FIELD NOTE / DJANGO

Panduan Memulai Belajar Django

Petruknisme · 30 Nov, 2016 · 5 min to read

Perlahan saya mulai melirik Python dan setahap demi setahap mulai membiasakan diri menggunakan Python. Karena saya pikir, sudah saat nya saya bermigrasi dan juga karena memang banyak sekali project yang sedang trend di Python, salah satunya adalah Artificial Intelligence, Machine Learning, Deep Learning dan lain-lain. Sebagai bahan belajar Python, maka kali ini saya akan membahas tentang membangun sebuah web menggunakan Django. Saya harap, artikel ini bisa bersambung karena yang akan saya bahas pada kesempatan kali ini hanyalah awalan dari sebuah pembelajaran.

Apa itu Django?

Django merupakan Free dan Open Source Web Framework yang dibangun menggunakan bahasa Python. Sebuah web framework yang didalamnya sudah terdapat komponen untuk lebih mempercepat dan memudahkan web development. Arsitektur yang digunakan oleh Django adalah model-template-view(MTV).

- Model adalah lapisan akses data. Pada lapisan ini terdapat tentang apa saja sebuah data itu: bagaimana cara akses, bagaimana cara validasi dan lain-lain.
- Template adalah lapisan representasi. Pada lapisan ini terdapat presentasi tentang bagaimana seharusnya web tersebut ditampilan.
- View adalah lapisan yang berurusan dengan logika. Pada lapisan ini terdapat logika yang dapat mengakses model dan juga template. Bisa juga disebut sebagai jembatan antara model dan template.

Django sendiri memiliki filosofi dasar yang bisa dibaca pada halaman berikut:
https://docs.djangoproject.com/en/1.10/misc/design-philosophies/
(https://docs.djangoproject.com/en/1.10/misc/design-philosophies/)

> *"The web framework for perfectionists with deadlines."*

Kenapa Django?

- Fast

Django telah dirancang untuk membantu developer membuat aplikasi dari konsep sampai selesai secepat mungkin.

- Flexible

Django cocok digunakan untuk skala project kecil hingga skala project besar.

- Fully Loaded

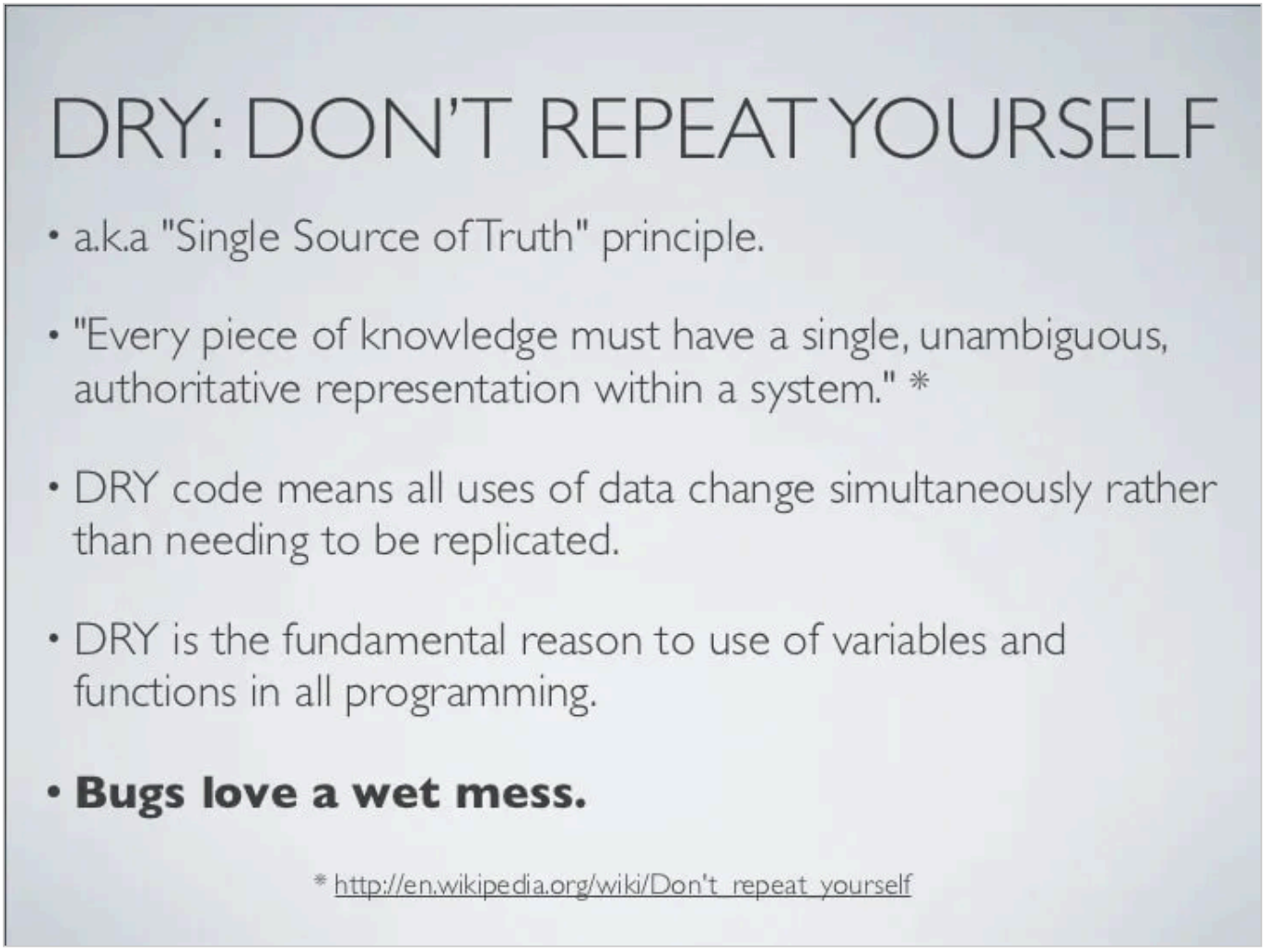
Licensed under CC BY-SA 4.0

Django menyediakan banyak komponen yang dibutuhkan untuk development aplikasi web atau pun mobile. Django itu sendiri lebih lengkap dibandingkan dengan framework python lain nya.

- Cross-Plaftorm

Karena Django ini menggunakan Python, dan kita tahu bahwa python ini bisa berjalan pada platform apapun yang sudah terpasang python.

- Django menggunakan prinsip D.R.Y: Don’t Repeat Yourself



- Good Documentation

Django memiliki web dengan dokumentasi yang sangat lengkap dan terstruktur. Sangat cocok untuk yang sedang belajar untuk tahap awal. Juga disediakan code examples sebagai bahan belajar.

- Secure

Sebenarnya Secure ini relatif dan tidak mutlak karena tidak ada sistem yang aman. Namun, Django sudah include pengamanan untuk serangan umum seperti : SQL Injection, XSS, CSRF dan clickjacking.

Pemasangan Django

Karena Django ini menggunakan Python sebagai bahasa yang digunakan untuk membuat framework. Maka, pastikan terlebih dahulu bahwa kamu sudah memasang python. Kalau belum, silahkan unduh dan pasang sesuai distro masing-masing.

```
$ python
Python 3.5.2 (default, Nov  7 2016, 11:31:36)
[GCC 6.2.1 20160830] on linux
Type "help", "copyright", "credits" or "license" for more information.
```

Lalu pasang Django dengan menggunakan pip, karena itu adalah cara paling mudah:

```
pip install django
```

Pada artikel ini akan menggunakan Django 1.10 dengan menggunakan Python 3.5, lakukan pengecekan terlebih dahulu dengan menggunakan:

```
$ python
Python 3.5.2 (default, Nov  7 2016, 11:31:36)
[GCC 6.2.1 20160830] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import django
>>> print(django.get_version())
1.10.3
```

Bila versi django yang terpasang versi nya kurang dari 1.10, lakukan upgrade python menjadi versi 3.x kalau kamu menggunakan python versi 2.x. Untuk Arch Linux dengan python versi 2.x sudah mendapatkan Django versi 1.10.

```
$ python2
Python 2.7.12 (default, Nov  7 2016, 11:55:55)
```

Licensed under CC BY-SA 4.0

```
[GCC 6.2.1 20160830] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> import django
>>> print(django.get_version())
1.10.3
```

Atau bisa juga cek dengan menggunakan perintah:

```
python -m django --version
python2 -m django --version
```

Yey! Sekarang kita sudah berhasil memasang Django.

Pada pembahasan kali ini saya tidak akan membahas lebih detail tentang Django, tapi hanya dasar-dasar dan pengenalan Django yang lain waktu akan disambung lagi.

Membuat Project Pertama Django

Setelah semua nya telah berhasil terpasang dan berjalan lancar. Tahap pertama yang akan kita lakukan adalah membuat project pertama Django kita. Caranya? Cukup ketikkan perintah:

```
django-admin startproject djangotutorial
```

Maka akan terbuat direktori djangotutorial seperti berikut:

Project structure

```
djangotutorial/
|-- djangotutorial
|   |-- __init__.py
|   |-- settings.py
|   |-- urls.py
|   `-- wsgi.py
`-- manage.py
```

Licensed under CC BY-SA 4.0

Yang dimana masing-masing tersebut adalah:

- **djangotutorial/** adalah direktori inti yang memuat sebuah project. Perlu diketahui bahwa nama tersebut tidak berpengaruh sama sekali dengan Django, jadi bisa ubah sesuka hati.
- **manage.py** sebuah utilitas yang digunakan untuk melakukan interaksi terhadap project Django dengan berbagai cara. Untuk lebih lengkapnya silahkan membaca https://docs.djangoproject.com/en/1.10/ref/django-admin/ (https://docs.djangoproject.com/en/1.10/ref/django-admin/)
- Direktori **djangotutorial/** kedua adalah Python package untuk project yang kita buat. Nama tersebut yang akan digunakan ketika melakukan import package (cont: djangotutorial.urls)
- **djangotutorial/__init__.py** adalah sebuah berkas kosong yang memberitahukan bahwa direktori tersebut merupakan Python package.
- **djangotutorial/settings.py** merupakan pengaturan atau konfigurasi untuk project Django itu sendiri. Lebih lengkap nya silahkan baca disini https://docs.djangoproject.com/en/1.10/topics/settings/ (https://docs.djangoproject.com/en/1.10/topics/settings/)
- **djangotutorial/urls.py** merupakan deklarasi URL untuk project Django; Berisi konfigurasi URL pada project yang kita buat. Akan kita bahas pada kesempatan selanjutnya untuk **URL Dispatcher** dan **URLConf**. Isi berkas urls.py:

djangotutorial/urls.py

```
"""djangotutorial URL Configuration

The `urlpatterns` list routes URLs to views. For more information please s
    https://docs.djangoproject.com/en/1.10/topics/http/urls/
Examples:
Function views
    1. Add an import:  from my_app import views
    2. Add a URL to urlpatterns:  url(r'^$', views.home, name='home')
Class-based views
    1. Add an import:  from other_app.views import Home
    2. Add a URL to urlpatterns:  url(r'^$', Home.as_view(), name='home')
Including another URLconf
    1. Import the include() function: from django.conf.urls import url, in
    2. Add a URL to urlpatterns:  url(r'^blog/', include('blog.urls'))
"""
from django.conf.urls import include,url
from django.contrib import admin

urlpatterns = [
    url(r'^admin/', admin.site.urls),
```

Licensed under CC BY-SA 4.0

```
21  ]
```

- **djangotutorial/wsgi.py** merupakan dukungan kompatibilitas **WSGI** dengan web server untuk menjalankan project Django. Nantinya bisa diintegrasikan dengan menggunakan web server pada umumnya seperti apache, nginx dan lain-lain.

Django itu sendiri sudah menyediakan web server yang dibuat dengan Python untuk melakukan testing dalam masa development. Jadi memudahkan kita untuk development sebelum memasukan server production. Perlu diingat bahwa web server yang disediakan tidak dianjurkan untuk digunakan pada masa production. Pada kesempatan selanjutnya akan membahas tentang integrasi antara WSGI dan web server.

Ok. Sekarang kita lakukan pengecekan untuk memastikan bahwa project Django kita berjalan dengan lancar. Lakukan perintah berikut di direktori utama yang sudah disebutkan diatas:

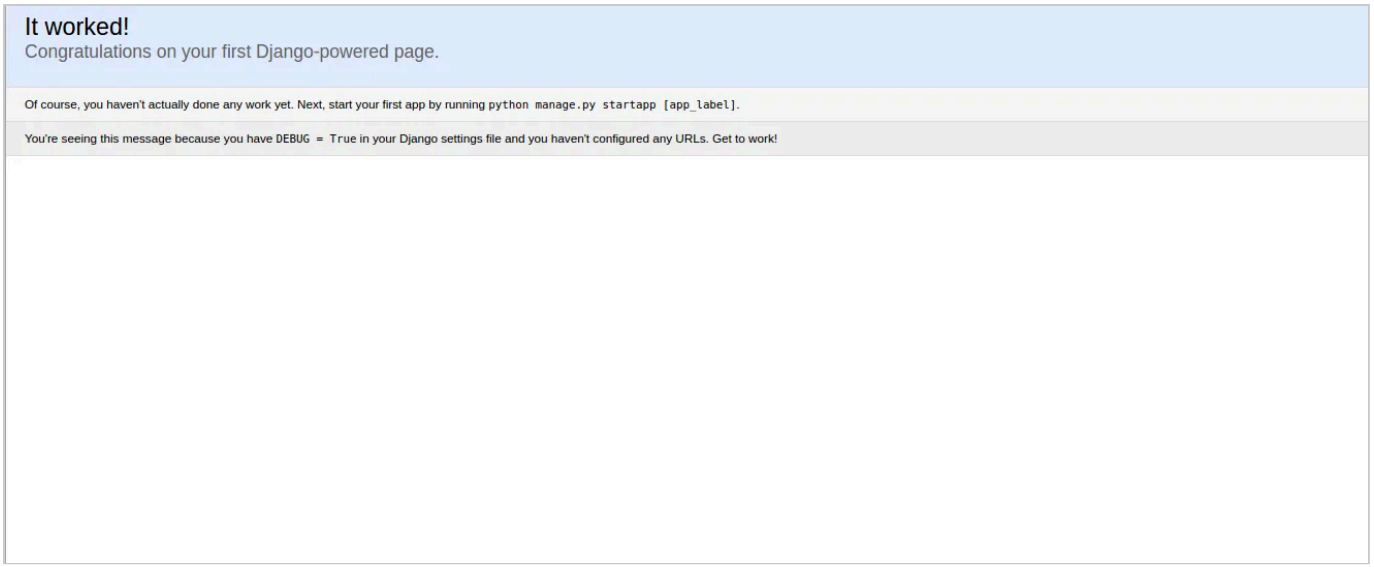
```
1  python manage.py runserver
```

Maka akan menampilkan pesan seperti berikut:

```
1  Performing system checks...
2
3  System check identified no issues (0 silenced).
4
5  You have 13 unapplied migration(s). Your project may not work properly unt
6  Run 'python manage.py migrate' to apply them.
7
8  November 30, 2016 - 01:06:48
9  Django version 1.10.3, using settings 'djangotutorial.settings'
10 Starting development server at http://127.0.0.1:8000/
11 Quit the server with CONTROL-C.
```

Abaikan peringatan yang menyebutkan bahwa ada beberapa bagian yang belum dimigrasi, karena memang kita belum saatnya melakukan itu. Pesan tersebut menyatakan bahwa project Django berjalan lancar dan bisa diakses melalui http://127.0.0.1:8000 (http://127.0.0.1:8000). Akan tampil seperti gambar berikut:

Licensed under CC BY-SA 4.0



Perlu diingat bahwa *runserver* secara default akan menjalankan web server dengan internal IP pada port 8000. Kita bisa mengubah nya dengan menggunakan:

```
python manage.py runserver 8989 # untuk ubah port
python manage.py runserver 0.0.0.0:8989 # untuk rubah ip dan port
```

Oh iya, web server yang sudah berjalan ini sudah otomatis reload ketika kita melakukan perubahan terhadap project. Jadi tidak usah restart web server ketika melakukan perubahan kecuali melakukan penambahan berkas. Lebih lengkap bisa baca disini https://docs.djangoproject.com/en/1.10/ref/django-admin/#django-admin-runserver (https://docs.djangoproject.com/en/1.10/ref/django-admin/#django-admin-runserver)

Mungkin itu saja yang bisa saya bahas pada artikel kali ini. Pada kesempatan selanjutnya saya akan membahas tentang membuat app pada project Django dan seputar URL Dispatcher. Semoga artikel ini bermanfaat. Koreksi bila ada kesalahan

Referensi:

- https://docs.djangoproject.com/en/1.10/intro/tutorial01/ (https://docs.djangoproject.com/en/1.10/intro/tutorial01/)
- https://docs.djangoproject.com/en/1.10/topics/install/#installing-official-release (https://docs.djangoproject.com/en/1.10/topics/install/#installing-official-release)
- http://djangostars.com/blog/why-we-use-django-framework/ (http://djangostars.com/blog/why-we-use-django-framework/)

Licensed under CC BY-SA 4.0

- http://learn.onemonth.com/ten-reasons-django-is-perfect-for-startups (http://learn.onemonth.com/ten-reasons-django-is-perfect-for-startups)
- http://djangobook.com/model-view-controller-design-pattern/ (http://djangobook.com/model-view-controller-design-pattern/)
- http://stackoverflow.com/questions/10079747/understanding-directory-structure-advice (http://stackoverflow.com/questions/10079747/understanding-directory-structure-advice)

Django # Python # Web # Tutorial

Licensed under CC BY-SA 4.0