

Sviluppare siti web in python

Introduzione al framework web “**django**”



Links

Documentazione ufficiale: <https://docs.djangoproject.com>

Repository github: <https://github.com/django/django>

Archivio app riusabili <https://djangopackages.org/>

Archivio di links su django:

<https://github.com/rosarior/awesome-django>

Ambiente di sviluppo

— — —

- Python 3
- pip
- virtualenv

<http://timmyreilly.azurewebsites.net/python-pip-virtualenv-installation-on-windows/>

Introduzione a python

Linguaggio interpretato

Multipiattaforma

Tipizzazione dinamica

Tutto è un oggetto - Introspezione

python2 vs python3

libreria standard

Introduzione a python

- tipi di variabili:
 - int, float, list, str, set, dict
 - <https://docs.python.org/3/library/datatypes.html>
- funzioni
- oggetti e classi
 - costruttore
 - ereditarietà
- moduli

Introduzione a python (2)

Costrutti del linguaggio:

- condizionali
- iterazioni
- built-ins <https://docs.python.org/3/library/functions.html>

Eccezioni

il framework django

— — —

- introduzione
- mvc
 - models
 - views
 - templates

Punti di forza:

- orm
- admin
- applicazioni riusabili

Creiamo un applicativo CRM

— — —

Gestiremo:

- Aziende
- Persone
- Sedi
- Interazioni
- Documenti

Struttura di un progetto django

— — —

- installazione e creazione di ambiente python e di un progetto django

```
pip install django  
django-admin.py startproject crm
```

- configurazione git
- settings
- applicazioni riusabili
- comandi di management

Modelli dati

— — —

- Modelli dati django
- Database backends
 - <https://docs.djangoproject.com/en/2.0/ref/settings/#std:setting-DATABASES>
 - <https://docs.djangoproject.com/en/2.0/ref/databases/#using-a-3rd-party-database-backend>
- ORM
- Migrazioni dati
- Django admin

Creiamo un applicativo CRM (2)

Creazione modelli dati

- Campi
- Campi relazionali
- Opzioni “meta”

Configurazione interfaccia di amministrazione

- registrazione modelli
- configurazione filtri
- configurazione liste
- configurazione viste “inline”

Applicazioni di terze parti

— — —

Debug: `django debug toolbar`

<https://github.com/jazzband/django-debug-toolbar>

REST: `django-rest-framework`

<http://www.django-rest-framework.org/>

URL e viste

ciclo di richiesta e risposta

views

<https://docs.djangoproject.com/en/2.0/topics/http/views/>

urls

<https://docs.djangoproject.com/en/2.0/topics/http/urls/#example>

Comandi base django

`python manage.py migrate`

`python manage.py createsuperuser` => crea un utente

`python manage.py runserver` => avvia il server di sviluppo

`python manage.py inspectdb [nometabella]` => “ispeziona” il db esistente

Django admin

`list_filter`

`list_display`

`search_fields`

`actions`

<https://docs.djangoproject.com/en/2.0/ref/contrib/admin/>

Django admin: filtro “dropdown”

— — —

<https://github.com/mrts/django-admin-list-filter-dropdown>

API / web services

— — —

Framework per creare API

<http://www.django-rest-framework.org/>

Tool di test

<https://www.getpostman.com/>