Django ORM — Very Simple Explanation

Step 1: Start With This Question

Ask students:

“Where do we store app data?”

Answer:
□ In a **database**

A database has:

- Tables
- Rows
- Columns

Like an Excel sheet.

Step 2: What is ORM?

Say this:

Django ORM lets us use **Python instead of SQL** to talk to the database.

That’s it.

No big definitions needed.

Step 3: One Simple Example

Tell them:

□ “In Django, a Python class becomes a database table.”

Example:

```
from django.db import models

class Student(models.Model):
    name = models.CharField(max_length=100)
    age = models.IntegerField()
```

Now explain in very simple terms:

- `Student` → Table
- `name` → Column
- `age` → Column
- Each student → Row

That’s ORM.

Step 4: Show Magic (This Excites Beginners)

After:

```
python manage.py makemigrations
python manage.py migrate
```

Tell them:

□ “Django just created a real database table from your Python code.”

That’s the magic.

Step 5: Show Basic Actions (Very Simple)

➜ Add Data

```
Student.objects.create(name="Ali", age=20)
```

Say:
□ This adds one row.

➜ Get Data

```
Student.objects.all()
```

Say:
□ This shows all rows.

➜ Change Data

```
s = Student.objects.get(id=1)
s.age = 25
s.save()
```

Say:
□ This updates a row.

➜ Delete Data

```
s.delete()
```

Say:
□ This removes a row.

Simple Way to Explain ORM

Use this one sentence:

ORM = “Write Python, Django handles the database.”

Or even simpler:

Python class → Database table