

Deploying a **Django project on Render** is one of the easiest ways to get your app live for free or cheap. Here's a step-by-step guide for uploading and deploying your Django app on **Render**

□ Step 1: Prepare Your Django Project

1. Make sure your project is using a virtual environment

If you don't have one:

Bash:-

```
python -m venv venv
source venv/bin/activate # or venv\Scripts\activate on Windows
```

2. Install necessary packages

Bash:-

```
pip install django gunicorn psycopg2-binary whitenoise
```

- gunicorn: production server for Python
- whitenoise: serves static files
- psycopg2-binary: PostgreSQL driver

3. Add requirements.txt

Bash:-

```
pip freeze > requirements.txt
```

4. Create Procfile (no file extension, capital P)

```
makefile
```

```
web: gunicorn your_project_name.wsgi
```

Replace `your_project_name` with your actual Django project name (folder containing `settings.py`).

5. Edit `settings.py`

a. Add allowed hosts:

```
ALLOWED_HOSTS = ['*']
```

b. Static file config:

At the bottom of `settings.py`, add:

```
import os
STATIC_ROOT = os.path.join(BASE_DIR, 'staticfiles')
STATICFILES_STORAGE =
'whitenoise.storage.CompressedManifestStaticFilesStorage'
MIDDLEWARE.insert(1, 'whitenoise.middleware.WhiteNoiseMiddleware')
```

□ Step 2: Push to GitHub

Create a new repo, then push your code to GitHub if you haven't already:

Bash:-

```
git init
git add .
git commit -m "Initial commit"
git remote add origin https://github.com/yourusername/your-repo.git
git push -u origin master # or main
```

□ Step 3: Deploy on Render

1. Go to <https://render.com> and sign up/log in.
2. Click "New +" → "Web Service"
3. Connect your GitHub repo
4. Fill out the deploy form:

Field	Value
Name	your-app-name
Environment	Python
Build Command	<code>pip install -r requirements.txt && python manage.py collectstatic --noinput && python manage.py migrate</code>
Start Command	<code>gunicorn your_project_name.wsgi</code>
Instance Type	Free or Starter (Free is good for testing)

5. Click "Create Web Service"
-

□ Step 4: Set Environment Variables

In your Render service → go to **"Environment" tab**, add:

- PYTHON_VERSION=3.10 (or your version)
 - DJANGO_SECRET_KEY=your-secret-key
 - DATABASE_URL (Render will auto-create this if you add a PostgreSQL database)
 - DEBUG=False
-

□ Optional: Add PostgreSQL

1. Go to **"Databases"** → **PostgreSQL** → **Create**
2. After creation, copy the **Internal Database URL**
3. Use that as your **DATABASE_URL** env variable

Update settings.py to use dj_database_url:

```
pip install dj-database-url

import dj_database_url

DATABASES = {
    'default': dj_database_url.config(conn_max_age=600)
}
```

□ Step 5: Done! □

Render will build and deploy your project, and you'll get a public URL to access your Django app.

Want me to generate a sample template you can plug in and use directly?

Let me know your project name and if you're using a database or not — I can give you a ready-to-go example.