serpent_tracker

Brian Hopkins

Mar 23, 2020

CONTENTS:

1	Getti	ng Up and Running Locally	1
	1.1	Setting Up Development Environment	1
	1.2	Setup Email Backend	2
	1.3	Celery	3
	1.4	Sass Compilation & Live Reloading	3
	1.5	Summary	3
2	Getti	ng Up and Running Locally With Docker	5
	2.1	Prerequisites	5
	2.2	Build the Stack	5
	2.3	Run the Stack	5
	2.4	Execute Management Commands	6
	2.5	(Optionally) Designate your Docker Development Server IP	6
	2.6	Configuring the Environment	6
	2.7	Tips & Tricks	7
3	Setti	ngs	9
	3.1	Other Environment Settings	10
4	Testi	ng	11
	4.1	Pytest	11
	4.2	Coverage	11
5	Dock	er Remote Debugging	13
	5.1	Configure Remote Python Interpreter	14
	5.2	Known issues	18
6	Indic	es and tables	21
In	dex		23

ONE

GETTING UP AND RUNNING LOCALLY

1.1 Setting Up Development Environment

Make sure to have the following on your host:

- Python 3.7
- PostgreSQL.
- · Redis, if using Celery

First things first.

1. Create a virtualenv:

\$ python3.7 -m venv <virtual env path>

2. Activate the virtualenv you have just created:

\$ source <virtual env path>/bin/activate

3. Install development requirements:

4. Create a new PostgreSQL database using createdb:

Note: if this is the first time a database is created on your machine you might need an initial PostgreSQL set up to allow local connections & set a password for the postgres user. The postgres documentation explains the syntax of the config file that you need to change.

5. Set the environment variables for your database(s):

Note: Check out the Settings page for a comprehensive list of the environments variables.

See also:

To help setting up your environment variables, you have a few options:

- create an .env file in the root of your project and define all the variables you need in it. Then you just need to have DJANGO_READ_DOT_ENV_FILE=True in your machine and all the variables will be read.
- Use a local environment manager like direnv
- 6. Apply migrations:

\$ python manage.py migrate

7. See the application being served through Django development server:

```
$ python manage.py runserver 0.0.0.0:8000
```

1.2 Setup Email Backend

1.2.1 MailHog

Note: In order for the project to support *MailHog* it must have been bootstrapped with use_mailhog set to y.

MailHog is used to receive emails during development, it is written in Go and has no external dependencies.

For instance, one of the packages we depend upon, django-allauth sends verification emails to new users signing up as well as to the existing ones who have not yet verified themselves.

- 1. Download the latest MailHog release for your OS.
- 2. Rename the build to MailHog.
- 3. Copy the file to the project root.
- 4. Make it executable:

```
$ chmod +x MailHog
```

5. Spin up another terminal window and start it there:

./MailHog

6. Check out http://127.0.0.1:8025/ to see how it goes.

Now you have your own mail server running locally, ready to receive whatever you send it.

1.2.2 Console

Note: If you have generated your project with use_mailhog set to n this will be a default setup.

Alternatively, deliver emails over console via EMAIL_BACKEND = 'django.core.mail.backends. console.EmailBackend'.

In production, we have Mailgun configured to have your back!

1.3 Celery

If the project is configured to use Celery as a task scheduler then by default tasks are set to run on the main thread when developing locally. If you have the appropriate setup on your local machine then set the following in config/settings/local.py:

```
CELERY_TASK_ALWAYS_EAGER = False
```

1.4 Sass Compilation & Live Reloading

If you'd like to take advantage of live reloading and Sass compilation you can do so with a little bit of preparation, see sass-compilation-live-reload.

1.5 Summary

Congratulations, you have made it! Keep on reading to unleash full potential of Cookiecutter Django.

TWO

GETTING UP AND RUNNING LOCALLY WITH DOCKER

The steps below will get you up and running with a local development environment. All of these commands assume you are in the root of your generated project.

Note: If you're new to Docker, please be aware that some resources are cached system-wide and might reappear if you generate a project multiple times with the same name (e.g. this issue with Postgres).

2.1 Prerequisites

- Docker; if you don't have it yet, follow the installation instructions;
- Docker Compose; refer to the official documentation for the installation guide.

2.2 Build the Stack

This can take a while, especially the first time you run this particular command on your development system:

\$ docker-compose -f local.yml build

Generally, if you want to emulate production environment use production.yml instead. And this is true for any other actions you might need to perform: whenever a switch is required, just do it!

2.3 Run the Stack

This brings up both Django and PostgreSQL. The first time it is run it might take a while to get started, but subsequent runs will occur quickly.

Open a terminal at the project root and run the following for local development:

\$ docker-compose -f local.yml up

You can also set the environment variable COMPOSE_FILE pointing to local.yml like this:

```
$ export COMPOSE_FILE=local.yml
```

And then run:

\$ docker-compose up

To run in a detached (background) mode, just:

```
$ docker-compose up -d
```

2.4 Execute Management Commands

As with any shell command that we wish to run in our container, this is done using the docker-compose -f local.yml run --rm command:

\$ docker-compose -f local.yml run --rm django python manage.py migrate \$ docker-compose -f local.yml run --rm django python manage.py createsuperuser

Here, django is the target service we are executing the commands against.

2.5 (Optionally) Designate your Docker Development Server IP

When DEBUG is set to True, the host is validated against ['localhost', 'l27.0.0.1', '[::1]']. This is adequate when running a virtualenv. For Docker, in the config.settings.local, add your host development server IP to INTERNAL_IPS or ALLOWED_HOSTS if the variable exists.

2.6 Configuring the Environment

This is the excerpt from your project's local.yml:

```
# ...
postgres:
    build:
        context: .
        dockerfile: ./compose/production/postgres/Dockerfile
    volumes:
            - local_postgres_data:/var/lib/postgresql/data
            - local_postgres_data_backups:/backups
    env_file:
            - ./.envs/.local/.postgres
# ...
```

The most important thing for us here now is env_file section enlisting ./.envs/.local/.postgres. Generally, the stack's behavior is governed by a number of environment variables (*env(s)*, for short) residing in envs/, for instance, this is what we generate for you:

.envs .local .django .postgres .production

(continues on next page)

(continued from previous page)

— .django — .postgres

By convention, for any service sI in environment e (you know someenv is an environment when there is a someenv.yml file in the project root), given sI requires configuration, a .envs/.e/.sI service configuration file exists.

Consider the aforementioned .envs/.local/.postgres:

```
# PostgreSQL
# -----
POSTGRES_HOST=postgres
POSTGRES_DB=<your project slug>
POSTGRES_USER=XgOWtQtJecsAbaIyslwGvFvPawftNaqO
POSTGRES_PASSWORD=jSljDz4whHuwO3aJIgVBrqEml5Ycbghorep4uVJ4xjDYQu0LfuTZdctj7y0YcCLu
```

The three envs we are presented with here are POSTGRES_DB, POSTGRES_USER, and POSTGRES_PASSWORD (by the way, their values have also been generated for you). You might have figured out already where these definitions will end up; it's all the same with django service container envs.

One final touch: should you ever need to merge .envs/production/* in a single .env run the merge_production_dotenvs_in_dotenv.py:

\$ python merge_production_dotenvs_in_dotenv.py

The .env file will then be created, with all your production envs residing beside each other.

2.7 Tips & Tricks

2.7.1 Activate a Docker Machine

This tells our computer that all future commands are specifically for the dev1 machine. Using the eval command we can switch machines as needed.:

\$ eval "\$(docker-machine env dev1)"

2.7.2 Debugging

ipdb

If you are using the following within your code to debug:

import ipdb; ipdb.set_trace()

Then you may need to run the following for it to work as desired:

\$ docker-compose -f local.yml run --rm --service-ports django

django-debug-toolbar

In order for django-debug-toolbar to work designate your Docker Machine IP with INTERNAL_IPS in local.py.

2.7.3 Mailhog

When developing locally you can go with MailHog for email testing provided use_mailhog was set to y on setup. To proceed,

- 1. make sure mailhog container is up and running;
- 2. open up http://127.0.0.1:8025.

2.7.4 Celery tasks in local development

When not using docker Celery tasks are set to run in Eager mode, so that a full stack is not needed. When using docker the task scheduler will be used by default.

If you need tasks to be executed on the main thread during development set CELERY_TASK_ALWAYS_EAGER = True in config/settings/local.py.

Possible uses could be for testing, or ease of profiling with DJDT.

2.7.5 Celery Flower

Flower is a "real-time monitor and web admin for Celery distributed task queue".

Prerequisites:

- use_docker was set to y on project initialization;
- use_celery was set to y on project initialization.

By default, it's enabled both in local and production environments (local.yml and production.yml Docker Compose configs, respectively) through a flower service. For added security, flower requires its clients to provide authentication credentials specified as the corresponding environments' .envs/.local/.django and .envs/.production/.django CELERY_FLOWER_USER and CELERY_FLOWER_PASSWORD environment variables. Check out localhost:5555 and see for yourself.

THREE

SETTINGS

This project relies extensively on environment settings which **will not work with Apache/mod_wsgi setups**. It has been deployed successfully with both Gunicorn/Nginx and even uWSGI/Nginx.

For configuration purposes, the following table maps environment variables to their Django setting and project settings:

Environment Variable	Django Setting	Development fault	De-	Production fault	De-
DJANGO_READ_DOT_ENV_FILE	READ_DOT_ENV_FILE	False		False	

Environment Variable	Django Setting	Development Default	Production Default
DATABASE_URL	DATABASES	auto w/ Docker; post-	raises error
		gres://project_slug w/o	
DJANGO_ADMIN_URL	n/a	'admin/'	raises error
DJANGO_DEBUG	DEBUG	True	False
DJANGO_SECRET_KEY	SECRET_KEY	auto-generated	raises error
DJANGO_SECURE_BROWSE	R <u>S</u> KSS_FILTER	n/a	True
	CURE_BROWSER_XS	S_FILTER	
DJANGO_SECURE_SSL_RED	I SE CT	n/a	True
	CURE_SSL_REDIREC	Т	
DJANGO_SECURE_CONTEN	T <u>S</u> EYPE_NOSNIFF	n/a	True
	CURE_CONTENT_TY	PE_NOSNIFF	
DJANGO_SECURE_FRAME_	D BN -Y	n/a	True
	CURE_FRAME_DEN	Y	
DJANGO_SECURE_HSTS_IN	C HSDE_BUCBDDDM ASNI	BIQMAINS	True
DJANGO_SESSION_COOKIE	ISTES PONLY	n/a	True
	SION_COOKIE_HTTH	ONLY	
DJANGO_SESSION_COOKIE	SHECSURE	n/a	False
	SION_COOKIE_SECU	JRE	
DJANGO_DEFAULT_FROM_I	ENDÆIL	n/a	"your_project_name <nore-< td=""></nore-<>
	FAULT_FROM_EMAI	L	ply@your_domain_name>"
DJANGO_SERVER_EMAIL	SERVER_EMAIL	n/a	"your_project_name <nore-< td=""></nore-<>
			ply@your_domain_name>"
DJANGO_EMAIL_SUBJECT_	PIENFAXL_SUBJECT_PR	EH7bX	"[your_project_name] "
DJANGO_ALLOWED_HOSTS	ALLOWED_HOSTS	['*']	['your_domain_name']

The following table lists settings and their defaults for third-party applications, which may or may not be part of your project:

Environment Variable	Django Setting	Development Default	Production De-
			fault
CELERY_BROKER_URL	CEL-	auto w/ Docker; raises	raises error
	ERY_BROKER_URL	error w/o	
DJANGO_AWS_ACCESS_KEY_II	AWS_ACCESS_KEY_ID	n/a	raises error
DJANGO_AWS_SECRET_ACCES	S_A&VESY_SECRET_ACCESS_H	K TE Xa	raises error
DJANGO_AWS_STORAGE_BUCK	EATWISASSATERAGE_BUCKET	ſ_ iMa ME	raises error
DJANGO_AWS_S3_REGION_NAM	MAWS_S3_REGION_NAME	n/a	None
DJANGO_GCP_STORAGE_BUCK	EG <u>SN</u> BANGKET_NAME	n/a	raises error
GOOGLE_APPLICATION_CREDE	ENHTALS	n/a	raises error
SENTRY_DSN	SENTRY_DSN	n/a	raises error
DJANGO_SENTRY_LOG_LEVEL	SENTRY_LOG_LEVEL	n/a	logging.INFO
MAILGUN_API_KEY	MAILGUN_API_KEY	n/a	raises error
MAILGUN_DOMAIN	MAIL-	n/a	raises error
	GUN_SENDER_DOMAIN		
MAILGUN_API_URL	n/a	n/a	"https://api.
			mailgun.net/v3"

3.1 Other Environment Settings

DJANGO_ACCOUNT_ALLOW_REGISTRATION (=True) Allow enable or disable user registration through *django-allauth* without disabling other characteristics like authentication and account management. (Django Setting: ACCOUNT_ALLOW_REGISTRATION)

FOUR

TESTING

We encourage users to build application tests. As best practice, this should be done immediately after documentation of the application being built, before starting on any coding.

4.1 Pytest

This project uses the Pytest, a framework for easily building simple and scalable tests. After you have set up to develop locally, run the following commands to make sure the testing environment is ready:

\$ pytest

You will get a readout of the *users* app that has already been set up with tests. If you do not want to run the *pytest* on the entire project, you can target a particular app by typing in its location:

\$ pytest <path-to-app-in-project/app>

If you set up your project to develop locally with docker, run the following command:

\$ docker-compose -f local.yml run --rm django pytest

Targeting particular apps for testing in docker follows a similar pattern as previously shown above.

4.2 Coverage

You should build your tests to provide the highest level of **code coverage**. You can run the pytest with code coverage by typing in the following command:

\$ docker-compose -f local.yml run --rm django coverage run -m pytest

Once the tests are complete, in order to see the code coverage, run the following command:

\$ docker-compose -f local.yml run --rm django coverage report

Note: At the root of the project folder, you will find the *pytest.ini* file. You can use this to customize the pytest to your liking.

There is also the *.coveragerc*. This is the configuration file for the coverage tool. You can find out more about configuring coverage.

See also:

For unit tests, run:

```
$ python manage.py test
```

Since this is a fresh install, and there are no tests built using the Python unittest library yet, you should get feedback that says there were no tests carried out.

DOCKER REMOTE DEBUGGING

To connect to python remote interpreter inside docker, you have to make sure first, that Pycharm is aware of your docker.

Go to *Settings > Build, Execution, Deployment > Docker*. If you are on linux, you can use docker directly using its socket *unix:///var/run/docker.sock*, if you are on Windows or Mac, make sure that you have docker-machine installed, then you can simply *Import credentials from Docker Machine*.

😣 🗈 Settings			
Q.	Build, Execution, D	ployment > Docker	Reset
Appearance & Behavior	+ -	Na <u>m</u> e: Docker	
Keymap ▶ Editor	🗇 Docker	API URL:	var/run/docker.sock
Plugins		Certificates folder:	· · · · · · · · · · · · · · · · · · ·
 Version Control Project: reddit 		Docker Compose executable: docker-	-compose
Build, Execution, Deployment		Import credentials from Docker Mach	.hine
 Debugger Python Debugger 		Docker Machine executable: dock	ker-machine Detect
▶ Deployment @		Machine:	
Buildout Support @ ▶ Console @		Pidenine.	
Coverage			
Docker Docker Registry			
► Languages & Frameworks			
▶ Tools			
			OK Cancel Apply Help

5.1 Configure Remote Python Interpreter

This repository comes with already prepared "Run/Debug Configurations" for docker.



But as you can see, at the beginning there is something wrong with them. They have red X on django icon, and they cannot be used, without configuring remote python interpreter. To do that, you have to go to *Settings* > *Build*, *Execution*, *Deployment* first.

Next, you have to add new remote python interpreter, based on already tested deployment settings. Go to *Settings* > *Project* > *Project Interpreter*. Click on the cog icon, and click *Add Remote*.

Qinterp	8	Project: reddit > Project Interpret	er @ For current project				
▼ Editor		Project Interpreter: 👼 2.7.6 (/usr/bin/python2.7)					
Colors & Fonts					AddLocal		
reStructuredText		Package	Version	Latest	Add Local		
Sass/SCSS		Diango	1.9	⇒ 1.9.4	Create Virtual Env		
Inspections	6	Jinja2	2.8	2.8			
Plugins		MarkupSafe	0.23	0.23	More		
v Project: reddit		Nautilus-PyExtensions	3.2				
Project Interpreter	6	PAM	0.4.2				
Build, Execution, Deployment		Pillow	2.3.0	➡ 3.1.1			
▼ Console	6	PyBluez	0.18	➡ 0.22			
Python Console		PyICU	1.5	➡ 1.9.2			
Vitrion Console Languages & Frameworks JavaScript Code Quality Tools		PyYAML	3.11	3.11			
	P	Twisted-Core	13.2.0				
	421	Twisted-Web	13.2.0				
Code Quality Tools		adium-theme-ubuntu	0.3.4				
7262		apt-xapian-index	0.45				
ESLint		argparse	1.2.1	➡ 1.4.0			
Bower		backports.ssl-match-hostname	3.4.0.2				
 TypeScript 		binaryornot	0.4.0	0.4.0			
TSLint		ccsm	0.9.11.3				
▼ Tools		chardet	2.0.1	➡ 2.3.0			
SSH Terminal	6	click	5.1	➡ 6.4			
		colorama	0.2.5	➡ 0.3.7			
		command-not-found	0.3				
		compizconfig-python	0.9.11.3				
		cookiecutter	1.3.0	➡ 1.4.0			
		debtagshw	0.1				
		derer	1.0.6	1.0.4			
		discolor	42.40	42.00			
				OK Capcel	Apply Holo		

🛚 🗉 Settings								
Qinterp	8	Project: reddi	> Project Interp	reter @ For current	t project			Reset
 Editor Colors & Fonts 		Project Interpr	eter: 🟓 2.7.6 (/u	sr/bin/python2.7)				• *
reStructuredText Sass/SCSS Inspections Plugins	ū	Django Jinja2 MarkupSafe	Package	1.9 2.8 0.23	Version	➡ 1.9.4 2.8 0.23	Latest	+ - +
🔻 Project: reddit		Nautilus-PyE×	tensions	3.2				
Project Interpreter	D	PAM		0.4.2				
 Build, Execution, Deployr Console Python Console Languages & Frameworks JavaScript Code Quality Tools JSCS ESLint Bower TypeScript TSLint Tools SSH Terminal 	Confi <u>V</u> agran Server: Configurati Service nam Python inte	gure Remote Py t O Deployme Docker on: /home/ksz ne: django rpreter path:	/thon Interpreter nt configuration umny/Devel/redd python	○ ∑SH Credentials	s () Docker () D	New		
	_	colorama	<i>c</i> 1	0.2.5		➡ 0.3.7		_
		command-no	t-found	0.3				
		cookiecutter	-python	1.3.0		➡ 1.4.0		
		debtagshw		0.1		- 1.1.0		
		defer		1.0.6		1.0.4		
		disconc		42.40		42.00		
						ок Са	ancel <u>A</u> pply	Help

Switch to Docker Compose and select local.yml file from directory of your project, next set Service name to django

Having that, click OK. Close Settings panel, and wait few seconds...

	_	_	Event Log	-

After few seconds, all Run/Debug Configurations should be ready to use.



Things you can do with provided configuration:

• run and debug python code

Eile Edit View Navigate Code Befactor Run	Iools VCS Window Help	
🖿 reddit 👌 🗈 config 👌 🗈 settings 👌 😼 common.py 👌	[Do	ocker: runserver 🔹 🕨 🗮 🍀 🕸 🐺 🌹 🚏 📁 🔍
🔐 Project 🔹 😳 💠 🕸 + 🕸 - !*	Recommon.py x	1
<pre></pre>	<pre># coding: wfr# Diamo settings for Medit (lone project. For once information on this file, see https://doc.djangeroject.com/en/doc/topics/settings/ For the full list of settings and their values, see https://doc.djangeroject.com/en/doc/topics/ import Exact Set Settings for Apple (list of settings/ http://doc.djangeroject.com/en/doc/topics/ settings/ How - environ.Brv() # APPC CONTONE # APPC CONTONE DIMONO_PPS - { figure_contrib.settings',</pre>	Debase
Debug 🔄 Docker: runserver		@ · ±.
G Debugger 🗉 Console + 🔚 👱 🖄 🖄 🦉		
Frames	Variables	→* 🐼 Watches →*
ImainThread • • - <td< td=""><td> Image: State of the state of th</td><td>w + − + + D w No watches</td></td<>	 Image: State of the state of th	w + − + + D w No watches
Python Console Terminal Stresson Console Python Console Stresson Console Construction Constructico Constru	itrol 🔍 3: Find 🕨 4: Run 💘 5: Debug 🐚 6: TODO	Event Log
En or running bocker: runserver: Can't run remo	te python interpreters raited to create endpoint reddit_gango_r on network reddit_default: Bind for 0.0.0.08000 failed: port is already allocated (5 minutes ago)	/ is:i Lr= UIH-8 GIC: mascer= % @

• run and debug tests

File Edit View Na	avidate Code Relactor Run	1000			
🗀 reddit 🕽 🛅 reddit) 🗈 users) 🗈 tests) 🔒 test vie	ws.py)		🖾 Docker; tests - all 🔹 🕨 🛞 🐺 🐺 🖞 🖤 🗇	Q
Project	· ○ ÷ ∲· !!	· 🔒 ma	odels.py × 🔒 urls.py × 🔒 views.py × 🔒 test_views.py ×		1
👹 🔻 🖿 reddit ~/co	oklecutter/reddit		from django.test import RequestFactory	,	Da
🛔 🕞 🖿 compose	2		from test plus test import TestCase		taba
Image: Second			from views import (se
Docs			UserRedirectView,		
	b		UserUpdateview		
k ⊨ static					
💙 💿 kaskap	pp	oļ	class BaseUserTestCase(TestCase):		
▷ templ	ates	o† 0.	def setUp(self):		
v 🗈 users			self.user = self.make_user() self.factory = RequestFactory()		
► E mig	grations				
• E (es	init .pv		class TestUserRedirectView(BaseUserTestCase):		
	Factories.py		<pre>def test_get_redirect_url(self):</pre>		
R	test_admin.py		<pre># Instantiate the view directly. Never do this outside a test/ view = UserRedirectView()</pre>		
R	test_models.py		<pre># Generate a fake request request = colf festers ast('/fake.url')</pre>		
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	test_views.py		# Attach the user to the request		11
"" □ adi	anters py		# Attach the request to the view		
adr	min.py		<pre>view.request = request # Expect: '/users/testuser/'. as that is the default username for</pre>		
🗟 ma	dels.py		# self.make_user()		
🔒 url	s.py		view.get_redirect_url(),		
🗟 vie	ws.py		'/users/testuser/'		
🧔 _init_	py				
Tequirent	num num		class TestUserUpdateView(BaseUserTestCase):		
.coverag	erc	•†	<pre>def setUp(self):</pre>		
.dockerig	gnore		# call BaseUserTestCase.setUp() super(TestUserUpdateViev, self).setUp()		
Run 🚺 Docker: tes	ts - all			&- ±	
🕨 💽 🗐 🐙 .	原王子 1 + 1 昭義		All 7 tests passed – 328ms		
p 💌 💌 Test Re	esults		szens reddit_pycha:python -u /opt/.pycharm_helpers/pycharm/django_test_manage.py test . /app	†	
🕼 🔻 🖲 rede	dit.users.tests.test_models.Test	User	89ms Creating test database for alias 'default'	+	
= et	est_str_ est get absolute url		Destroying test database for alias 'default' 44ms		
📰 🔻 🖲 rede	dit.users.tests.test_views.TestUs	serRedir	ectView 40ms Process finished with exit code 0	E	
	est_get_redirect_url		40ms	10	
X v en rede	dit.users.tests.test_views.TestUs est.get.object	serUpda	teView 96ms	<u></u>	
	est get success url		54ms		
g 🔹 🔹 rede	dit.users.tests.test_admin.TestM	lyUserC	reation! 103ms		
2 (B)	est_clean_username_false		53ms		
*	esc_ctean_osername_soccess		30115		
🖏 Python Console	e 🗵 Terminal 🛛 🧕 🤉: Version Con	trol (1, <u>3</u> : Find ▶ <u>4</u> : Run 🗰 <u>5</u> : Debug 🐏 <u>6</u> : TODO	Event Log	
File Falls Marco Ma	ularta Cada Defentes Due 3	Teels 1			
Eile Edit View Na	avigate <u>C</u> ode <u>Befactor</u> Run] ;) to users) to tests)] test_vie	Tools \ ews.py	ICS Window Help	[ЩDocken.tests-all™] ► 💐 🕸 🕸 🛱 😰 😨 🖄	Q
Eile Edit View Na reddit) reddit	avigate <u>C</u> ode <u>B</u> efactor <u>Run</u>] :)	Tools N ews.py	CS Window Help	[@Docker.tests-all ≠ 🕨 ରେ କ 🕸 ହି ହି ଛି ୦	0
Eile Edit View Na reddit) = reddit Project reddit ~/co	avigate Code Befactor Run] :) E users) E tests) & test_vie • © # #• !** ioklecutter/reddit	Tools N ews.py) Remo	CS_Window_Help dels.py x & urls.py x & vews.py x & test_views.py x & common.py x effer setup(sel1): sel1user = sel1uske_user()	[@Docker.tests_all⊻ ▶ 《 ※ ※ ♡ ♡ ♡ ♡ ♡ ♡ ♡ ♡ ♡ ♡ ♡ ♡ ♡ ♡ ♡ ♡ ♡	o 🖉 Datat
Eile Edit View Na reddit) E reddit Project Project F compose E config	avigate <u>C</u> ode <u>R</u> efactor <u>Run</u>] :) <u>E</u> users <u>E</u> tests <u>B</u> test_vie - <u>B</u> + <u>B</u>	Tools N ews.py Reprint	CS Window Help dels.py x @ urls.py x @ views.py x @ test_views.py x @ common.py x def setup(set); setf.setup.r = setf.sete_user() setf.setup.r = NetworkTectory()	[@Docker.tests-al.* ▶ ≪ 등 등 두 두 두 만 5	o 🖉 Database
Eile Edit View Na reddit) Ereddit Project Project Project Project Compose F config F config F config	avigate Code Refactor Run] :) Enusers) Entests) & test_vie • Other Armonia (Construction) • Other Armonia (Construction)	Tools News.py	CS Window Help dels.py x @ urls.py x @ vlews.py x @ test_vlews.py x @ common.py x def settp(set1): setf.user = setf.rake user() setf.ractory = RequestFactory() dear TestPointentine(insettine(in	[@Docker.tests-all*] ▶ 戦 ⊕ 命 耳 寧 寧 말 ⊃	of Database
Eile Edit View Na reddit) Ereddit Project Project Project F compose F config F codig F codig	avigate _Code _Refactor _Run _] :) E users) E tests)	Tools V ews.py	CS Window Help dels py x @ virs.py x @ views.py x @ test_views.py x @ common.py x def setUb(ref): setUt common restrictions user() setUt common restrictions () et ass TestUserRedirectView(BaseUserTestCase):	[@Docker.tests-all ▼ ▶ 戦 ŵ ŵ 등 및 및 및 한 ⊃	o Database
Elle Edit View Na reddit) reddit Project Project reddit -/cco F compose Compose reddit -/cco F compose reddit -/cco F compose F confis	avigate _Code Befactor Run] ;)	Tools V ews.py) · Common et et	CS Window Help dels.py x @ urts.py x @ vlews.py x @ test_vlews.py x @ common.py x eff settp(sel1): sett.setry = RequestInstructy() sett.setry = RequestInstructy() class TestUserRedirectView(BaseUserTestCase): del test_sqt_redirect_uv[(sel1): sel1; test_pet_redirect_url (redd_t.users.test_test_views.TestUserRedirectView) # firstraites the view of this outside a test/	[@Docker.tests-al." ► ৠ ৩ জ জ ে ফ ফ ফ b	of Database
Elle Edit View Na reddit) reddit Project reddit -/cco F config F confi	avigate Code Befactor Run]) Crusers) Entests) Brest view -	Tools V ews.py Remo	CS Window Help dels.py x @ urls.py x @ Wews.py x @ test_views.py x @ common.py x eff settig.fcs(?): eff settig.fcs(?): setti.fcs(row) = Repetificatory() class TestUserRedirectView(BaselserTestCase): def test_get_redirect_url(setl): self.test_get_redirect_url (reddit.users.tests.test_views.TestUserRedirectView) a finismizing the view direct(Vi.wver do this outside a test): def test_get_redirect_arl(setl): self.test_get_redirect_url (reddit.users.tests.test_views.TestUserRedirectView) a finismizing the view direct(Vi.wver do this outside a test): def test_get_redirect_arl(setl): self.test_get_redirect_url (reddit.users.tests.test_views.TestUserRedirectView) a finismizing the view direct(Vi.wver do this outside a test): def test_get_redirect_arl(setl): self.test_get_redirect_url (reddit.users.tests.test_views.TestUserRedirectView) a finismizing the view direct(Vi.wver do this outside a test): def test_get_redirect_url(setl): self.test_get_redirect_url (reddit.users.tests.test_views.TestUserRedirectView) a finismizing the view direct(Vi.wver do this outside a test): def test_get_redirect_url(setl): self.test_get_redirect_url (reddit.users.tests.test_views.TestUserRedirectView) a finismizing the view direct(Vi.wver do this outside a test): def test_get_redirect_url(setl): self.test_get_redirect_url (reddit.users.tests.test_views.TestUserRedirectView) a finismizing the view direct(Vi.wver do this outside a test): def test_get_redirect_url(setl): self.test_get_redirect_url (reddit.users.tests.test_views.tests.test_views.tests.test_views.tests.test_views.tests.test_views.tests.test_views.tests.test_views.tests.test_views.tests.test_views.tests.test_views.tests.test_views.tests.test_views.tests.test_views.tests.test_views.tests.test_views.tests.test_views.tests.test_views.tests.tests.test_views.tests	@Docker.tests-all > ► ৠ ৩ জ झ জ জ ৫ এ ১	0' 🔮 Database
Elle Edit View Na reddit) reddit Project reddit /ccompose F config F c	wigate <u>Code</u> <u>B</u> efactor <u>Run</u>)) <u>D</u> users) <u>D</u> tests) <u>G</u> test vie) <u>D</u> users) <u>D</u> tests <u>vie</u> C vie C vie Vie C vie Vi Vie 	Iools V ews.py) Bond I I	CS Window Help dels py x @ vlews.py x @ test_vlews.py x @ common.py x def sttlp(srif): stlf.factory = RequestFactory() class TestBarRedirectView(BaseBarTestCas): def test_get_redirect_url(srlf): solf test_get_redirect_url (reddit.users.tests.test_views.TestDarRedirectView) stlf.factory = solf.settory of this outside a test! f demonster a flate request request = solf.factory.get(/fade=utl')	[@Docker.tests-all: ▶ ≪ 용 등 두 두 두 두 5	0 🔮 Database
Elle Edit View NA reddit) = reddit Project P	avigate <u>Code</u> <u>B</u> efactor <u>Run</u>]) <u>b</u> users) <u>b</u> tests) <u>b</u> tests) <u>b</u> test vie v v v v v v v v v v	Iools V ews.py) Remo	CS Window Help dels.py x @ urts.py x @ vlews.py x @ test_vlews.py x @ common.py x off settp(set1): sett settp(set1): sett settp(set1): sett settp(set1): sett settp(set1): sett settp(set1): sett settp(set1): sett settp(set1): settp	[@Dockentests-all > ▶ ≪ 등 등 약 약 만 5	0 Database
Elle Edit View bly reddit) breddit Preddit) breddit Preddit / breddit Preddit -/cc Preddit -/ccc Preddit -/cc Preddit -/cc Preddit -/cc Preddit -	avigate Code Befactor Run 1) Disers) Ditests) Ditests) Ditest (best view oklecutter/reddit b b pp ates grations	Iools V ews.py) mo	CS Window Help dels py x @ urts py x @ vews py x @ test_views py x @ common py x eff setty (scif): eff setty (scif): sett factor = Repertence(or)() class TestUserRedirectView(BaseUserTestCase): eff test_get_redirect_view(BaseUserTestCase): eff test_get_	@Docker.tests-all > ► ৠ ৩ জ জ জ জ জ জ জ জ জ জ জ জ জ জ জ জ জ জ	0 La Database
Elle Edit View bk	avigate <u>Code</u> <u>B</u> efactor <u>Run</u> <u>)</u>) <u>Dusers</u>) <u>Dtests</u>) B test vie) <u>Dusers</u>) <u>Dtests</u>) B test vie v <u>Objective</u> okliecutter/reddt b b pp ates grations ts pp	Tools V ews.py) Remo	CS Window Help delspy x & Wurkspy x & Vewspy x & test_views.py x & common.py x def stub(stif): stif.factory = RequestFactory() class TestUserRedirectView(BaskUserTestCase): def test_get_redirect_url(ss(1): self.test_get_redirect_url (reddit.users.tests.test_views.TestUserRedirectView) * stif.factory = self.testroy.get('fade-url') request_self.testroy.get('fade-url') request_self.testroy.get('fade-url') request_self.testroy.get('fade-url') request_self.testroy.get('fade-url') request_self.testroy.get('fade-url') request_self.testroy.get('fade-url') request_self.testroy.get('fade-url') request_self.testroy.self.testroy	[@Docker.tests-al." ► ¾ ୫ ର ∓ ଅଅଅର ୨	0 🔮 Database
Ele Edit View bit reddit > 0:reddit Droddit > 0:reddit Project	avigate <u>Code</u> <u>Befactor</u> <u>Run</u>]) <u>D</u> users) <u>D</u> tests() \overline{constrainty} (Constrainty) (Iools N ews.py ⟩ · ≩ ma • ™	CS Window Help dels py x @ wits py x @ vlews py x @ test_vlews.py x @ common py x def setup(sil): silf.factory = RequestFactory() silf.factory = RequestFactory() def test_eqt_redirect_vle(set): set_eqt_redirect_url (reddit.users.tests.test_views.TestWserRedirectView) # finitesite the view for test_get_redirect_url (reddit.users.tests.test_views.TestWserRedirectView) # finitesite the view for test_get_redirect_url (reddit.users.tests.test_views.TestWserRedirectView) # finitesite the view for test_get_redirect_url (reddit.users.tests.test_views.TestWserRedirectView) # finitesite the view for test_set_views for test_get_redirect of test_get_redirect of test_get_redirect of test_get_redirect of test_get_views for test_get_redirect of test_get_views for test_get_redirect(test_get_redirect) of test_get_redirect(test_get_redirect) of test_get_redirect(test_get_redirect) of test_get_views for test_get_redirect(test_get_redirect) of test_get_redirect) of test_get_redirect(test_get_redirect) of test_get_redirect(test_get_redirect) of test_get_redirect) of test_get_redirect(test_get_redirect) of test_get_redirect) of test_get_redirect(test_get_redirect) of test_get_redirect) of test_get_re	[@Docker.tests-al." ► ৠ ৩ জ জ ৼ ৼ ৼ ৳ >	0 Database
Ele Edit View bit	avigate <u>Code</u> <u>Befactor</u> <u>Run</u> <u>3</u>) <u>Discos</u> ; <u>Discos</u>	Tools V ews.py) Particular Partic	CS Window Help dels py x @ uris py x @ views py x @ test_views py x @ common py x effective self.rate user() self.ratery = RequestFactory() class TestBerRedirectView(BaselBerTestCase); effective trained are the view directly, Rever do this outside a test/ error endotrective() effective trained are the view directly, Rever do this outside a test/ error endotrective() error endotrective	[@Docker.tests-al." ▶ ≪ ⊗ ⊛ ∓ ♥ ♥ ♥ ⊃	O Database
Elle Edit View bit reddit > Dreddit Broject P	avigate <u>Code</u> <u>Befactor</u> <u>Run</u> <u>7</u>) <u>Dusers</u>) <u>Dtests</u>) Betst , <u>Best</u> , <u>B</u>	Tools V ews.py)	CS Window Help dels py x @ vufs.py x @ vlews.py x @ test_vlews.py x @ common.py x eff stilp(stil): stil.factory = RequestFactory() class TestBacRedirectView(BaseBacrEstEase): def test_get_redirect_url(stil): soif test_get_redirect_url (reddit.users.tests.test_views.TestDacRedirectView) # finalizes the size class clas	[@Docker.tests-all⊻ ▶ ৠ ড় জ জ ঢ় ঢ় ঢ় ঢ় b s	0 🔮 Database
Elle Edit View bit reddit > Dreddit - / Creddit Project	avigate Code Befactor Run 1 busers) Entests) & test view busers) Entests) & test view constructor/reddt b b pp ates grations tis initopv factories pv test_admin.pv test_admin.pv test_admin.pv test_admin.pv	Tools \ ews.py > • @ ma • • •	<pre>CS Window Help dels py x @ wits py x @ vlews py x @ test_vlews.py x @ common py x def setup(set): set.factory = RequestFactory() set.factory = RequestFactory() set.factory = RequestFactory() def test_eqt_redirect_vlew(BaselberTestCase): def test_eqt_redirect_vlew(BaselberTestCase): def test_eqt_redirect_vlew(red bits outside a test) request = set.factory.get(rfake-unt') # infinition = set.factory.get(rfake-unt') # set.factory.exer(bits outside a test) request = set.factory.exer(bits outside a test) # set.facto</pre>	[@Docker.tests-al." ▶ ≪ ♥ 중 루 ♡ ♡ 만 > >	0 🔮 Database
Elle Edit Mew Bit reddit b) reddit b) Preddit b) reddit b) Preddit b) reddit Preddit b) reddit Preddit b) Preddit b) Pr	avigate Code Befactor Run 1) Durses) Ditests) & Cest, Vie) Durses) Ditests) & Cest, Vie v 0 + 0 + 0 okiecutter/reddt ates grations ts 	Iools \ ews.py > mm e e	CS Window Help dels py * @ uris py * @ views py * @ test_views py * @ common py * dels restly_del(1): sist_ist_ist_ist_ist_ist_ist_ist_ist_ist	[@Docker.tests-al." ▶ ≪ ⊗ ⊛ ∓ ♥ ♥ ♥ > >	0 🔮 Database
Elle Edit View bit reddit > Dreddit = /creddit B Project B Project P reddit = /creddit = /cre > D compiss > D compis	avigate <u>Code</u> <u>Befactor</u> <u>Run</u> <u>1</u>) <u>Dusers</u>) <u>Dtests</u>) <u>Betasts</u> <u>Best</u> <u>Vie</u>) <u>Dusers</u>) <u>Dtests</u> <u>Best</u> <u>Vie</u> <u>Other</u> <u>Code</u> <u>All</u>	Iools \ ews.py > e mm et el	CS Window Help dels py x @ wirks py x @ views py x @ test_views py x @ common py x def setb(set1):	[@Docker.tests-all ≥ ▶ ≪ ♥ ⊕ ∓ ♥ ♥ ♥ D	0 🔮 Database
Elle Edit View bit reddit > Dreddit = / Creddit Project	avigate Code Befactor Run 1 busers) Entests) & test view busers) Entests) & test view okiecutter/reddt b b pp ates grations tts initopv factories pv test_admin.pv te	Iools \ ews.py) e and an all e all e	<pre>CS Window Help dels py x @ wits py x @ vlews py x @ test_vlews.py x @ common py x def setup(sit): sit.factory = RequestFactory() sit.factory = RequestFactory() def test_qet_redirect_vlew(fileselberTestCase): def test_qet_redirect_vlew(fileselberTestCase): def test_qet_redirect_vlew(fileselberTestCase): def test_qet_redirect_vlew(fileselberTestCase): def test_qet_redirect_vlew(fileselberTestCase): def test_qet_redirect_vlew(fileselberTestCase): def test_qet_redirect_vlew(fileselberTestCase): def setUp(setUp(fileselberTestCase): def setUp(setUp(fileselberTestCase): def setUp(setUp(fileselberTestCase): def setUp(setUp(fileselberTestCase): def setUp(setUp(fileselberTestCase)): def setUp(setUp(fileselberTestCase)): def setUp(setUp(fileselberTestCase)): def setUp(setUp(fileselberTestCase)): def setUp(setUp(fileselberTestCase)): def setUp(setUp(fileselberTestCase)): def setUp(setUp(fileselberTestCase)): def setUp(setUp(filesetUp(filesetUp(filesetUp(filesetUp(filesetUp(filesetUp(filesetUp(filesetUp(filesetUp(filesetUp(filesEUp</pre>	[@Docker.tests-al." ▶ ≪ ♥ ⊕ 事 ♥ ♥ ♥ ⊃ >	0 🔮 Database
Elle Edit Miew Bit reddit > Dreddit > Dreddit = /co Preddit > Dreddit = /co Preddit = /co Pr	avigate Code Befactor Run 1 Conserved to the set of th	Iools \ ews.py) emseted end end end end end end end end end e	CS Window Help delspy * @ urispy * @ viewspy * @ test_viewspy * @ common.py * delspy * @ urispy * @ viewspy * @ test_viewspy * @ common.py * def settue(set1): settates restrictive viewspy * @ test_viewspy * @ common.py * def settue(setties): def testue(setties): def setties): def setties: def setti	Image: Contract of the second of the sec	O Database
Elle Edit View bit reddit > Dreddit = /creddit B Project B Project P reddit = /creddit = /cre > D compas > D	avigate Code Befactor Run 1) Dusers) Dtests) & test view) Dusers) Dtests) & test view (Iools \ www.py) imm of ei	<pre>CS Window Help dels py x @ wins py x @ wiews py x @ test_views py x @ common py x def setip(set): def setip(set): sif.factory = RequestFactory() class TestUserRedirectView(BaseUserTestCase): def test_qet_redirect_url(set): set of test_get_redirect_url (reduit.users.tests.test_views.TestUserRedirectView) a finging the view dictory, where do this outside a test! def setip(set): set is the request to be view view.request = setif.setory.get('fide-url') # statch the user to be view view.request = request it issues(request): def setup(set): def setup(set)</pre>	[@Docker.tests-al ≥ ▶ ≪ № ⊕ ∓ ♥ ♥ ♥ > > >	O Patabase
Elle Edit View bit reddit > Dreddit = / Creddit Project	avigate Code Befactor Run 1 Conservent Code Befactor Run 1 Conservent Code Befactor Run 1 Code Befactor Run 1 Code Befactor Run 1 Code Befactor Run 1 Befactor Run 1 Be	Iools \ www.py> i in a i i i i i i i i i i i i i i i i i i i	<pre>CS Window Help dels py x @ wits py x @ views py x @ test_views.py x @ common py x def setup(set): sit.factory = RequestFactory() sit.factory = RequestFactory() class TestUserRedirectView(BaseUserTestCase): def test_eqt_redirect_view(CaseUserTestCase): view userRedirectView(CaseUserTestCase): view userRedirectView(CaseUserTestCase): view userRedirectView(CaseUserTestCase): view userRedirectView(CaseUserTestCase): view userRedirectView(CaseUserTestCase): view userRedirectView(CaseUserTestCase): view userRedirectView(CaseUserTestCase): view userRedirectView(CaseUserTestCase): ef satisfication with userTestCase(): ef satisfication the user user() subject(TestUserTestCase): def stil(sati): set(TestUserTestCase): def stil(satisfic the view of this outside a test/ sift.ster(); redirect(), forew of this outside a test/ sift.ster(); redirect(), forew of this outside a test/ sift.ster(); redirect(); redirect(), forew of this outside a test/ sift.ster(); redirect(); red</pre>	[@Docker.tests-al." ▶ 戦 № 乐 平 平 만 つ 、	0 Patabase
Elle Edit Miew Bit reddit > Dreddit > Dreddit = /cco Product > Dreddit > Dreddit = /cco Product > Dreddit > Dreddit = /cco Product > Dreddit > Dredd	avigate _Code Befactor Run _]) Dusers) Dt tests) & tests / & test / & e }) Dusers) Dt tests / & test / & e } o & e & e	Iools \ Www.py)	CS Window Help dedspy * @ urispy * @ viewspy * @ test_viewspy * @ common.py * def settue(set1): setf.factory = RequestFactory() class TestbarRedirective((usastbarTestCase): def test_views_factory(); forer do fils outside a test? * demarks informative information: * demarks informative information: * demarks informative information: * demarks informative information: * demarks informative: * demarks i	Image: Contract of the second of the sec	0 Database
Elle Edit View bit reddit > Dreddit > Dreddit = / Creddit B Project B Project P reddit = / Creddit = / Cre > D compose > D compos	avigate Code Befactor Run 1 b Dusors Ditexts & Cest Vie comments and the set of the se	Iools \ www.py) erei erei	<pre>CS Window Help dels py x @ wins py x @ wiews py x @ test_views py x @ common py x def setip(si1):</pre>	[@Docker.tests-al ≥ ▶ ≪ ♥ ⊕ 耳 ♥ ♥ ♥ ⊃	0 Database
Elle Edit Miew Isk reddit)∋ reddit → Endet Paredit → Compose Denset Paredit → Compose Paredit →	avigate Code Befactor Run 1 Conservery Entersty & Cest Viet Conservery Entersty & Cest Viet Conservery Entersty & Cest Viet Conservery Entersty & Cest Viet Conservery Cest Conservery Cest Cest Conservery Cest Cest Conservery Cest Cest Cest Conservery Cest C	Iools \ Yws.py))	<pre>CS Window Help dedspy x @ urispy x @ vewspy x @ test_vewspy x @ common.py x eff setU(set1): stif.factory = NegestBactory() stif.factory = NegestBactory() eff setup(setU(setUserTestLass)): eff setup(setUserTestLass): eff setUserTestLassTestLass): eff setUserTestLassTestLass(): eff setUserTestLassTestLass(): eff setUserTestLassTestLass(): eff setUserTestLassTestLass(): eff setUserTestLassTestLass(): eff setUserTestLassTest</pre>	Image: Docker.tests-al ▶ ● ● ● ● ● >	0 🖉 Database
Elle Edit View bit reddit) preddit Preddit) preddit Preddit / preddit Preddit / preddit Preddit / compare Preddit / com	avigate Code Befactor Run 2 Code Befactor Run 2 C	Tools X www.py) Rendered ef ef ef ef	CS Window Help dedspy * @urispy * @viewspy * @ test_viewspy * @ common py * def study(mil): stif.factory = RequestFactory() def study(mil): self:test_get_redirect_url (reddift.users.tests.test_views.TestUserRedirectWiew/ def study(mil): frequest.user self.testroy(ft/facurt') frequest.user self.testroy(ft/facurt') if encodes for the view diverty.test is the default username for if factoristicstestreet def setup(mil): frequest.estUser(MaetUer(TestCase): def setup(mil): frequest.estUser(MaetUer(TestCase): def setup(mil): frequest.estUser(MaetUer(TestCase): def setup(mil): frequest.estUser(MaetUer(TestCase): def setup(mil): frequest.estUser(TestCase): def setup(mi	Image: Docker: tests=alt ▶ 4% ● ● ■ ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ●	0 Database
Elle Edit View bit reddit > 0: reddit D'Project Projec	avigate Code Befactor Run 1 b conserved to the set of		CS Window Help delspy x & urspy x & vewspy x & test_vewspy x & common py x def setip(s(1): stif.factory = RequestFactory() stif.factory = RequestFactory = RequestFactory = RequestFactory() stif.factory = RequestFactory = RequestFacto	Image: Contract tests - all ▶ ● ● > <t< td=""><td>O" Database</td></t<>	O" Database
Elle Edit View Dit reddit) Dreddit () Preddit) Composi- Preddit) Composi- Preddit -/co Preddit -/co Predd	avigate Code Befactor Run 1 Conservent Code Befactor Run 1 Conservent Code Befactor Run 1 Code Befactor Run 1	Iools V www.py> Iome I I I Iome I Iome I Iome I Iome I Iome I Iome I Iome I Iome I Iome I Iome I Iome Iome	CS Window Help deds py * @ uris py * @ views py * @ test_views py * @ common py * def still (still: still factory = RequestFactory() effective redirective(Baselaertestcase): def set user are redirective(Baselaertestcase): def set(Baselaertestcase.set(D)): def set(Baselaertestcase.set(D)): set(Faselaertestcase.set	Image: Docker: tests - all ▶ ● ● □ > > 0 □ > > 0 □ > □ > □	O Database
Ele Edit View Dit reddit) Ereddit Breddit) Ereddit Preddit - (co Preddit -	avigate _Code _Befactor Run _2) Dusers) Dt tests) & tests) & tests (& test vie) Dusers) Dt tests) & test vie) Dusers) Dt tests) & test vie okiecutter/reddt b b pp atates arations ts 		CS Window Help ddispy * @ urfspy * @ vlewspy * @ test_vlewspy * @ common.py * def stip(cni); eff stip(cni);	Image: Docker.tests-all ▶ ● ● ● ● ><	O Patabase
Elle Edit View bit reddit > 0: reddit D'Project Projec	avigate Code Befactor Run 1 Conservery Entrests & Creat View Conservery Entrests & Creat View Conservery Entrests & Creat View Conservery Conserve	Iools V Bonnesser Iools Py Ioons	<pre>CS Window Help delspy x @ urispy x @ viewspy x @ test_viewspy x @ common.py x delspy x @ urispy x @ viewspy x @ test_viewspy x @ common.py x def setty_common_setty = sett_ase_user() sit_factory = RequestFactory() def test_active.trainet_user() sett_fact_act_redirect_usel (reddit_users.tests.test_views.TestUserRedirectView) # fortanizet the user of trainet user() # denotes a fact request # common py x = test_viewspow = denotes a test/ # denotes a fact request # def et[desUserRedrectView] test_get_redrect_ut(reddt.users.tests.test_views.TestUserRedrectView) # denotest fact request # def et[desUserRedrectView] test_get_redrect_ut(reddt.users.tests.test_views.TestUserRedrectView] # def et[desUserRedrectView] test_get_redrect_uter[def et[desUserRedrectView] # def et[desU</pre>	Image: Contract tests-al ▶ <td>O Database</td>	O Database
Elle Edit View Dit reddit) Dreddit Dreddit) Dreddit Pradit -/co Pradit -/co	avigate Code Befactor Run 1 Composed Code Befactor Run 1 Composed Code Befactor Run 1 Code Befactor Run 1 Composed Code Befactor Run 1 Code Befactor Run 1 C		CS Window Help delspy * @ urispy * @ viewspy * @ test_viewspy * @ common.py * def setUs(en1): setF.setUserPaires.setF.set@ user() setF.setTory = RequestTractory() def setup(en1): setF.setTory = RequestTractory() def setup(en1): setF.setUserPaires.setWindow(): def setup(ent): setF.setUserPaires.setWindow(): request.setF.setUserPaires(): def setup(ent): request.setF.setUserPaires(): def setup(ent): setF.setUserPaires(): def setUserPaires(): def setUserPaires(): setUserPaires(): def setUserPaires(): setUserPaires(): def setUserPaires(): def setUserPaires(): def setUserPaires(): def setUserPaires(): def setUserPaires(): def setUserPaires(): def setUserPaires(): def setUserPaires(): def setUserPaires(): setUserPaires(): def setUserPaires(): setUserPaires(): def setUserPaires(): setUserPaires(): setUserPaires(): def setUserPaires():	Image: Docker.tests-alt ▶ ● ■ ♥ ♥ ♥ ● > <td>O Database</td>	O Database
Elle Edit View Dit reddit) Dreddit Dreddit) Dreddit Dreddit) Dreddit Dreddit - /co Dreddit - /co Dredd	avigate Code Befactor Run 1 Dusres Ditests) Bit tests (Bit cest, Vie Dusres Ditests) Bit tests (Bit cest, Vie Cest) Bit tests (Bit cest, Vie Autor State		CS Window Help ddispy x @ urfspy x @ vlewspy x @ test_vlewspy x @ common py x def setig(m1): eff setig(m1): sif.factory = RequestFactory() class TestUserRedirectView(BaseUserTestCase): def test_get_redirect_url(bil): soff test_get_redirect_url (reddit.users.tests.test_views.TestUserRedirectView) # financial factory of (/fate-url') # denotes a fast request # denotes a fas	【型Docken.tests-all ▶ 戦 ゆ み ま ア マ ひ う	C Database
Elle Edit View Dit reddit) D:reddit Preddit) D:reddit Preddit) D:reddit Preddit) D:reddit Preddit / Co Preddit / Co P	avigate Code Befactor Run 1 Conserve Ditests Bitest Me Conserve Ditests Bitest Me Conserve Ditests Bitest Me Conserve Ditests Bitest Me Poly Bitest Me Bitest Admin Py test, models py te		<pre>CS Window Help delspy * @ urispy * @ viewspy * @ test_viewspy * @ common.py * delspy * @ urispy * @ viewspy * @ test_viewspy * @ common.py * delse results are solf-asia guer() silf-factory = RequestFactory() class TestUserRedirectView(BaseLaerTestCase); del test gueration & factory end (factory); f elserstate & factor view (factory); f elserstate & factory get(factor); f elserstate & fac</pre>	Image: Content tests-all ▶ ● ● ● ●	C Database
Elle Edit View 144 reddit) Dreddit reddit) Dreddit reddit) Dreddit reddit) Dreddit reddit re	avigate Code Befactor Run 1 Code Befactor Run 1 Console - Befactor Run 1 Con		CS Window Help ddsspy * @ uris py * @ viewspy * @ test_views py * @ common py * def setUp(sif): sif.factory = RequestFactory() def setup(sif): saff: test_get_redirect_url (reddit.users.tests.test_views.TestUserRedirectView) def test_get_rediret_url(saff): saff: test_get_redirect_url (reddit.users.tests.test_views.TestUserRedirectView) def test_get_rediret_url(saff): saff: test_get_redirect_url (reddit.users.tests.test_views.TestUserRedirectView) def test_get_rediret_url(saff): saff: test_get_redirect_url (reddit.users.tests.test_views.TestUserRedirectView) def test_get_rediret_url(saff): saff: test_get_redirect_url(saff): request_set_set_set_url(saff): def setup(saff): f constituent patients to the view if request_user = saff.tectory.get(rfateurl): f constituent patients (saff): def setup(saff): f constituent patients (saff): f constituent patients (saff): def setup(saff): f constituent patients (saff): f constituent	Image: Docker.tests-all ▶ ● ● ><	
Elle Edit View DA reddit) Dreddit Dreddit) Dreddit Dreddit) Dreddit Dreddit -/Co Dreddit -/C	avigate Code Befactor Run 1 Dusres Ditests) Bit tests (Bit cest, View Comparison of the set of th		<pre>CS Window Help dels py x @ wlews py x @ test_vlews py x @ common py x def setip(si1):</pre>	○ ● 上 ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ●	d 🔮 Database
Elle Edit View Dit reddit) D:reddit Pereddit) D:reddit Pereddit) D:reddit Pereddit) D:reddit Pereddit) D:reddit Pereddit -/co Pereddit -/co Pereeddit -/co Pereeddit -/co Pereeddit -/co Pereeddit -/co	avigate Code Befactor Run 1 Conservent Code Befactor Run 1 Conservent Code Befactor Run 1 Conservent Code Befactor Run 1 Code Befact Code Befactor Run 1 Code Befactor Run 1 C		<pre>CS Window Help dedspy * @ urispy * @ viewspy * @ test_viewspy * @ common py * def still (active) still factory = RequestFactory() class TestBerRedirectView(BestBerTestCase): def test quert endirectury(Legget redirect_url (reddit users tests test_views.TestBerRedirectView) # containing the view directly. Never do this outside a fest # containing the view directly. Never do this outside a fest # containing the view directly. Never do this outside a fest # containing the view directly. # containing the view dir</pre>	Image: Docker.tests-all ▶ ● ● ●<	C Database
Elle Edit View Dit reddit) Dreddit Dreddit) Dreddit Dreddit) Dreddit Dreddit () Dreddit () Dreddit Dreddit () Dreddit () Dreddi	avigate Code Befactor Run 2 Discreptions Discretified b b b b b c c c c c c c c c c c c c		CS Window Help ddssyy * @urfspy * @vlewspy * @ test_vlewspy * @common.py * def setUp(sil): sif.factory = RequestFactory() def setup(sil): sall.test_get_redirect_url (redit.users.tests.test_vlews.TestUserRedirectVlew) def set.get_redirectView(BastBartestrast): def setUserRedirectView(BastBartestrast): def setUserRedirectView(BastBartestrast): def setUserRedirectView(BastBartestrast): def setUserRedirectView(IsstBastBartestrast): def setUserRedirectView(IsstBastBartestrast): def setUserRedirectView(IsstBastBartestrast): def setUserRedirectView(IsstBastBartestrast): def setUserRedirectView(IsstBastBartest): def setUserRedirectView(Isst_get_redirect_url(reddit.users.tests.test_views.TestUserRedirectView) setF = {TestUserRedirectView}test_get_redirect_url(reddit.users.tests.test_views.TestUserRedirectView)	Image: Docker.tests-all ▶ ● ■ ● □ >	C Catabase
Elle Edit View Dit reddit) Dreddit Preddit) Dreddit Preddit) Dreddit Preddit -/co Preddit -/	avigate Code Befactor Run 1 Conservery Entrests & Creat View Conservery Entrests & Creat View Conservery Entrests & Creat View Conservery Conservery Conservery Conservery Conserve		CS Window Help dels py X @ wins py X @ wiews py X @ test_views py X @ common py X eff setig(si1): stif.factory = RequestFactory() class TestUserRedirectView(BaseUserTestCase): eff ist_car_redirect_url(si1): asif test_get_redirect_url (reddit.users.tests.test_views.TestUserRedirectView) # Simple Site Site Site Site Site Site Site Sit	● Let Less	Q Dechasse

• run and debug migrations or different django management commands

File Fqlt New Davidate Code Belactor Bin Tools ACZ Millight								
🖿 reddit areddit areddir ared	04_auto_20160327_2222.py 〉	🖾 Docker: migrate 🔹 🕨 🕷 🕸 🐺 🍟	v∰ 🕑 🖕 Q					
_ ⊡ Project • ③ ÷ ♦• !	🗟 models.py × 📓 0004_auto_20160327_2222.py ×		1					
<pre></pre>	0 = ** coding: utf-8 =*- 3 € Generate D Jango 1.9.4 on 2016-03-27 22:22 Ciport :		>					
2 v Ereddit	Class Migration(migrations.Higration):							
y = reddt v = contrib b = sites p = ntpy v = static b = fonts fonts fonts fonts fon	<pre>ef dependencies - [('users', ''04]_auto_20160227_2220'), orerations = [setimations = [setimations = [setimations', nase="mame", feld=models.churField(blank=True, max_length=255, verbose_mame="Mame of User"),),] </pre>							
bahua 🖽 Daskaa miasata			AL. 1					
Debug og Docker: migrace								
G Debugger 🖭 Console 🕂 😫 🛓 🖄 🖄								
Frames	📲 Variables	📲 🔐 Watches	→ *					
II MainThread • II MainThread • III MainThread • IIII MainThread • IIII MainThread • IIII MainThread • IIIII MainThread • IIIII MainThread • IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII		n alls, 'help': Type help() L. View + - + + () rations/_initpyc'> No watches	Event Log					
Tests Passed: 0 passed (vesterday 22:21)		8:1 LF UTF-8 Gib	master: a 🕀					

• and many others ..

5.2 Known issues

• Pycharm hangs on "Connecting to Debugger"

test_views.py		
1 Docker: runserver	Background Tasks	÷- ;
gubocker.numerver sugger E Console ~ NE 포 또 또 진 것 E newpr5_djanspython -u /opt/.pycharm_helpers/pydev/pydevd.py -=multiprocqt-supportclient 172.17.0.1port 51501file /app/manage.py runserver 0.0.0.0:8000 varming: Debugger speedups using cython not found. Run *-/ <u>usr/local/bin/python* //opt/.pycharm_helpers/pydev/setup_cython.py* bulld_ext</u> inplace* to bulld.	Background Tasks	\$2* ,
ts Passed: 0 passed (3 minutes ago)	€1 process running 1:1 LF÷ UTF-8 Git: master÷	ъ 👳 🛙

This might be fault of your firewall. Take a look on this ticket - https://youtrack.jetbrains.com/issue/PY-18913

• Modified files in .idea directory

Most of the files from *.idea*/ were added to *.gitignore* with a few exceptions, which were made, to provide "ready to go" configuration. After adding remote interpreter some of these files are altered by PyCharm:

In theory you can remove them from repository, but then, other people will lose a ability to initialize a project from provided configurations as you did. To get rid of this annoying state, you can run command:

\$ git update-index --assume-unchanged serpent_tracker.iml

SIX

INDICES AND TABLES

- genindex
- modindex
- search

INDEX

D

Docker,5

Ρ

pip,1 PostgreSQL,1

V

virtualenv,1