Django model data types and fields list

Last Updated : 04 Jan, 2022

The most important part of a model and the only required part of a model is the list of database fields it defines. Fields are specified by class attributes. Be careful not to choose field names that conflict with the models API like clean, save, or delete.

Example:

```
from django.db import models

class Musician(models.Model):
    first_name = models.CharField(max_length=200)
    last_name = models.CharField(max_length=200)
    instrument = models.CharField(max_length=200)

class Album(models.Model):
    artist = models.ForeignKey(Musician, on_delete=models.CASCADE)
    name = models.CharField(max_length=100)
    release_date = models.DateField()
    num_stars = models.IntegerField()
```

Setting a field for storing any type of data is like deciding a data type in C/C++ for storing a particular integer, char etc. Fields in Django are the data types to store a particular type of data. For example, to store an integer, IntegerField would be used. These fields have in-built validation for a particular data type, that is you can not store “abc” in an IntegerField. Similarly, for other fields. This post revolves around major fields one can use in Django Models.

Here are some key attributes one should be aware of before starting to use Django Fields.

Field types

Each field in the model should be an instance of the appropriate Field class. Django uses field class types to determine a few things:

- The column type, which tells the database what kind of data to store (e.g. INTEGER, VARCHAR, TEXT).
- The default HTML widget to use when rendering a form field (e.g. <input

Django Views Model Template Forms Jinja Python SQLite Flask Json Postman Interview Ques

- The minimal validation requirements, used in Django's admin and in automatically-generated forms.

Django ships with dozens of built-in field types which can be used to save any type of data from number to entire HTML file too. Here is a list of all Field types used in Django.

Basic model data types and fields list

Field Name	Description
AutoField	It is an IntegerField that automatically increments.
BigAutoField	It is a 64-bit integer, much like an AutoField except that it is guaranteed to fit numbers from 1 to 9223372036854775807.
BigIntegerField	It is a 64-bit integer, much like an IntegerField except that it is guaranteed to fit numbers from -9223372036854775808 to 9223372036854775807.
BinaryField	A field to store raw binary data.
BooleanField	A true/false field. The default form widget for this field is a CheckboxInput.
CharField	A field to store text-based values.

Field Name	Description
DateField	A date, represented in Python by a datetime.date instance
DateTimeField	It is used for date and time, represented in Python by a datetime.datetime instance.
DecimalField	It is a fixed-precision decimal number, represented in Python by a Decimal instance.
DurationField	A field for storing periods of time.
EmailField	It is a CharField that checks that the value is a valid email address.
FileField	It is a file-upload field.
FloatField	It is a floating-point number represented in Python by a float instance.
ImageField	It inherits all attributes and methods from FileField, but also validates that the uploaded object is a valid image.
IntegerField	It is an integer field. Values from -2147483648 to 2147483647 are safe in all databases supported by Django.
GenericIPAddressField	An IPv4 or IPv6 address, in string format (e.g. 192.0.2.30 or 2a02:42fe::4).
NullBooleanField	Like a BooleanField, but allows NULL as one of the options.
PositiveIntegerField	Like an IntegerField, but must be either positive or zero (0).

Field Name	Description
PositiveSmallIntegerField	Like a PositiveIntegerField, but only allows values under a certain (database-dependent) point.
SlugField	Slug is a newspaper term. A slug is a short label for something, containing only letters, numbers, underscores or hyphens. They're generally used in URLs.
SmallIntegerField	It is like an IntegerField, but only allows values under a certain (database-dependent) point.
TextField	A large text field. The default form widget for this field is a Textarea.
TimeField	A time, represented in Python by a datetime.time instance.
URLField	A CharField for a URL, validated by URLValidator.
UUIDField	A field for storing universally unique identifiers. Uses Python's UUID class. When used on PostgreSQL, this stores in a uuid datatype, otherwise in a char(32).

Relationship Fields

Django also defines a set of fields that represent relations.

Field Name	Description
ForeignKey	A many-to-one relationship. Requires two positional arguments: the class to which the model is related and the on_delete option.

Field Name	Description
ManyToManyField	A many-to-many relationship. Requires a positional argument: the class to which the model is related, which works exactly the same as it does for ForeignKey, including recursive and lazy relationships.
OneToOneField	A one-to-one relationship. Conceptually, this is similar to a ForeignKey with unique=True, but the “reverse” side of the relation will directly return a single object.

N Nave...

Previous Article

Django Models

Next Article

Built-in Field Validations - Django Models

Similar Reads

How to Get a List of the Fields in a Django Model

When working with Django models, it's often necessary to access the fields of a model dynamically. For instance, we might want to loop through the fields or...

2 min read

Django Form | Data Types and Fields

When gathering user information to store in a database, we employ Django forms. Django offers a variety of model field forms for different uses, and these...

6 min read

Django Dynamic Model Fields

In web development, Django is a framework that helps to build awesome websites. One of the main features of Django is dynamic model fields so we will...

5 min read

Convert Django Model Object to Dict with all of the Fields Intact

Django, a high-level Python web framework, simplifies the process of building web applications by providing a robust ORM (Object-Relational Mapping) syste...

4 min read

Include Related Model Fields In Django Rest Framework

In Django Rest Framework (DRF), it is common to work with related models, especially when building APIs that expose complex data structures. We might...

3 min read

Date and time fields in serializers - Django REST Framework

In Django REST Framework the very concept of Serializing is to convert DB data to a datatype that can be used by javascript. Every serializer comes with some...

7 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

Intermediate fields in Django | Python

Prerequisite: Django models, Relational fields in DjangoIn Django, a many-to-many relationship exists between two models A and B, when one instance of A ...

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

Render Django Form Fields Manually

Django form fields have several built-in methods to ease the work of the developer but sometimes one needs to implement things manually for...

5 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

Company

Languages

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

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
Engineering Maths
Software Development
Software Testing

DevOps

Git
Linux
AWS
Docker
Kubernetes
Azure
GCP
DevOps Roadmap

System Design

High Level Design
Low Level Design
UML Diagrams
Interview Guide
Design Patterns
OOAD

Inteview Preparation

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

System Design Bootcamp

Interview Questions

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