**Django Built-in Admin Panel**

Django comes with a **ready-made web interface** to manage your database tables.  
It is called the **Admin Panel** and is included by default when you create a Django project.

**1. Purpose of Admin Panel**

* Allows you to **add, edit, and delete** database records without writing SQL queries.
* Works for all models you register.
* Very useful for **testing, managing data, and content** for your site.

**2. How to Enable the Admin Panel**

When you create a Django project, the admin system is already included in:

* INSTALLED\_APPS inside **settings.py**

INSTALLED\_APPS **=** [

    'django.contrib.admin'**,**       *# Admin site*

    'django.contrib.auth'**,**        *# Authentication system*

    'django.contrib.contenttypes'**,**

    'django.contrib.sessions'**,**

    'django.contrib.messages'**,**

    'django.contrib.staticfiles'**,**

]

* Default **URL** for the admin site is added in your project’s urls.py:

**from** django**.**contrib **import** admin

**from** django**.**urls **import** path

urlpatterns **=** [

    path('admin/'**,** admin**.**site**.**urls)**,**

]

**3. Accessing the Admin Panel**

1. Start your Django development server:

python manage**.**py runserver

1. Visit:

http**://**127.0**.**0**.**1**:**8000**/**admin

1. It will ask for **username** and **password** (superuser account).

**Creating a Superuser**

A **superuser** is an admin account with full permissions to manage everything in the admin panel.

**Steps to Create a Superuser**

1. Run the command:
2. python manage**.**py createsuperuser
3. Django will ask for:

Username**:** admin  (**or** any name you choose)

Email address**:** admin**@**example**.**com

Password**:** **\*\*\*\*\*\*\*\***

Password (again)**:** **\*\*\*\*\*\*\*\***

Password must be strong (at least 8 characters). Django warns if it’s too weak, but you can skip warning by typing y.

1. You will see:

Superuser created successfully**.**

**4. Logging into Admin Panel**

* Go to: http://127.0.0.1:8000/admin
* Login with **username** and **password** you just created.

**Registering Models in Admin**

By default, **only built-in models** appear in the admin panel.  
To see your **custom models**, you must register them in admin.py of your app.

Example:

*# admin.py*

**from** django**.**contrib **import** admin

**from** **.**models **import** Student

admin**.**site**.**register(Student)

Now, the **Student** model will appear in the admin panel.

**Summary Table for Notes**

| **Step** | **Command / File Change** | **Purpose** |
| --- | --- | --- |
| Enable admin | 'django.contrib.admin' in INSTALLED\_APPS | Activates admin system |
| Add admin URL | path('admin/', admin.site.urls) in urls.py | Allows access to admin page |
| Create superuser | python manage.py createsuperuser | Creates admin account |
| Run server | python manage.py runserver | Starts project to open admin page |
| Register model | admin.site.register(ModelName) in admin.py | Shows model in admin panel |