Practical Question: Sales Management

Create a Django application named "Sales Management" where users can add sales records and view sales reports.

1. Model

Use the same `Sale` model as before:

- product_name → CharField(max_length=200)
- category → CharField(max_length=100)
- unit_price → DecimalField(max_digits=10, decimal_places=2)
- quantity → PositiveIntegerField()
- discount_percent → DecimalField(max_digits=5, decimal_places=2, default=0)
- tax_percent → DecimalField(max_digits=5, decimal_places=2, default=0)
- total_price → DecimalField(max_digits=12, decimal_places=2, blank=True, null=True)
- sale_date → DateField(auto_now_add=True)

$$total\_price = (unit\_price \times quantity) - \left((unit\_price \times quantity) \times \frac{discount\_percent}{100}\right) + \left((unit\_price \times quantity) \times \frac{tax\_percen}{100}\right.$$

2. Templates

- add_sale.html → Form to create a sale.
- sale_list.html → Table to list all sales with an **Edit button** next to each record.

3. Views & URLs (No Django Forms)

- **Add Sale (Create):**
  - User fills an HTML form.
  - On submission, total_price is **calculated manually in views.py** and saved
- **View Sales (Read):**
  - Show all sales in a table.
  - **Each row shows:** Product, Category, Unit Price, Quantity, Discount %, Tax %, Total Price, Date.

Product	Category	Unit Price	Quantity	Discount %	Tax %	Total Price	Date
Laptop	Electronics	1000.00	2	10	5	1900.00	2025-09-23
Phone	Electronics	500.00	3	5	10	1575.00	2025-09-23