



## Section : 1. GeoNode

### Module : 1.21. Administration using python management commands



#### Interactions with GeoNode from the command line

*“GeoNode has additional python commands which is only accessible on the host where GeoNode is installed”*

GeoNode provides command-line options for doing various administrative tasks in GeoNode. These commands can be run by an administrator with programming knowledge. These commands also help in the automatic ingestion of the large raster into GeoNode. The management commands are an extension of the admin interface as they allow user to do additional tasks that are not available in the admin interface.

In this module, we explore the management interface from the command line.

[localhost:8080/admin/index.html#geonode-management-commands](http://localhost:8080/admin/index.html#geonode-management-commands)

#### GeoNode Management Commands

- Migrate GeoNode Base URL
- Update Permissions, Metadata, Legends and Download Links
  - Management Command `sync_geonode_layers`
  - Management Command `sync_geonode_maps`
  - Management Command `set_all_layers_metadata`
- Loading Data into GeoNode
  - Management Command `importlayers`
  - Management Command `updatelayers`
    - Data from a PostGIS database
  - Using `GDAL` and `OGR` to convert your Data for use in GeoNode
    - OGR (Vector Data)
    - GDAL (Raster Data)
    - Other Raster Data Use Cases
    - Process Raster Datasets Programmatically
- Create Users and Super Users
- Batch Sync Permissions
  - Usage examples:
- Delete Certain GeoNode Resources
  - Configuration File
  - CLI

#### You try:

**Goal: To explore the some management commands**

#### Docker or Kubernetes

- Navigate to the server where the GeoNode is installed.
- Locate the running **django** container and exec the container.
- Run the following command

```
python manage.py --help
```

- Download the raster using the command line tool like get or curl

```
mkdir raster; wget https://github.com/kartoza/docker-mapserver/blob/master/map/E020N40.tif -O raster/E020N40.tif
```

- Inspect the command to use to ingest the raster into GeoNode

```
python manage.py importlayers -n "East Africa" -v 3 raster
```

- Inspect the layer in GeoNode to see if has been published.

## Virtual Env

- Change directory to where the source code for GeoNode is installed
- Activate the virtual env.
- Run the management command as mentioned in the docker section

## Check your results

When you are in the Django container you should be able to run a command without getting an error message. Execute `python manage.py createsuperuser` to ensure that you set up a master account on GeoNode.

Name	Expectation
GeoNode Install method	Rancher, Docker-compose, Ansible, Virtual Env
Docker container lookup	<code>docker ps -a</code>
Django Container Log in (Docker)	<code>docker exec it django bash</code>
Python Management Commands lookup	<code>python manage.py --help</code>

### More about management commands

GeoNode is built with Django and Python with a backend of either Geoserver or QGIS server and a PostgreSQL database backend. Django has an admin interface that allows users to manipulate all the data and other related components relating to permissions for layers. The management command provides a set of management tools that extend the functionality of the Django admin.

Example of management commands on the command-line. Assume all commands start with `python manage.py $name` of the command. Where the `$name` of command could be

- `createsuperuser`
- `importlayers`
- `updatelayers`

- sync\_geonode\_maps
- set\_all\_layers\_metadata
- 

For a full list of all the management commands execute `python manage.py -help` and choose each individual command to understand how it can be executed for example `python manage.py importlayers -help`

Probably the two main reasons you would need the management commands are:

- to upload and publish very large layers that time out through the web interface
- to publish layers that are available in the backend but not yet in GeoNode

### Check your knowledge:

1. What components make up GeoNode :
  - a. QGIS Desktop, Django and Celery
  - b. GeoServer or QGIS Server, Django, PostgreSQL
  - c. Django, python3, QGIS Desktop
2. Why would you want to use the command line management commands vs admin interface accessible on the browser:
  - a. It does not make sense, the browser is much more friendly
  - b. The command-line should not be used because it is complex
  - c. The command-line allows a user to do more tasks in an automated way which the admin interface will not be able to achieve
  - d. The admin interface is simpler and easy to use and has more commands
3. How would you use the command line management tools:
  - a. Yes, Only if you have installed management software like rancher
  - b. It is impossible, you need server access

### Further reading:

- admin/admin\_mgmt\_commands <https://docs.geonode.org/en/master/admin/index.html#geonode-management-commands>