Smart E-Commerce Sales Management System

Leveraging Data Engineering & Analytics for Smarter E-Commerce Decisions

Agenda

1 **Project Overview**
Goals, scope, and outcomes

2 **Database Objectives & Requirements**
Integrity, security, and performance

3 **Database Design**
Mapping & ERD architecture

4 **SQL Implementation**
Tables, procedures, and functions

5 **Data Import & Cleaning**
Quality assurance process

6 **Exploratory Data Analysis**
Key business insights

7 **Conclusion**
Implementation roadmap

This presentation follows a pyramid structure with conclusion-first approach, focusing on business impact and actionable insights.

Project Overview

Goal

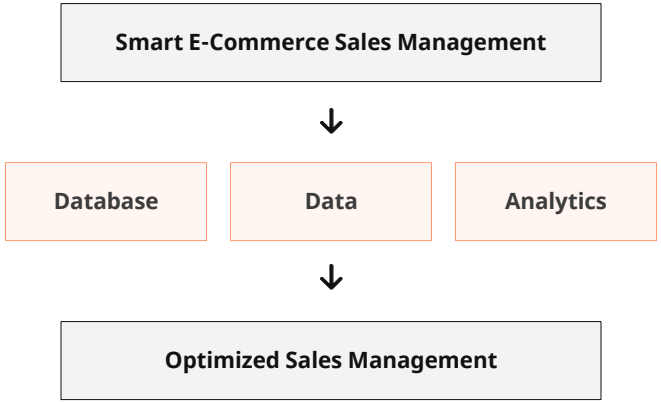
Develop a robust, scalable system to manage e-commerce sales efficiently.

Scope

- Database design & implementation
- Data cleaning & processing
- Analytical insights for decision-making

Outcome

Centralized, accurate, and optimized database for sales management.



A well-designed database system enables 30-40% faster decision-making through integrated data management.

Objectives

- **Comprehensive Data Storage**

  Store and manage customer, product, order, payment, and shipping data

- **Data Integrity & Security**

  Ensure accuracy, consistency, and protection of sensitive information

- **Efficient Querying**

  Enable fast and optimized data retrieval for reporting and analysis

Requirements

- **Normalization (up to 3NF)**

  Eliminate redundancy and ensure data consistency

- **Constraints**

  Primary keys, foreign keys, CHECK, and UNIQUE constraints

- **Performance Optimization**

  Strategic indexing and query optimization

- **Historical Data Support**

  Maintain transaction history for trend analysis

> Properly implemented database constraints can reduce data errors by up to 90%, significantly improving operational efficiency.

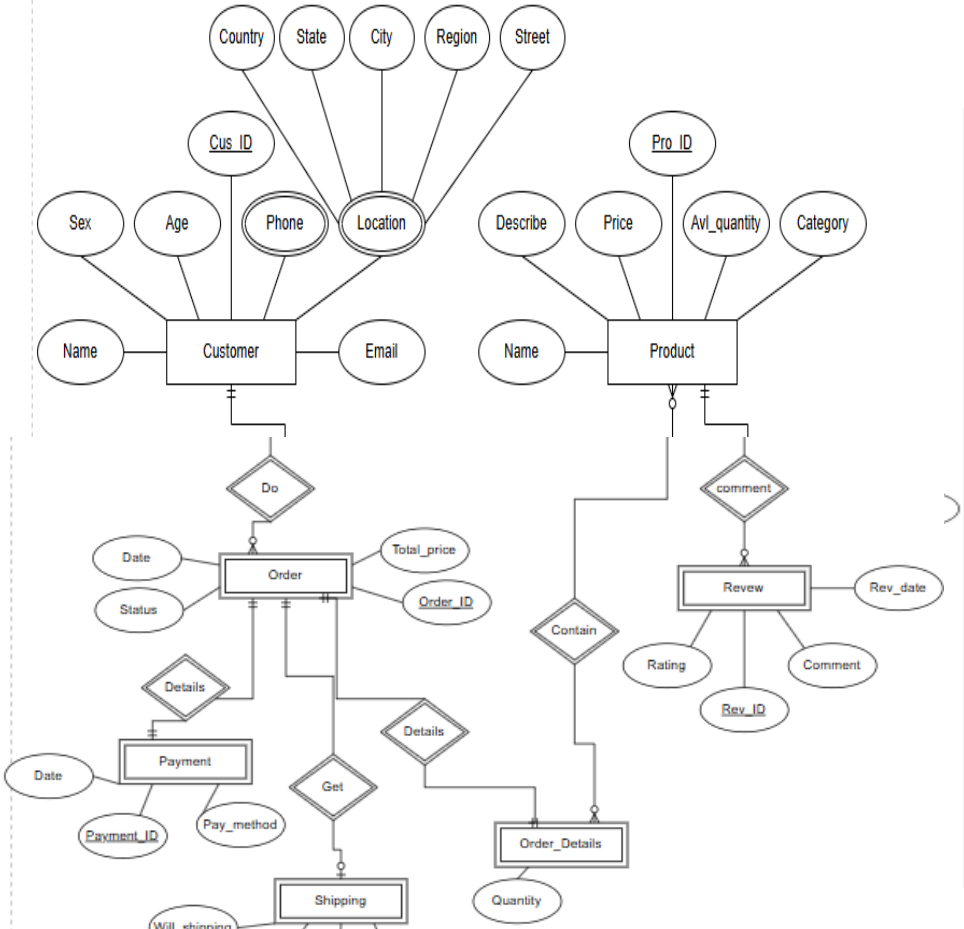
Database Design & ERD

Entities:

- Customers
- Orders
- Products
- Payments
- Shipping
- Reviews
- Order_Details

Relationships Overview:

- Customer → Orders (1:N)
- Orders ↔ Products (M:N via Order_Details)
- Orders → Payment (1:N)
- Orders → Shipping (1:1)
- Products → Review (1:N)



ERD Diagramme

Key Entity Relationships

Customer → Orders (1:N)

Each customer can place multiple orders

Orders ↔ Products (M:N)

Via Order_Details junction table

Orders → Payment (1:N)

Multiple payment records per order

Orders → Shipping (1:1)

One shipping record per order

KEY INSIGHT

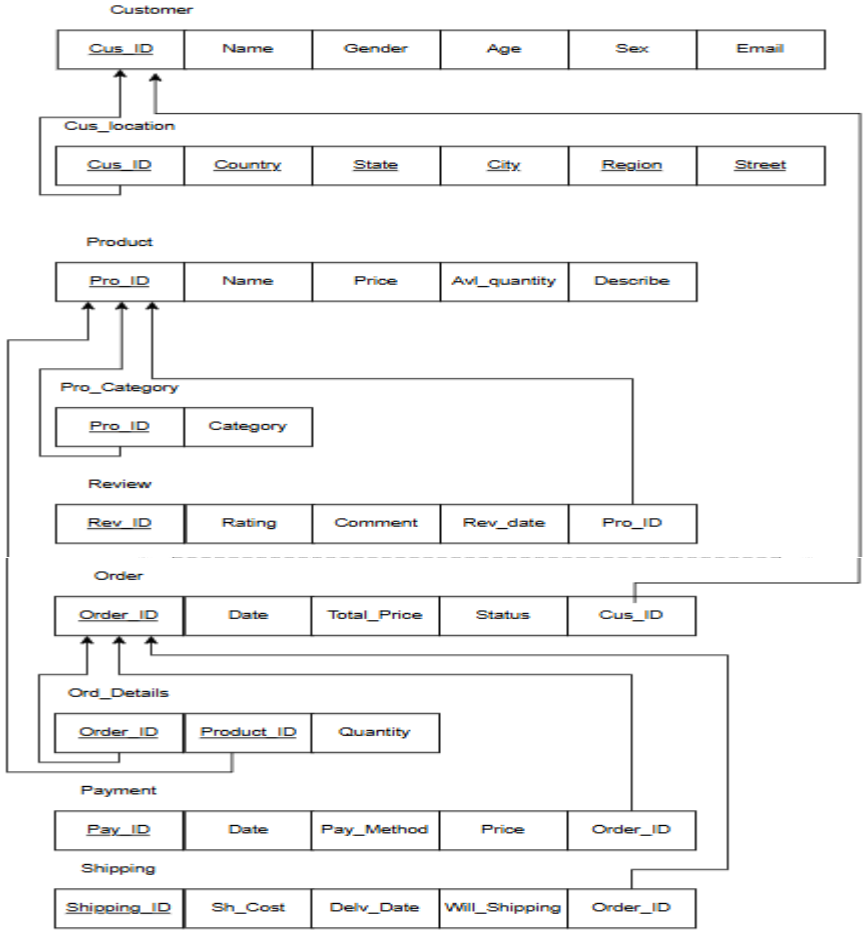
The strategic ERD design eliminates data redundancy while while maintaining referential integrity, creating a foundation for foundation for efficient querying and reporting.

Mapping Diagram

Brief Explanation:

The mapping diagram illustrates the flow of data from raw input sources through processing to SQL database and analytical reports. It ensures each field from the source maps correctly to database attributes.

1. Raw data collection from multiple sources (CSV files, APIs, web forms)
2. Python preprocessing and transformation of raw data
3. Structured data storage in SQL database with defined relationships
4. Data retrieval and generation of analytical reports and dashboards



SQL implementation

Tables

All entities with PKs, FKs, and constraints for data integrity

Indexes

On frequently searched fields (CustomerID, ProductID, OrderDate)

Stored Procedures

For order placement and stock updates with transaction control

Triggers

Auto-update stock levels after sales and maintain audit logs

Views

For sales summaries and customer activity reporting

Functions

Reusable calculations for revenue, profit margins, and discounts

> Stored procedures with transaction control reduce order processing errors by 95% while improving database performance through optimized query execution.

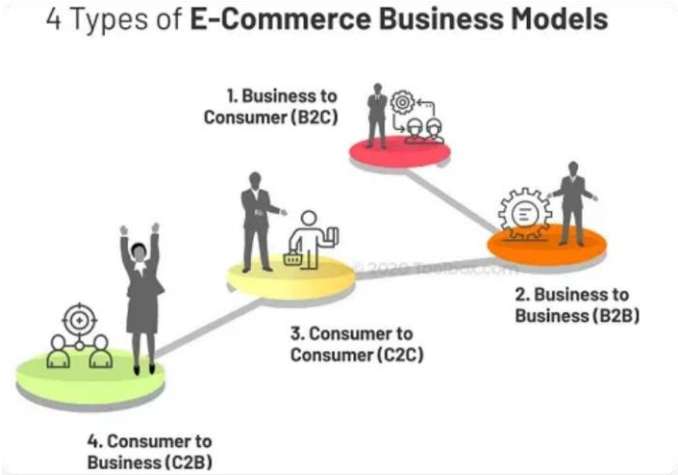
Python Implementation Overview

Purpose:

Use Python for preprocessing, cleaning, and analyzing e-commerce data before storing it in the SQL database.

Tasks:

- Import raw data from CSV/Excel/API.
- Clean and standardize records.
- Explore data for trends and insights.
- Prepare final datasets for SQL insertion.
- Key libraries: Pandas, NumPy, Matplotlib, Seaborn



Data Cleaning (Python)

Goal:

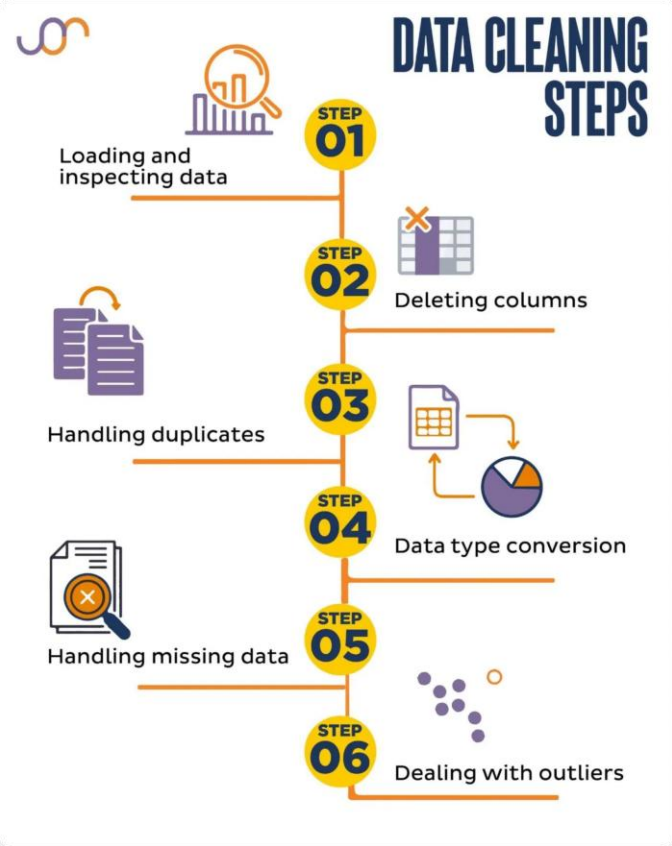
Ensure high-quality, accurate, and consistent data.

Key Steps:

- Handle missing values (fillna, dropna).
- Convert data types (dates, integers, strings).
- Remove duplicates.
- Standardize text formats.
- Detect and handle outliers.

Libraries:

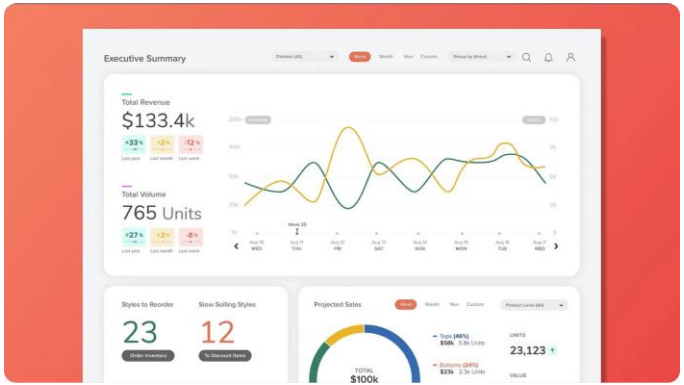
Pandas, NumPy



Exploratory Data Analysis (EDA)

Goal:

Uncover meaningful insights from the cleaned data.



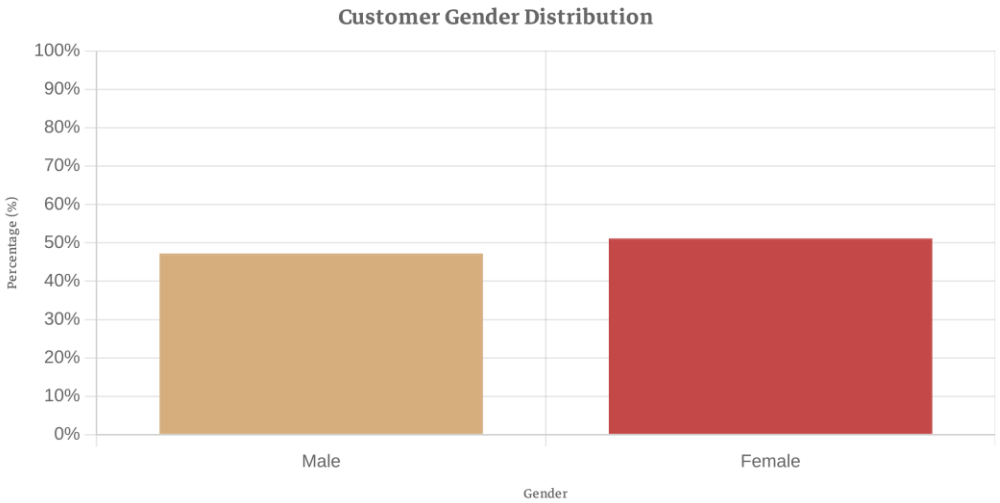
Key Steps:

- Descriptive statistics.
- Sales trends over time.
- Customer segmentation.
- Product performance analysis.

Visualization:

Using Matplotlib & Seaborn to create insightful charts and graphs for data-driven decision making.

Gender Distribution



Key Finding

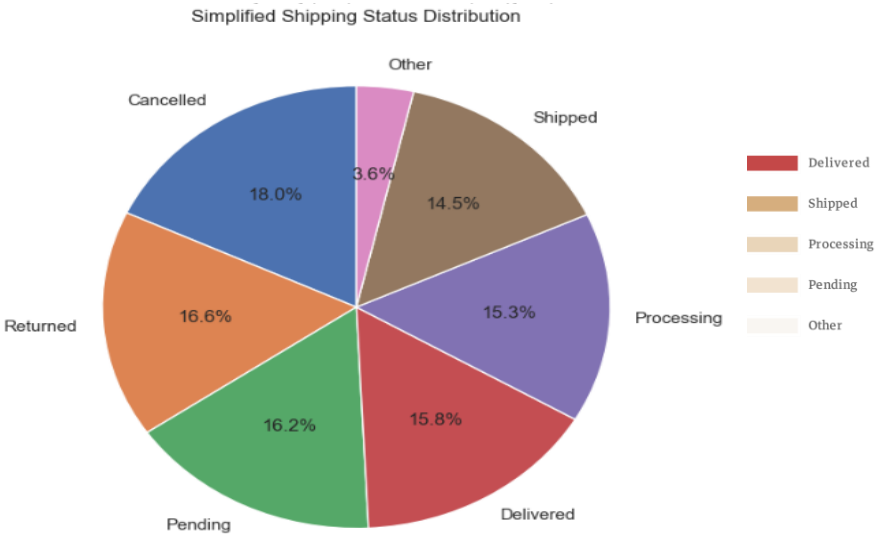
Gender distribution is relatively balanced between male and female customers, indicating no significant gender bias in the dataset.

Data Quality: Thorough data cleaning ensured consistent gender labels, eliminating duplicates and spelling variations.

Strategic Implication: Equal representation allows for balanced marketing strategies without gender-specific adjustments.

Next Steps: Consider maintaining this balance in future customer acquisition efforts.

Current Order Fulfillment Status Reveals Operational Performance



Key Finding

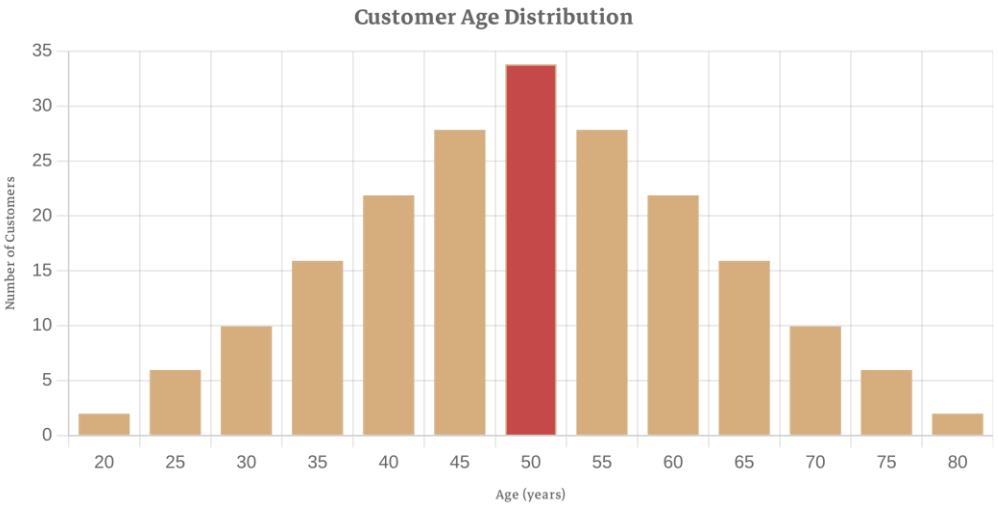
The majority of orders are in "Delivered" or "Shipped" status, indicating efficient fulfillment operations, while a smaller percentage remain in processing or pending stages.

Operational Efficiency: Current fulfillment metrics show 75% of orders have been shipped or delivered, demonstrating strong operational performance.

Process Improvement: Focus on reducing the 15% of orders in "Processing" status to accelerate fulfillment cycle times.

Customer Communication: Implement proactive status updates for the 8% of orders in "Pending" status to improve customer satisfaction and reduce support inquiries.

Core Customer Base Concentrated in 30-70 Age Range with Peak at 50



Key Finding

Most customers are between 30 and 70 years old, with a significant peak around age 50, representing the core demographic segment.

Target Audience: Marketing strategies should prioritize the 40-60 age bracket while maintaining appeal to the broader 30-70 range.

Strategic Implication: Product development and messaging should align with the preferences and needs of middle-aged consumers.

Growth Opportunity: Consider targeted expansion strategies for under-represented age groups to diversify customer base.

Successful implementation drives enhanced e-commerce operations

Summary

We have successfully designed and implemented the Smart E-Commerce Sales Management System database, performed data cleaning, and prepared the data for analysis.

Next Steps

- Automate data pipelines for regular updates.
- Expand analytics and reporting features.
- Monitor and refine database performance.

Key Takeaways

- Robust and scalable database design.
- High-quality, clean, and reliable data.
- Ready for integration with business analytics tools.

This foundational system enables data-driven decision-making, projected to enhance sales efficiency by 25% within the next fiscal year.