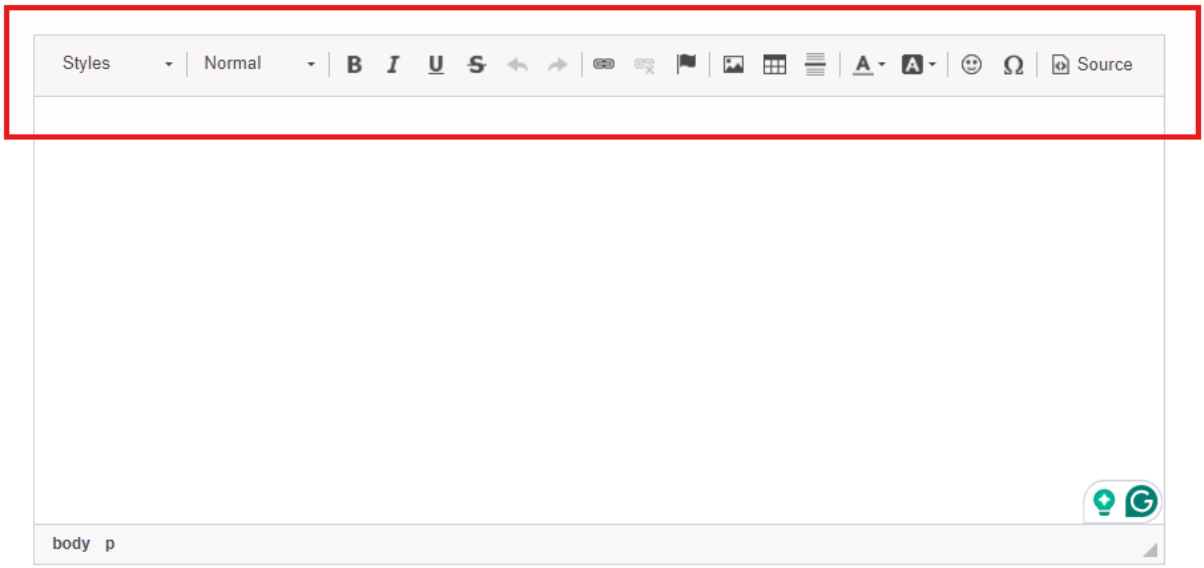
Topic: Bulk Email Tool 19: Integrating Rich Text Editor

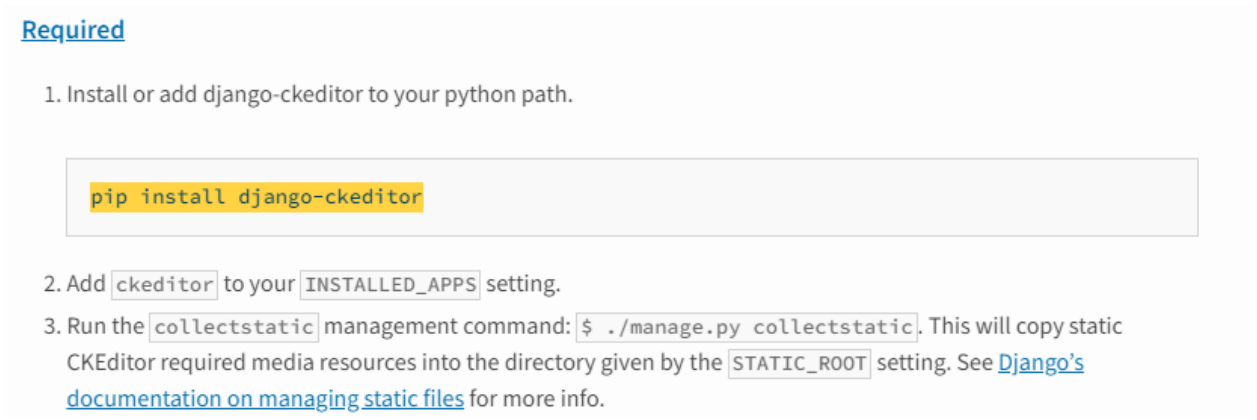
Speaker: Udemy Instructor Rathan Kumar | Notebook: Django: Automating Common Tasks

---

The email's body should have a formatting toolbar just like the image below and we can use CKEDITOR with this. This is the same tool we used in our Django Blog and this Digital Notebook website.



1. See this CKEditor documentation for the details of installation and configuration.



Required

1. Install or add django-ckeditor to your python path.

```
pip install django-ckeditor
```

2. Add `ckeditor` to your `INSTALLED_APPS` setting.
3. Run the `collectstatic` management command: `$ ./manage.py collectstatic`. This will copy static CKEditor required media resources into the directory given by the `STATIC_ROOT` setting. See Django's documentation on managing static files for more info.

Install CkEditor in your Django-bash terminal

$ pip install django-ckeditor

2. Add CkEditor in the INSTALLED APPS in SETTINGS.PY

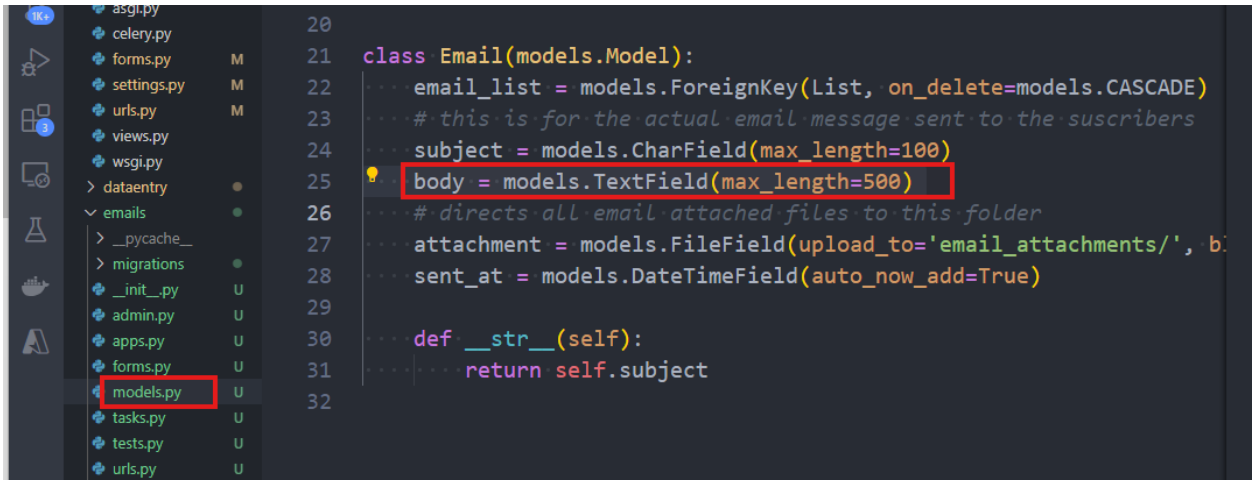
```
INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'dataentry',
    'uploads',
    'crispy_forms',
    'crispy_bootstrap5',
    'emails',
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

3. Run the COLLECTSTATIC COMMAND.

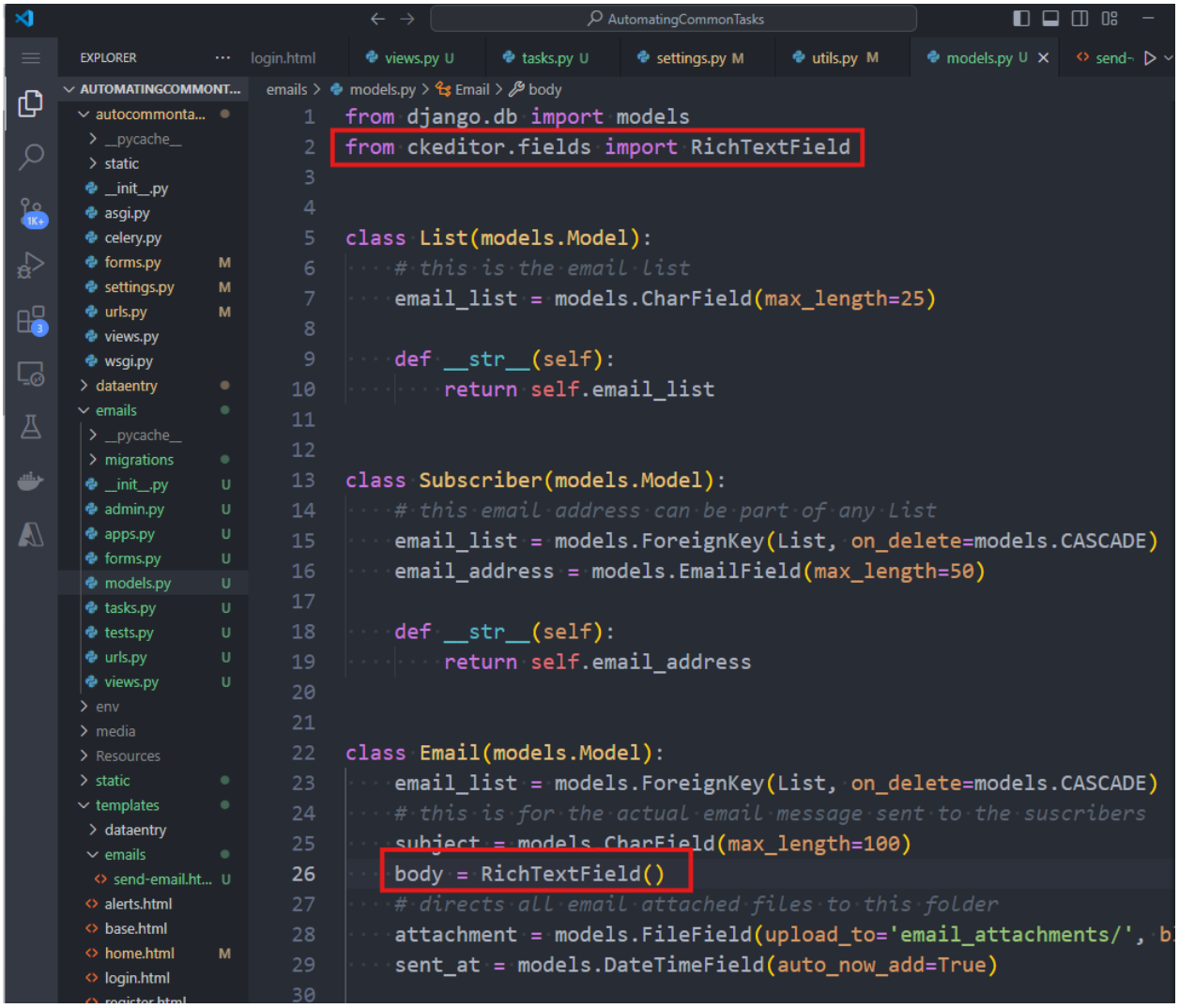
$ python manage.py collectstatic

4. Now update the EMAIL model in EMAILS\MODELS.PY, that will require Rick Text Editor:

FROM:



TO:

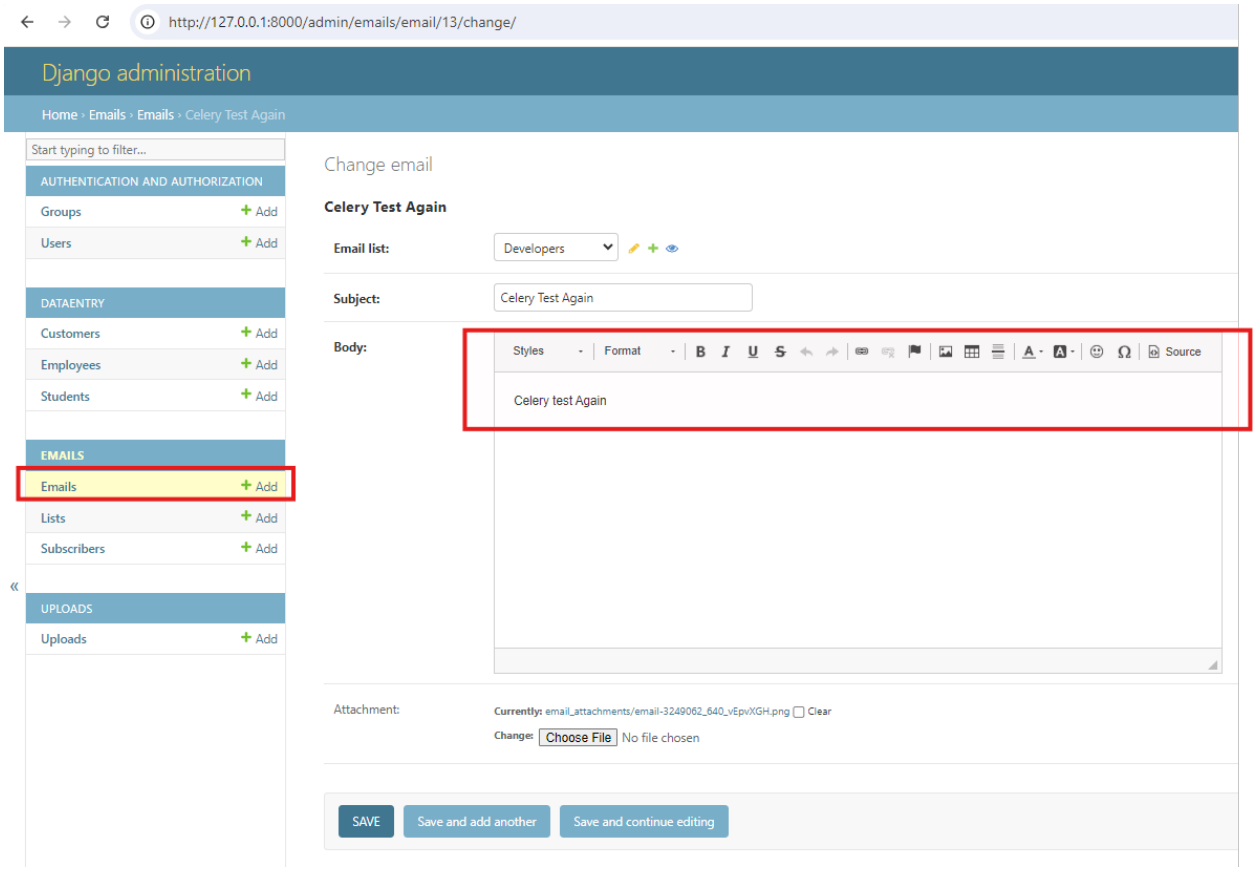


5. Since we have made some changes in our model, run the migration commands.

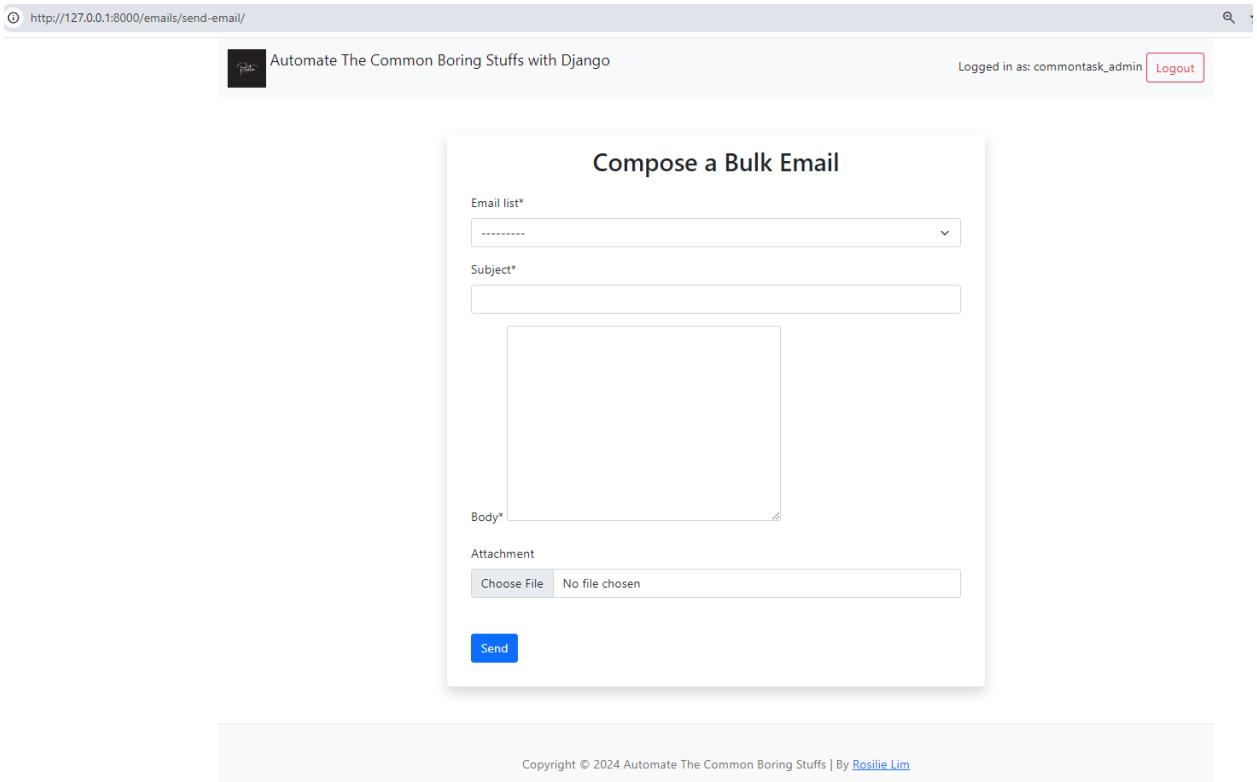
$ python manage.py makemigrations

$ python manage.py migrate

6. Run your Django server and see your admin dashboard Email model to see the effect of CKEditor.



7. Now we want the CKEditor to load as well on our front-end HTML page. When we load the SEND-EMAIL page, this is now the after-result:



8. In the CKEditor, copy the JAVASCRIPTS LINKS and save this in BASE.HTML

<script type="text/javascript" src="{% static "ckeditor/ckeditor-init.js" %}"></script>

<script type="text/javascript" src="{% static "ckeditor/ckeditor/ckeditor.js" %}"></script>

Outside of django admin

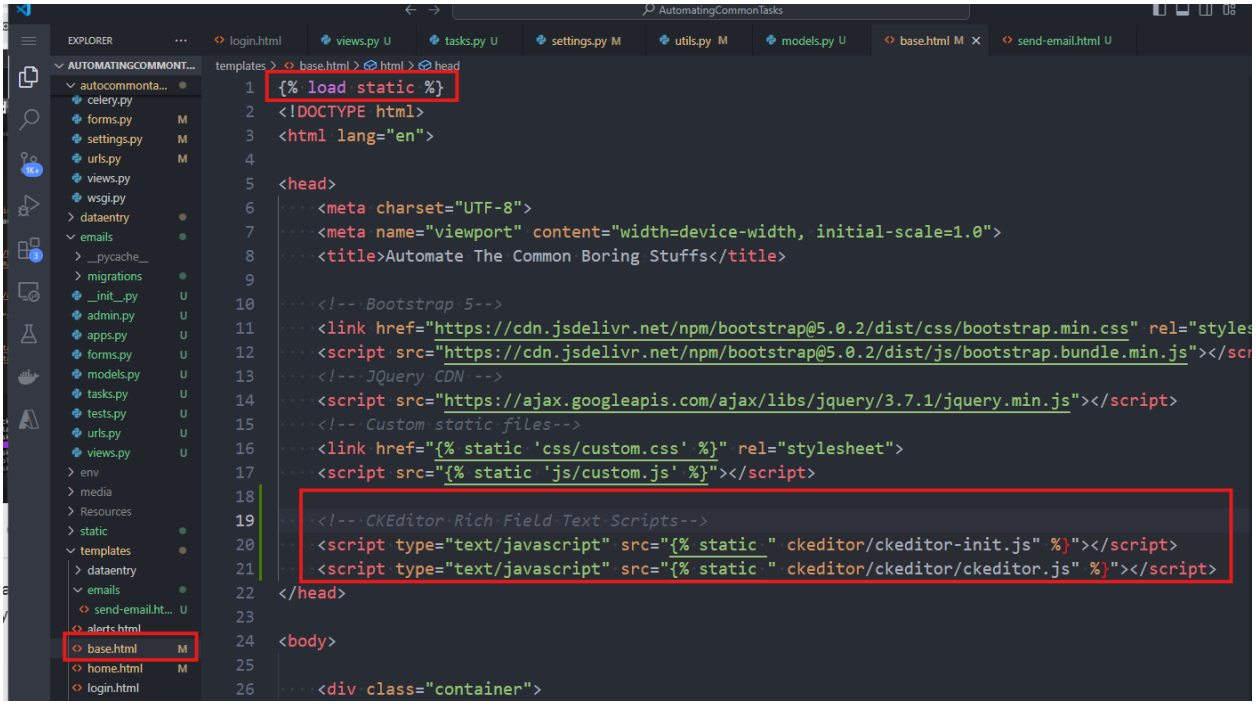
When you are rendering a form outside the admin panel, you'll have to make sure all form media is present for the editor to work. One way to achieve this is like this:

```
<form>
    {{ myform.media }}
    {{ myform.as_p }}
    <input type="submit"/>
</form>
```

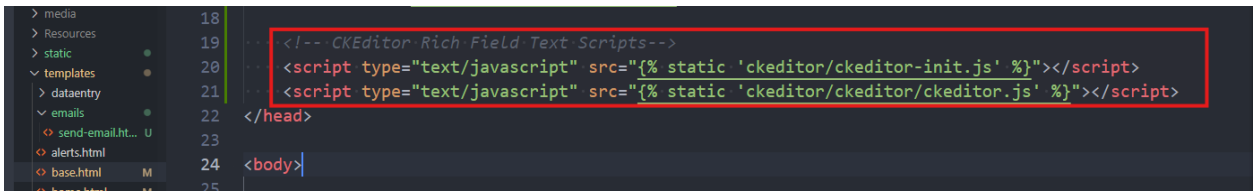
or you can load the media manually as it is done in the demo app:

```
{% load static %}
<script type="text/javascript" src="{% static "ckeditor/ckeditor-init.js" %}"></script>
<script type="text/javascript" src="{% static "ckeditor/ckeditor/ckeditor.js" %}"></script>
```

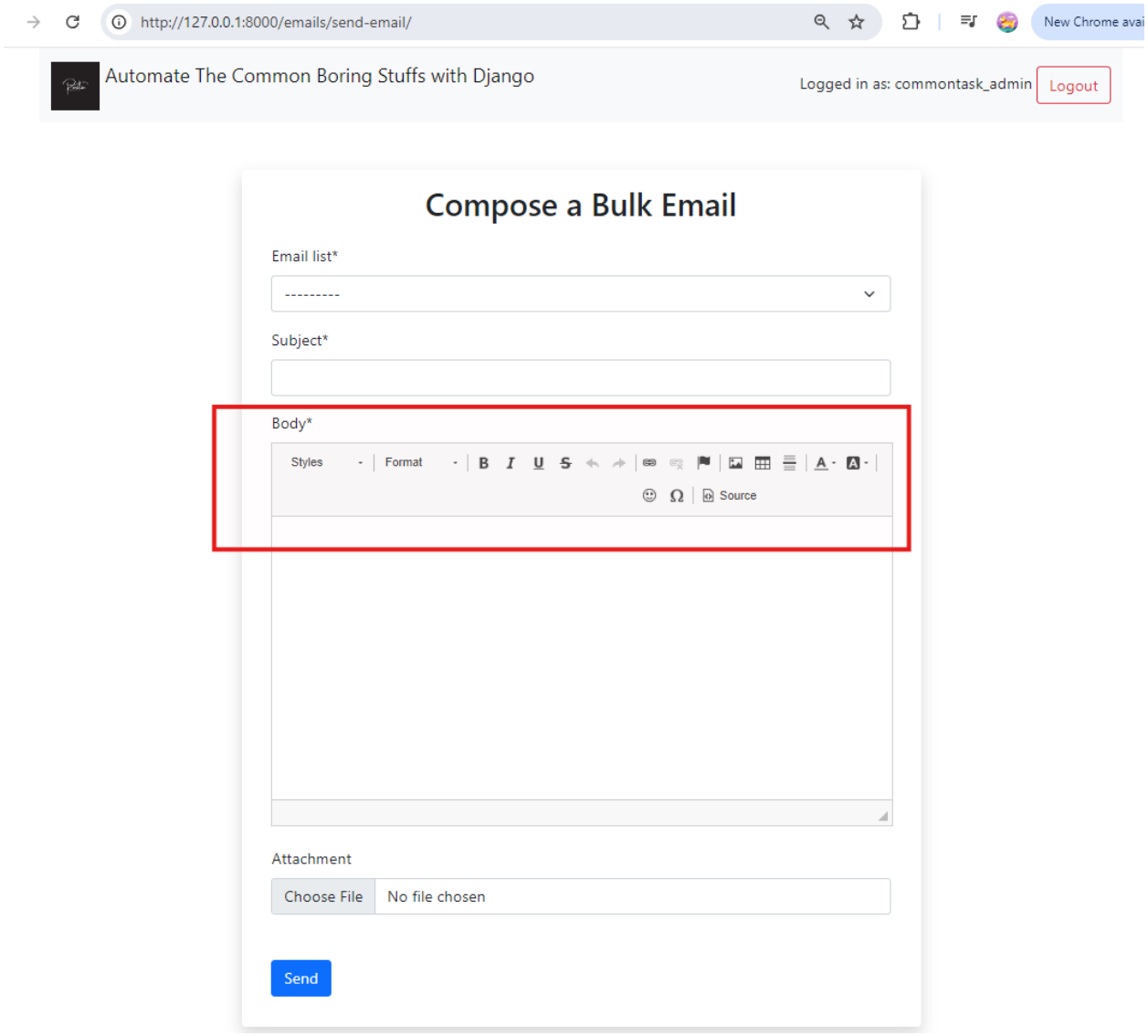
When you need to render `RichTextField`'s HTML output in your templates safely, just use `{{ content|safe }}`, Django's safe filter



The CKEditor scripts are showing red-highlighted symbols or words. This is because of the double " " used in the line. So modify it as:



9. Now refresh our SEND-EMAIL page and the Rich Field Toolbar should show up:

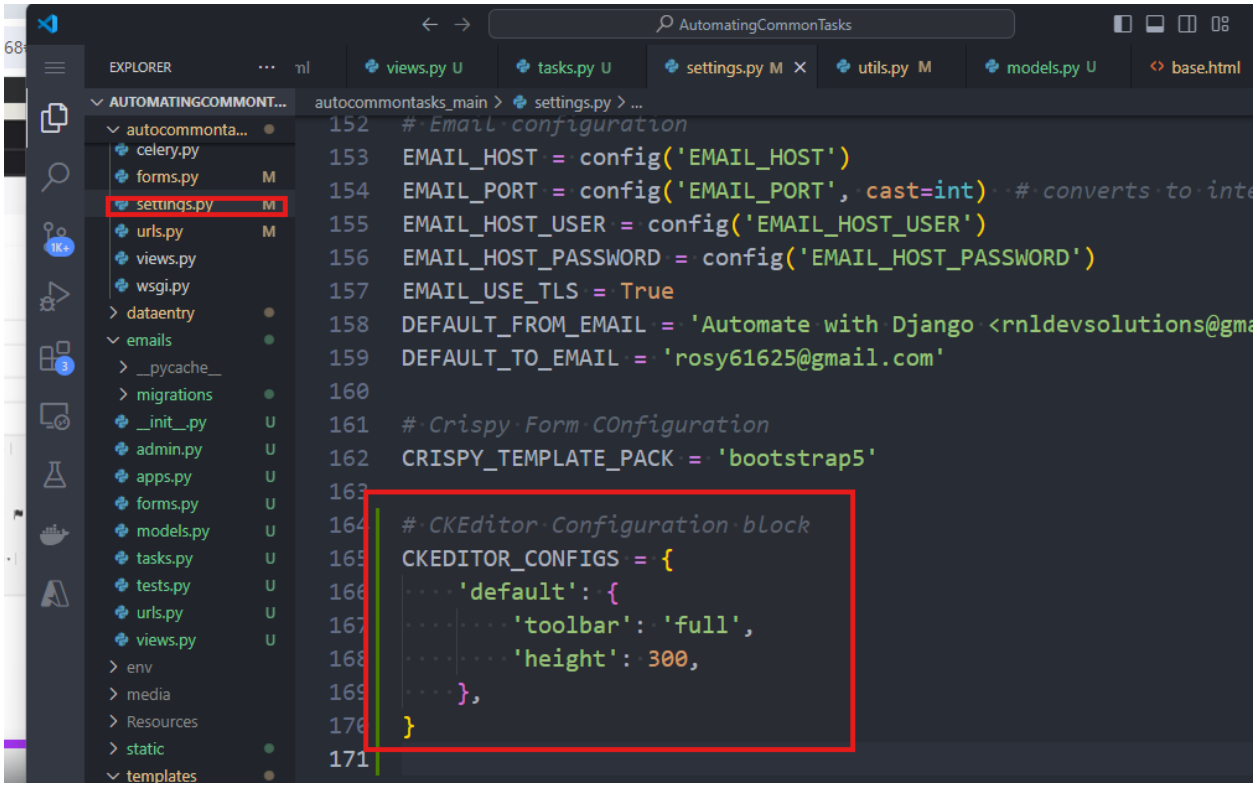


10. The CKEditor has a default size window which can be resizeable but if you want a different height, we can configure it as:

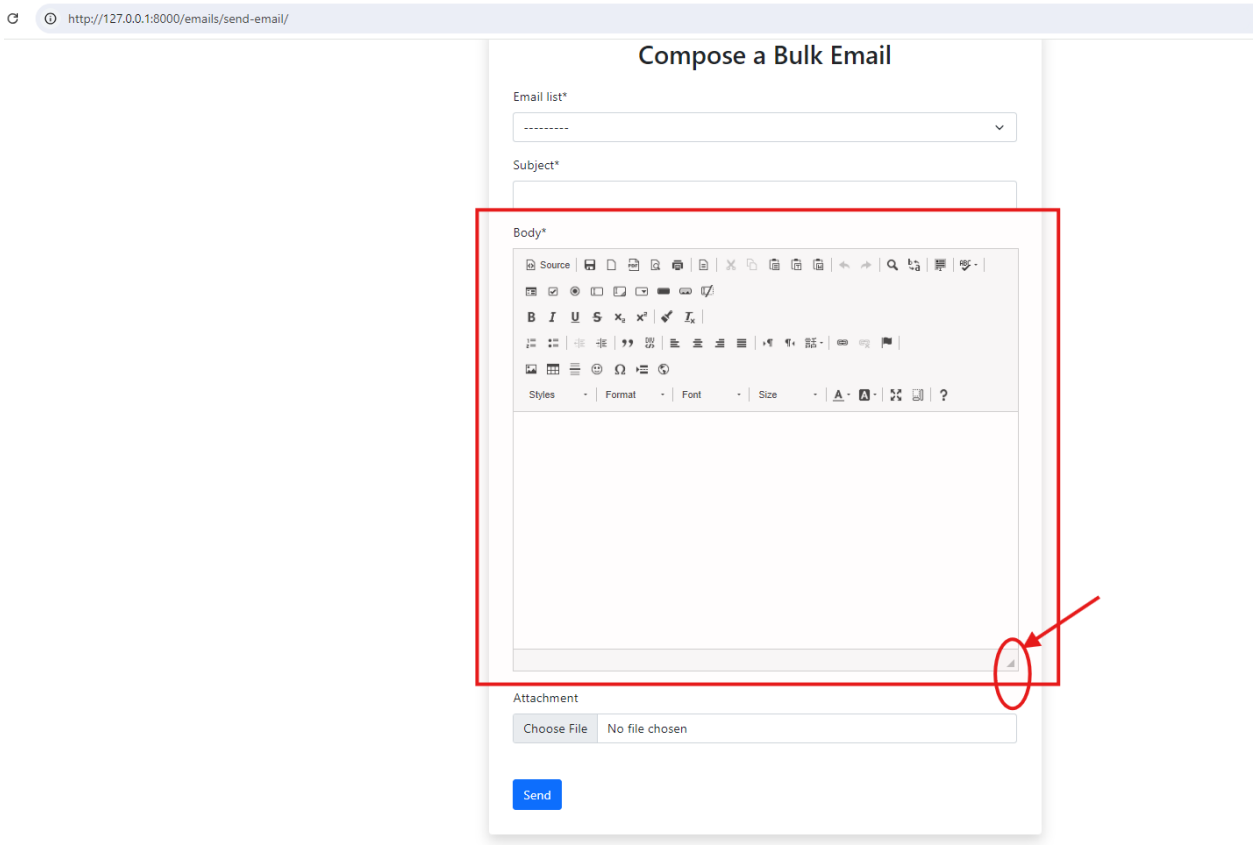
By specifying a set named `default` you'll be applying its settings to all RichTextField and CKEditorWidget objects for which `config_name` has not been explicitly defined

```
CKEDITOR_CONFIGS = {
    'default': {
        'toolbar': 'full',
        'height': 300,
        'width': 300,
    },
}
```

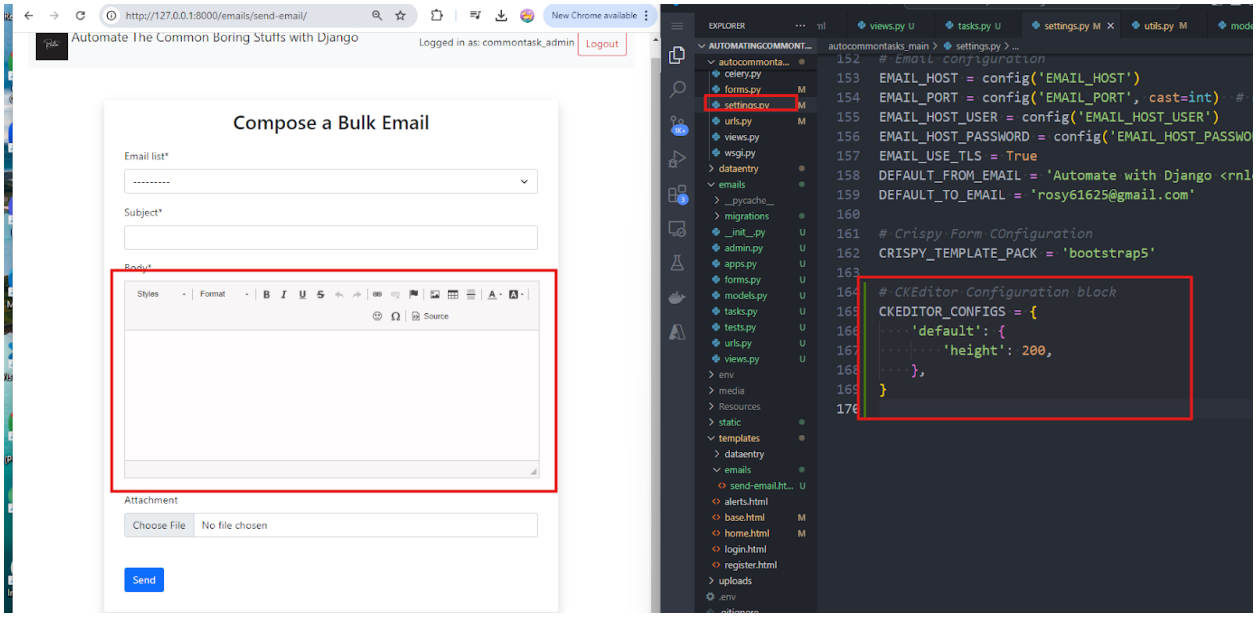
So, go to SETTINGS.PY and add the above configuration block.



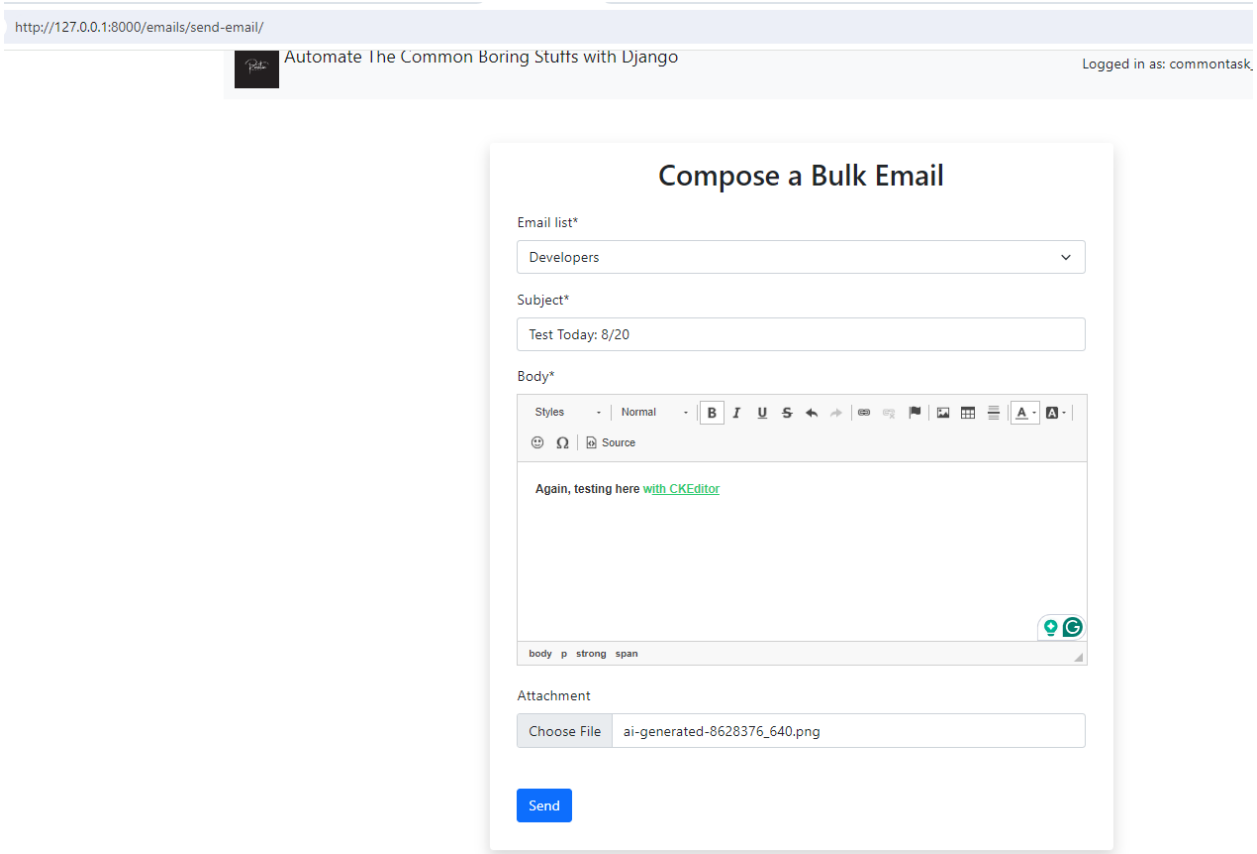
Reload the page and you can see the complete tools with a height. Take note that we can resize the window using the bottom right corner.



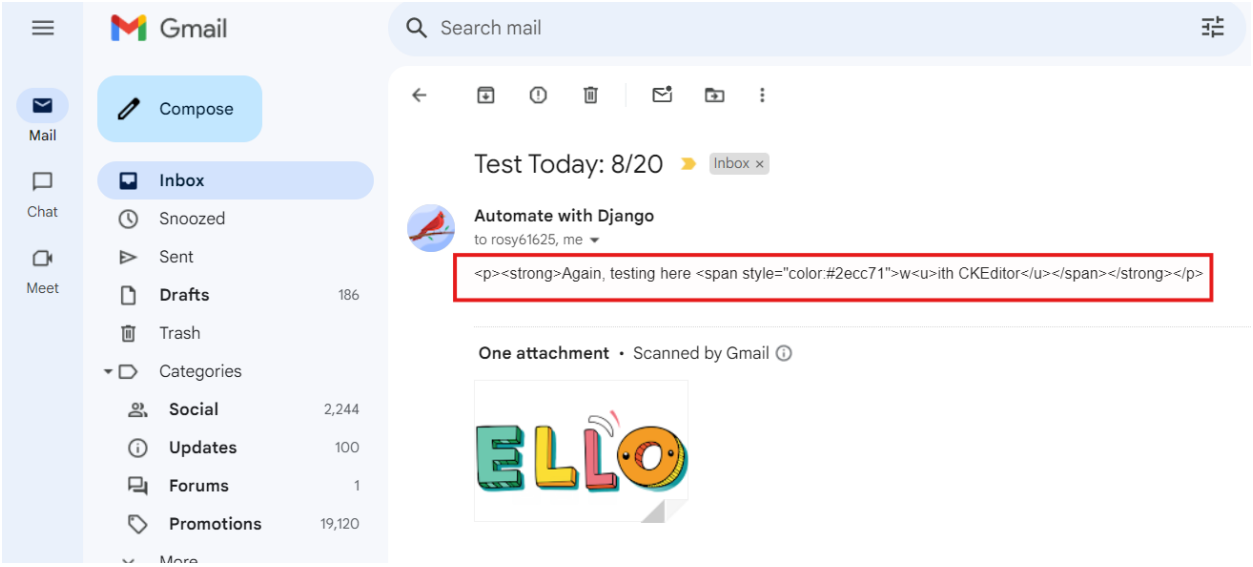
11. Showing a complete toolbar would be too much, so we simplify it by using the default and with reduced height.



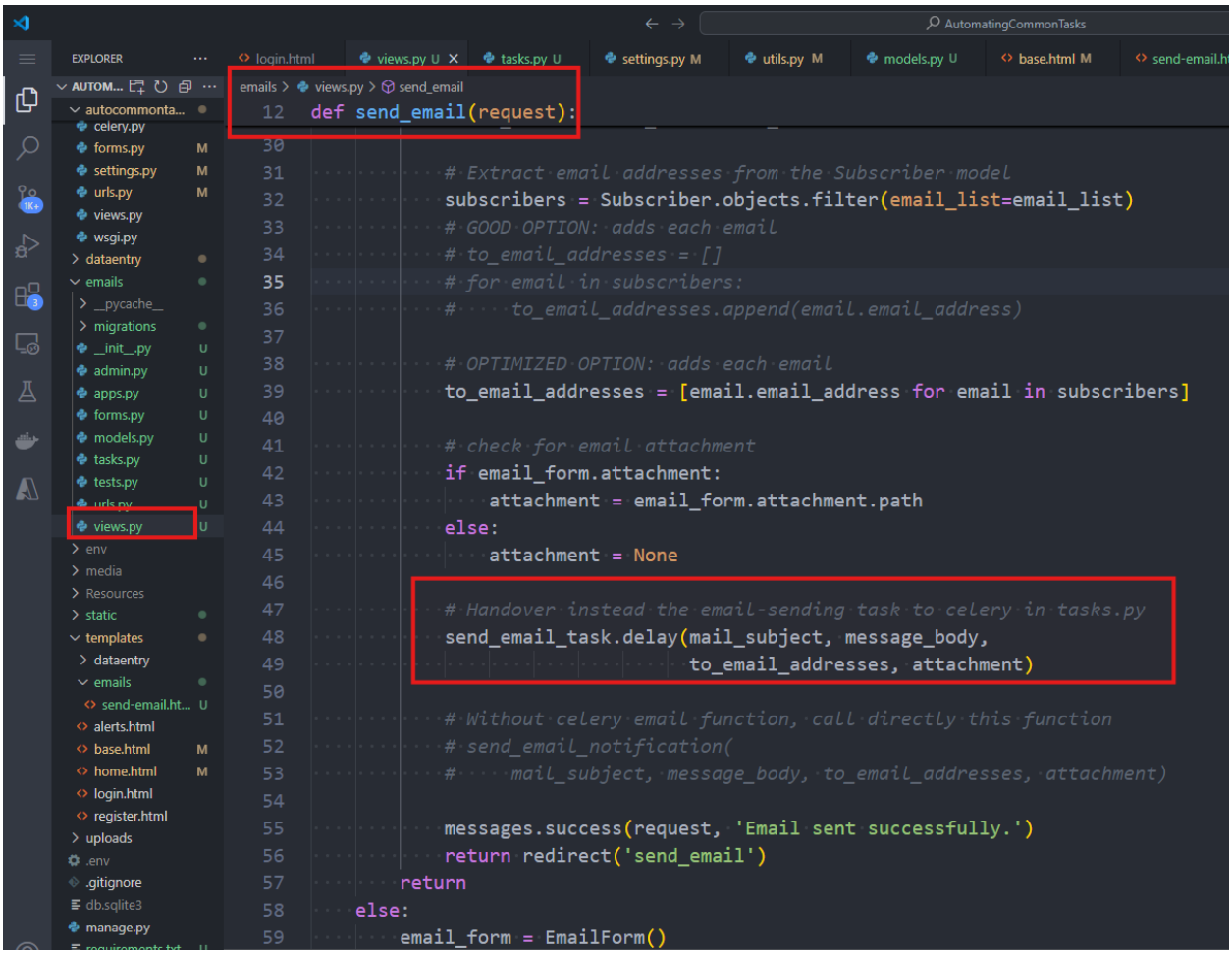
12. However, when we send the emails, we will be receiving it with HTML tags.



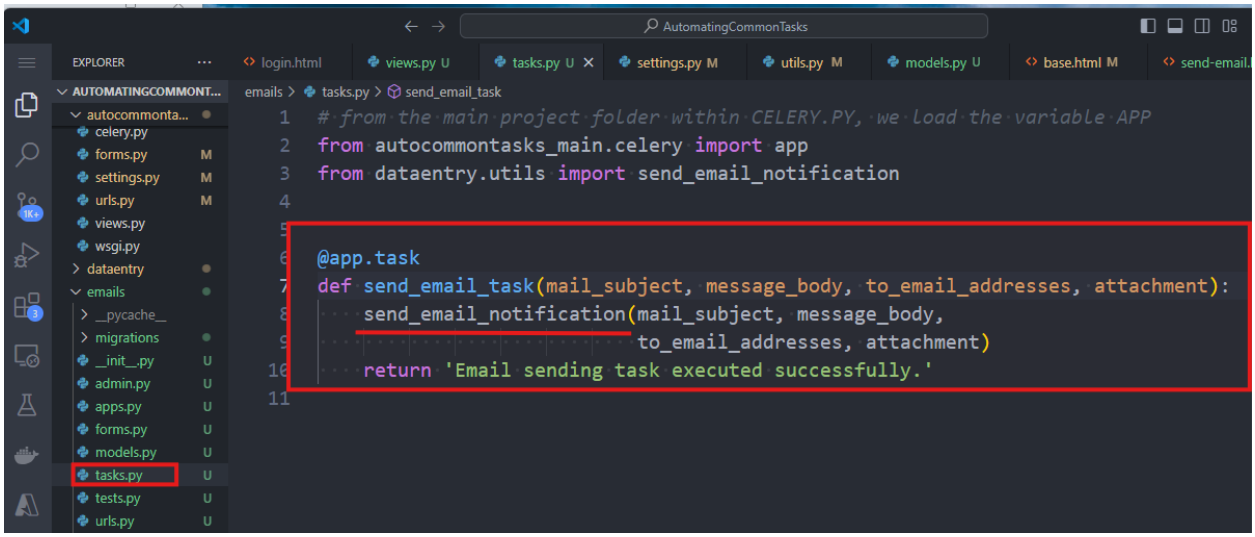
Checking our email:



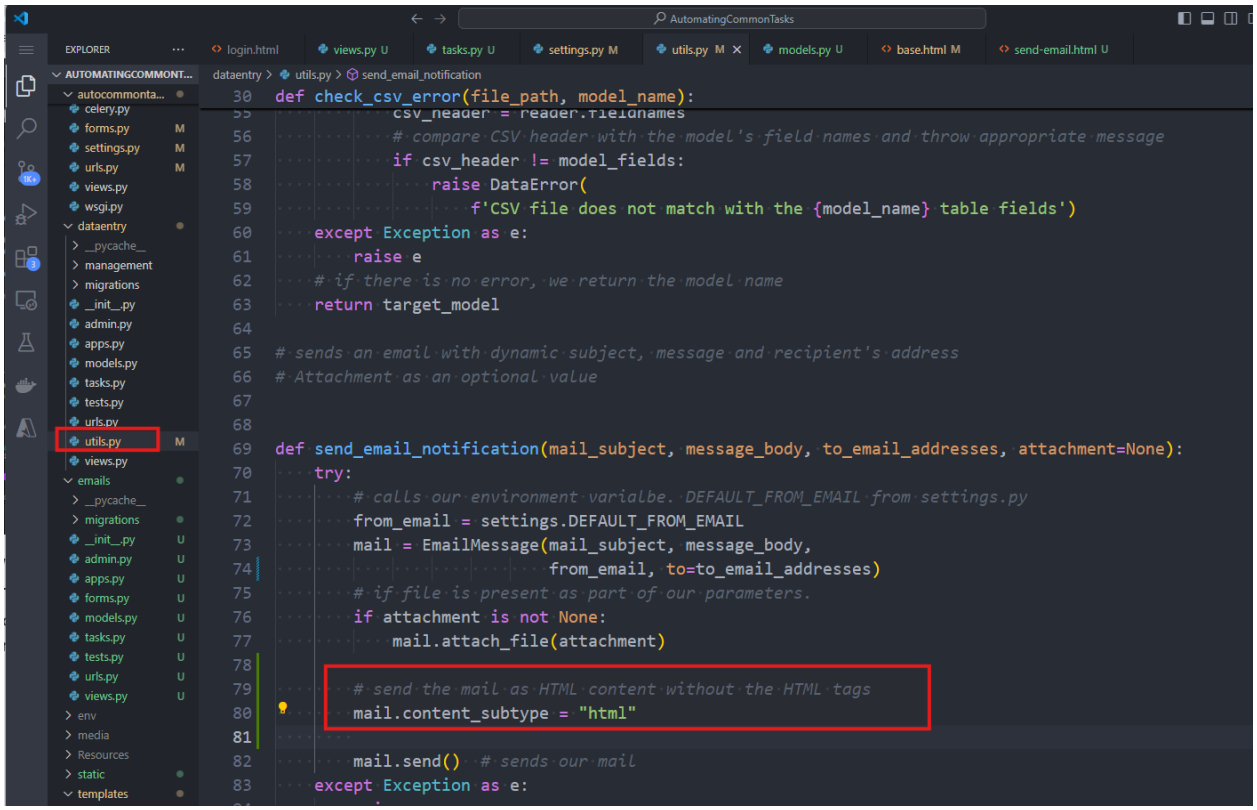
13.Go to your SEND_EMAIL function in EMAILS\VIEWS.PY and press CTRL + click the link on the SEND_EMAIL_TASK to bring you to the function.



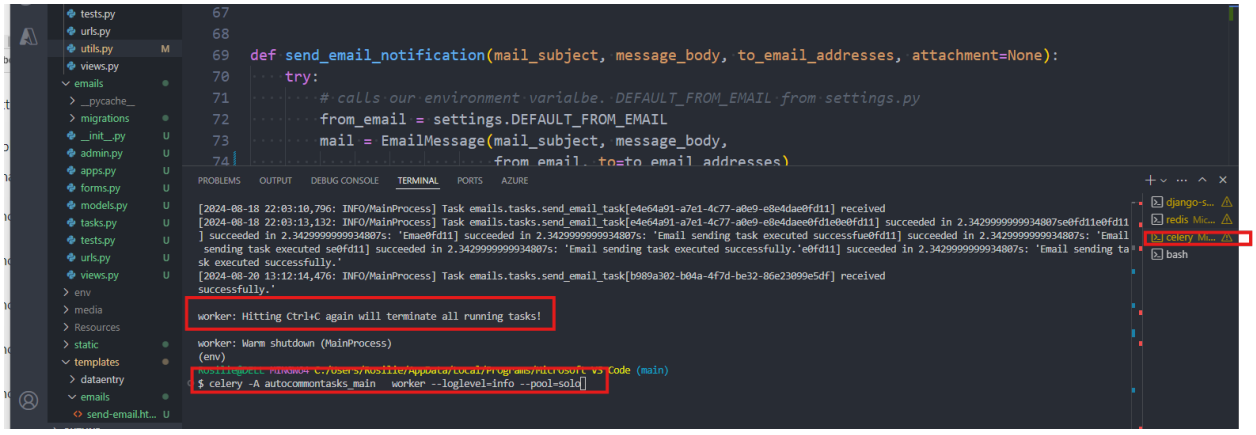
In EMAILS\TASKS.PY, press CTROL + LEFT CLICK on our SEND_EMAIL_NOTIFICATION function to bring us to another function:



In DATAENTRY\UTILS.PY, we update our function to show that the email body is sent as HTML without the HTML tags.

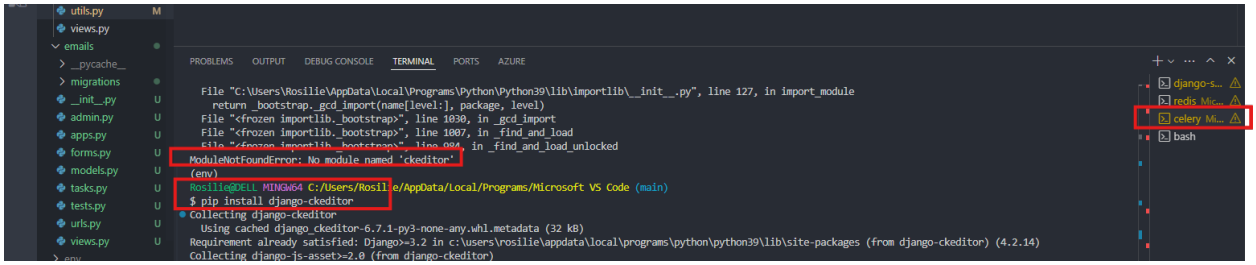


14. We have to reload our CELERY bash terminal code since we have made some changes in our Celery function. Press CTRL + C to stop it from running:

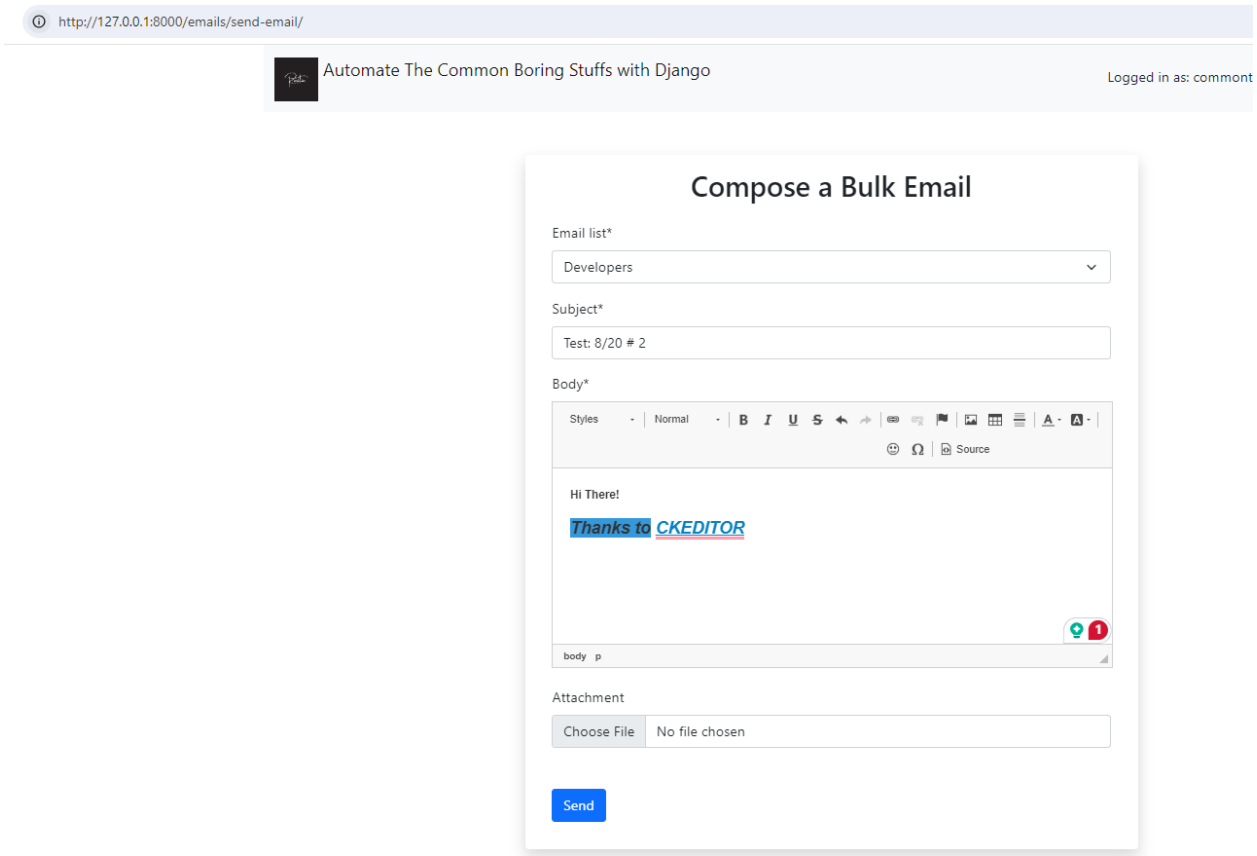


We encountered an error: CKEditor MODULE NOT FOUND Error, so we installed it as well:

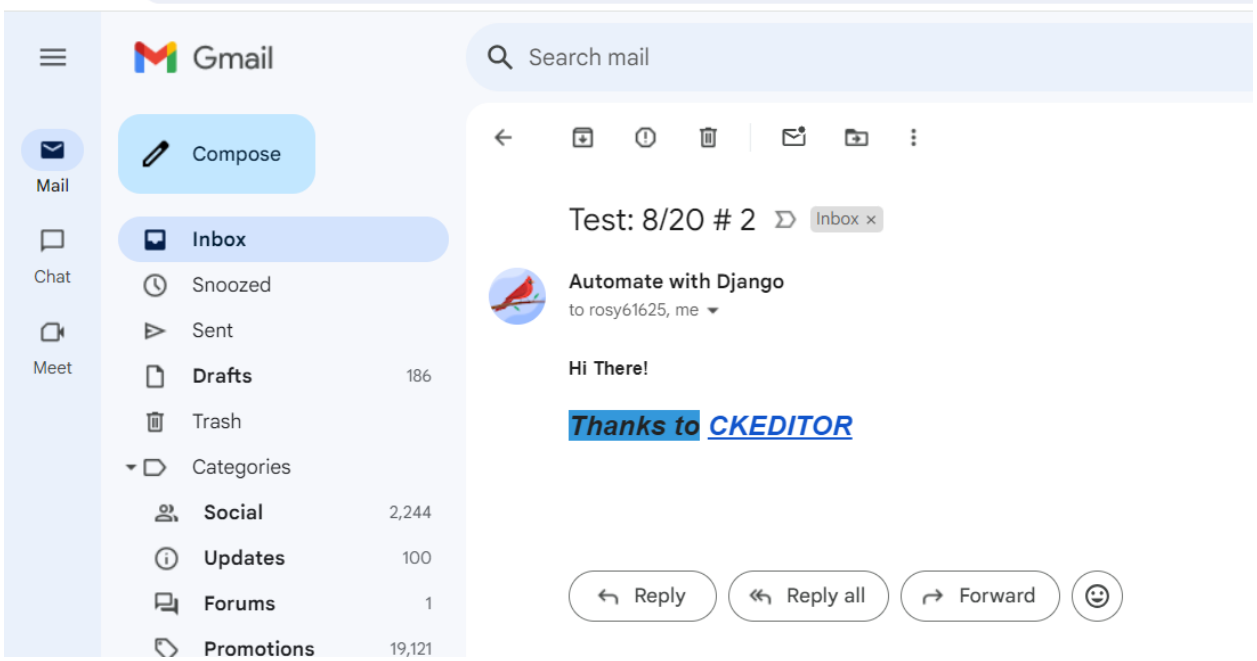
$ pip install django-ckeditor



15. Sending our email again,



In our email:



16. Push our changes in Github.