GE03 Technical Documentation

# Migrations

<https://www.geeksforgeeks.org/django-basic-app-model-makemigrations-and-migrate/>

Makemigrations



Migrate

A screenshot of a computer program

Description automatically generated

|  |  |
| --- | --- |
| https://marketsplash.com/tutorials/django/how-to-delete-migrations-in-django/#Reverting%20Mistakes | How to delete migrations that were not working correctly and causing problems |

Migration

<https://docs.djangoproject.com/en/4.2/topics/migrations/#dependencies>

Shows the migration process and how they work, how to do them, when to do them.Also shows the files they contained such as dependencies and operations

# Models

|  |  |
| --- | --- |
| https://www.geeksforgeeks.org/textfield-django-models/ | How to set up the different fields in a model |

Model’s relationships

<https://www.pythontutorial.net/django-tutorial/django-one-to-many/>

Helped me understand the types of relationships between model’s objects and the work behind them

Lecture 5 intro to SaaS and lecture 10 Django models, databases and queries.

Help me understand the concepts of models, queries and databases and how they work and more

1. one-to-one relationship

<https://docs.djangoproject.com/en/4.2/topics/db/examples/one_to_one/>

To define a one-to-one relationship, use OneToOneField.

1. many-to-one relationship

<https://docs.djangoproject.com/en/4.2/topics/db/examples/many_to_one/>

To define a many-to-one relationship, use ForeignKey.

# Active Record Pattern and BDD

|  |  |
| --- | --- |
| https://www.departmentofproduct.com/blog/writing-bdd-test-scenarios/ | Given-when-then BDD scenario structure |

BDD and active pattern record

<https://www.agilealliance.org/glossary/bdd/#q=~(infinite~false~filters~(postType~(~'page~'post~'aa_book~'aa_event_session~'aa_experience_report~'aa_glossary~'aa_research_paper~'aa_video)~tags~(~'bdd))~searchTerm~'~sort~false~sortDirection~'asc~page~1>)

Help me understand behavior-driven development and agile environment

# 

# Create a Superuser

|  |  |
| --- | --- |
| https://docs.djangoproject.com/en/4.2/intro/tutorial02/ | Setting up the database, setting up a superuser admin account and logging in |

Create superuser

A computer screen with white text

Description automatically generated

Superuser and admin log in

<https://codinggear.blog/how-to-access-django-admin/>

Shows how to create a super user and log-in to django admin app

Resource: <https://www.youtube.com/watch?v=mOu9fpfzyUg&list=PL-51WBLyFTg2vW-_6XBoUpE7vpmoR3ztO&index=5>

Database Models & Admin Panel |Django Framework (3.0) Crash Course Tutorials (pt 5)

A close-up of a computer screen

Description automatically generated

When I run the command 'py -3 manage.py createsuperuser' to create the super user account, I got the following error:

(django\_virtual\_env) d:\Emily\Education\UCCS\fall 2023\CS3300\portfolio>py -3 manage.py createsuperuser

You have 18 unapplied migration(s). Your project may not work properly until you apply the migrations for app(s): admin, auth, contenttypes, sessions.

Run 'python manage.py migrate' to apply them.

How to fix it: run 'py -3 manage.py migrate' first, then run 'py -3 manage.py createsuperuser'

Then run the command 'py -3 manage.py runserver'.

# Starting the Server

Start Server

A black screen with white text

Description automatically generated

# Logging into Admin App

Log into admin

A screenshot of a login screen

Description automatically generated

# Setting Up the Python Shell

Activate the virtual environment and enter the python shell:

A screenshot of a computer code

Description automatically generated

|  |  |
| --- | --- |
| https://www.youtube.com/watch?v=rn3HDfx5nWA | How to set up the python shell to run database queries |

# Queries

|  |  |
| --- | --- |
| https://docs.djangoproject.com/en/4.2/topics/db/queries/ | How to construct database queries in the Python shell |

A screenshot of a computer

Description automatically generated

A screenshot of a computer program

Description automatically generated

Note: when doing the query for the objects of Project, we should activate the virtual environment for Django, and change the directory to portfolio folder, then use the command ‘py -3 -m manage shell' to start python shell. Then import Project (From portfolio\_app.models import Project). Then we can use queryset methods, like all(), get(), etc.

Import the model class first to be able to access the database table values

A black screen with white text

Description automatically generated