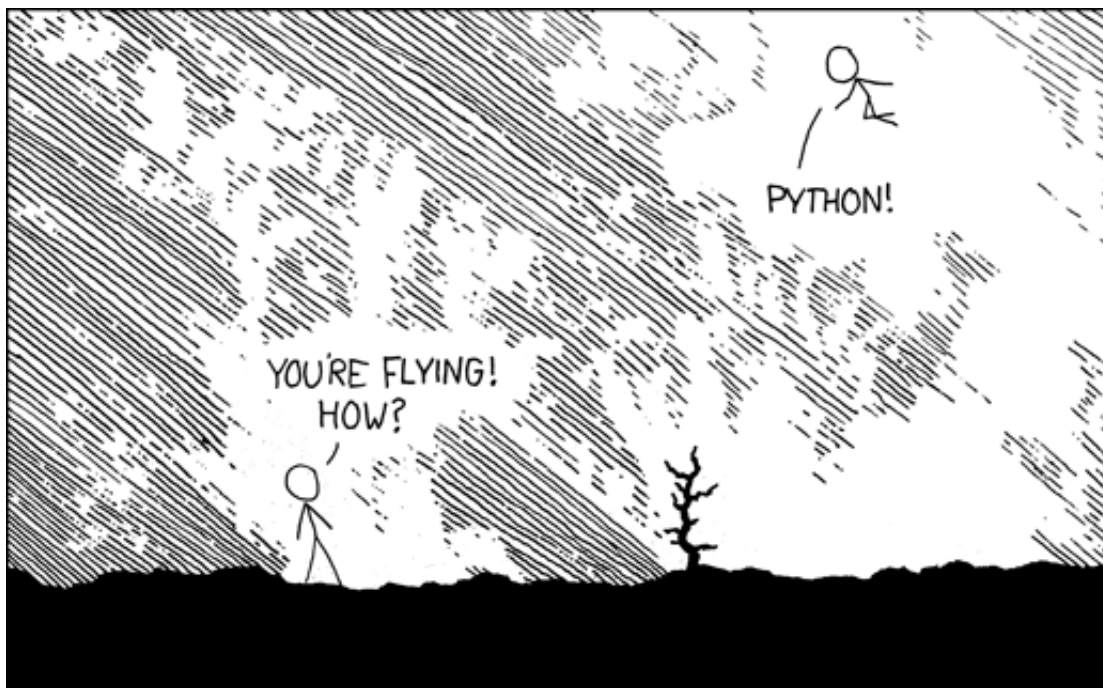
2

When I started web development, PHP seemed the easiest language (and the obvious choice) to get little things done. I would first submit simple data through a form and store it in a database. Eventually, with that knowledge and a bit of AJAX, I came up with a full application, which worked well, but only I knew deep down how bad my procedural programming was. That was two years ago.

Soon thereafter, I shifted to Django for my needs and life has been great ever since. And I wonder it took me two years to write a blog post on the same.

Django creates tables automatically. I mean bloody hell! You just type in the models and run `python manage.py syncdb`. Voila! The tables are created. Django is perhaps the only framework which makes you make great apps using dbs without much knowledge of SQL!

The Form Generator is awesome! Another great feature of Django is the form generator. You make a form class and Django takes care of the rest, right from validation of the data to saving the data by attaching a model to it!



Django is based on Python, the most beautiful language. Development is really fast in Python. Anecdotal evidence suggests that a Python developer can write code in 2 months which would do the same task that two C++ developers would write in a year.

Django Admin. As if the SQL was not enough, Django provides a great admin interface for managing your data.

Django Shell. If you prefer the terminal, you can do the same tasks of Django Admin in the powerful django shell. In short, you can make a kickass application in Django with minimal or almost no knowledge of SQL! Isn't that great?

Generic Views. Detail View, List View - all of these are there in Django by default. You don't need to write the exact same lines of code for these tasks.

Django is not so great with applications that require server push like features like chat. Node.js comes to the picture and works great for bringing such features to life. A chat application using Node.js within the Django framework would work wonders!

To schedule tasks automatically, Celery comes into the picture. For step wise tasks which require to be performed one after the other, Celery is your friend. For a detailed explanation of sending mails through Celery, check out [this blog post](#).

I would be lying if I say Django is the perfect! However, since it is open source and with the development going on in full swing, Django makes a great platform to base your basic app on and then add certain features through other languages or modules.

At the end of the day, with the speed at which developers can make apps in Python/Django, I think it has an edge over any other language!

Like { 0 } Tweet { 0 } G+1 { 3 }

AUTHOR



Shaumik Daityari

 Coordinator, Product Rese...

 Roorkee

 1 note

Write Note

My Notes

Drafts

TRENDING NOTES

[Strings And String Functions](#)

written by Vinay Singh

[Segment Tree and Lazy Propagation](#)

written by Akash Sharma

[Number Theory - II](#)

written by Tanmay Chaudhari

[Matrix exponentiation](#)

written by Mike Koltsov

[Graph Theory - Part II](#)

written by Pawel Kacprzak

[more ...](#)

ABOUT US

[Blog](#)
[Engineering Blog](#)
[Updates & Releases](#)
[Team](#)
[Careers](#)
[In the Press](#)

HACKEREARTH

[API](#)
[Chrome Extension](#)
[CodeTable](#)
[HackerEarth Academy](#)
[Developer Profile](#)
[Resume](#)
[Campus Ambassadors](#)
[Get Me Hired](#)
[Privacy](#)
[Terms of Service](#)

DEVELOPERS

[AMA](#)
[Code Monk](#)
[Judge Environment](#)
[Solution Guide](#)
[Problem Setter Guide](#)
[Practice Problems](#)
[HackerEarth Challenges](#)
[College Challenges](#)

RECRUIT

[Developer Sourcing](#)
[Lateral Hiring](#)
[Campus Hiring](#)
[FAQs](#)
[Customers](#)
[Annual Report](#)

REACH US



Map data ©2015 Google
IIIrd Floor, Salarpuria Business Center,
4th B Cross Road, 5th A Block,
Koramangala Industrial Layout,
Bangalore, Karnataka 560095, India.

✉ contact@hackerearth.com

☎ +91-80-4155-4695

☎ +1-650-461-4192



© 2015 HackerEarth