How to track number of hits/views for particular objects in Django | Django Packages Series #2

#python   #django   #tutorial   #beginners

coderasha   Oct 7, 2019 *Updated on Oct 11, 2019* • 3 min read

By Reverse Python

What's up DEVs?

Welcome to the second post of Django Packages Series. In this quick tutorial we are going to learn how to track number of views/hits for specific objects.

Tracking is really useful functionality. When user enters your application, number of views can attract user to click specific posts and read it. Also it's good to show users most popular posts.

I saw some solutions on internet related with this topic. Some of developers using **F()** expression to handle view count:

```
Post.objects.filter(pk=post.pk).update(views=F('views') + 1)
```

Bad Approach!

By using F() and update views every time when user clicks on object is not good solution to create view counter. So, it will not count clicks properly, views will increase every time when same user clicks to same post.

Let me introduce you **django-hitcount**. Django-Hitcount allows you to track the number of hits (views) for a particular object. This isn't meant to be a full-fledged tracking application or a real analytic tool; it's just a basic hit counter.

Hitcounter will detect IPs and prevent from unreal views. So, views will count once for each specific user.

Create new Django project named "blog" and create app named "posts". Then run the following command to install the package.

```
pip3 install django-hitcount
```

Add django-hitcount to your INSTALLED_APPS:

settings.py

```
INSTALLED_APPS = (
    ...
    'hitcount'
)
```

There are several strategies for using django-hitcount but in this tutorial I will show you the best way to implement it in your app. Now, let's create our models:

models.py

```python
from django.db import models
from hitcount.models import HitCountMixin, HitCount
from django.contrib.contenttypes.fields import GenericRelation
from django.utils.encoding import python_2_unicode_compatible


@python_2_unicode_compatible
class Post(models.Model):
    title = models.CharField(max_length=100)
    description = models.TextField()
    published = models.DateField(auto_now_add=True)
    slug = models.SlugField(unique=True, max_length=100)
    hit_count_generic = GenericRelation(HitCount, object_id_field='object_pk',
      related_query_name='hit_count_generic_relation')


    def __str__(self):
        return self.title


    def save(self, *args, **kwargs):
        if not self.slug:
            self.slug = slugify(self.title)
        return super(Post, self).save(*args, **kwargs)
```

You are not required to do anything specific with your models; django-hitcount relies on a GenericForeignKey to create the relationship to your model's HitCount.

Then, let's create our views:

views.py

```python
from django.shortcuts import render
from django.views.generic.list import ListView
from hitcount.views import HitCountDetailView
from .models import Post


class PostListView(ListView):
    model = Post
    context_object_name = 'posts'
    template_name = 'post_list.html'



class PostDetailView(HitCountDetailView):
    model = Post
    template_name = 'post_detail.html'
    context_object_name = 'post'
    slug_field = 'slug'
    # set to True to count the hit
    count_hit = True

    def get_context_data(self, **kwargs):
        context = super(PostDetailView, self).get_context_data(**kwargs)
        context.update({
        'popular_posts': Post.objects.order_by('-hit_count_generic__hits')[:3],
        })
        return context
```

```
            return context
```

The **HitCountDetailView** can be used to do the business-logic of counting the hits by setting count_hit=True. Also we can filter the most viewed posts as shown above.

Now, let's configure our urls:

urls.py

```python
from django.contrib import admin
from django.urls import path, include
from posts.views import PostListView, PostDetailView


urlpatterns = [
    path('admin/', admin.site.urls),
    path('', PostListView.as_view(), name='posts'),
    path('<slug:slug>/', PostDetailView.as_view(), name='detail'),
    path('hitcount/', include(('hitcount.urls', 'hitcount'), namespace='hitcount')),
]
```

To display total views for related object we will use *get_hit_count* template tag.

post_list.html

```html
{% extends 'base.html' %}
{% load hitcount_tags %}
```

```
{% block content %}
  <h2>Posts List</h2>

  <ul>
    {% for post in posts %}
        <h3 class="mt-5"><a href='{% url "detail" post.slug %}'>{{post.title}}</a></h3>
        <p class="lead">{{post.description}}</p>
        <p>{{post.published}}</p>
        <p>Views: {% get_hit_count for post %}</p>
    {% endfor %}
  </ul>
{% endblock %}
```

post_detail.html

```
{% extends 'base.html' %}
{% load hitcount_tags %}

{% block content %}
  <h2>{{post.title}}</h2>
  <ul>
        <p class="lead">{{post.description}}</p>
        <p>Published: {{post.published}}</p>
        <p>Views: {% get_hit_count for post %}</p>
  </ul>

  <h3>Popular Posts</h3>
  {% for p in popular_posts %}
  <p>{{p.title}}</p>
```

```
    {% endfor %}
{% endblock %}
```

You can download or clone the project on my GitHub

○ **raszidzie** / **Django-Hitcount-Tutorial**

Simple Blog App using django-hitcount package

Django-Hitcount-Tutorial

Simple Blog App using django-hitcount package

Get Started

Install dependencies by following commad:

```
pip3 install requirements.txt
```

and run the project

```
python3 manage.py makemigrations posts
python3 manage.py migrate
```

View on GitHub

That's it! Make sure you are following me on social media and please support me by buying me a

That's it! Make sure you are following me on social media and please support me by buying me a coffee☕ so I can upload more tutorials like this. See you in next post DEVs!

Reverse Python
Instagram
Twitter

Posted on Oct 6 '19 by:

coderasha
@coderasha

Mainly focused on Python but love React also. Currently learning Data Science and ML.

Follow

Discussion

Subscribe

Add to the discussion

PREVIEW   SUBMIT

▼

Muhammad Sajjad 🐙   Jun 20 •••

After python mange.py migrate
i get this error
_mysql.connection.query(self, query)
django.db.utils.OperationalError: (1170, "BLOB/TEXT column 'object_pk' used in key specification without a key length")

♡ 1   REPLY

▼

Mahmudul Hassan 🐙   Jul 24 •••

Did you solved this error??

♡ 1   REPLY

▼

Beka 🐙   Jun 18 •••

How I can get the number views of one object in django shell?
I am getting error that says:
Traceback (most recent call last):
File "", line 1, in
AttributeError: 'GenericRelatedObjectManager' object has no attribute 'hits'

for this error the query is: john_wick.hit_count_generic.hits

john_wick is Movie instance. And my model is

class Movie(models.Model):
title = models.CharField(max_length=100)
description = models.TextField(max_length=2000)
image = models.ImageField(upload_to='movie_upload', null=False, blank=False)
year_of_production = models.IntegerField()
iframe = models.CharField(max_length=1500, verbose_name='iframe src')
hit_count_generic = GenericRelation(HitCount, object_id_field='object_pk',
related_query_name='hit_count_generic_relation')
created_at = models.DateTimeField(auto_now_add=True)

Please help me figure out.

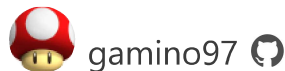♡ 1   REPLY

▼

                                                                                                                                            May 21 •••
    bos1313 ○

Great tutorial and it works fin with Sqlite database but when I try to use it with MySQL database I got error when
do migration
150 "Foreign key constraint is incorrectly formed")')
I use user field as well
user = models.ForeignKey(settings.AUTH_USER_MODEL, default=1, on_delete=models.CASCADE)

♡ 1   REPLY

▼

                                                                                                                                             Jul 25 •••
    gamino97 ○

For those that have the error "(errno: 150 "Foreign key constraint is incorrectly formed")')", I've found an improvise solution, simply go to the installation folder of django-hitcount named "hitcount", go to the

migrations folder, and erase everything except the **init**.py, run python manage.py makemigrations and python manage.py migrate and it's solved, sorry my bad English

♡ 1
REPLY

▼

Billy Okeyo 🐙
Apr 29 •••

Can't i run it without python 2

♡ 2
REPLY

▼

Crow Pigeon
Jul 19 •••

run only in Django 2x

♡ 1
REPLY

▼

cavinblake 🐙
Mar 8 •••

hi, will it work with the autoplay videos. Can i work with it..if a video autoplays the view will be counted for that post

♡ 1
REPLY

▼

CodeSnail (Arpit) ○   Sep 3 •••

Ok worked!! but how to test it. Initially showing 0. I opened my post but count not increase

♡ 1   REPLY

Code of Conduct • Report abuse

Read next

🖼️ 🚀 React Component Patterns
Alexi Taylor 👾 - Aug 26

🎉 32 Code Tutorials for Visual Learners 🖼️
ender minyard - Aug 31

Studying a programming language: tips from a former language teacher
Madeline - Sep 16

"CSS Flexbox Cheatsheet" v3
Dzhavat Ushev - Sep 9

More from coderasha

I made a bot that can solve multiple choice question from the picture

#python   #productivity   #career

Scraping data with Selenium (Python) | Upwork Series #2

#python   #database   #webdev   #tutorial

Web Scraper & Data Extraction with Python | Upwork Series #1

#python   #webdev   #database   #tutorial

DEV

A constructive and inclusive social network. Open source and radically transparent.

DEV Community copyright 2016 - 2020

Built on Forem — the open source software that powers DEV and other inclusive communities.

Made with love and Ruby on Rails.

Home

Listings

Podcasts

Videos

Tags

Sign In/Up

Code of Conduct

FAQ

About

Privacy policy

Terms of use

Contact

Sponsors

DEV Shop