## 1. What Are Models in Django?

Models in Django are Python classes that represent database tables. Each attribute of the model class represents a column, and each instance represents a row.

**Example:**

from django.db import models

class Student(models.Model):

name = models.CharField(max\_length=100)

age = models.IntegerField()

enrolled = models.BooleanField(default=True)

## 2. What is ORM (Object-Relational Mapping)?

ORM allows developers to interact with the database using Python code instead of SQL queries. It improves safety, portability, and readability of database operations.

**Example:**

-- Without ORM

SELECT \* FROM student WHERE age > 18;

# With ORM

Student.objects.filter(age\_\_gt=18)

## 3. Integrating MySQL with Django

**Install MySQL client:**

pip install mysqlclient

**Configure** settings.py**:**

DATABASES = {

'default': {

'ENGINE': 'django.db.backends.mysql',

'NAME': 'school\_db',

'USER': 'root',

'PASSWORD': 'your\_password',

'HOST': 'localhost',

'PORT': '3306',

}

}

**Apply migrations:**

python manage.py makemigrations

python manage.py migrate

## 4. Sending Model Records as JSON Without Serializers

You can send Django ORM query results as JSON directly using JsonResponse and values().

**Multiple Records Example:**

from django.http import JsonResponsefrom .models import Student

def get\_students(request):

students = list(Student.objects.values())

return JsonResponse({'students': students})

**Single Record Example:**

def get\_single\_student(request, id):

student = Student.objects.filter(id=id).values().first()

if student:

return JsonResponse(student, safe=False)

else:

return JsonResponse({'error': 'Student not found'}, status=404)

**Create Record Example:**

from django.views.decorators.csrf import csrf\_exemptimport jsonfrom .models import Student

@csrf\_exemptdef create\_student(request):

if request.method == 'POST':

data = json.loads(request.body)

name = data.get('name')

age = data.get('age')

enrolled = data.get('enrolled', True)

student = Student.objects.create(name=name, age=age, enrolled=enrolled)

return JsonResponse({'message': 'Student created', 'id': student.id})

## 5. Summary

**Models** define your database structure.

**ORM** makes database operations easier and safer.

**MySQL integration** connects Django with real-world databases.

**JsonResponse** allows sending data as JSON without serializers.