

# How to Dump Your Django Database and Load It into a New Project



Roxanna Coldiron  
WEB DEVELOPER  
@roxcoldiron  
Oct. 2, 2020

```
$ python manage.py dumpdata
```

```
$ python manage.py loaddata
```

Image credit: CodeRed LLC

Sometimes you just need to transfer your Django project database into a fresh environment with a new database. This could be the case if you are changing from one type of database to another, or if you want to share an entire copy of a project with someone else. You may not want to have to add in all of your content all over again. One of the easiest solutions to this problem is Django fixtures and two commands `dumpdata` and `loaddata`.

It took hours of Internet searching and several trial runs before I got it right. I want to share how to dump your Django database content so that you can use it in another project or in a different database.

## What are Django Fixtures?

Django fixtures allow you to provide initial data for the models in your project; however, you can also use it to upload your data from another Django app or site. Database fixtures include the data from your database and are output in a JSON, XML, or YAML file.

When you go to load the data into a database, Django will look for a file in the Fixtures folder, which you create in your django app's folders. You specify the output file type and the indentation of your fixture in the Command Line.

Read more about Django fixtures in their documentation: [Django Fixtures](#).

## Exporting the Data

The `manage.py dumpdata` command dumps the data from your database either in the command line or in a file that you specify. You do not want it to dump everything, however, because it will make it difficult to load the data into another database.

For generally dumping from one project and loading into another, you'll need the `dumpdata` command to exclude ContentType objects and Auth Permissions since those will often prevent loading the data the first time. I also want it to go into a JSON file and have indentation to make it more readable. The command for this is:

```
$ python manage.py dumpdata --natural-foreign --natural-primary -e
```

It created the file with the dumped data. But before I could use it, I needed to change the UTF encoding. It will not load into another project if the UTF encoding is wrong. The `dumpdata` command defaults to UTF8-BOM, which will cause an error when you try to load the data.

**Simple Fix:** All I did was create an empty JSON file in the Fixtures folder, copied all of the `dump.json` contents, pasted them into the new JSON file and saved it in VS Code. Then I got rid of the original JSON file since I was not going to use it. Now my `dump.json` file had the correct UTF encoding.

**Easier Fix:** You can actually change the encoding of the fixture file that is output by `dumpdata` in Visual Studio Code without having to make a whole new file. If you look down at the bottom of your view screen for VSCode in the bottom right status bar, there is an option to change or select the encoding (UTF). **Your file needs to be set to UTF-8.** Now save the file again.

## Importing the Data

In your new project, you will want to create its own virtual environment and pip install requirements as you usually would. If you're using MySQL or Postgres, you will need to set up a database like you would for any other Django project. A database needs to exist and be connected to your project in order to load the fixture.

Next run the migrations as normal to set up the database. The final step is to copy the file path of your fixture and load it into the database. The last part should be replaced by the actual path of your fixture file.

```
$ python manage.py migrate
$ python manage.py loaddata "path/to/fixture/file"
```

**Did it work?** Check it by starting the server with `python manage.py runserver` and going to `localhost:8000`. You should see everything from your data dump now loaded into the new project!

**CODERED**  
CODERED, LLC © 2012 - 2022  
ALL RIGHTS RESERVED

55 PUBLIC SQUARE  
SUITE 1710  
CLEVELAND, OH 44113

FOLLOW US



BUILD A BETTER WEBSITE

CODERED CMS