Published in DataDrivenInvestor

You have **2** free member-only stories left this month. Sign up for Medium and get an extra one

George Studenko   Follow
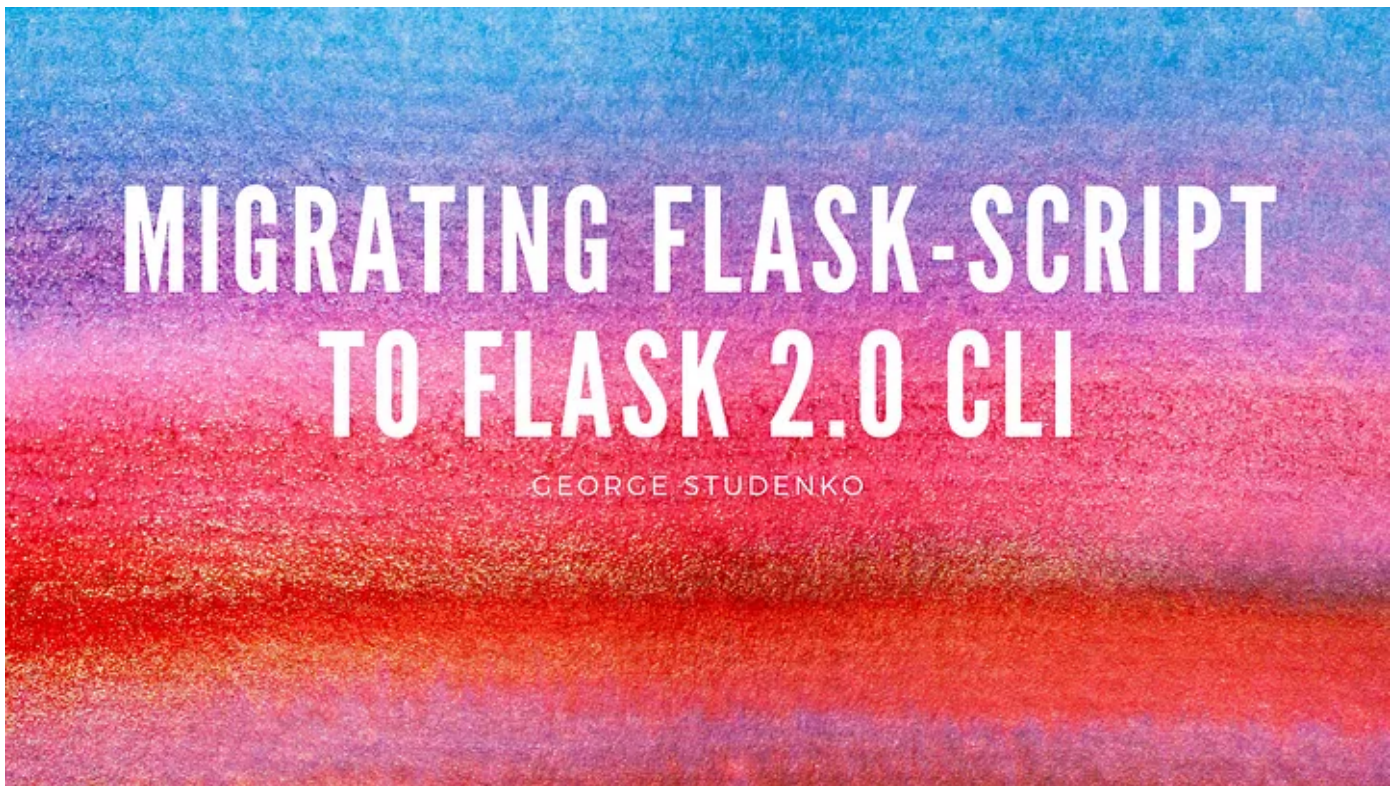
Jul 2, 2021 · 3 min read · ✦ · ▶ Listen

🔖 Save   𝕏   f   in   🔗

How to Migrate Flask-Script to Flask 2.0 CLI in 3 Easy Steps

Flask 2.0 was released recently, and got a lot of great improvements, but as you should already know with new versions also come things that become obsolete and

deprecated.

If you were managing your dev instance of flask with Flask-Script + Flask Migrate in a nice manage.py file where you had all of your commands such as run, test, coverage and db upgrade, db migrate then you will run into some issues when upgrading.

Step one: Cleaning it up

Flask script is no longer supported so you will not longer be able to do this:

```python
from flask_script import Manager
from myapp import app

manager = Manager(app)
```

So the first thing you will need to do it to get rid of flask_script, so go ahead and delete the import and the line where you were creating the instance of Manager.

Also remember to delete Flask-Script from your requirements.txt file, because you had it there right?

Then get rid of flask_migrate in manage.py

```python
from flask_migrate import Migrate, MigrateCommand

migrate = Migrate(app, db)
manager.add_command('db', MigrateCommand)
```

We will do the migrations later through the main app class.

My approach is to use now the Flask 2.0 CLI instead, but there are probably other solutions that could fit you as well, if you know a different one I would be very interested in knowing about it, so if you do please share it in a comment.

Step two: Add the new imports and create the cli

Alright!, let's get coding, so first let's import what is needed and create an instance of the cli:

```
from flask.cli
import FlaskGroup

from myapp import app

cli = FlaskGroup(app)
```

     Sign up   Sign In

◐◖   🔍 Search Medium

```
@cli.command('test')
@click.argument('test_case', default='test*.py')
def test(test_case='test*.py'):
```

And by the way in case you didn't notice above:

```
@manager.option(...)

# Becomes:

@click.argument(...)
```

Also notice that in @click.argument there is no 'help' argument, so just delete it or you will have an error when running the script.

One thing to notice is that the cli will some default commands added by default, including the run command, so I completely deleted my run command in manage.py, and I had to slightly refactor how you start the app in my main class.

Call the new cli when executing the class

So to finish with manage.py you will need to call cli() when running the file:

```
if __name__ == "__main__":
    cli()
```

Run the app

```
python manage.py run
```

Step three: Add flask_migrate to the main app class

Since we are no longer using the MigrateCommand in manage.py we will have to
manage migrations through the main app class.

To do so we will add Migrate to it:

```
from flask_migrate import Migrate

migrate = Migrate()
migrate.init_app(application, db)
```

Now we can use the flask db command to create migrations and upgrade the db:

```
export FLASK_APP=app

flask db init
flask db migrate -m 'Migration message'
flask db upgrade
```

Then just make sure that in your main class you are handling the initialisation of your
app, for example:

```
if __name__ == "app":
    app = create_app(env, args)
```

And there you have it, the migration is done!

Did you find this article useful?

Share your comments and experience!, let us know what works and what did not work for you.

Make sure to give this post some claps (50 or so is a good number!) if you enjoyed this post and want to see more.

0

And to keep up to date make sure to follow, until next time!

Flask   Python   API   Script   Database

Enjoy the read? Reward the writer. [Beta]

Your tip will go to George Studenko through a third-party platform of their choice, letting them know you appreciate their story.

Give a tip

Get notified whenever George publishes a new entry.

Hey there! thanks for your support, if you want to be notified when I publish a new article, you can subscribe here.

By signing up, you will create a Medium account if you don't already have one. Review our Privacy Policy for more information about our privacy practices.

⌷⁺ Subscribe

About   Help   Terms   Privacy

Get the Medium app