Topic: Image Compression 25: Setup, Model & Logic

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

---

Overview

The Python Imaging Library adds image processing capabilities to your Python interpreter.

This library provides extensive file format support, an efficient internal representation, and fairly powerful image processing capabilities.

The core image library is designed for fast access to data stored in a few basic pixel formats. It should provide a solid foundation for a general image processing tool.

1. Go to Pillow documentation and install the Django package in the Djngo server terminal and in Celery terminal

```
$ pip install pillow
```

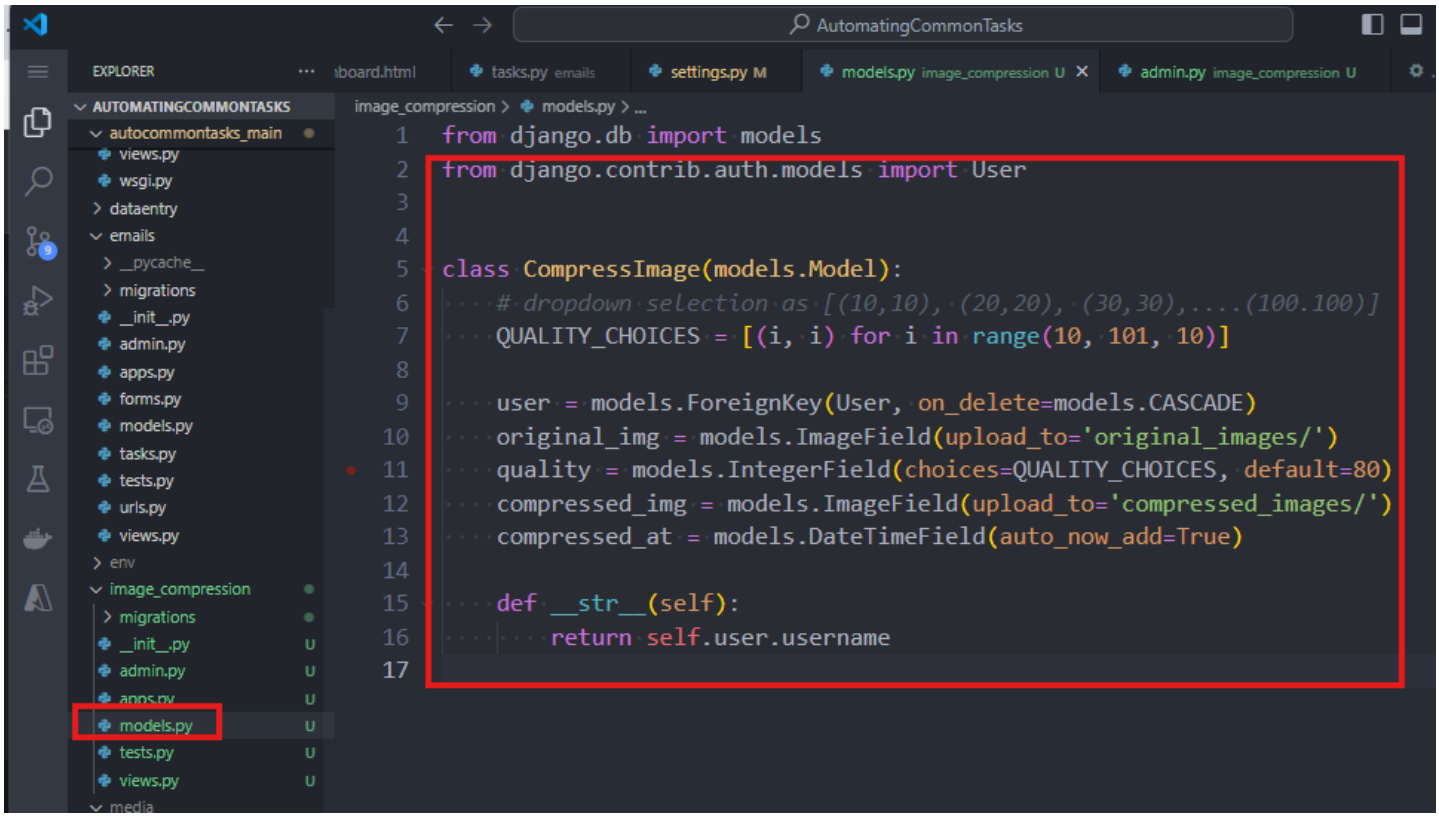
2. In the Django server, we create a new app IMAGE_COMPRESSION

```
$ python manage.py startapp image_compression
```

3. Register the new app in SETTINGS.PY INSTALLED_APPS

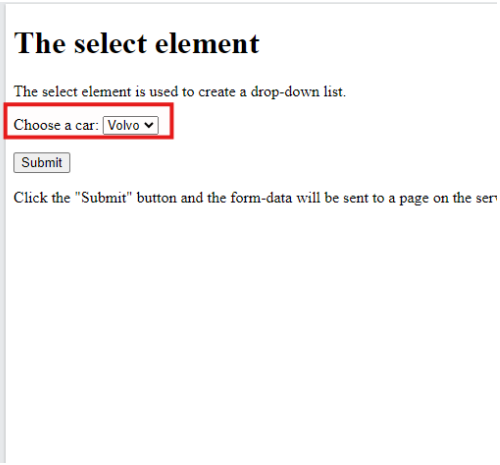
```python
# Application definition

INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'dataentry',
    'uploads',
    'crispy_forms',
    'crispy_bootstrap5',
    'emails',
    'ckeditor',
    'anymail',
    'image_compression',
]
```

4. Create the model in MODELS.PY

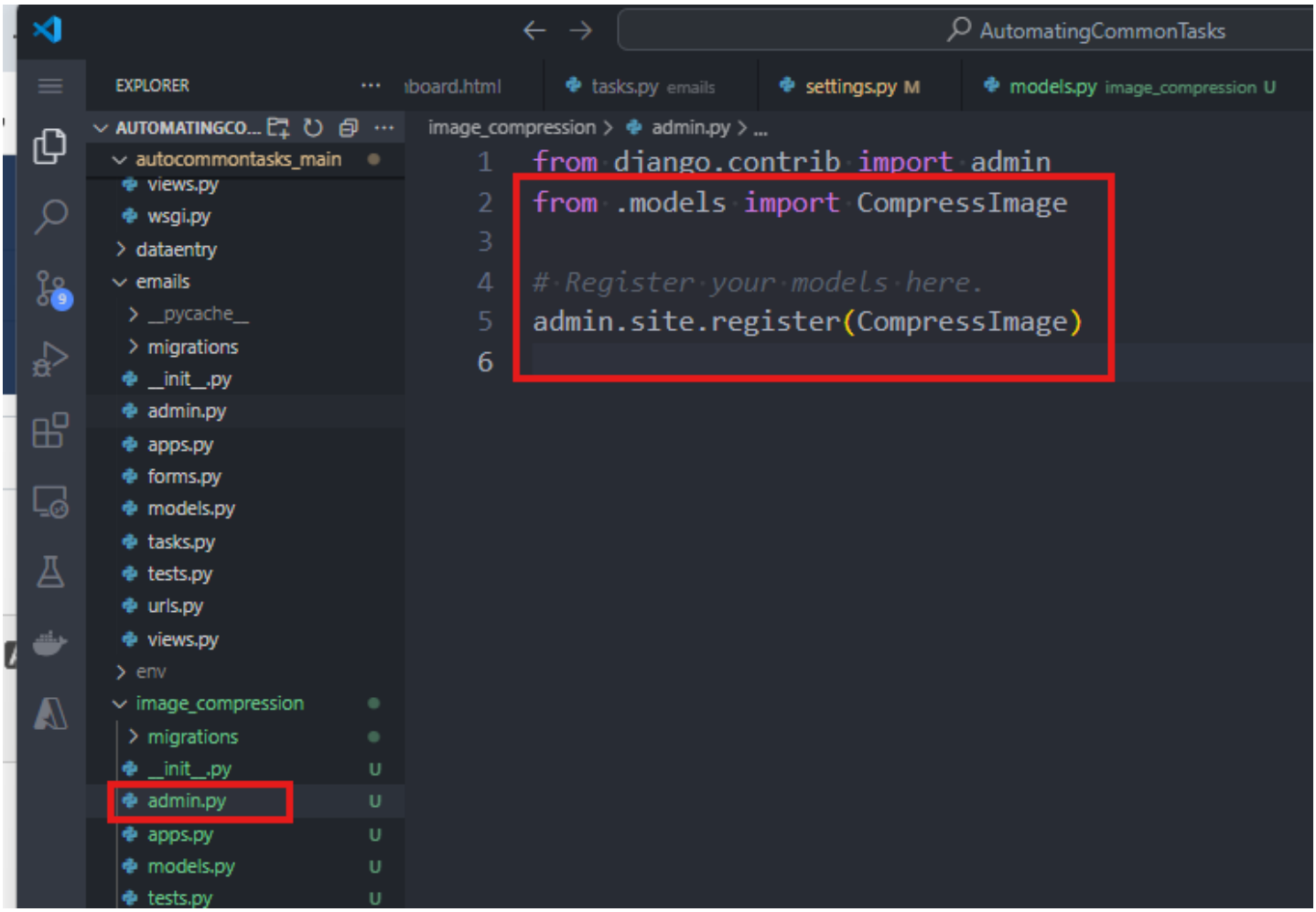


To make the QUALITY field a drop-down option, we use the SELECT TAG. Where we see the VALUE AND THE LABEL (which is displayed as a user option in the dropdown)

```html
<!DOCTYPE html>
<html>
<body>

<h1>The select element</h1>

<p>The select element is used to create a drop-down list.</p>

<form action="/action_page.php">
  <label for="cars">Choose a car:</label>
  <select name="cars" id="cars">
    <option value="volvo">Volvo</option>
    <option value="saab">Saab</option>
    <option value="opel">Opel</option>
    <option value="audi">Audi</option>
  </select>
  <br><br>
  <input type="submit" value="Submit">
</form>

<p>Click the "Submit" button and the form-data will be sent to a page on the
server called "action_page.php".</p>

</body>
</html>
```



5. Register the model for our ADMIN panel. Update ADMIN.PY:

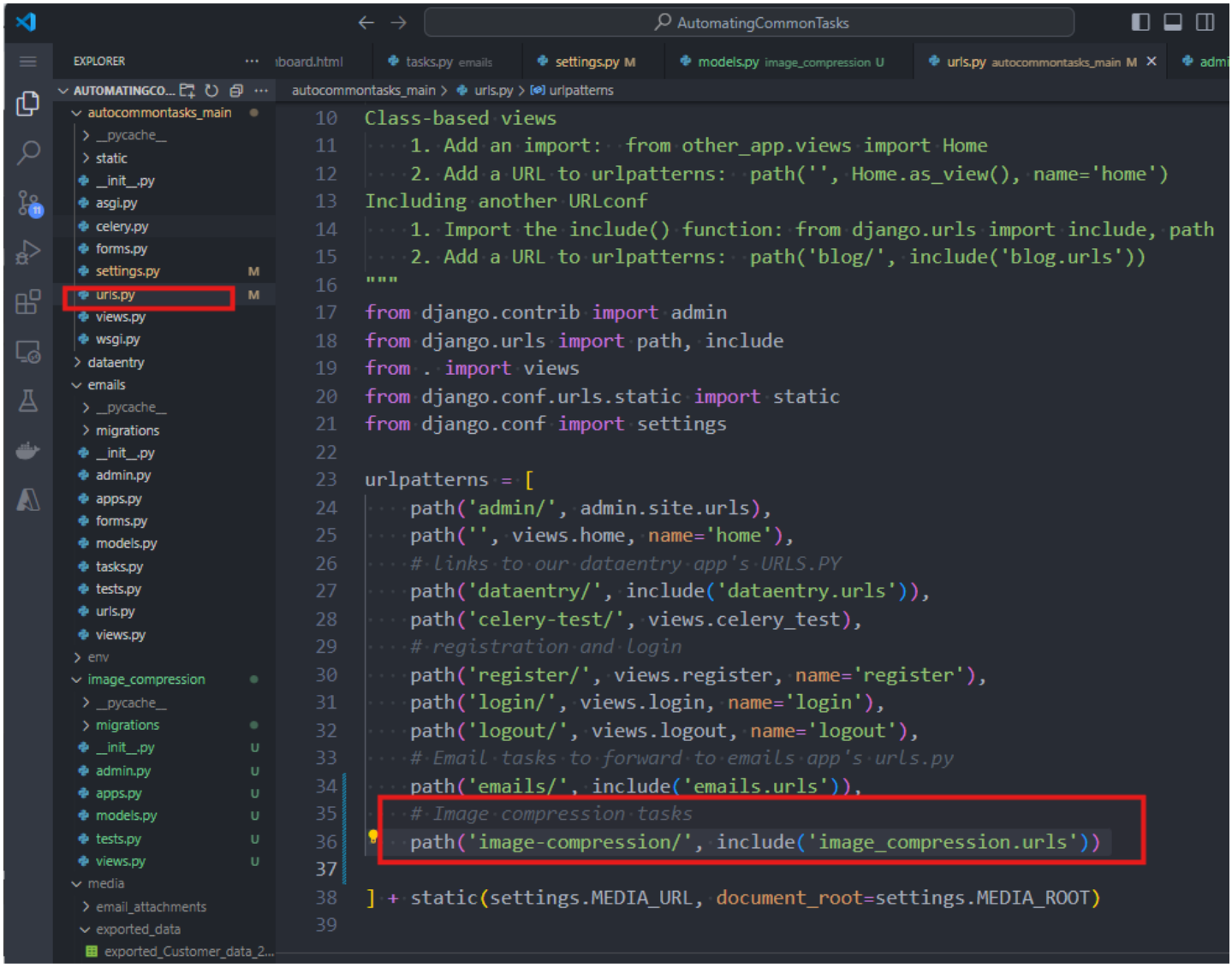


6. Make the necessary model migrations

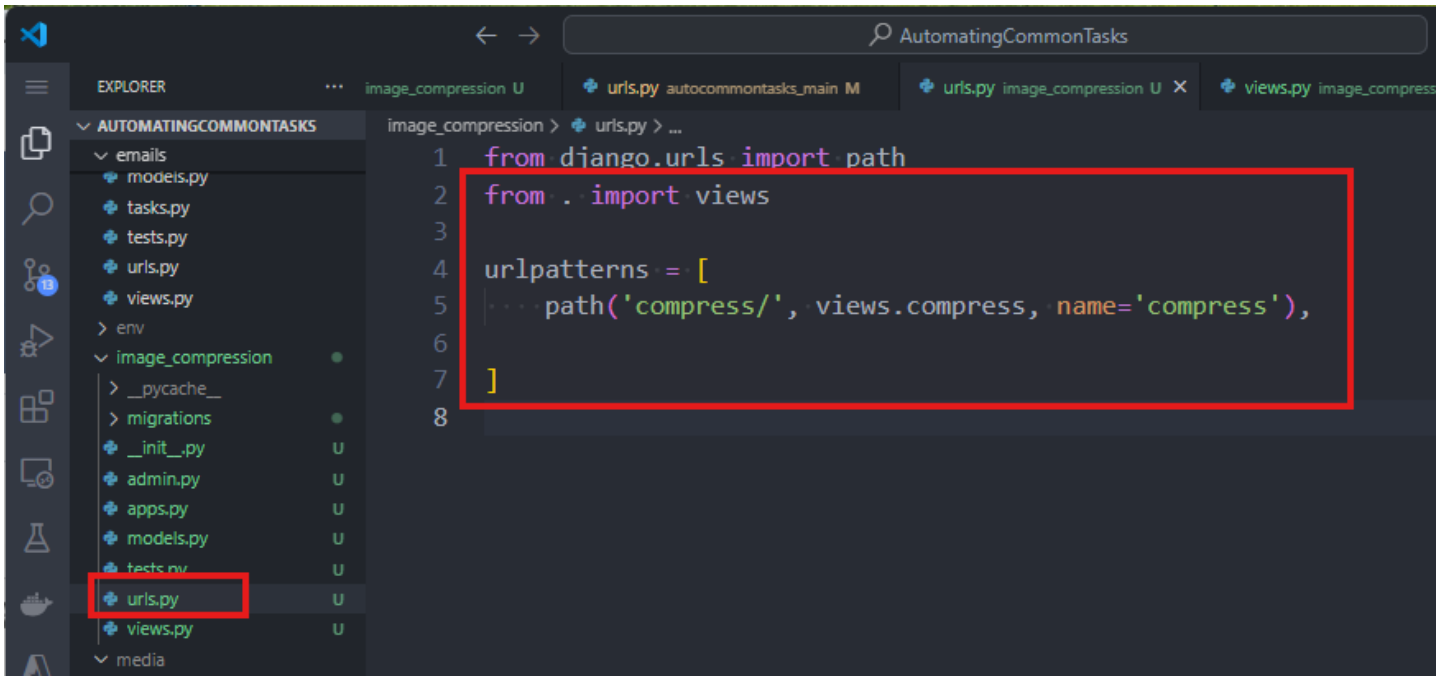
```
$ python manage.py makemigrations
```

```
$ python manage.py migrate
```

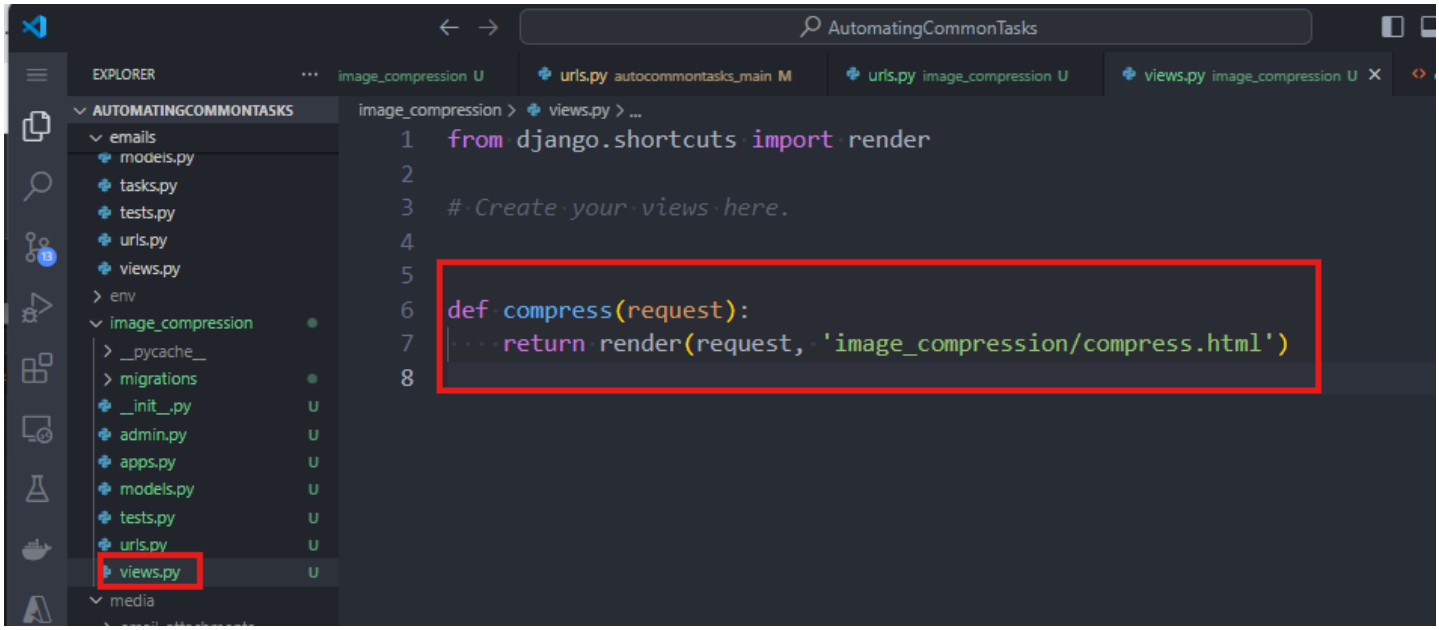
7. Create the new URL pattern. Since we have a new app, we need to create a new pattern in our main project's URLS.PY.



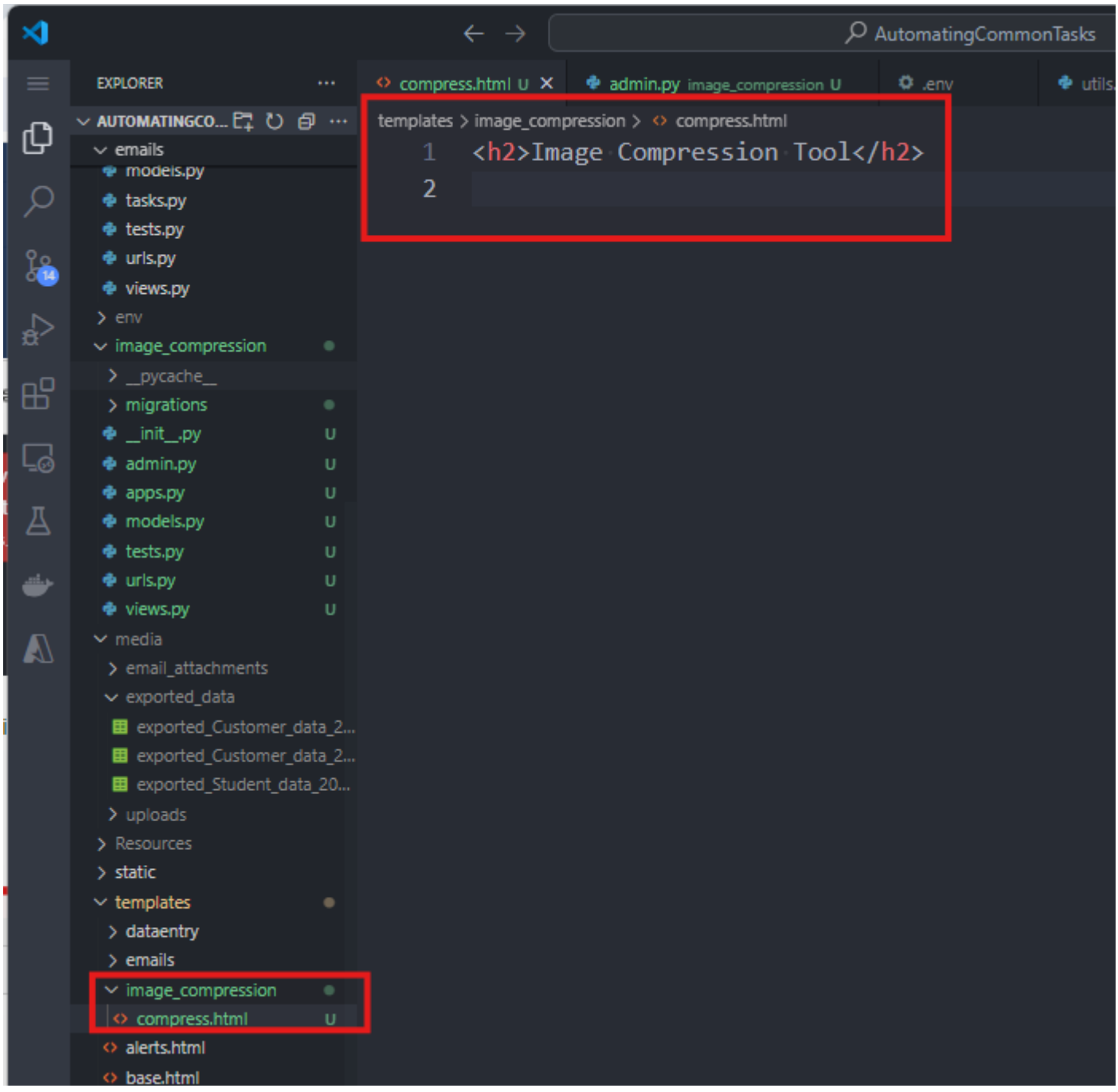
Then, create a new URLS.PY file in our new app for image-compression-related URL paths.



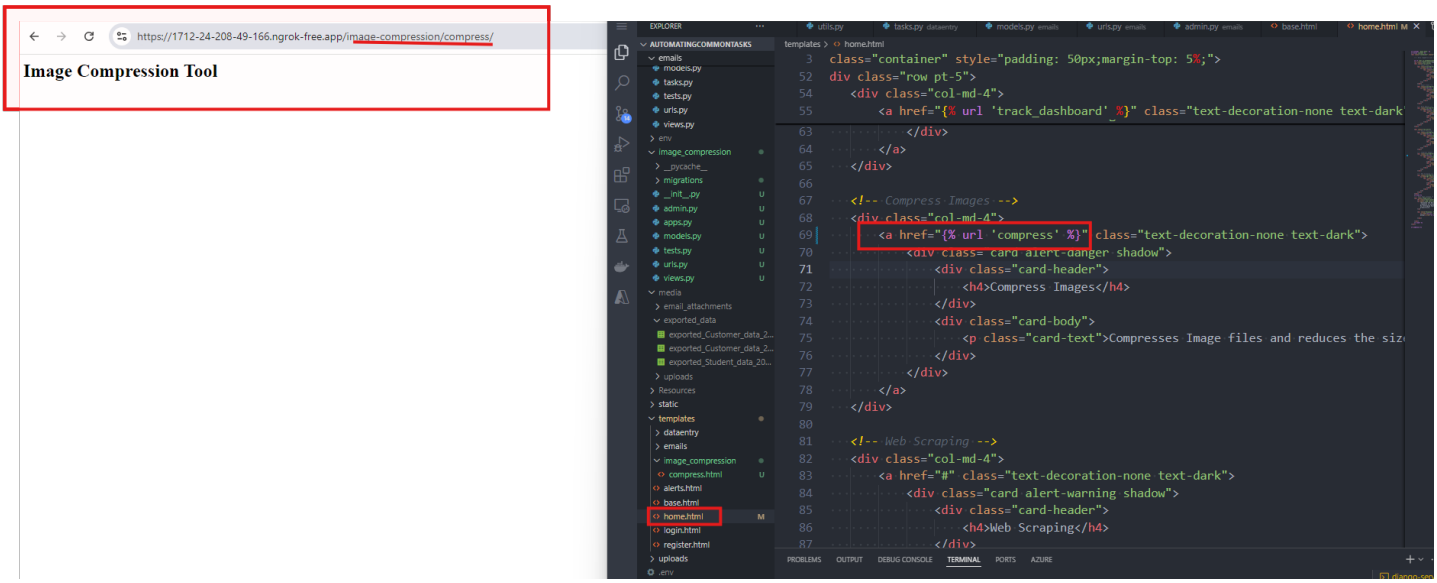
8. Create the function in the VIEWS.PY



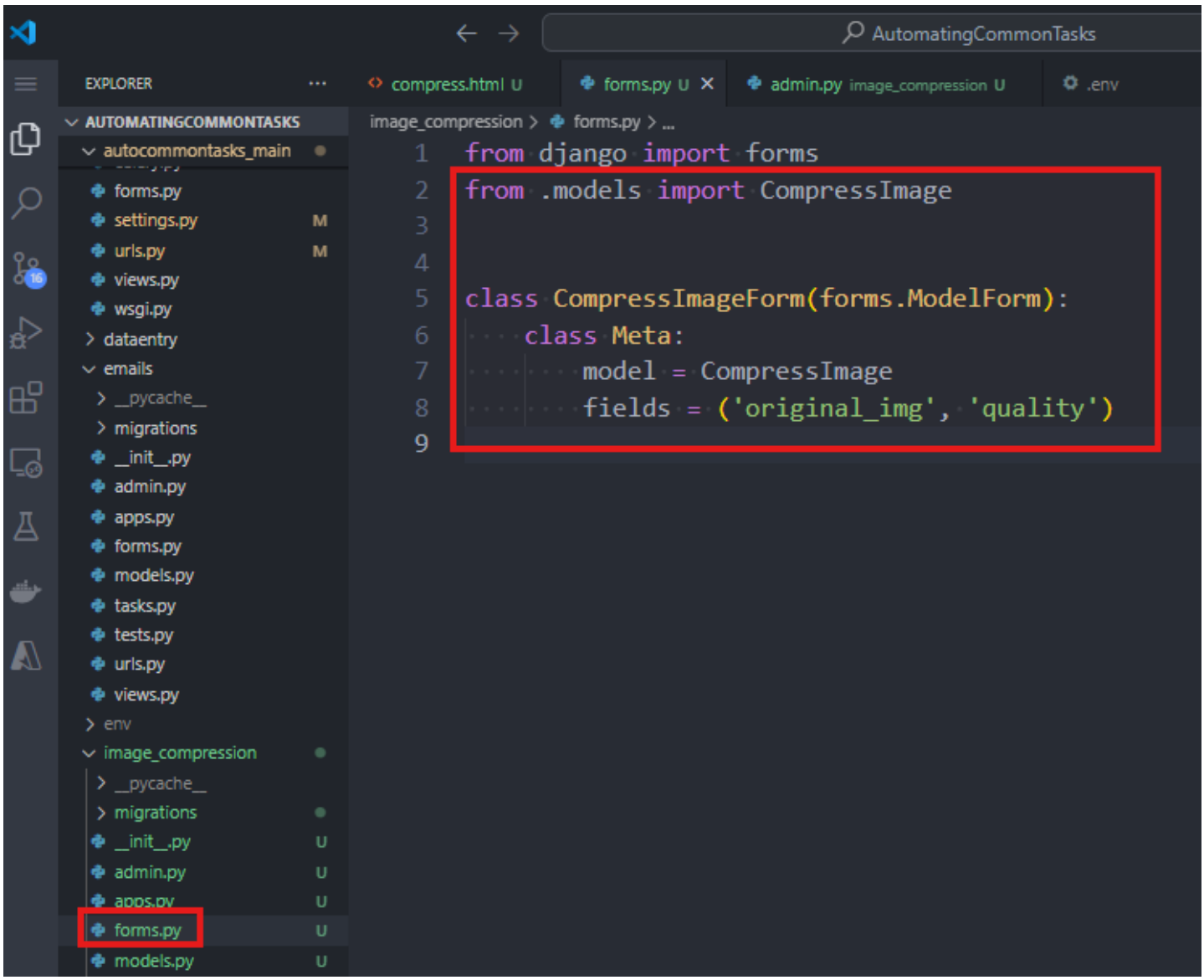
9. Create a new FOLDER called IMAGE_COMPRESSION, and in this folder, create a new file, COMPRESS.HTML



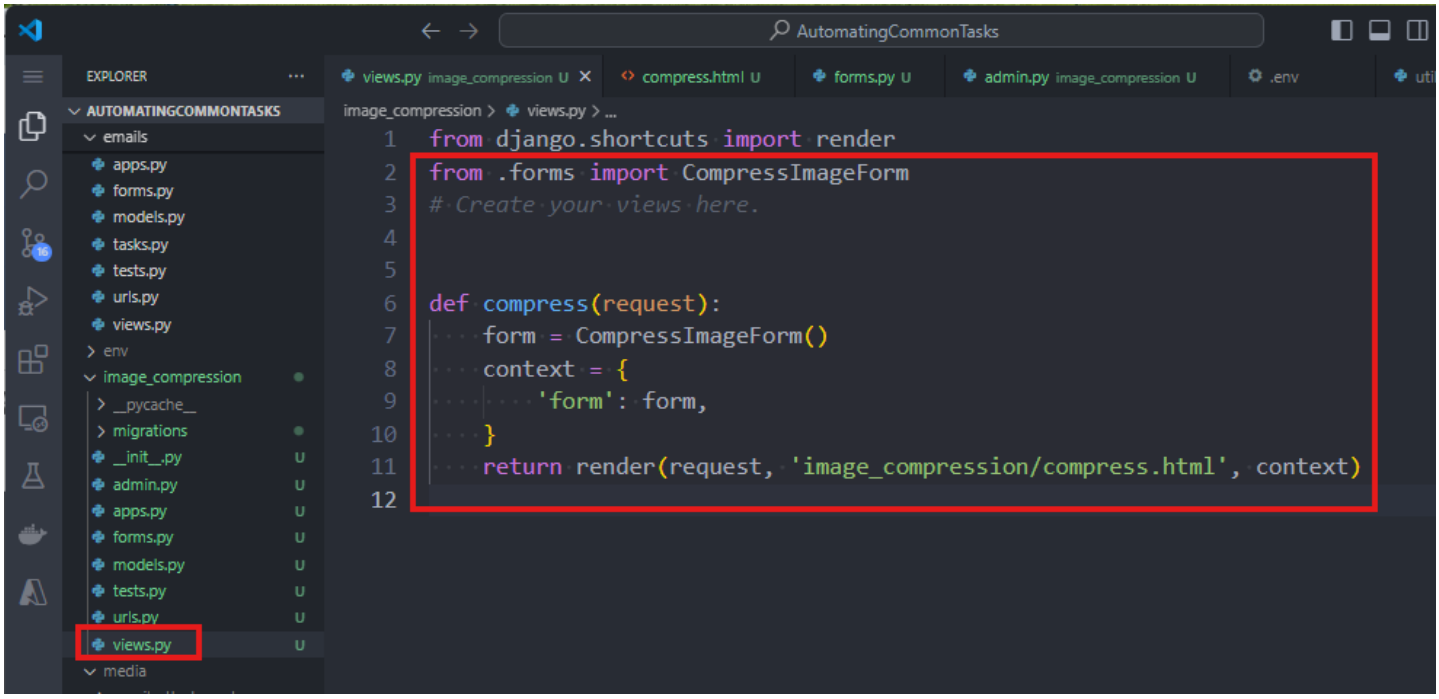
10. Update the HOME.HTML to call this new web page.



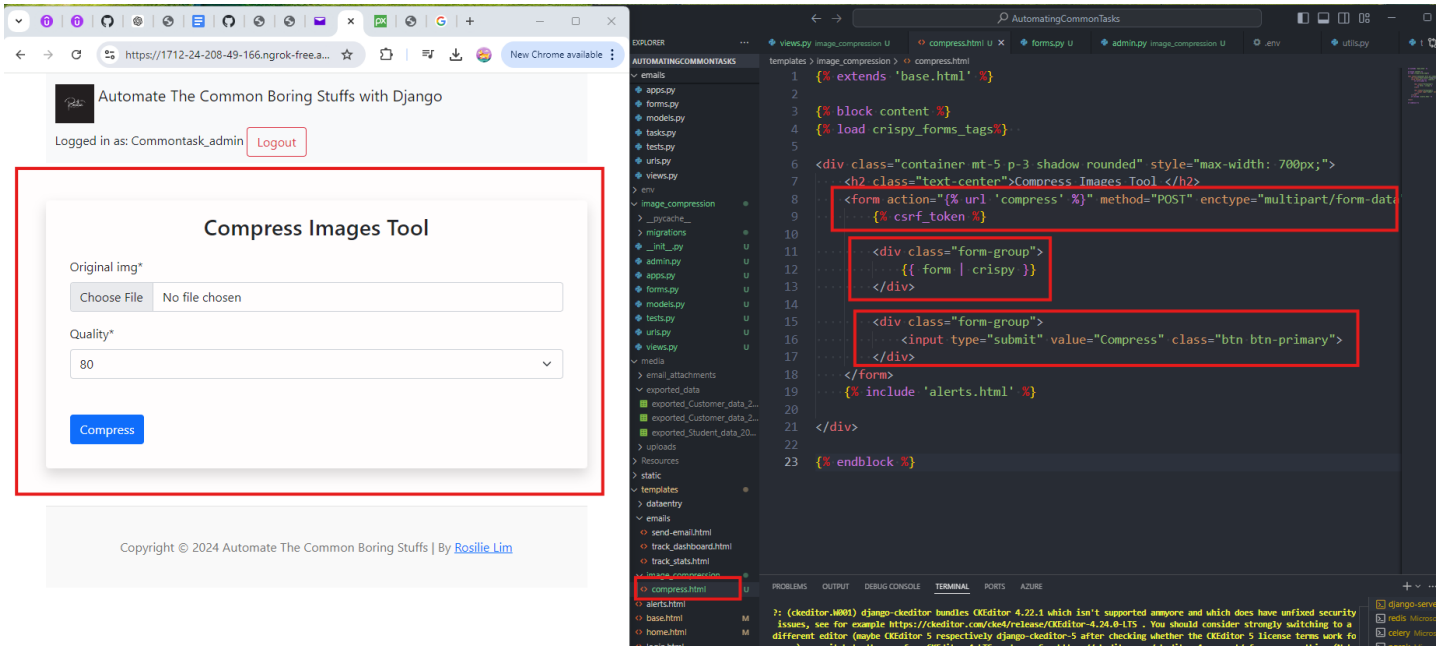
11. Create a FORMS.PY



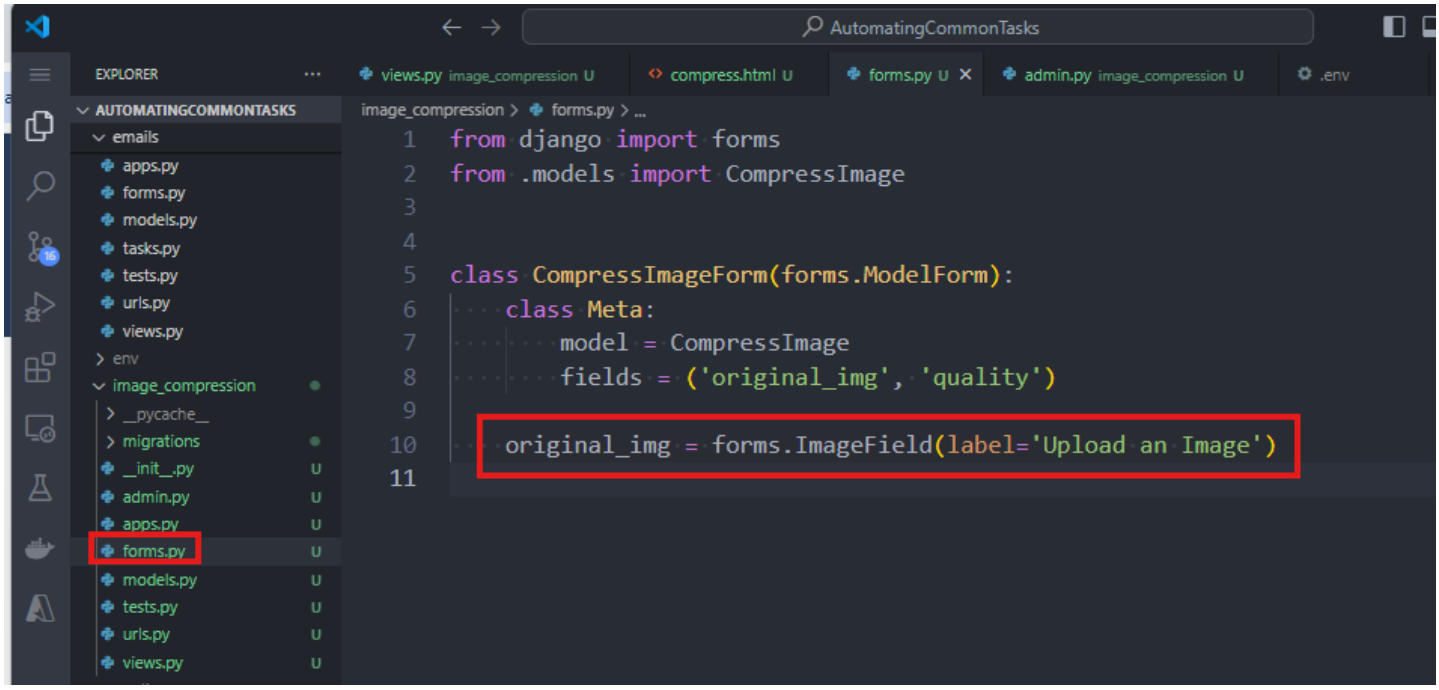
12. Call our form using our VIEWS.PY:



13. In the COMPRESS.HTML, update as:

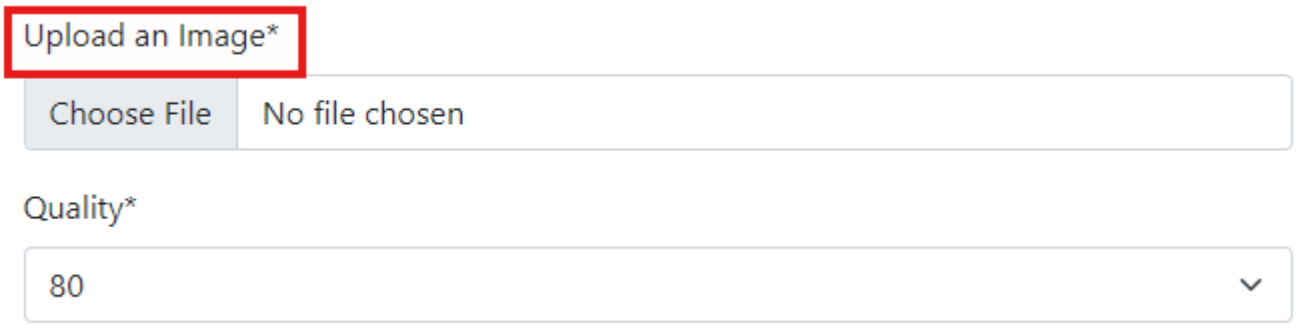


14. We can change the label of the field on the form, so update FORMS.PY and add the line:



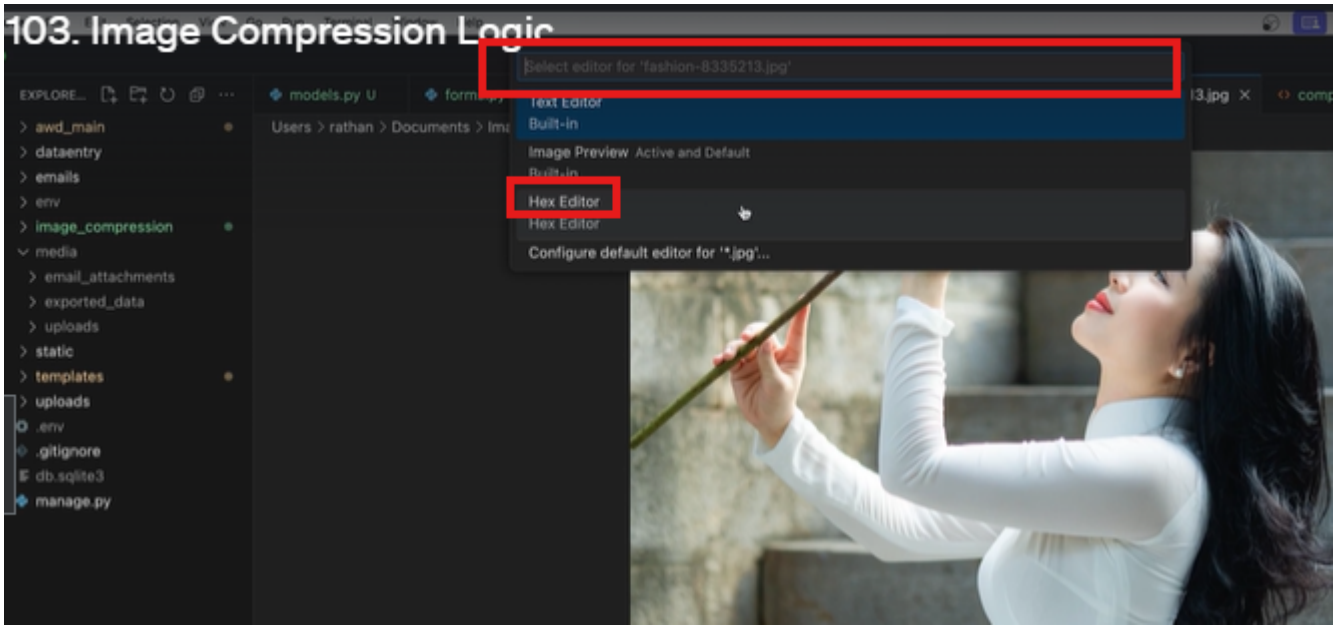
FROM:

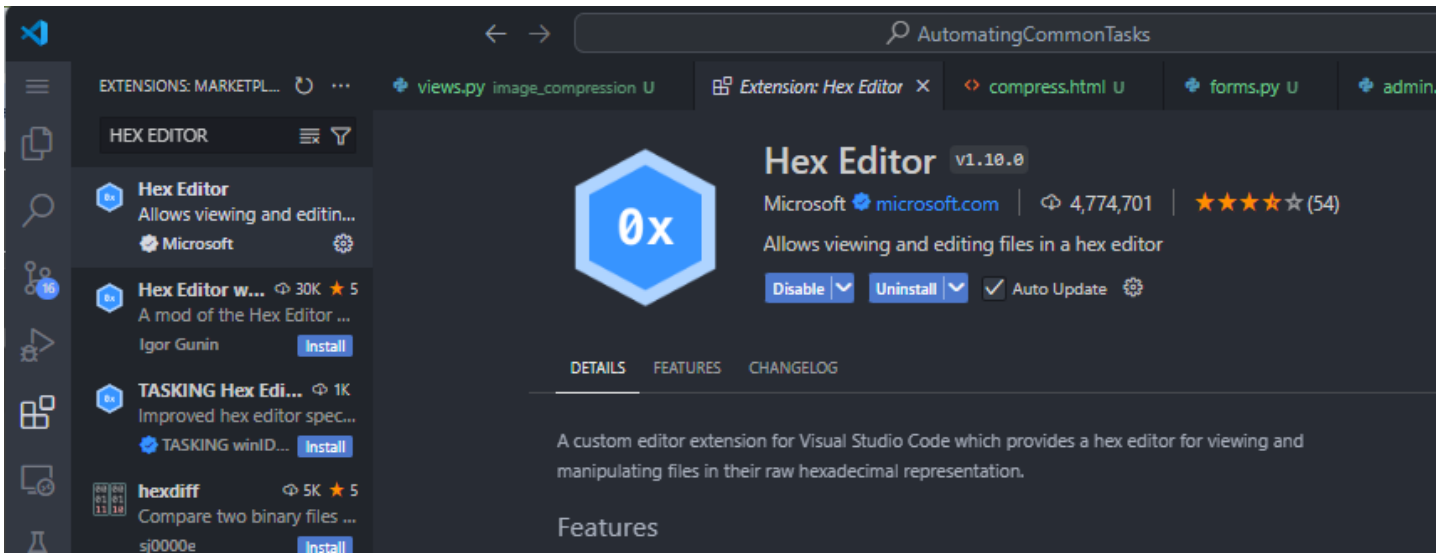
TO:

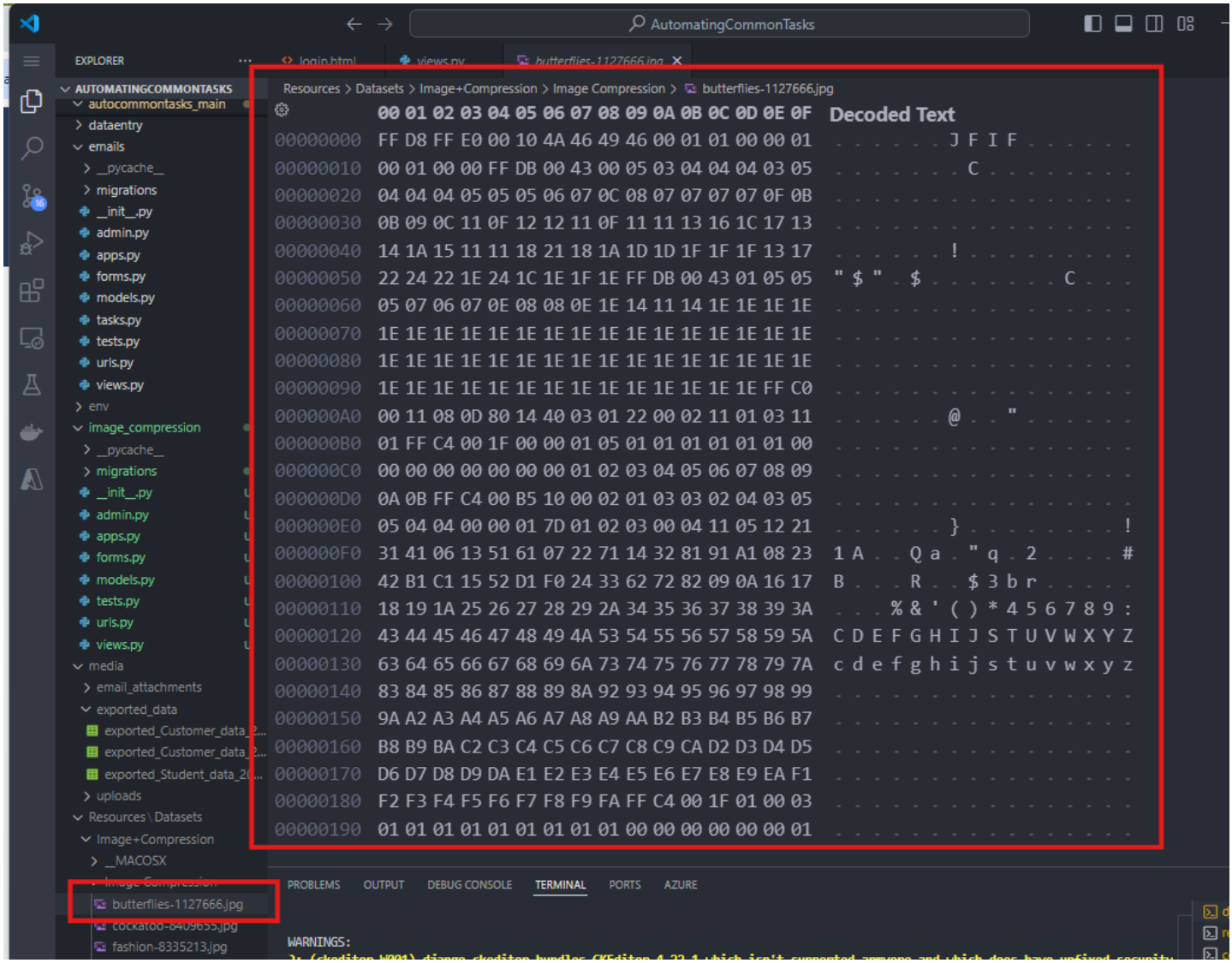


15. When we use io.BytesIO, we get bytes value of the image. To see what is the visual representation of these bytes, you can add the EXTENSION 'HEX EDITOR'.

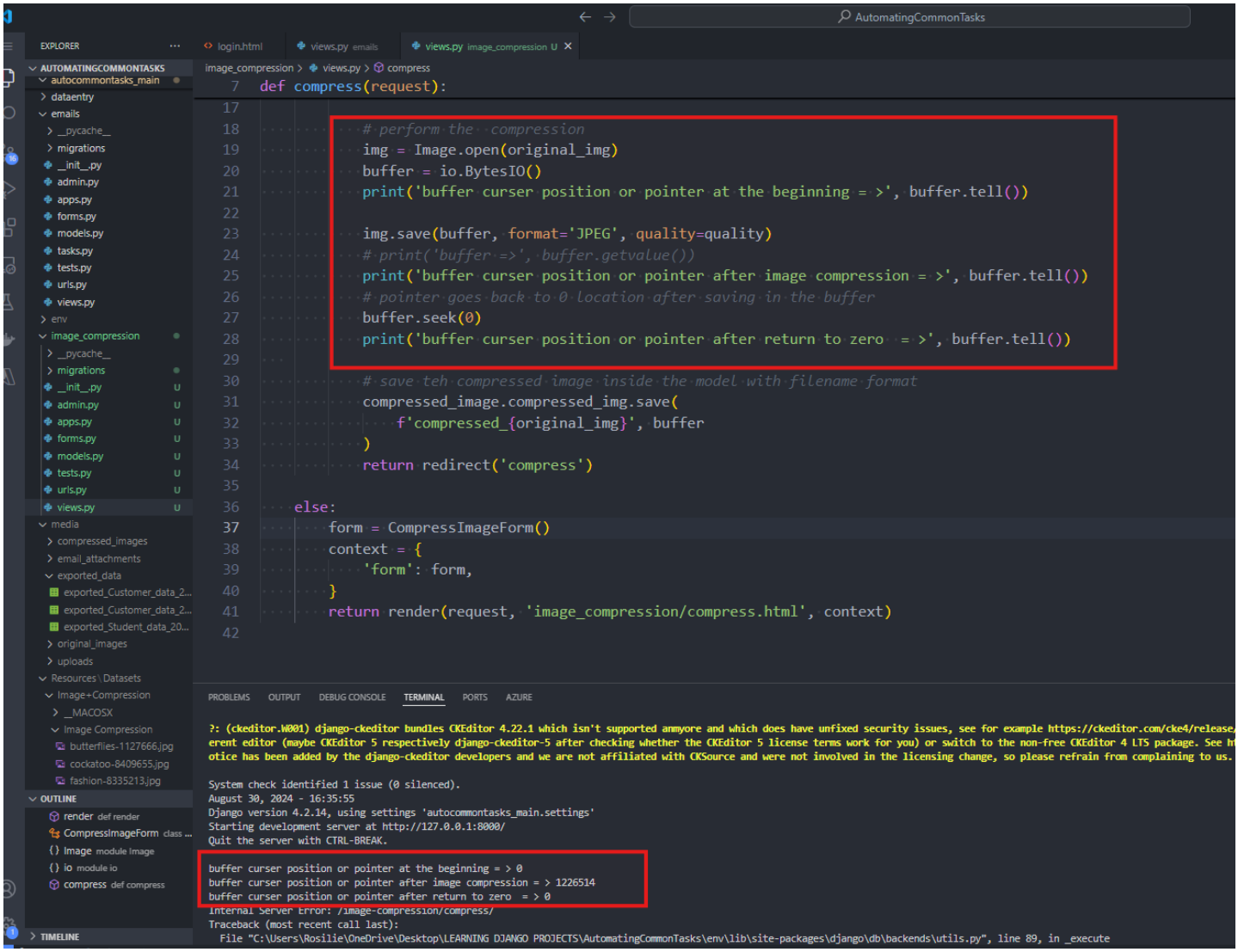
Now open an image file. On the tab of this image, right-click, select 'REOPEN EDITOR WITH', and select HEXEDITOR and you will see the HEX value of the file.


103. Image Compression Logic
Select editor for 'fashion-8335213.jpg'
Text Editor
Built-in
Image Preview Active and Default
Hex Editor
Hex Editor
Configure default editor for '*.jpg'...
EXPLORE...
models.py U
Users > rathan > Documents
awd_main
dataentry
emails
env
image_compression
media
email_attachments
exported_data
uploads
static
templates
uploads
.env
.gitignore
db.sqlite3
manage.py

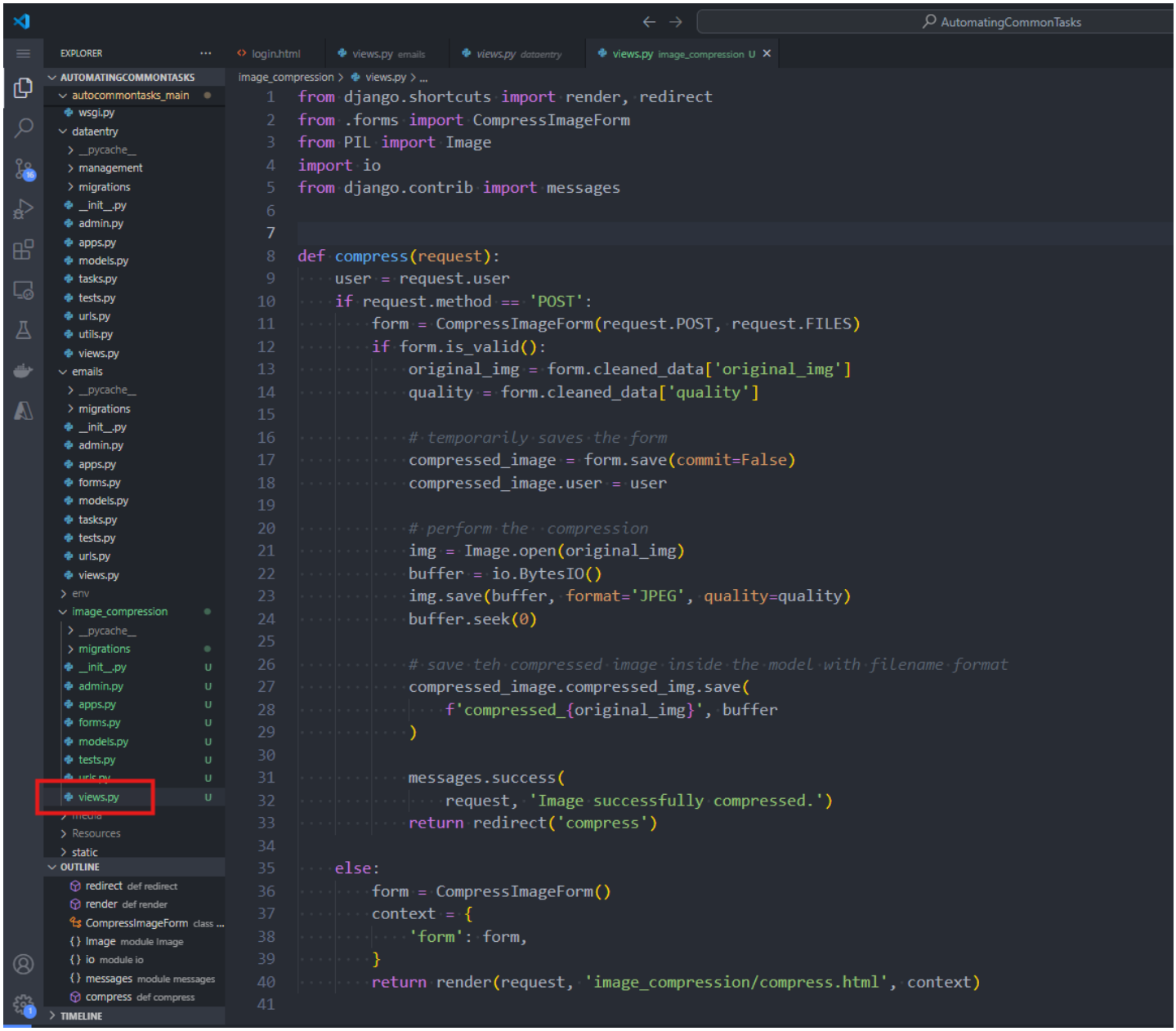

AutomatingCommonTasks
EXTENSIONS: MARKETPL...
HEX EDITOR
Hex Editor
Allows viewing and editin...
Microsoft
Hex Editor w... 30K 5
A mod of the Hex Editor ...
Igor Gunin
Install
TASKING Hex Edi... 1K
Improved hex editor spec...
TASKING winID...
Install
hexdiff 5K 5
Compare two binary files ...
sj0000e
Install
views.py image_compression U
Extension: Hex Editor
compress.html U
forms.py U
admin
Hex Editor v1.10.0
Microsoft microsoft.com
4,774,701
(54)
Allows viewing and editing files in a hex editor
Disable
Uninstall
Auto Update
DETAILS
FEATURES
CHANGELOG
A custom editor extension for Visual Studio Code which provides a hex editor for viewing and manipulating files in their raw hexadecimal representation.
Features



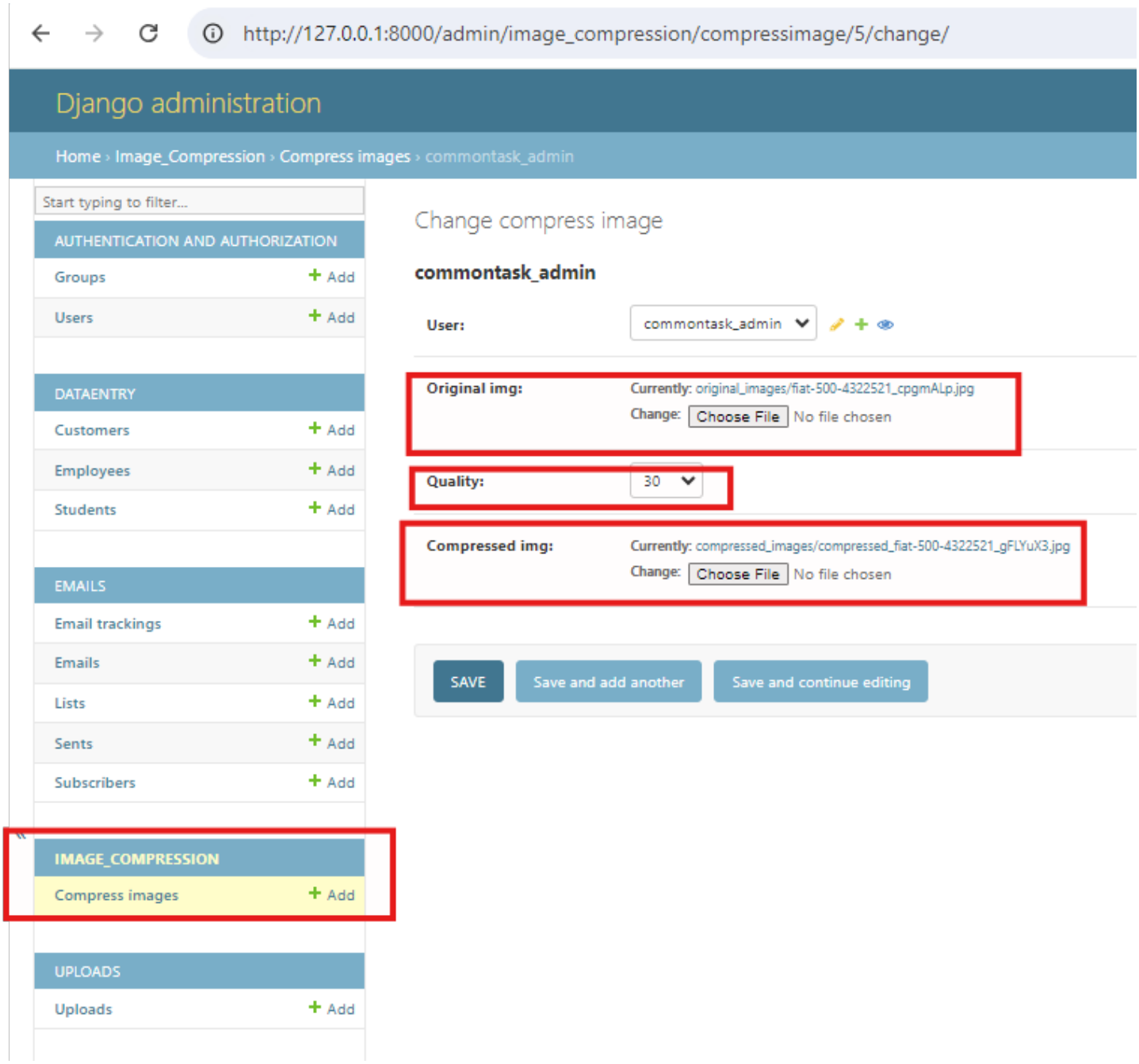
16. We use BUFFER.SEEK(0) to make sure that after we save, we set our curser position back to 0.



17. The VIEWS.PY shall be:



18. Checking our ADMIN panel



19. To set the image format to any format not just JPEG, we update our VIEWS.PY AS:

```
    # perform the compression
    img = Image.open(original_img)

    # set the image format based on the uploaded image's format
    output_format = img.format

    buffer = io.BytesIO()
    img.save(buffer, format=output_format, quality=quality)
    buffer.seek(0)

    # save teh compressed image inside the model with filename format
```