

---

# **Django admin generator Documentation**

***Release 2.6.0***

**Rick van Hattem**

**Mar 27, 2023**



---

## Contents

---

<b>1</b>	<b>Readme</b>	<b>3</b>
1.1	Introduction . . . . .	3
1.2	Links . . . . .	3
1.3	Install . . . . .	4
1.4	Usage . . . . .	4
<b>2</b>	<b>django_admin_generator package</b>	<b>5</b>
2.1	Subpackages . . . . .	5
2.1.1	django_admin_generator.management package . . . . .	5
2.1.1.1	Subpackages . . . . .	5
2.1.1.2	Module contents . . . . .	6
2.2	Module contents . . . . .	6
<b>3</b>	<b>Indices and tables</b>	<b>7</b>
	<b>Python Module Index</b>	<b>9</b>
	<b>Index</b>	<b>11</b>



Contents:



## 1.1 Introduction

Build status:

Coverage:

The Django Admin Generator is a project which can automatically generate (scaffold) a Django Admin for you. By doing this it will introspect your models and automatically generate an Admin with properties like:

- `list_display` for all local fields
- `list_filter` for foreign keys with few items
- `raw_id_fields` for foreign keys with a lot of items
- `search_fields` for name and slug fields
- `prepopulated_fields` for slug fields
- `date_hierarchy` for `created_at`, `updated_at` or `joined_at` fields

## 1.2 Links

- **Documentation**
  - <http://django-admin-generator.readthedocs.org/en/latest/>
- **Source**
  - <https://github.com/WoLpH/django-admin-generator>
- **Bug reports**

- <https://github.com/WoLpH/django-admin-generator/issues>
- **Package homepage**
  - <https://pypi.python.org/pypi/django-admin-generator>
- **My blog**
  - <http://w.wol.ph/>

## 1.3 Install

To install:

1. Run `pip install django-admin-generator` or execute `python setup.py install` in the source directory
2. Add `django_admin_generator` to your `INSTALLED_APPS`

If you want to run the tests, run `py.test` (requires `pytest`)

## 1.4 Usage

To generate an admin for a given app:

```
./manage.py admin_generator APP_NAME >> APP_NAME/admin.py
```

To generate an admin for a given app with all models starting with user:

```
./manage.py admin_generator APP_NAME '^user' >> APP_NAME/admin.py
```



---

### django\_admin\_generator package

---

## 2.1 Subpackages

### 2.1.1 django\_admin\_generator.management package

#### 2.1.1.1 Subpackages

django\_admin\_generator.management.commands package

Submodules

django\_admin\_generator.management.commands.admin\_generator module

```
class django_admin_generator.management.commands.admin_generator.AdminApp(app,  
                                                                           model_res,  
                                                                           **op-  
                                                                           tions)
```

Bases: `object`

```
class django_admin_generator.management.commands.admin_generator.AdminModel (model,
                                                                    raw_id_threshold=10,
                                                                    date_hierarchy_thres=
                                                                    list_filter_threshold=
                                                                    search_field_names=
                                                                    'slug'),
                                                                    date_hierarchy_name=
                                                                    'up-
                                                                    dated_at',
                                                                    'cre-
                                                                    ated_at'),
                                                                    pre-
                                                                    pop-
                                                                    u-
                                                                    lated_field_names=(
                                                                    ),
                                                                    no_query_db=False,
                                                                    **op-
                                                                    tions)

Bases: object

PRINTABLE_PROPERTIES = ('list_display', 'list_filter', 'raw_id_fields', 'search_fields',
                           'name')

class django_admin_generator.management.commands.admin_generator.Command
Bases: django_utils.management.commands.base_command.CustomBaseCommand

add_arguments (parser)
    Entry point for subclassed commands to add custom arguments.

can_import_settings = True

handle (app=None, *args, **kwargs)
    The actual logic of the command. Subclasses must implement this method.

handle_app (app, model_res, **options)

help = 'Generate a `admin.py` file for the given app (models)'

requires_system_checks = ('__all__',)

warning (message)
    Log a message with severity 'WARNING' on the root logger. If the logger has no handlers, call basic-
    Config() to add a console handler with a pre-defined format.

django_admin_generator.management.commands.admin_generator.get_apps()

django_admin_generator.management.commands.admin_generator.get_models(app)
```

## Module contents

### 2.1.1.2 Module contents

## 2.2 Module contents

## CHAPTER 3

---

### Indices and tables

---

- `genindex`
- `modindex`
- `search`



### d

`django_admin_generator`, [6](#)

`django_admin_generator.management`, [6](#)

`django_admin_generator.management.commands`,  
[6](#)

`django_admin_generator.management.commands.admin_generator`,  
[5](#)



## A

`add_arguments()` (`django_admin_generator.management.commands.admin_generator.Command` attribute), 6

`AdminApp` (class in `django_admin_generator.management.commands.admin_generator`), 5

`AdminModel` (class in `PRINTABLE_PROPERTIES` `django_admin_generator.management.commands.admin_generator` attribute), 6

## C

`can_import_settings` (`django_admin_generator.management.commands.admin_generator.Command` attribute), 6

`Command` (class in `django_admin_generator.management.commands.admin_generator`), 6

## D

`django_admin_generator` (module), 6

`django_admin_generator.management` (module), 6

`django_admin_generator.management.commands` (module), 6

`django_admin_generator.management.commands.admin_generator` (module), 5

## G

`get_apps()` (in module `django_admin_generator.management.commands.admin_generator`), 6

`get_models()` (in module `django_admin_generator.management.commands.admin_generator`), 6

## H

`handle()` (`django_admin_generator.management.commands.admin_generator.Command` method), 6

`handle_app()` (`django_admin_generator.management.commands.admin_generator.Command` method), 6

`help` (`django_admin_generator.management.commands.admin_generator.Command` attribute), 6

## N

`requires_system_checks` (`django_admin_generator.management.commands.admin_generator.Command` attribute), 6

## P

## R

## W

`warning()` (`django_admin_generator.management.commands.admin_generator.Command` method), 6