



Django Models | Set – 1

Last Updated : 21 Jun, 2019

Prerequisites : [Django Creating apps](#)

Models –

According to [Django Models](#), A model is the single, definitive source of information about your data. It contains the essential fields and behaviors of the data you're storing. Generally, each model maps to a single database table.

The basics:

- Each model is a Python class that subclasses `django.db.models.Model`.
- Each attribute of the model represents a database field.
- With all of this, Django gives you an automatically-generated database-access API.

Django models are used as a structure to define fields and their types which will be saved in the database. Whatever changes we want to make in the database and want to store them in the database permanently are done using Django Models. A table for a phone in the database can be imagined as:

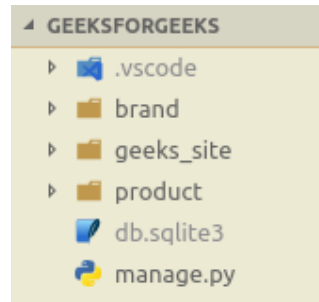
Phone	
Attribute	Value
Price	9999/-
Brand	Yiaomi
RAM	3GB
ROM	32GB
Camera	20MP Front 22MP Rear
Battery	4000 MaH
Color	Black
Quantity Available	100

Create table in database using Django –

We need to create a new app named **product** so that we can define all properties of phone described in the above-provided image. Open your terminal and run following command:

```
python manage.py startapp product
```

Now, our directory will be:



After creating the app, don't forget to mention it in `geeks_site/settings.py` under **INSTALLED_APPS**.

```
INSTALLED_APPS = [  
    'django.contrib.admin',  
    'django.contrib.auth',  
    'django.contrib.contenttypes',  
    'django.contrib.sessions',  
    'django.contrib.messages',  
    'django.contrib.staticfiles',  
  
    'brand',  
    'product',  
]
```

Also, register it with admin by adding following line of code to **product/admin.py**. By registering it with admin, you make sure that admin of the site will be aware that new table schema for database has been prepared.

```
from django.contrib import admin  
  
# Register your models here.  
from product.models import Phone  
  
admin.site.register(Phone)
```

Now, navigate to **product/models.py**. You will see a file with following lines:

```
from django.db import models

# Create your models here.
```

We use a python class to define models which inherit parent class named **Model** defined in **django.db.models** package.

Add the following lines of code to this file:

```
from django.db import models

# Create your models here.
class Phone(models.Model):
    Price = models.IntegerField()
    RAM = models.IntegerField()
    ROM = models.IntegerField()
    Front_camera = models.IntegerField()
    Rear_camera = models.IntegerField()
    Battery = models.IntegerField()
    Screen_size = models.DecimalField(max_digits = 2, decimal_places = 1)
    Color = models.CharField(max_length = 120)
    Quantity = models.IntegerField()
```

Are you ready to elevate your web development skills from foundational knowledge to advanced expertise? Explore our [Mastering Django Framework - Beginner to Advanced Course](#) on GeeksforGeeks, designed for aspiring developers and experienced programmers. This comprehensive course covers everything you need to know about Django, from the basics to advanced features. Gain practical experience through **hands-on projects** and real-world applications, mastering essential Django principles and techniques. Whether you're just starting or looking to refine your skills, this course will empower you to build sophisticated web applications efficiently. Ready to enhance your web development journey? Enroll now and unlock your potential with Django!



anku...



Previous Article

IntegerField - Django Models

Next Article

Similar Reads

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

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

BigIntegerField - Django Models

BigIntegerField is a 64-bit integer, much like an IntegerField except that it is guaranteed to fit numbers from -9223372036854775808 to...

4 min read

AutoField - Django Models

According to documentation, An AutoField is an IntegerField that automatically increments according to available IDs. One usually won't need to use this directl...

4 min read

BigAutoField - Django Models

BigAutoField is a 64-bit integer, much like an AutoField except that it is guaranteed to fit numbers from 1 to 9223372036854775807. One can create a...

4 min read

BooleanField - Django Models

BooleanField is a true/false field. It is like a bool field in C/C++. The default form widget for this field is CheckboxInput, or NullBooleanSelect if null=True. The...

4 min read

CharField - Django Models

CharField is a string field, for small- to large-sized strings. It is like a string field in C/C++. CharField is generally used for storing small strings like first name, last...

4 min read

DateField - Django Models

DateField is a field that stores date, represented in Python by a datetime.date instance. As the name suggests, this field is used to store an object of date...

5 min read

DateTimeField - Django Models

DateTimeField is a date and time field which stores date, represented in Python by a datetime.datetime instance. As the name suggests, this field is used to stor...

5 min read

DecimalField - Django Models

DecimalField is a field which stores a fixed-precision decimal number, represented in Python by a Decimal instance. It validates the input using...

4 min read

Article Tags : [Python](#) [Python Django](#)

Practice Tags : [python](#)



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



Company

[About Us](#)
[Legal](#)
[In Media](#)
[Contact Us](#)
[Advertise with us](#)
[GFG Corporate Solution](#)
[Placement Training Program](#)
[GeeksforGeeks Community](#)

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](#)

Web Technologies

[HTML](#)
[CSS](#)
[JavaScript](#)
[TypeScript](#)
[ReactJS](#)
[NextJS](#)
[Bootstrap](#)
[Web Design](#)

Languages

[Python](#)
[Java](#)
[C++](#)
[PHP](#)
[GoLang](#)
[SQL](#)
[R Language](#)
[Android Tutorial](#)
[Tutorials Archive](#)

Data Science & ML

[Data Science With Python](#)
[Data Science For Beginner](#)
[Machine Learning](#)
[ML Maths](#)
[Data Visualisation](#)
[Pandas](#)
[NumPy](#)
[NLP](#)
[Deep Learning](#)

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
Engineering Maths
Software Development
Software Testing

System Design

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

School Subjects

Mathematics
Physics
Chemistry
Biology
Social Science
English Grammar
Commerce
World GK

DevOps

Git
Linux
AWS
Docker
Kubernetes
Azure
GCP
DevOps Roadmap

Interview Preparation

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

GeeksforGeeks Videos

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

@GeeksforGeeks, Sanchhaya Education Private Limited, All rights reserved