

# Django

# Environment

- For the sake of this demo, I'm going to use **Eclipse** and **PyDev**.
- If you have time, it's also worth installing **Aptana Studio 3** (the Eclipse plugin version). It's a full-fledged web programming environment for Django, as well as PHP/Rails/...



# Server-side requirements

- **python** (2.4 - 2.7)
- **django** ( $\geq 1.2.4$ )
- **python-twitter**, which depends on httplib2, oauth2 and simplejson (most recent versions)
- Install these with **pip**. To install pip (if you don't already have it), use *easy\_install pip* after setting up python.

# Client-side requirements (javascript)

- These are all provided already (linked) in *app/templates/app/main.html* in the demo project.
- We're going to use **jQuery** and jQueryUI (and the jQuery plugin *hashchange*), as well as a visualization library from Google.



# Importing the Project

1. Go to [github.com/rtirrell/scrapr](https://github.com/rtirrell/scrapr), click on **Downloads** to download a zip or clone the repository with git.
2. **Install** the dependencies with *pip* explained on [github.com/rtirrell/scrapr](https://github.com/rtirrell/scrapr).
3. If you're going to be running from Eclipse, you can create a new Project.
  1. Make sure to create one from existing source, and use the Project type in general.
  2. After creating it, add the PyDev and django natures by right-clicking in the explorer/browser at left.

# Big Ideas

- django is an **MTV** framework (MVC by another name).
  - Developed by Jacob Kaplan-Moss and Adrian Holovaty at the Lawrence Journal-World (a newspaper), released to the world in 2005.
- Most of the big buzzy web frameworks at least adhere to the general principle of separating **persistence** (database) from **processing** (python) from **presentation** (HTML/CSS).
- But the lines can be pretty blurry: “smart models” straddle MV and Javascript the VC.

# Benefits

- django comes with a builtin **admin interface**. This is usually an important component of a large web application, and getting it for free is pretty nice.
- It also ships with an **auth/auth** (authentication/authorization) system. That is hard to do right and easy to do wrong, so it's reassuring to know it's pro.
- Common question: the auth framework is nice, but I want to add x to my user models: see <http://docs.djangoproject.com/en/dev/topics/auth/#storing-additional-information-about-users>.

# django resources

- Help
  - **Mailing lists/IRC** are great resources.  
Same goes for jQuery.
- Apps and extensions
  - Search: haystack, Facebook: *pyfacebook*, CMS: *feincms/django-cms*.
  - **800+ packages** in *pip* (use *pip search*) with 'django' in the name.



# Rich Internet Applications

- Cross-compiling x to javascript
  - **GWT**: java-to-javascript cross-compiler, plus widgets and mechanisms for internationalization, unit testing, history management, browser compatibility.
  - **pyjamas**: port of GWT to python!
- HTML5
  - <http://diveintohtml5.org/> by Mark Pilgrim (also authored <http://diveintopython.org/>, a python tutorial).

