





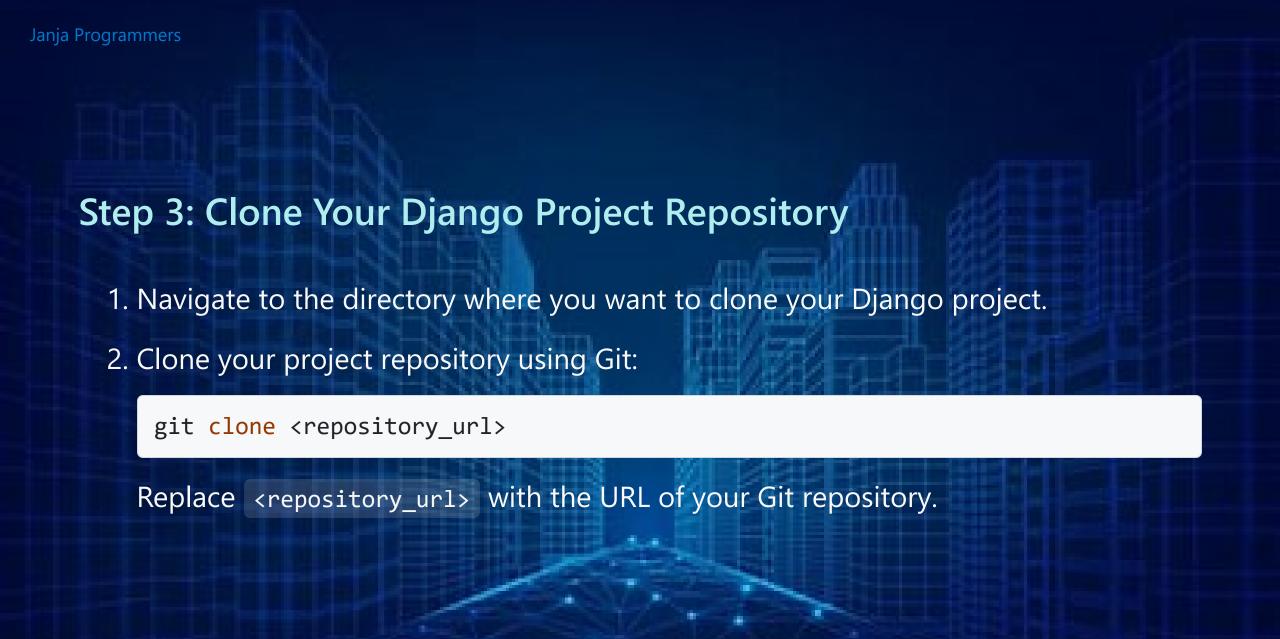


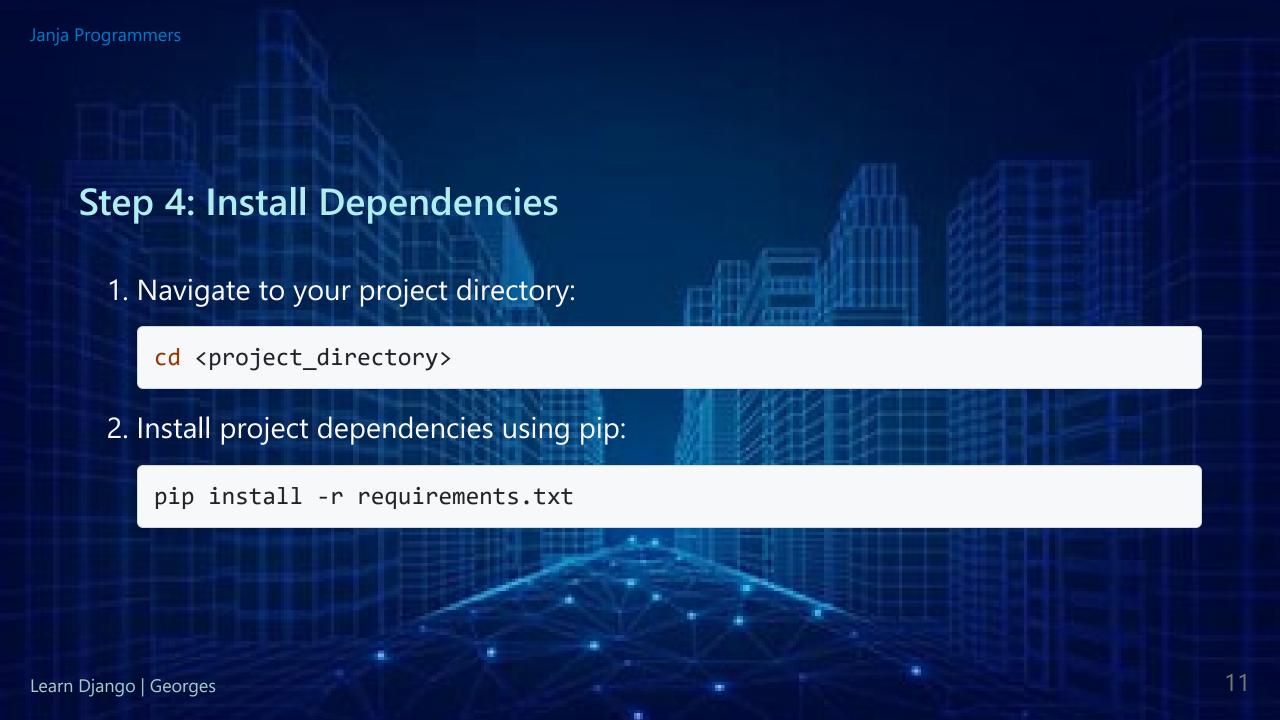
Step 2: Set Up a Virtual Environment

- 1. Once logged in to your PythonAnywhere account, navigate to the Dashboard.
- 2. Click on the "Consoles" tab and open a Bash console.
- 3. Create a virtual environment for your Django project:

mkvirtualenv --python=/usr/bin/python3.8 myenv

Replace myenv with the name you prefer for your virtual environment.





Step 5: Configure Static Files and Media Settings

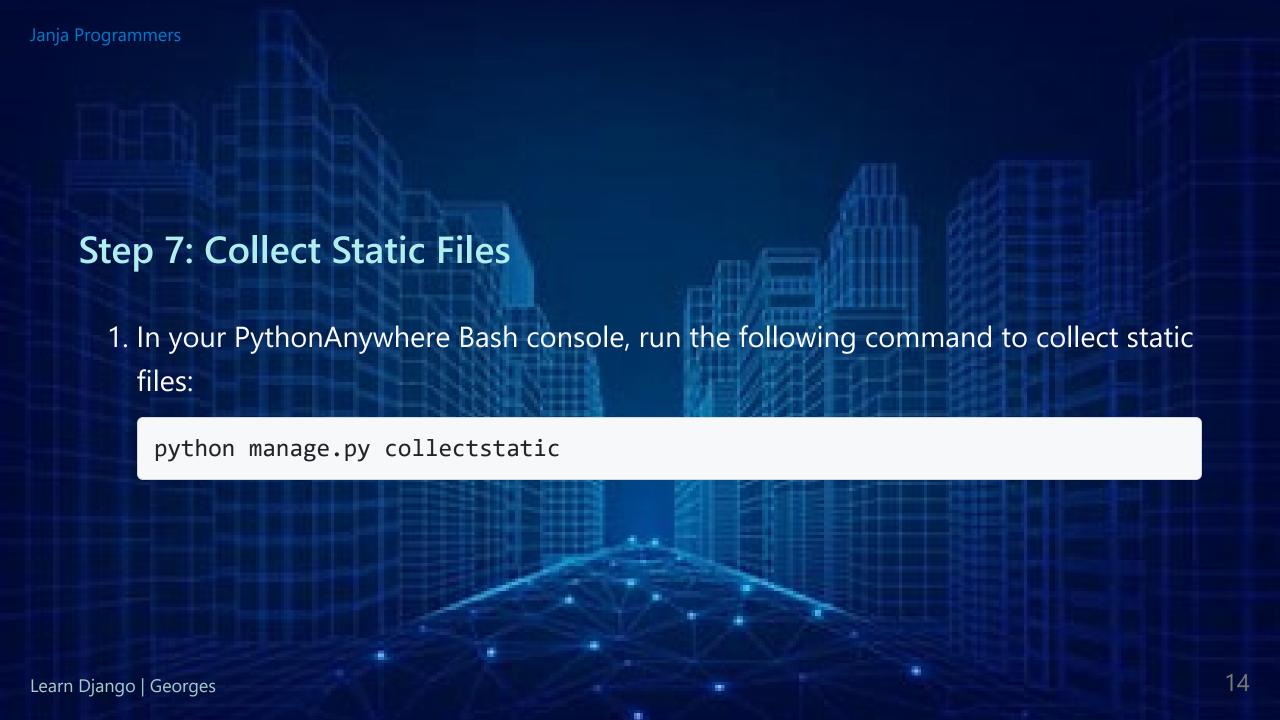
- 1. Open your Django project settings file (settings.py).
- 2. Configure the STATIC_URL, STATIC_ROOT, MEDIA_URL, and MEDIA_ROOT settings to point to the appropriate directories.

```
STATIC_URL = '/static/'
STATIC_ROOT = 'os.path.join(BASE_DIR, "static")'

MEDIA_URL = '/media/'
MEDIA_ROOT = 'os.path.join(BASE_DIR, "media")
```

Learn Django | Georges 12

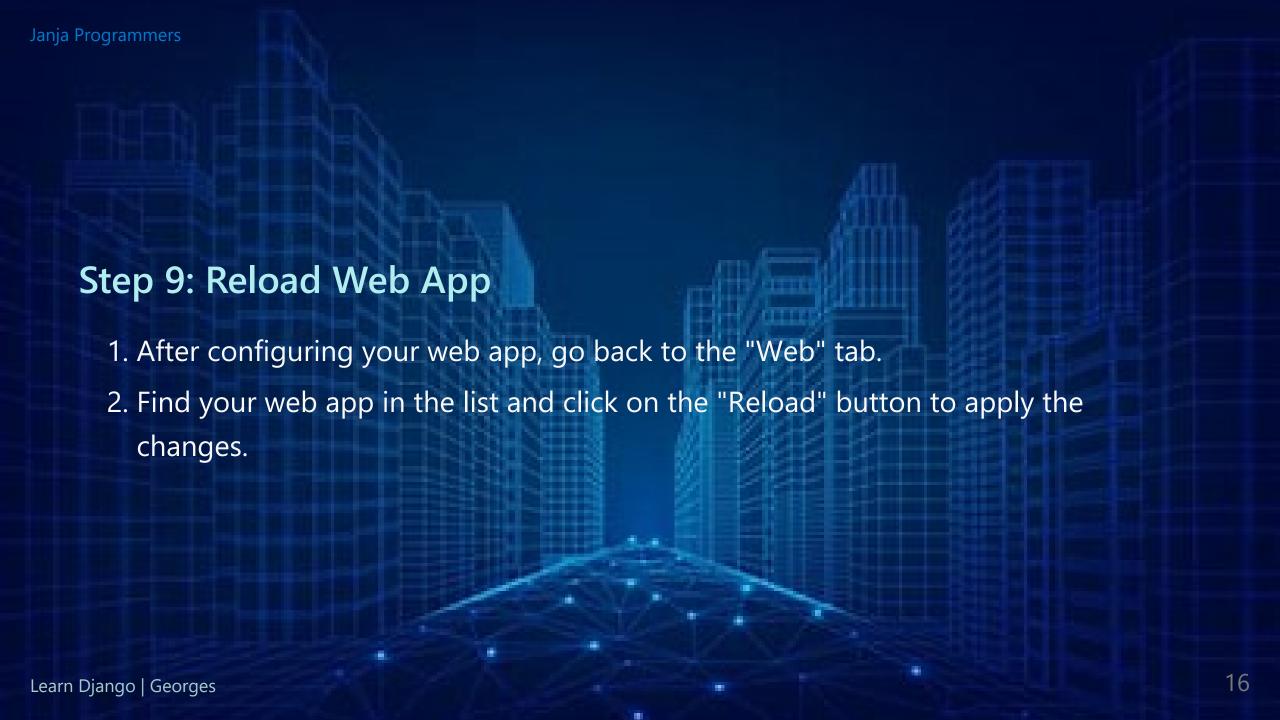


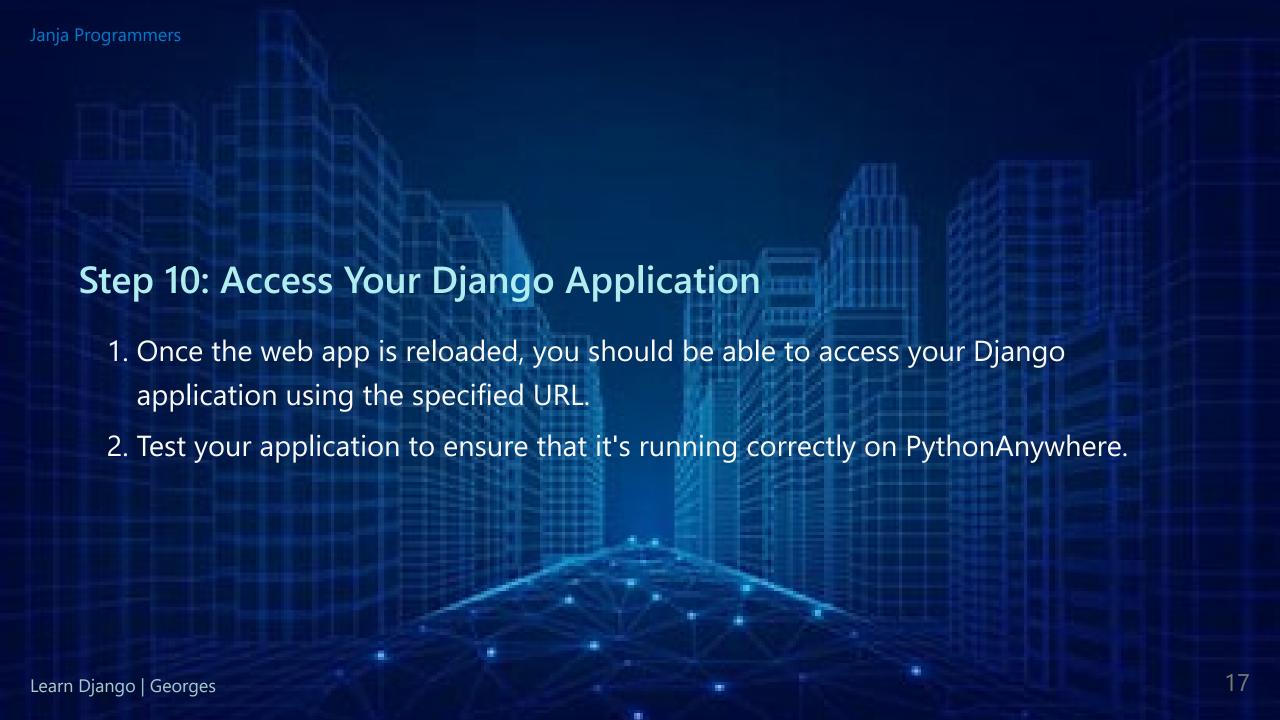


Step 8: Configure Web App on PythonAnywhere

- 1. Navigate to the "Web" tab on the PythonAnywhere Dashboard.
- 2. Click on the "Add a new web app" button.
- 3. Follow the wizard to configure your web app:
 - Choose the manual configuration option.
 - Specify the Python version and select the virtual environment you created.
 - Specify the URL for your web app.
 - Save your changes.

15









Summary

Deploying Django projects is a crucial step in bringing your web applications to production environments. By choosing the right deployment options and following best practices, you can ensure secure, scalable, and reliable deployments. PythonAnywhere offers a simple and free option for deploying Django projects, making it an excellent choice for beginners and small projects.

19