Start →

Django Series ────────→
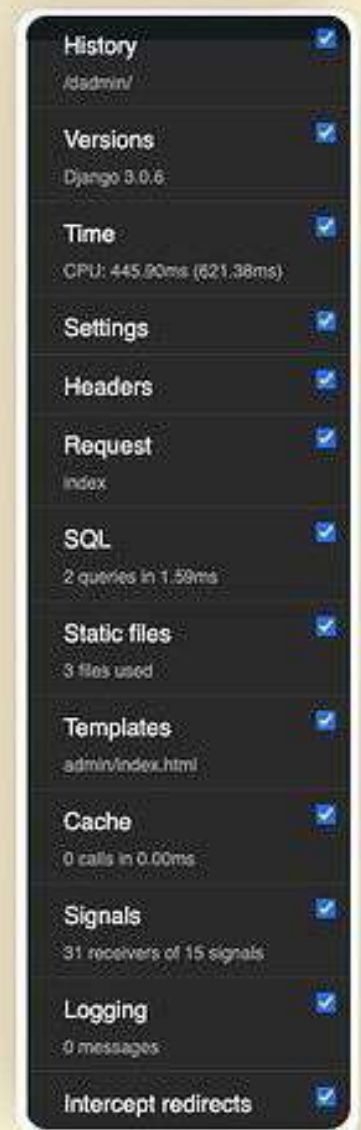
How To Create Debug Toolbar In Django

Start →

So, let's create a debug toolbar in django, that display various debug information about the current request/response and when clicked, display more details about the panel's content.

PYTHONWORLD

PYTHONWORLD

Here you will get these various option that you could use to debug various information

History /dadmin/	☑
Versions Django 3.0.6	☑
Time CPU: 445.90ms (621.38ms)	☑
Settings	☑
Headers	☑
Request index	☑
SQL 2 queries in 1.59ms	☑
Static files 3 files used	☑
Templates admin/index.html	☑
Cache 0 calls in 0.00ms	☑
Signals 31 receivers of 15 signals	☑
Logging 0 messages	☑
Intercept redirects	☑

PYTHONWORLD

Step - 01

Install This Package

You have to install **django-debug-toolbar** package in your system, by just hitting this command in Terminal

```
pip install django-debug-toolbar
```

Step - 02

Configure settings.py file

open settings.py file and include this package in
INSTALLED _ APPS

```python
INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'myapp.apps.MyappConfig',
    'debug_toolbar',
]
```

here include this

Add this middleware too

```python
MIDDLEWARE = [
    'debug_toolbar.middleware.DebugToolbarMiddleware',
]
```

Add this middleware

Step - 03

Just add this in settings.py file

```
INTERNAL_IPS = [

    '127.0.0.1',
]
```

just include this code in settings.py file

Step - 04

Open urls.py file and include this

```
from django.conf import settings        urls.py
import debug_toolbar

urlpatterns = [
    path('admin/', admin_site.urls
    path('__debug__/', include(debug_toolbar.urls)),
]
```

import this

just include this

Now, Just Open The Admin Panel, You will get the Debug Toolbar Option

Did You Find This Helpful ?

Let me know in the comment section

Aashish Kumar

IF IT DOSN'T CHALLENGE YOU
IT DOSN'T CHANGE YOU