



Admin control

Admin control of components



Add Admin Control to web page components

1. Define model class in `models.py`
2. Register model (`admin.py`)
3. Migrate
4. Add data from admin panel
5. Retrieve the data (`views.py`)
6. Fetch data in webpage



Serving files uploaded by a user

In urls.py of project, update url pattern

```
urlpatterns = [  
    # ... the rest of your URLconf goes here ...  
  
] + static(settings.MEDIA_URL, document_root=settings.MEDIA_ROOT)
```



Model

- A model is the single, definitive source of information about your data.
- It contains the essential fields and behaviors of the data you're storing.
- Generally, each model maps to a single database table.
- Each model is a Python class that subclasses `django.db.models.Model`.
- Each attribute of the model represents a database field.

<https://docs.djangoproject.com/en/4.1/topics/db/models/>



Customizing how model is displayed on admin site

1. Create model admin class in admin.py
2. Add relevant details to it using `ModelAdmin.inlines`
3. Register it



ModelAdmin

The **ModelAdmin** class is the representation of a model in the admin interface.

How the model has to be displayed on admin panel can be controlled using `ModelAdmin.inlines`.

- **list_display** -> control the fields that are displayed on the change list page of the admin.
- **list_display_links** -> control if and which fields in **list_display** should be linked to the “change” page for an object.
- **list_filter** -> activate filters in the right sidebar of the change list page of the admin.
- **search_fields** -> enable a search box on the admin change list page. This should be set to a list of field names that will be searched whenever somebody submits a search query in that text box.

<https://docs.djangoproject.com/en/4.1/ref/contrib/admin/>