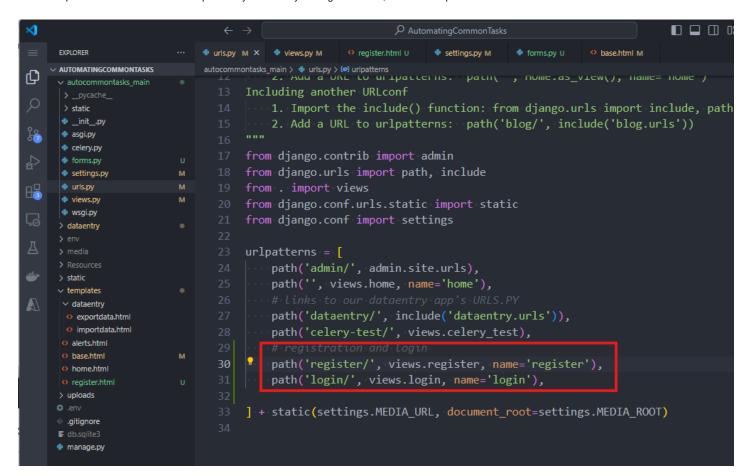
Topic: Login / Logout & User Restrictions Module Part 15

Speaker: Udemy Instructor Rathan Kumar | Notebook: Django: Automating Common Tasks



1. We completed the REGISTER module previously. To make your Login link work, we need to update our URLS.PY



2. Create our LOGIN function in the VIEWS.PY

```
def login(request):
    return render(request, 'login.html')

def logout(request):
    return render(request, 'home.html')
```

```
×
                                                            ◇ login.html U X ◇ register.html U ❖ settings.py M ❖ forms.py U
                                              a views py M
                                                                                                                         O base.html M
    V AUTOMATINGCOMMO... □ U ☐ ··· templates > ◊ login.html > ② div.container.mt-5
Ð
     v autocommontasks_main
1 {%extends 'base.html' %}
                                         {% block content %}
      > static
      __init__.py
      asgi.py
      celery.py
      forms.py
                                         <div class="container mt-5" style="max-width: 600px;">
      urls.py
      views.py
                                              <form action="{% url 'login' %}" method="POST">
      > dataentry
                                                   <div class="form-group">
      > static
      templates

✓ dataentry

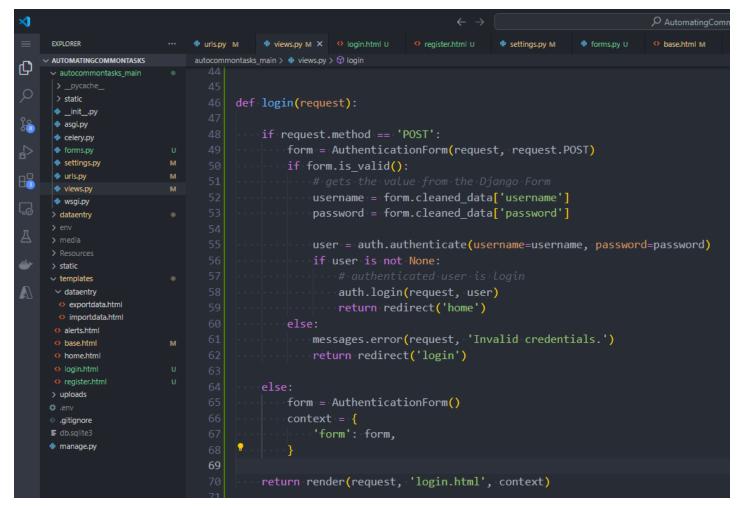
       exportdata.html
                                                   <div class="form-group">
       o importdata.html
                                                        <input type="submit" value="Login" class="btn btn-primary">

    alerts.html

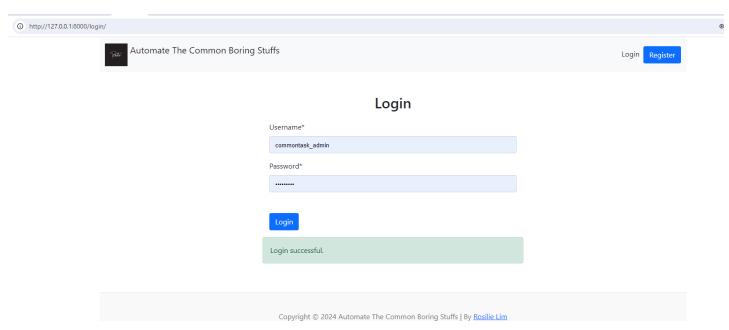
      o base.html
                                                   </div>
      O home.html
       login.html
                                    20
      > uploads
      .gitignore
     ■ db.sqlite3
      manage.py
```

4. Update our BASE.HTML

5. Update our VIEWS.PY



Run the code:

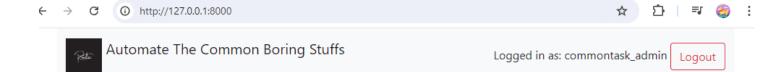


6. We should not show REGISTER button if we are logged in, so we update our BASE.HTML AS:

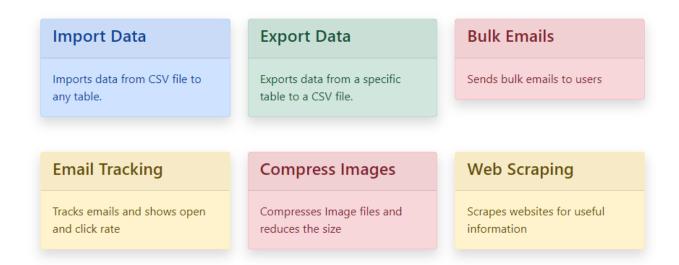
7. To make our logout button work, we update our VIEWS.PY AS:

```
def logout(request):
    auth.logout(request)
    return redirect('home')
```

8. Test our code:



Automate The Boring Stuff With Django



Copyright © 2024 Automate The Common Boring Stuffs | By Rosilie Lim

9. To make see only the tools when they log in, we must update our HOME.HTML. We add the IF ELSE statement so that the tools would only appear if we log in otherwise we see the homepage with some information instead.

```
AutomatingCommonTasks
                                                                                                                                       O bas
     EXPLORER
                             urls.pv M
                                         views.pv M
                                                                                               # custom.css M
                                                                                                             settings.pv M
                                                                                                                           forms.pv U

✓ AUTOMATINGCOMMONTASKS

ф

✓ static

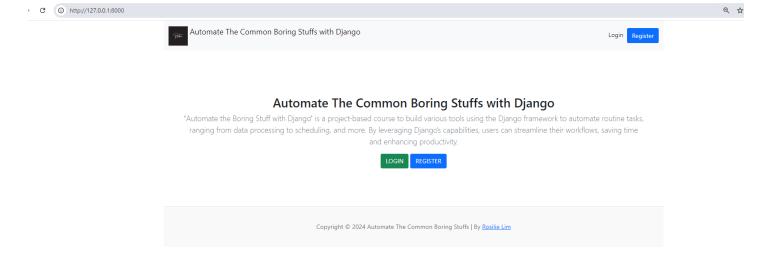
                                    <div class="container" style="padding: 50px;margin-top: 5%;">
      > images
     _init_.py
     asgi.py
                                        <h2 class="text-center">Automate The Common Boring Stuffs with Django</h2>
     celery.py
                                         <div class="row pt-5">
     settings.py
     urls.py
                                             <div class="col-md-4">
                                                  <a href="{% url 'import_data' %}" class="text-decoration-none text-dark">
     wsgi.py
                                                      <div class="card alert-primary shadow">
     > dataentry
                                                          <div class="card-header">
                                                              <h4>Import Data</h4>
                                                          <div class="card-body">
     templates
                                                              Imports data from CSV file to any table.

√ dataentry

      o exportdata.html
      importdata.html
      ○ home.html
     O login.html
                                             <div class="col-md-4">
                                                 <a href="{% url 'export_data' %}" class="text-decoration-none text-dark">
                                                      <div class="card alert-success shadow">
     .gitignore
                                                          <div class="card-header">
    manage.py
```

```
views.py M
                                                                                                      settings.py M
                           urls.py M
                           templates > 0 homehtml > 0 div.container > 0 div.container > 0 div.container > 0 div.col.md.4 > 0 atext-decoration-nonetext-dark > 0 div.card.alert-prin 3 <div class="container" style="padding: 50px;margin-top: 5%;">
Q
      # custom.css
                                          <div class="col-md-4">
    <div class="card-body">
                                                          Scrapes websites for useful information
                                     {% else %}
                                      <div class="home-info">
                                         <a href="https://www.class="text-center">Automate The Common Boring Stuffs with Django</h2>
                                          "Automate the Boring Stuff with Django" is a project-based course to build various tools using the Django
                                              By leveraging Django's capabilities, users can streamline their workflows, saving time and enhancing
     > uploads
     .gitignore
                                          {% endif %}
                            116 {% endblock %}
```

9. Load the page:



From here, you can login or register to see the Django tools.

We have made some custom CSS for our footer and home-info: We run COLLECTSTATIC to make sure that this external CSS is added to our root folder for deployment.

```
×
      EXPLORER
                                   urls.py M
                                                 views.py M
                                                                login.html U
                                                                              home.html M
                                                                                                o register.html U
     ∨ automatingcommo... 📴 🖔 🗐 …
                                   autocommontasks_main > static > css > # custom.css > 4 .home-info

✓ static

✓ CSS

                                            .blog-footer {
        # custom.css
                              М
                                                 padding: 2.5rem 0;
        > images
        ) js
                                                 color: □ #727272;
       __init__.py
                                                 text-align: center;
       asgi.py
                                                 background-color: #f9f9f9;
       celery.py
                                                 border-top: .05rem solid ■#e5e5e5;
      forms.py
settings.py
      urls.py
      views.py
                              М
                                            .blog-footer p:last-child {
                                                margin-bottom: 0;
      > dataentry
      > env
      > static
      templates

√ dataentry

       o exportdata.html
                                            .form-group {
       importdata.html
                                                 padding: 15px !important;
      o alerts.html
      O base.html
      O home.html
       o login.html
      > uploads
      .gitignore
      ■ db.sqlite3
                                            .home-info {
      manage.py
                                                 padding: 20px;
```