Change Object Display Name using __str__ function – Django Models | Python

Last Updated : 13 Feb, 2020

How to Change Display Name of an Object in Django admin interface? Whenever an instance of model is created in Django, it displays the object as **ModelName Object(1)**. This article will explore how to make changes to your Django model using `def __str__(self)` to change the display name in the model.

Object Display Name in Django Models

Consider a project named `geeksforgeeks` having an app named `geeks`.

> *Refer to the following articles to check how to create a project and an app in Django.*

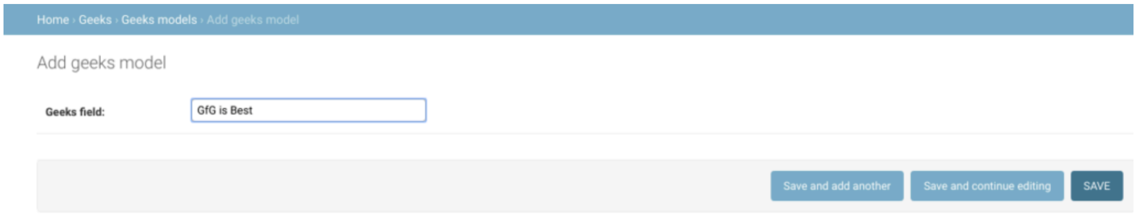
- *How to Create a Basic Project using MVT in Django?*
- *How to Create an App in Django ?*

Enter the following code into `models.py` file of **geeks** app.

```
from django.db import models
from django.db.models import Model
# Create your models here.

class GeeksModel(Model):
    geeks_field = models.CharField(max_length = 200)
```

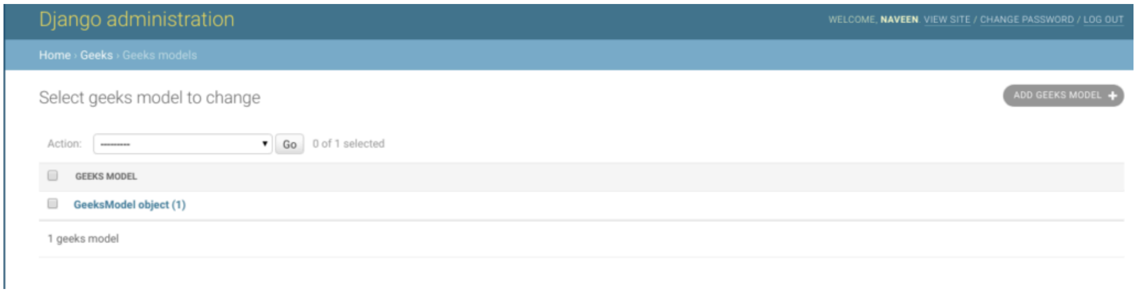
After running makemigrations and migrate on Django and rendering above model, let us try to create an instance using string "**GfG is Best**".


Home › Geeks › Geeks models › Add geeks model
Save and add another
Save and continue editing
SAVE

Add geeks model

Geeks field: GfG is Best

Now it will display the same object created as **GeeksModel object(1)** in the admin interface.


Django administration
WELCOME, NAVEEN. VIEW SITE / CHANGE PASSWORD / LOG OUT
Home › Geeks › Geeks models
ADD GEEKS MODEL +
Go

Select geeks model to change

Action: --------- 0 of 1 selected

GEEKS MODEL

GeeksModel object (1)

1 geeks model

How to change display name of Model Instances in Django admin interface?

To change the display name we will use `def __str__()` function in a model. **str** function in a django model returns a string that is exactly rendered as the display name of instances for that model.

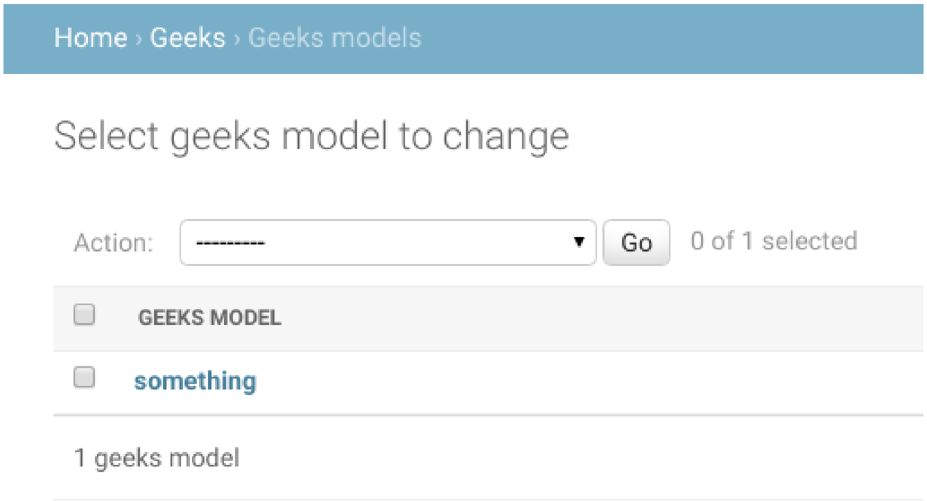
For example, if we change above models.py to

```
from django.db import models
from django.db.models import Model
# Create your models here.

class GeeksModel(Model):
    geeks_field = models.CharField(max_length = 200)

    def __str__(self):
         return "something"
```

This will display the objects as **something** always in the admin interface.



Home › Geeks › Geeks models

Select geeks model to change

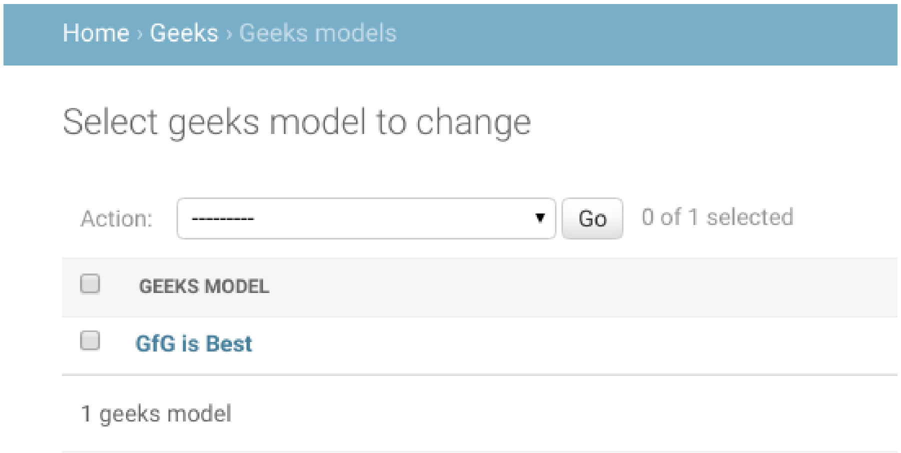
Action: --------- ▾ Go 0 of 1 selected

□ GEEKS MODEL

□ something

1 geeks model

Most of the time we name the display name which one could understand using `self` object. For example `return self.geeks_field` in above function will return the string stored in geeks_field of that object and it will be displayed in admin interface.



Home › Geeks › Geeks models

Select geeks model to change

Action: --------- ▾ Go 0 of 1 selected

□ GEEKS MODEL

□ GfG is Best

1 geeks model

.

Note that **str** function always returns a string and one can modify that according to one’s convenience.

Also Check –

- Render Model in Django Admin Interface
- Built-in Field Validations – Django Models

N Nave...

Previous Article

Python | Uploading images in Django

Next Article

Custom Field Validations in Django Models

Similar Reads

How to Fix Django "ImportError: Cannot Import Name 'six' from...

After Django 3.0, django.utils.six was removed. The "ImportError: Cannot Import Name 'six' from 'django.utils'" error occurs because of the dependency issue. Thi...

3 min read

Change the Name of a Django Application

Changing the name of a Django app might be necessary for various reasons, such as rebranding, restructuring, or simply improving the clarity of your project's...

3 min read

How to Change Field Name in Django REST Framework Serializer

When working with Django REST Framework (DRF), we may encounter situations where we need to change the name of a field in a serializer. This can ...

3 min read

Creating multiple user types and using proxy models in Python Django

In this article, we will understand the concept of the proxy model and the implementation of multiple user types depending on the requirements of your...

9 min read

How to Add created_at and updated_at Fields to All Django Models Using...

In Django applications, tracking when records are created and last updated in the database is common. Two of the most frequently used fields for this purpose are...

4 min read

Using UUID As The Primary Key In Django Models

When building Django applications, we often rely on the auto-incrementing primary key (id) generated by the database. However, there are cases where...

4 min read

Python | Relational fields in Django models

Prerequisite: Django models Django models represent real-world entities, and it is rarely the case that real-world entities are entirely independent of each other....

4 min read

GUI application to search a country name from a given state or city name…

In these articles, we are going to write python scripts to search a country from a given state or city name and bind it with the GUI application. We will be using t...

2 min read

Django Models | Set - 1

Prerequisites : Django Creating apps Models - According to Django Models, A model is the single, definitive source of information about your data. It contains...

2 min read

Django Models | Set - 2

Model Fields - Model fields define the datatype which will be stored in the provided variable. To store price, the integer type is more suitable. To store heig...

2 min read

Article Tags : Python Django-models Python Django

Practice Tags : python

GeeksforGeeks
Sanchhaya Education Private Limited

Corporate & Communications Address:-
A-143, 9th Floor, Sovereign Corporate
Tower, Sector- 136, Noida, Uttar Pradesh
(201305) | Registered Address:- K 061,

Tower K, Gulshan Vivante Apartment,
Sector 137, Noida, Gautam Buddh
Nagar, Uttar Pradesh, 201305

GET IT ON Google Play

Download on the App Store

Company

- About Us
- Legal
- In Media
- Contact Us
- Advertise with us
- GFG Corporate Solution
- Placement Training Program
- GeeksforGeeks Community

Languages

- Python
- Java
- C++
- PHP
- GoLang
- SQL
- R Language
- Android Tutorial
- Tutorials Archive

DSA

- Data Structures
- Algorithms
- DSA for Beginners
- Basic DSA Problems
- DSA Roadmap
- Top 100 DSA Interview Problems
- DSA Roadmap by Sandeep Jain
- All Cheat Sheets

Data Science & ML

- Data Science With Python
- Data Science For Beginner
- Machine Learning
- ML Maths
- Data Visualisation
- Pandas
- NumPy
- NLP
- Deep Learning

Web Technologies

- HTML
- CSS
- JavaScript
- TypeScript
- ReactJS
- NextJS
- Bootstrap
- Web Design

Python Tutorial

- Python Programming Examples
- Python Projects
- Python Tkinter
- Web Scraping
- OpenCV Tutorial
- Python Interview Question
- Django

Computer Science

- Operating Systems
- Computer Network
- Database Management System
- Software Engineering
- Digital Logic Design

DevOps

- Git
- Linux
- AWS
- Docker
- Kubernetes

Engineering Maths
Software Development
Software Testing

Azure
GCP
DevOps Roadmap

System Design
High Level Design
Low Level Design
UML Diagrams
Interview Guide
Design Patterns
OOAD
System Design Bootcamp
Interview Questions

Inteview Preparation
Competitive Programming
Top DS or Algo for CP
Company-Wise Recruitment Process
Company-Wise Preparation
Aptitude Preparation
Puzzles

School Subjects
Mathematics
Physics
Chemistry
Biology
Social Science
English Grammar
Commerce
World GK

GeeksforGeeks Videos
DSA
Python
Java
C++
Web Development
Data Science
CS Subjects