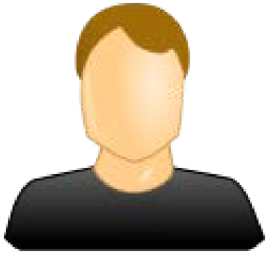
Adding a User Home Page

Reindert-Jan Ekker
nl.linkedin.com/in/rjekker/
@rjekker

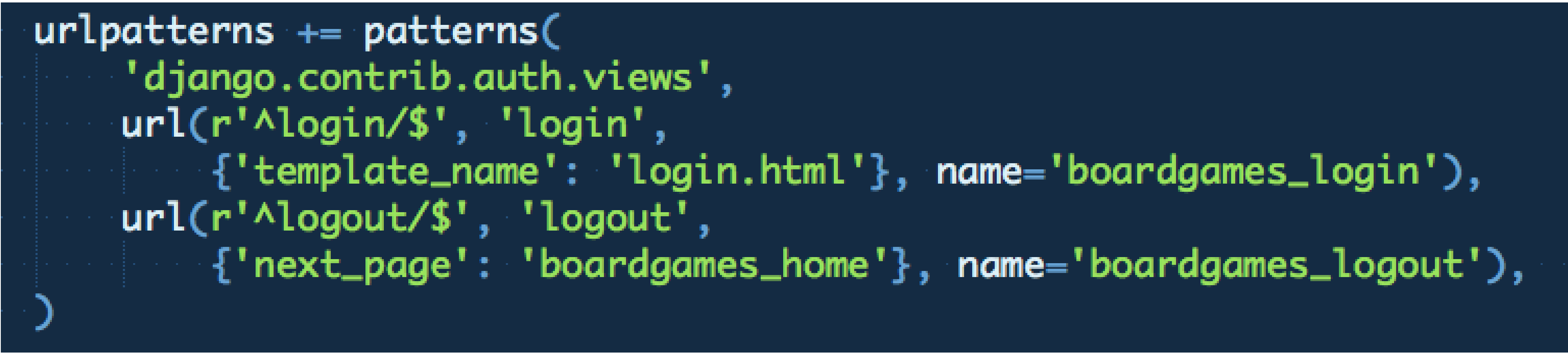
pluralsight
hardcore dev and IT training

Overview

- **User Home Page**
  - Display contents from database
- **Authentication**
- **Template Language**
  - Logic
  - URLs
  - Inheritance
- **URL Mappings**



More About URL Mappings

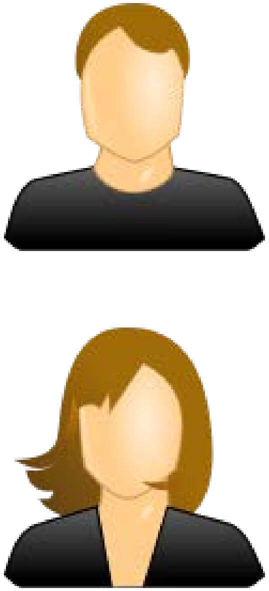


```
urlpatterns += patterns(
    'django.contrib.auth.views',
    url(r'^login/$', 'login',
        {'template_name': 'login.html'}, name='boardgames_login'),
    url(r'^logout/$', 'logout',
        {'next_page': 'boardgames_home'}, name='boardgames_logout'),
)
```

- Add patterns with +=
- `patterns` function has a prefix string as a first argument
- Views can receive keyword arguments
- URLs can be named

Login and Logout Views

- **`django.contrib.auth.views.login`**
  - Have to provide template yourself
- **`django.contrib.auth.views.logout`**
  - Have to give “next” page as an argument
- **`settings.py`**
  - `LOGIN_URL, LOGOUT_URL, LOGIN_REDIRECT_URL`
- **`{% if user.is_authenticated %}`**
- **More about Django authentication: http://goo.gl/qswNHN**



Templates: Tags and Variables

- **Variables: `{{ var }}`**
  - Will render the value of the item “var” from the context
- **Tags: `{% tag %}`**
  - Built-in tags: for, if, url, etc.
  - Built-in tag reference: http://goo.gl/l7RJri
- **Custom Tags**
  - From third-party apps
  - Write them yourself

Templates: if, url

```
{% if test %}
     <html_if_true>
  {% elif %}
     <html_if_false>
  {% else %}
     <html_else>
 {% endif %}
```

- **{% url ‘login’ %}**
  - Will generate a url for the URL named “login”

App URL Configuration

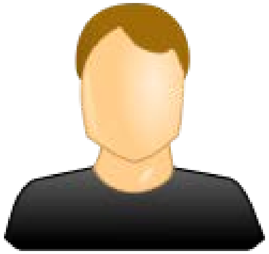
- **App has its own urls.py**
- **Include it in project urls.py:**
  - `url(r'^prefix/', include(‘myapp.urls’))`

Template Inheritance

- `{% extends “base.html” %}`
  - Extend a base template
  - Must be first tag
- `{% block content %}…{% endblock %}`
  - Defines a block that can be overridden by child templates
- **More info: http://goo.gl/nk1b1L**

Login_required

- **`django.contrib.auth.decorators.login_required`**
  - Decorate View functions
  - Non-logged in users will be redirected to `LOGIN_URL`



Template Context

- **Data is provided to the template by a RequestContext**
  - We add our own values as a dictionary
  - `render(request, 'myapp/index.html', {"foo": "bar”})`
- **All data you want to display should be present in the context**
  - Use the view and model to retrieve all data and do business logic
- **Template documentation: http://goo.gl/DKsS6H**
  - More technical details: http://goo.gl/B0VM0P

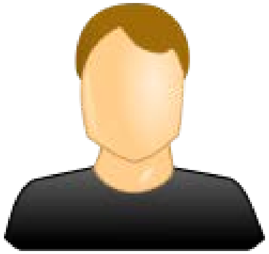
Templates: For, Include

-
  ```
  {% for item in list %}
      <html_for_item>
   {% empty %}
      <html_if_empty>
   {% endfor %}
  ```

- {% include "tictactoe/game_list_snippet.html" with header="Other active games" games_list=other_games %}

Summary

- **User Home Page**
  - Display contents from database
- **Authentication**
- **Template Language**
  - Logic
  - URLs
  - Inheritance
- **URL Mappings**



App URL Configuration

- **App has its own urls.py**
- **Include it in project urls.py:**
  - `url(r'^prefix/', include(‘myapp.urls’))`
- **Use named groups in your expression to capture parts of URL**
  - `(?P<name>expr)`
  - `(?P<id>\d+)`
  - `r’^blog/(?P<year>\d{4})/(?P<month>\d{2})/(?P<day>\d{2})/$’`
- **Captured values are passed to view as keyword arguments**
  - `def get(self, request, year, month, day)`

Templates: Lookup

- `{{ user.name }} or {% for m in game.move_set %}`
- **A dot causes:**
  - Dictionary lookup
  - Attribute lookup
  - Method call
  - List-index lookup

```
{{ list.5 }} # list[5]

{% if dict.key == “somevalue” %}

    # dict[“key”]

{% if g.move_set.count > 3 %}

    # g.move_set.count()
```