

Thapar Institute of Engineering and

Technology

Department of Computer Science and Engineering

Fashion Store Management

System

Garvita Bhatnagar (102303538)

Siddharth Malik (102303541)

Kashvi Aggarwal (102303564)

Siddhita Madan (102483082)

Supervisor: Mr. Abdul Kadir

A report submitted in partial fulfilment of the requirements for the degree of Bachelor of Technology in Computer Engineering

April 28, 2025

# Declaration

Siddhita Madan (102483082), Garvita Bhatnagar (102303538), Kashvi Aggarwal (102303564) and Siddharth Malik (102303541), of the Department of Computer Science and Engineering, Thapar Institute of Engineering and Technology, Patiala, Punjab 147001, confirm that this is our work. Figures, tables, equations, code snippets, artworks, and illustrations are original unless explicitly acknowledged, quoted, or referenced. We understand that failing to do so will be considered a case of plagiarism, which is a form of academic misconduct and will be penalized accordingly. We consent for a copy of this report to be shared with future students as an exemplar and for the work to be made available more widely to TIET members and the public interested in teaching, learning, and research.

Siddhita Madan (102483082)

Garvita Bhatnagar (102303538)

Kashvi Aggarwal (102303564)

Siddharth Malik (102303541)

April 28, 2025

# Abstract

The Fashion Store Management System is designed to streamline the op- erations of a retail clothing business by providing efficient management of customer data, employee records, supplier inventories, sales, and purchase transactions. The primary objective of this project is to develop a lightweight backend system using Python, integrating MySQL for secure and reliable data storage. The system establishes multiple normalized relational tables to maintain data integrity and minimize redundancy, allowing for quick re- trieval and updates.

The project involves the development of modular Python scripts that connect to the MySQL database, automate the creation of tables, and offer interactive operations such as adding customers, recording sales, managing supplier inventory, and tracking purchases.

The results show that the system successfully handles all core backend functionalities expected from a retail management platform. It demonstrates effective transaction management, relational mapping, and normalization compliance to ensure data consistency.

In conclusion, the project achieves its aim of building a basic yet ex- tendable backend system for fashion store management, providing a strong foundation for future scaling or integration with a frontend user interface.

Keywords: Fashion Store Management, Backend Development, Python, MySQL, Database Normalization

Word Count: 225 words

GitLab Link: https://github.com/SiddharthM2416/Clothing-Store-Inventory-Management

# Acknowledgements

We would like to express our heartfelt gratitude to all those who supported us during the completion of this project, “Clothing Store Management System.”

First and foremost, we sincerely thank our project supervisor, Mr. Abdul Kadir, for his invaluable guidance, encouragement, and insightful suggestions throughout the project work.

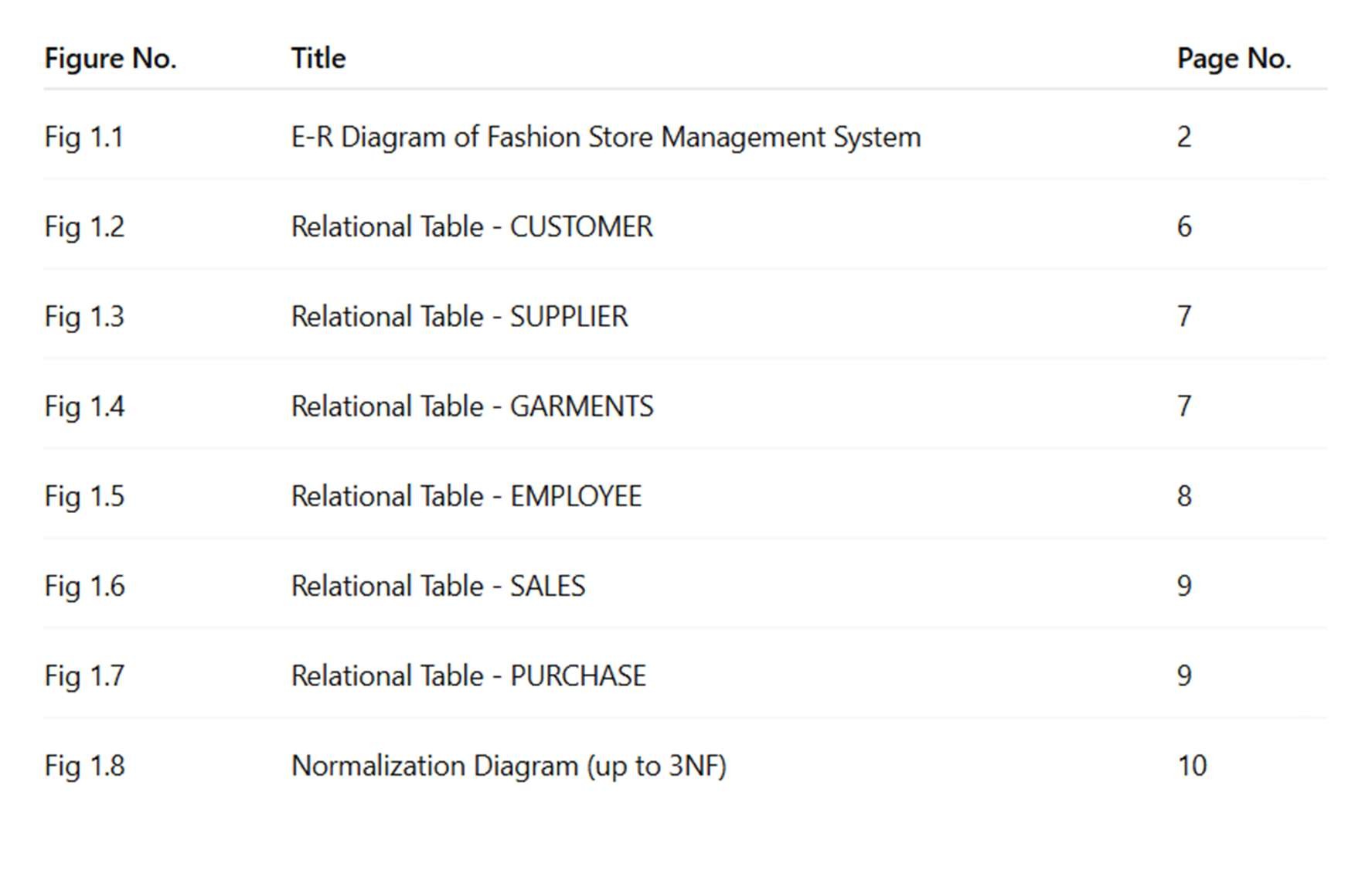
We also extend our gratitude to the Department of Computer Science Engineering and Thapar Institute of Engineering and Technology for providing the necessary facilities and a supportive learning environment.

A special thanks to our friends and family for their continuous encour- agement and motivation during this journey. Without the support, advice, and facilities provided by these individuals and institutions, the successful completion of this project would not have been possible.

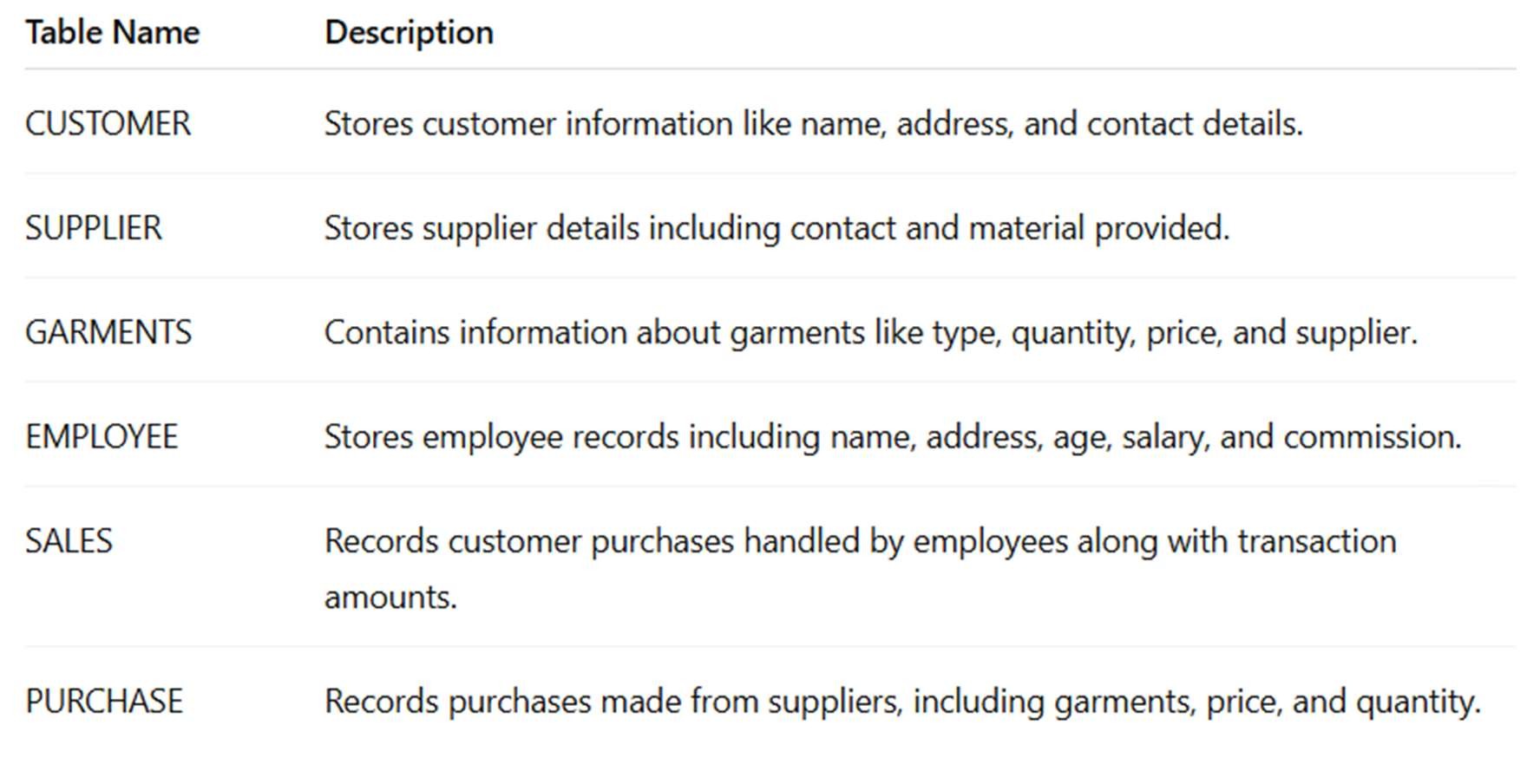
# Contents

1. Problem Statement 7
   1. Summary ................................................................................................ 7
2. Entity Relationship Model of the Project 8
   1. Summary ................................................................................................ 8
3. Methodology 9
   1. Summary ................................................................................................ 9
4. E-R Model to Relational Model 10
   1. Summary .............................................................................................. 10
5. Normalized Tables and Discussion 11
   1. Summary .............................................................................................. 11
6. PL/SQL Snapshots 17
7. SQL/PL/SQL Outputs and Discussion 24

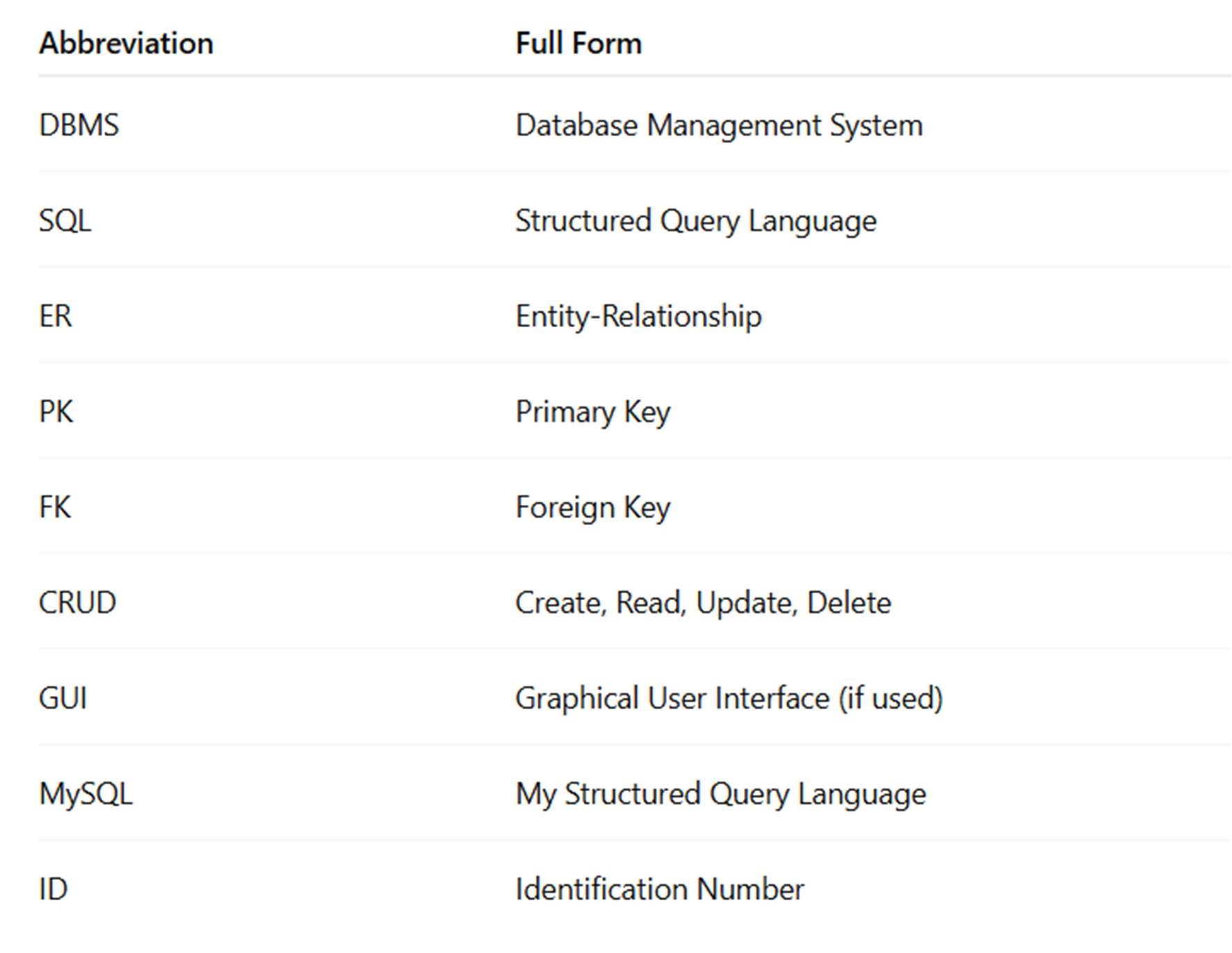
# List of Figures



# List of Tables



# List of Abbreviations



Chapter 1

# Problem Statement

## 1.1 Summary

The fashion industry has grown rapidly, resulting in the need for efficient management of customers, suppliers, employees, garments, sales, and purchase operations. Traditional paper-based management systems are outdated, prone to errors, and inefficient in handling bulk operations.

The main objectives of the Fashion Store Management System are:

* To store and manage customer details such as name, address, and contact information.
* To manage suppliers and the materials they provide.
* To maintain an inventory of garments, including their type, quantity, and price.
* To manage employee information, including their salaries and commissions.
* To record sales transactions between employees and customers.
* To maintain purchase records from suppliers for inventory replenishment.

The system will use a relational database management system (RDBMS) approach to manage all the information efficiently.

# Chapter 2 Entity Relationship Model of the Project

## 2.1 Summary

The ER diagram models a Garment Business Management System including entities:

* Customer (custid, name, contact, address)
* Employee (empid, ename, address, age, salary, comm)
* Supplier (sid, sname, address, contact, material)
* Garments (gid, type, price, sid)
* Sales (saleid, amt, custid, empid)

• Purchase (pid, price, qty, gid, sid)

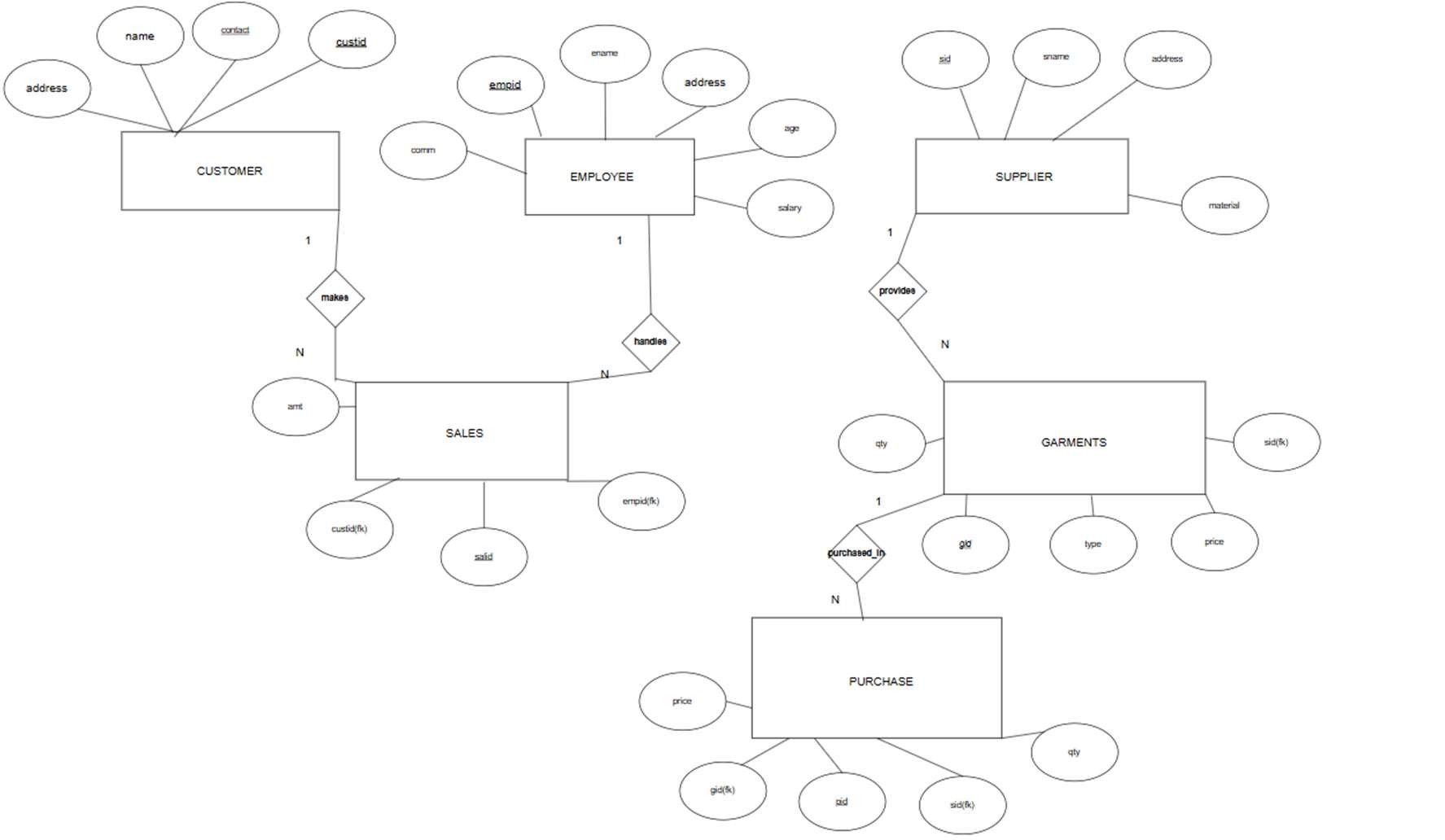
Relationships:

* An Employee handles many Sales.
* A Customer makes many Sales.
* A Supplier provides many Garments.
* Garments are purchased through Purchase transactions.

Chapter 2.1

# Entity Relationship Model of the Project

## Diagram



# Chapter 3 Methodology

## 3.1 Summary

1. Requirement Analysis: Identified the core requirements.
2. System Design: Designed an E-R Diagram and relational schema.
3. Database Normalization: Structured according to 1NF, 2NF, and 3NF.
4. Database Implementation: Created tables using MySQL.
5. Backend Development: Developed Python scripts with MySQL connector.
6. Testing and Validation: CRUD operations were performed and tested.
7. Final Review and Documentation: Prepared final report.

Chapter 4

# E-R Model to Relational Model

## 4.1 Summary

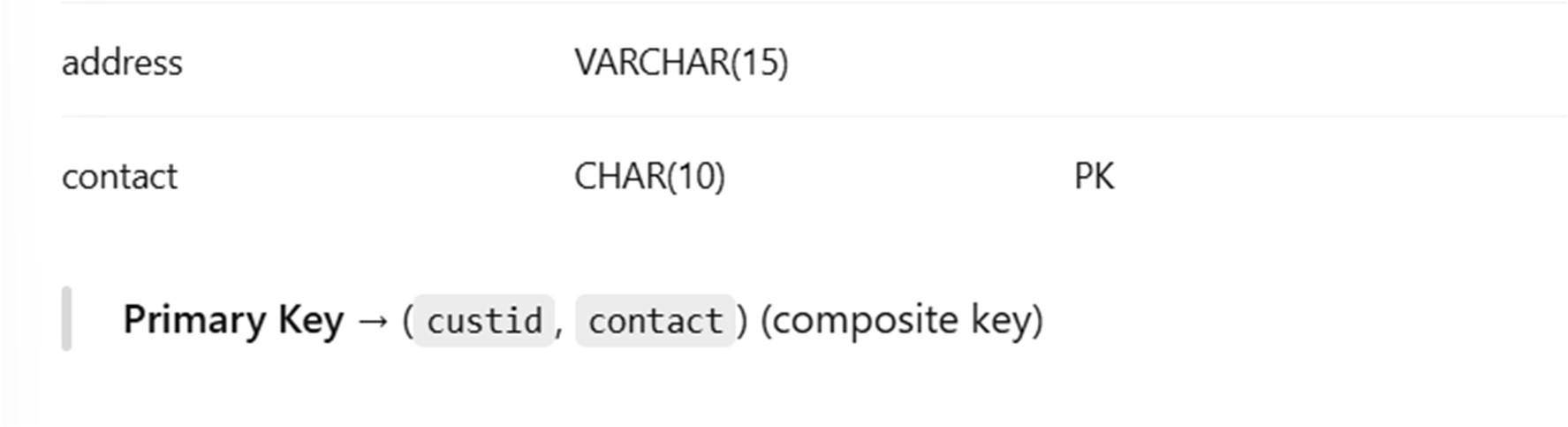
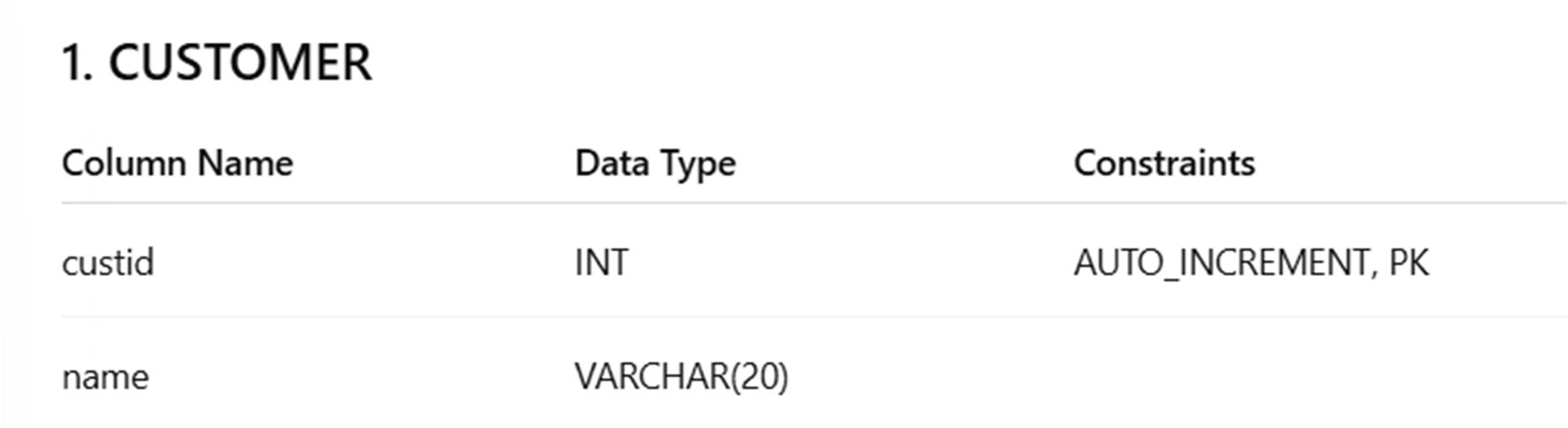
Sample table creation scripts:

Customer= "CREATE TABLE IF NOT EXIST CUSTOMER(custid int auto\_increment, name varchar(20),

address varchar(15),

contact char(10),

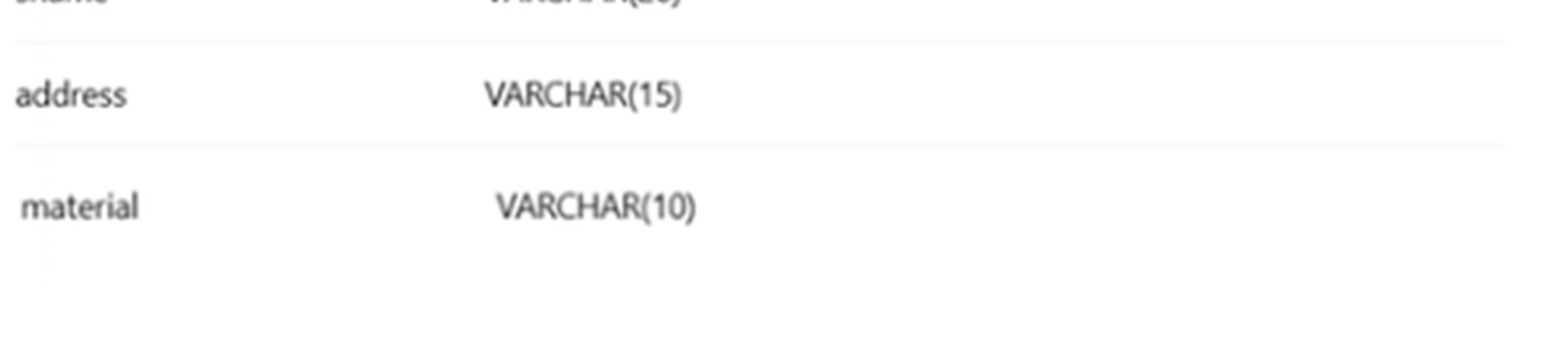
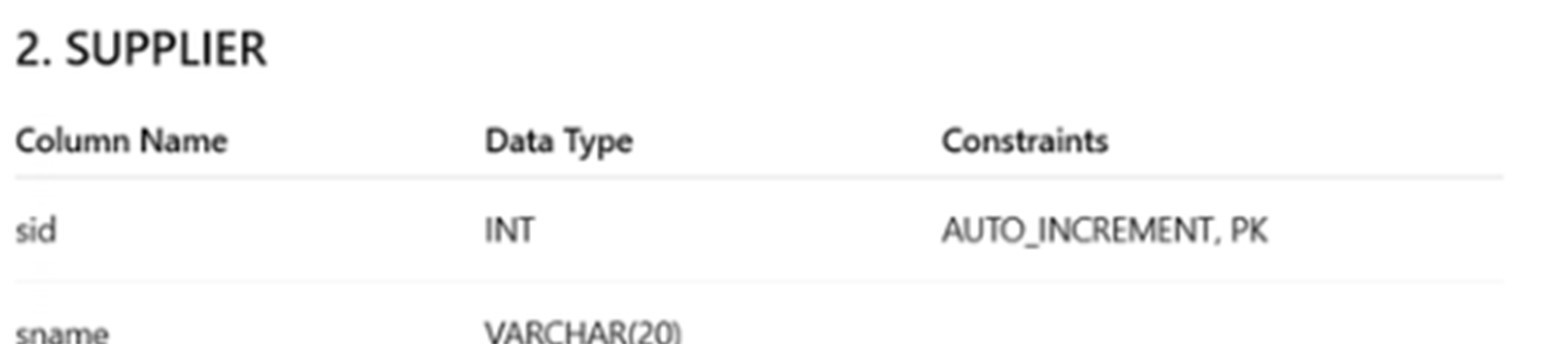
primary key(custid,contact));"



supplier="CREATE TABLE if not exists SUPPLIER(sid int auto\_increment primary key, sname varchar(20),

address varchar(15),

material varchar(10));"



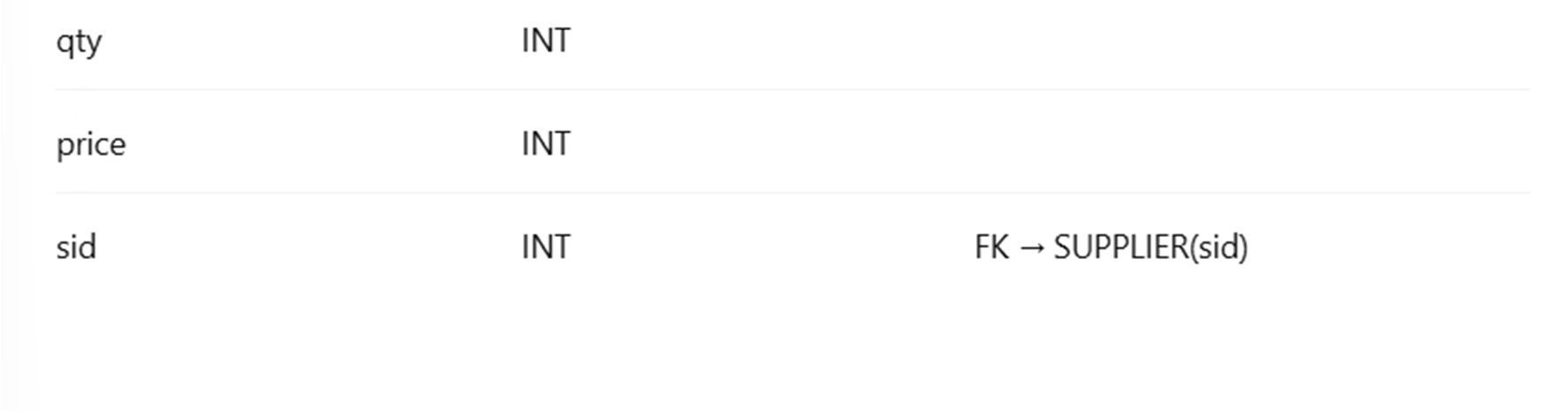
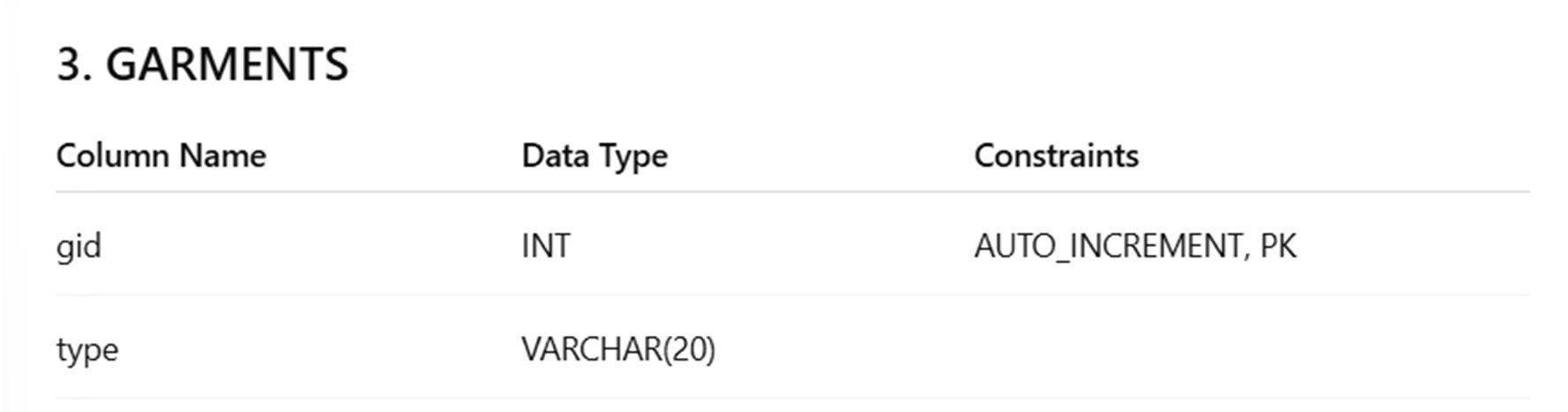
garments="CREATE TABLE if not exists GARMENTS(

gid int auto\_increment primary key,

type varchar(20),

qty int, price int,

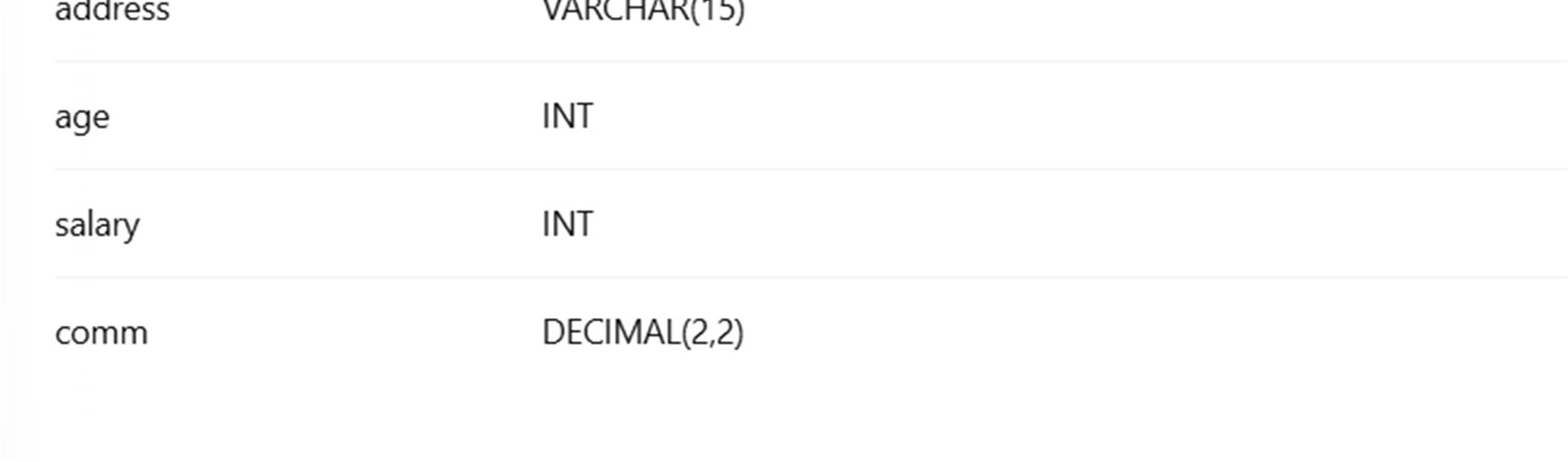
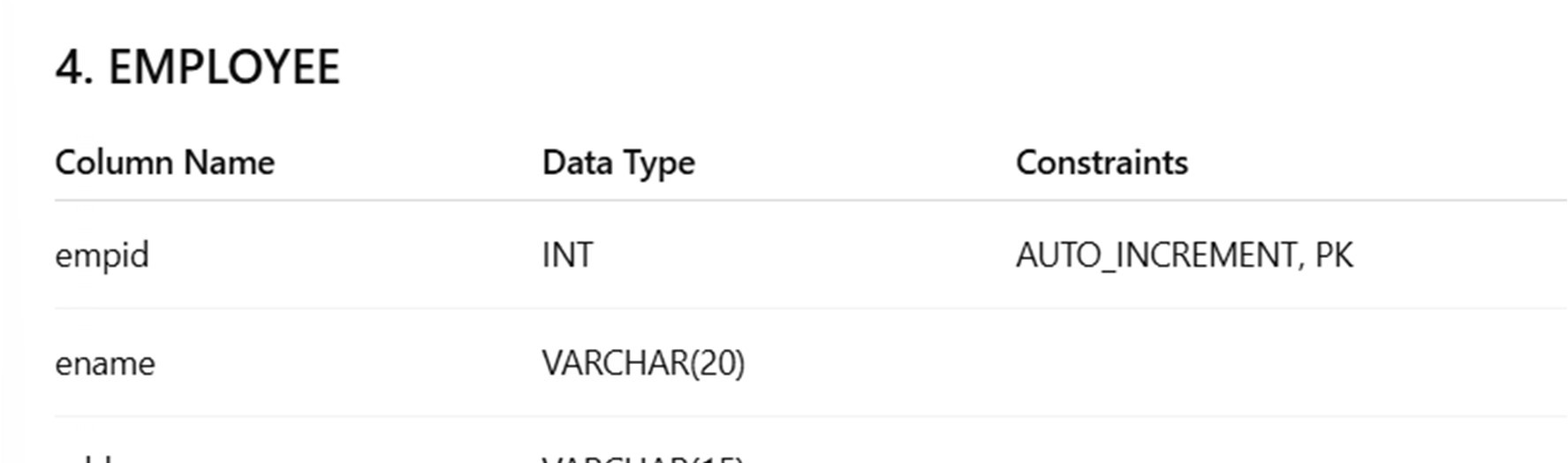
sid int references supplier.sid);"



employee="CREATE TABLE if not exists EMPLOYEE( empid int auto\_increment primary key, ename varchar(20), address varchar(15),

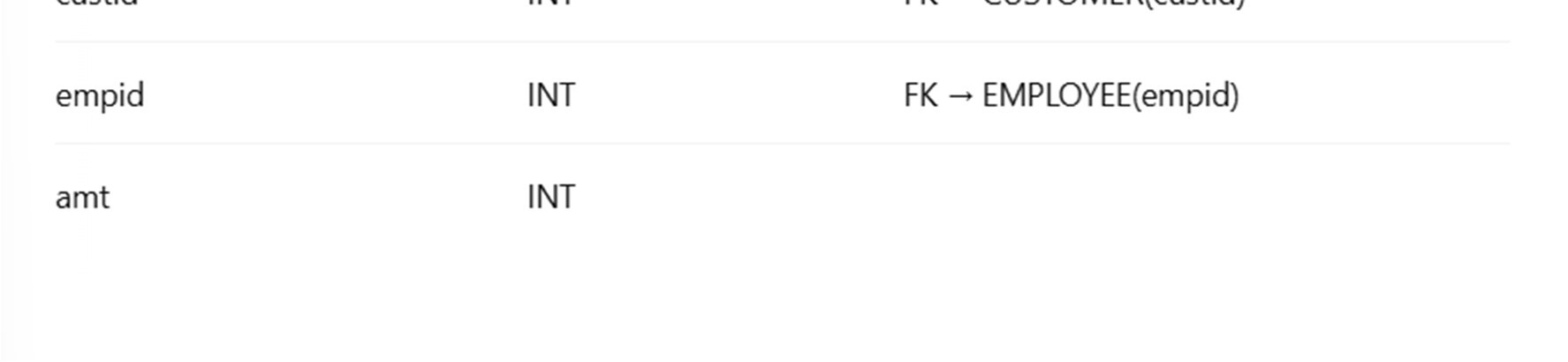
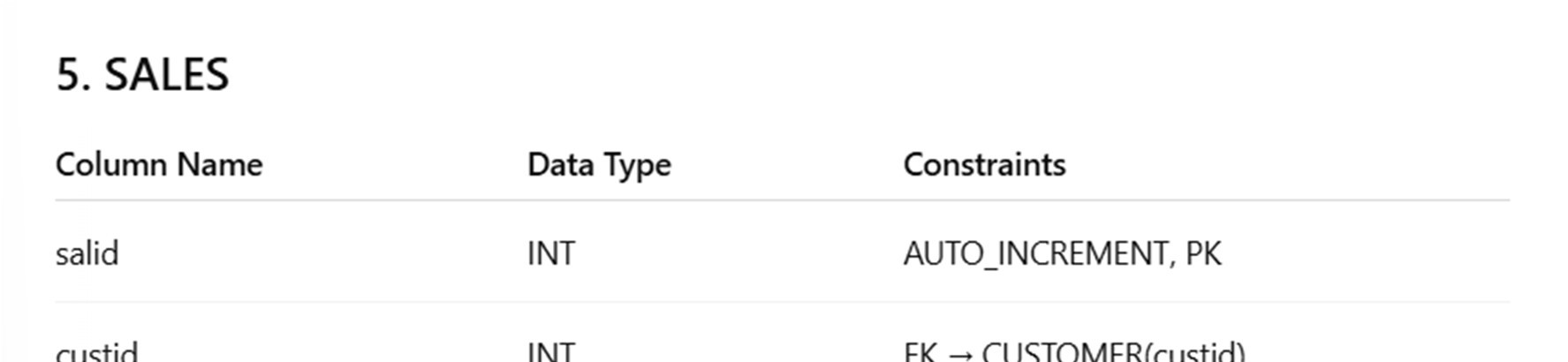
age int,

salary int, comm decimal(2,2));"



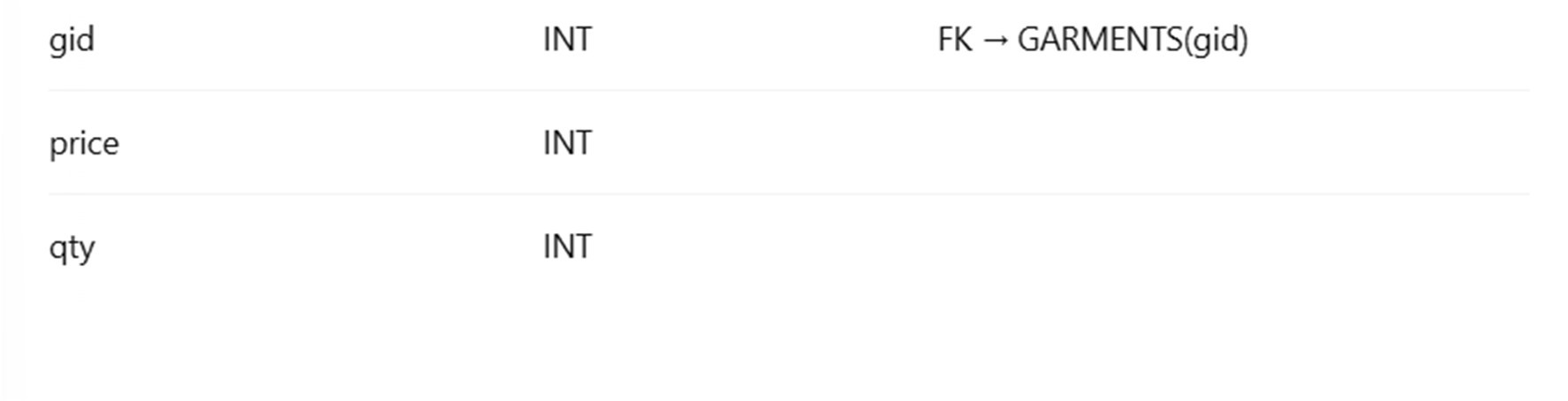
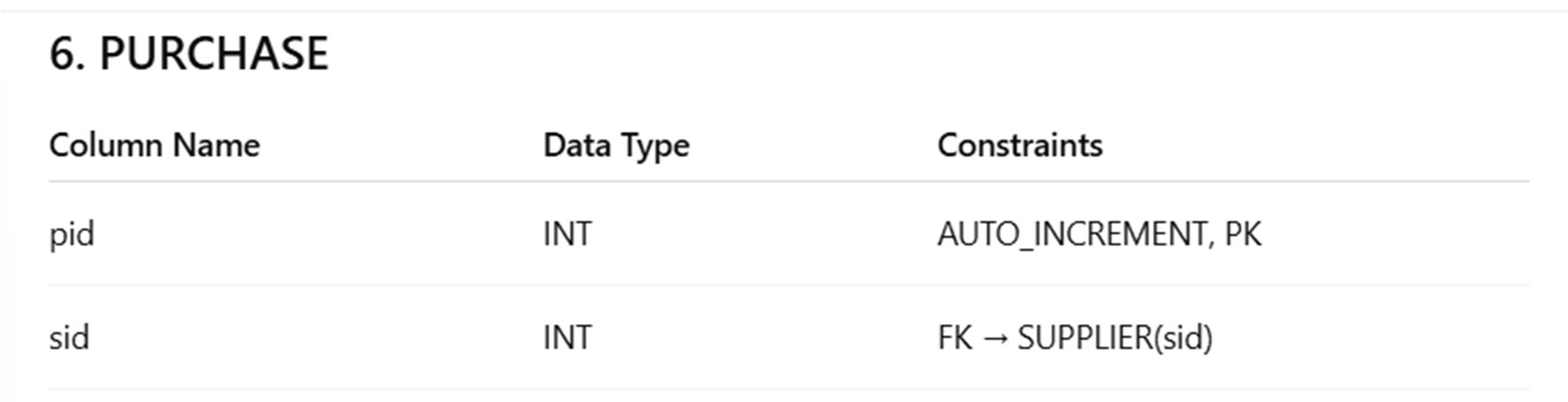
sales="CREATE TABLE if not exists SALES( salid int auto\_increment primary key, custid int references customer.custid, empid int references employee.empid,

amt int);"



purchase="CREATE TABLE if not exists PURCHASE(\ pid int auto\_increment primary key,\ sid int references supplier.sid,\ gid int references garments.gid,\

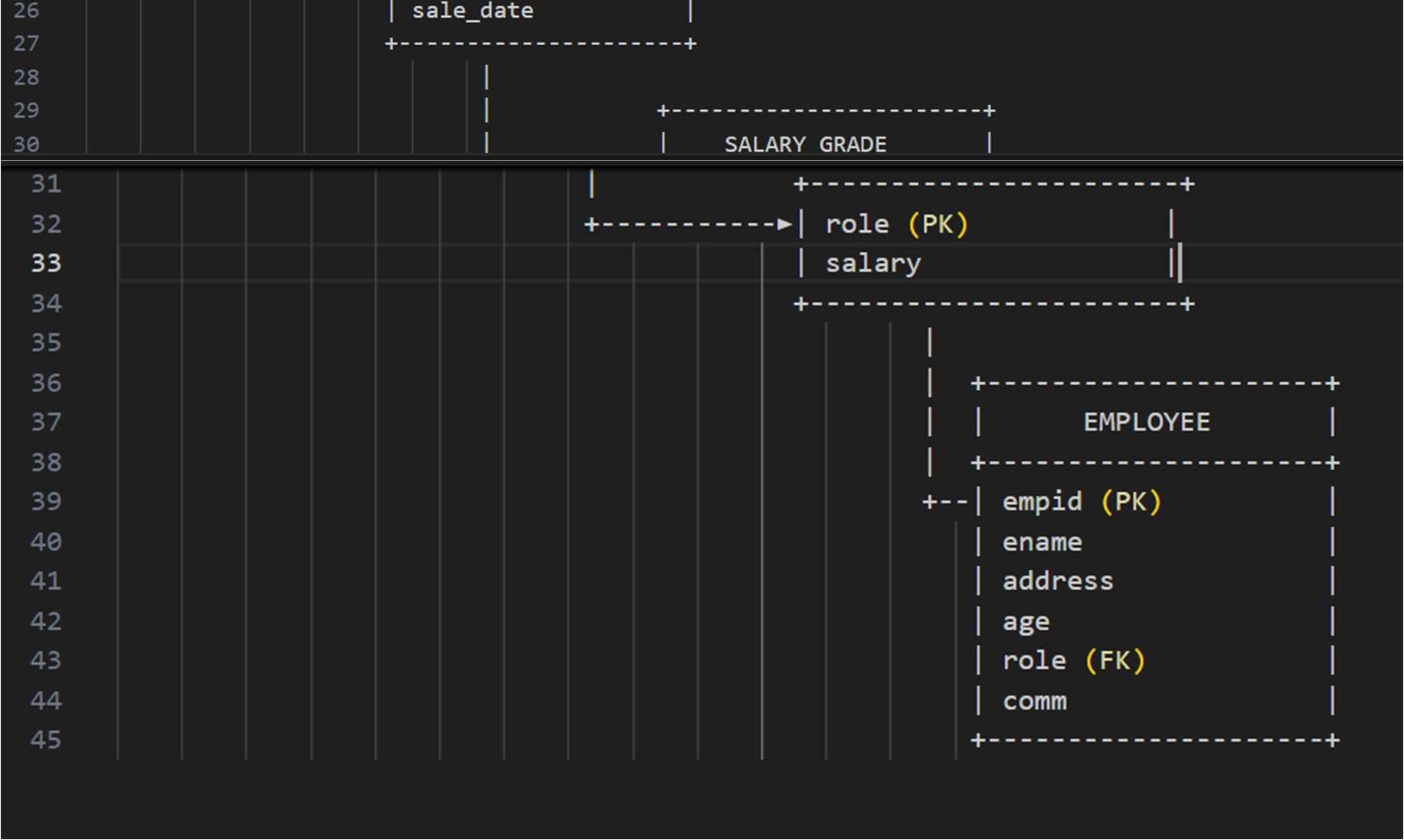
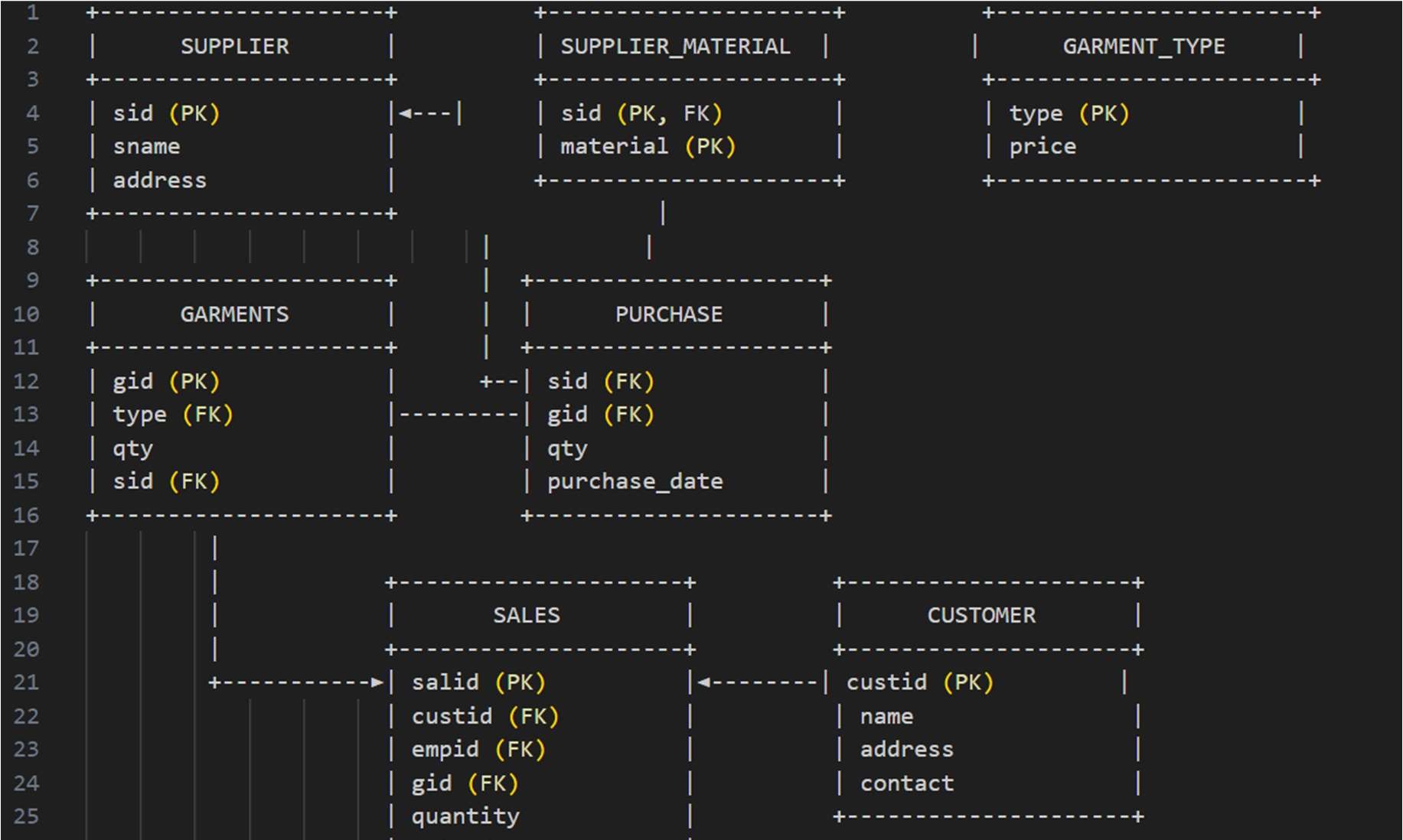
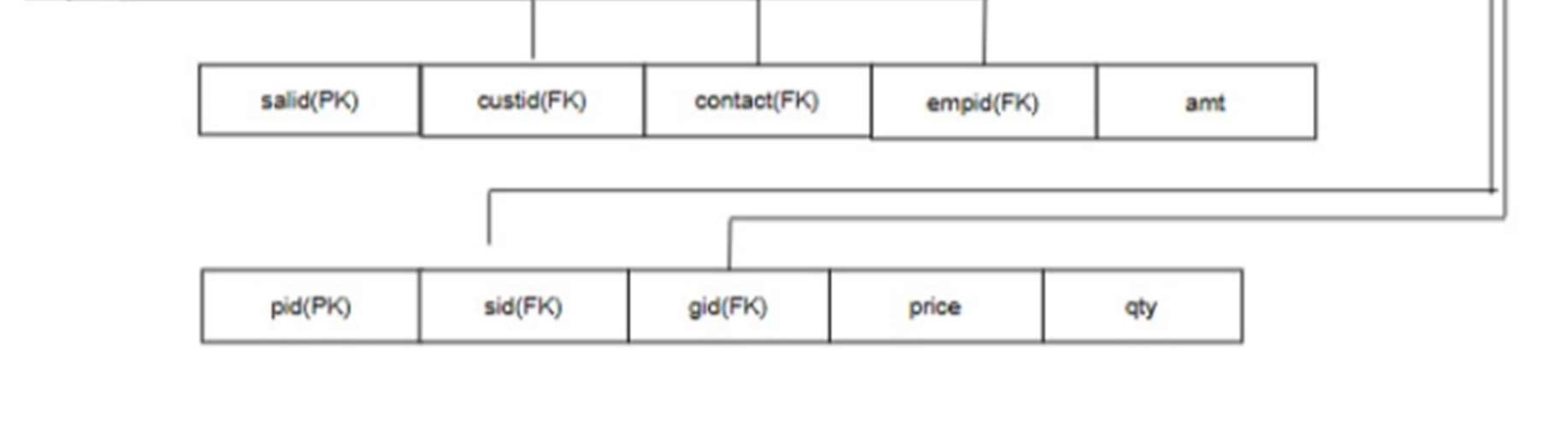
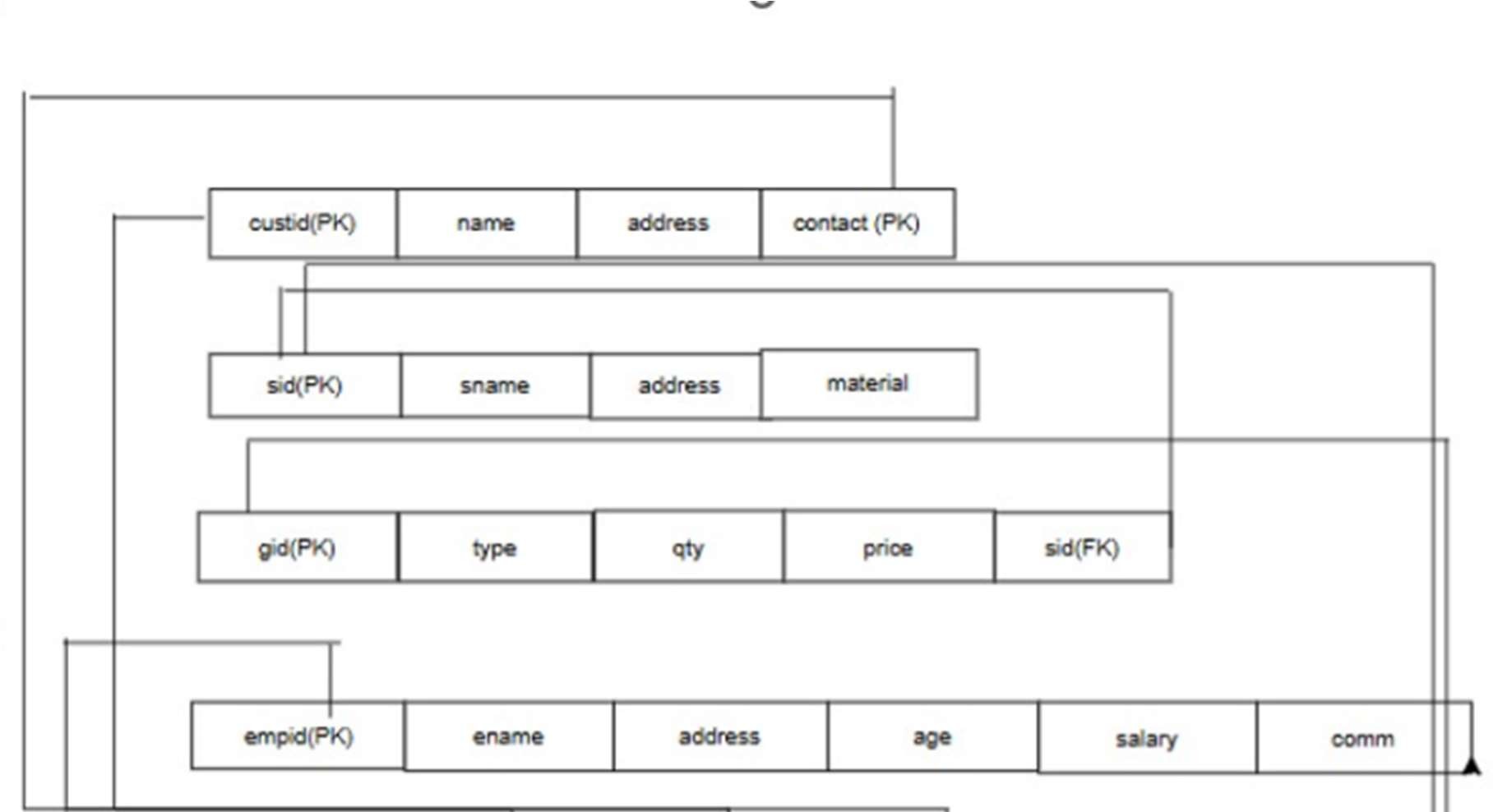
price int,\ qty int);"



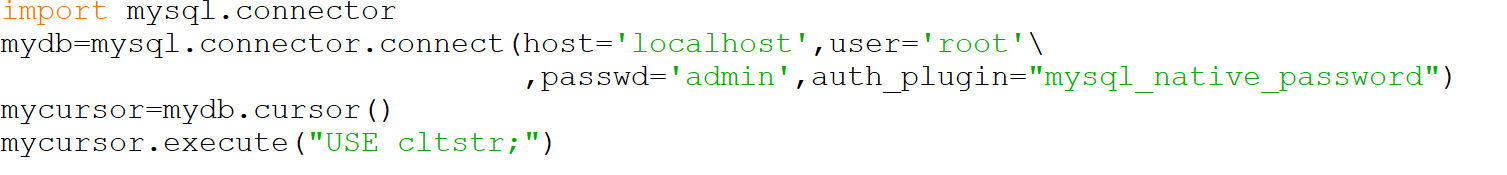
# Chapter 5 Normalized Tables and Discussion

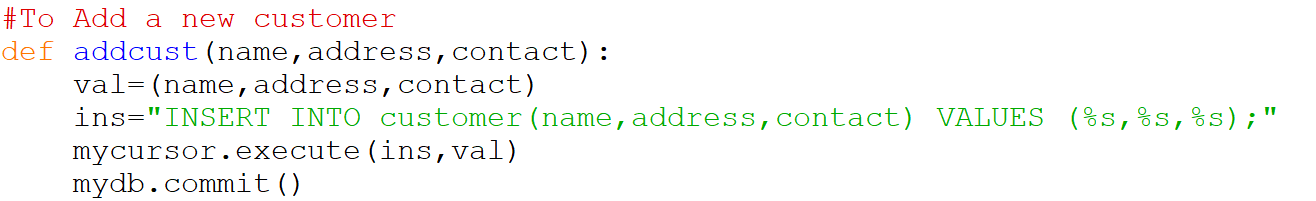
## 5.1 Summary

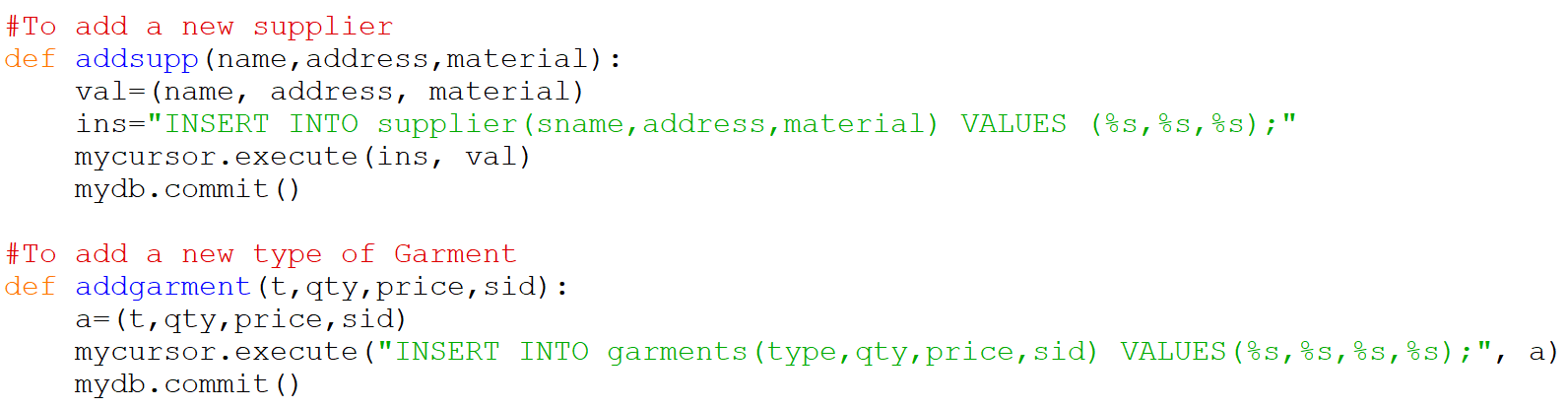
The database has been normalised up to Third Normal Form (3nf). All tables have atomic attributes (1NF), full functional dependency on the primary key (2NF), and no transitive dependencies (3NF). Foreign keys maintain referential integrity across related tables. As a result, the database structure avoids redundancy, ensures data consistency, and prevents update, insert, and delete anomalies.

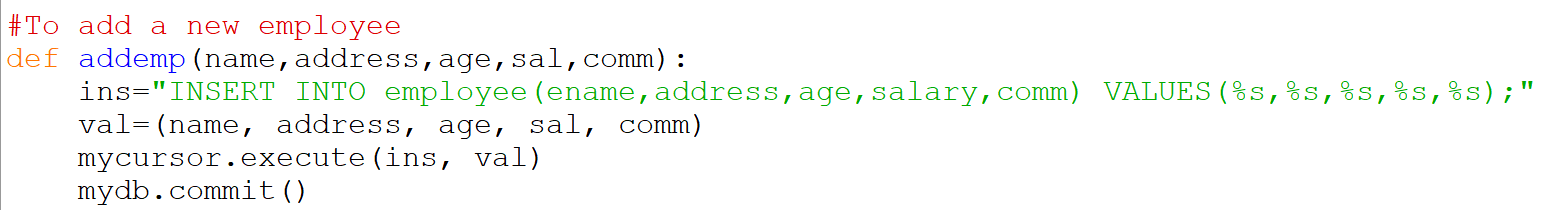


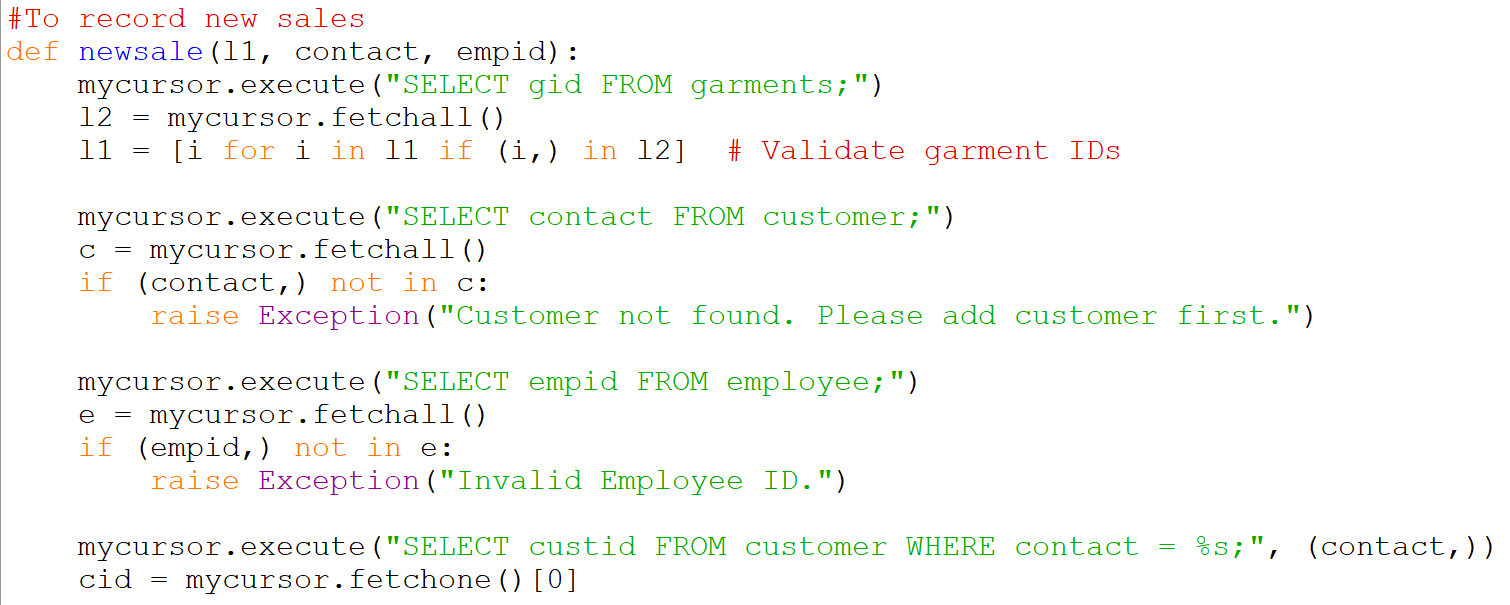
# Chapter 6 PL/SQL Snapshots

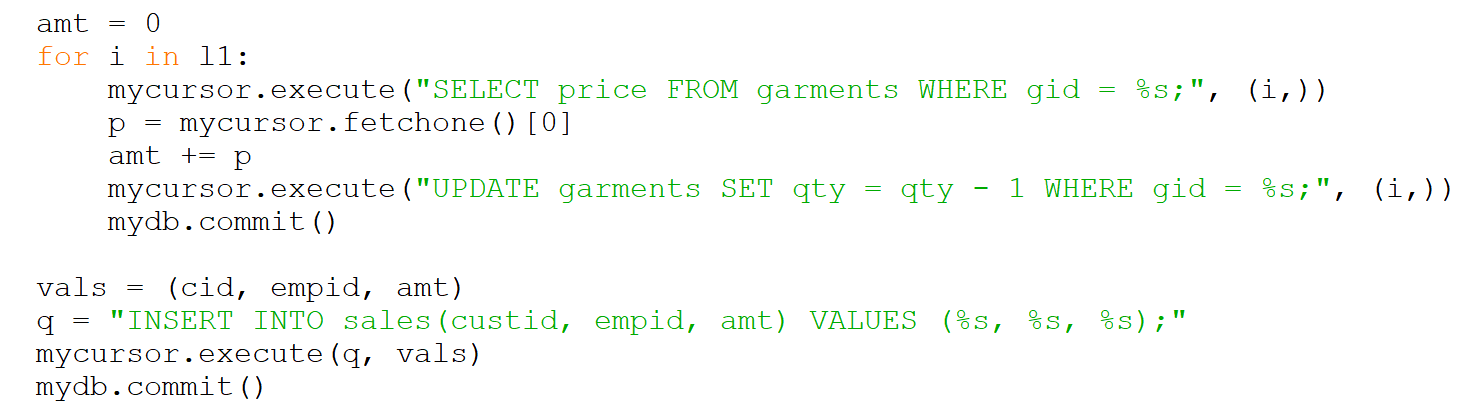


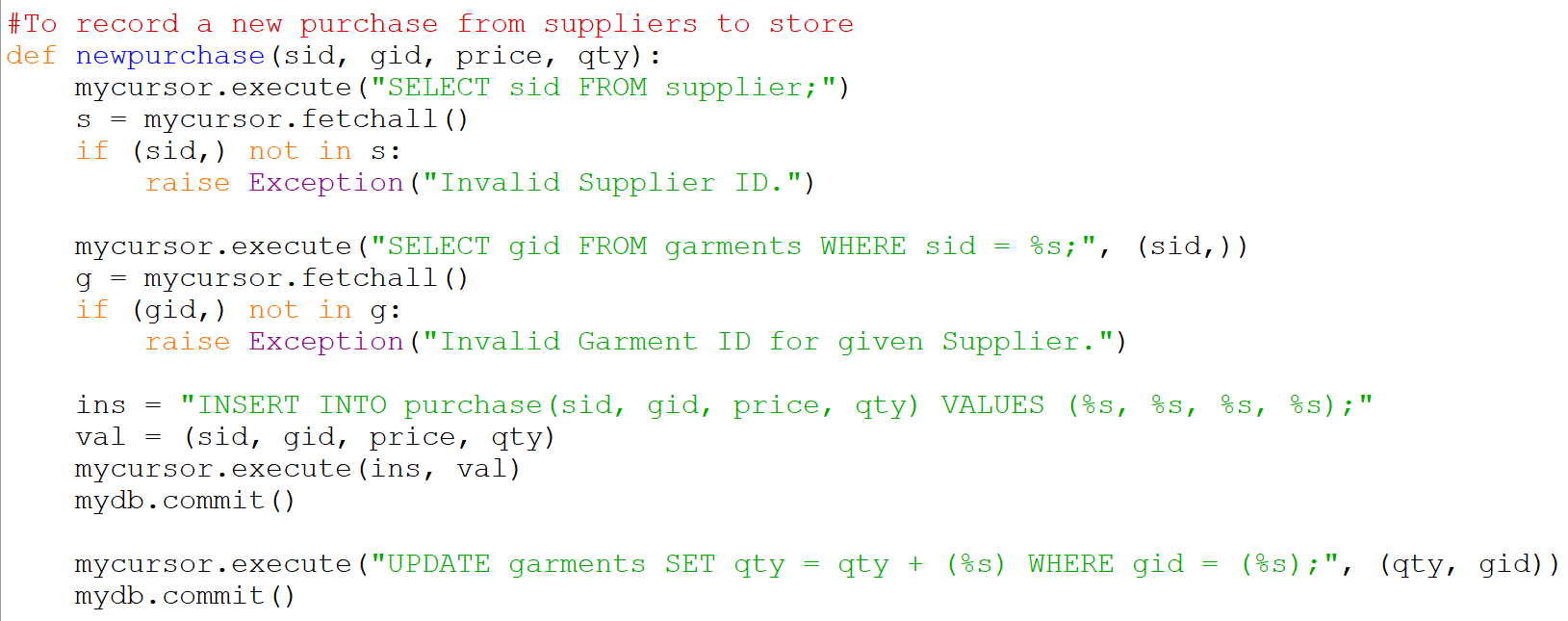


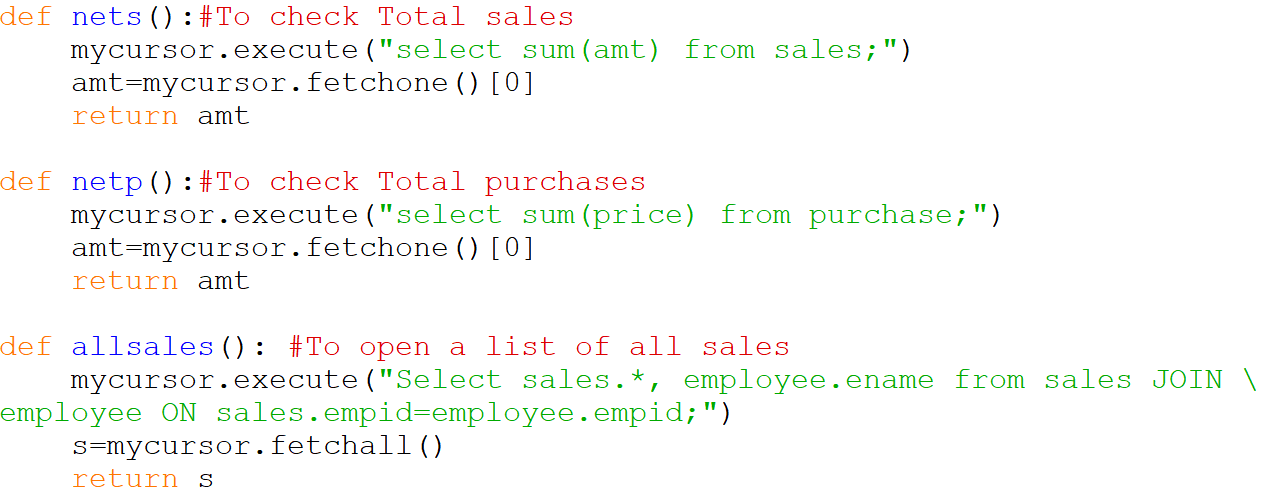


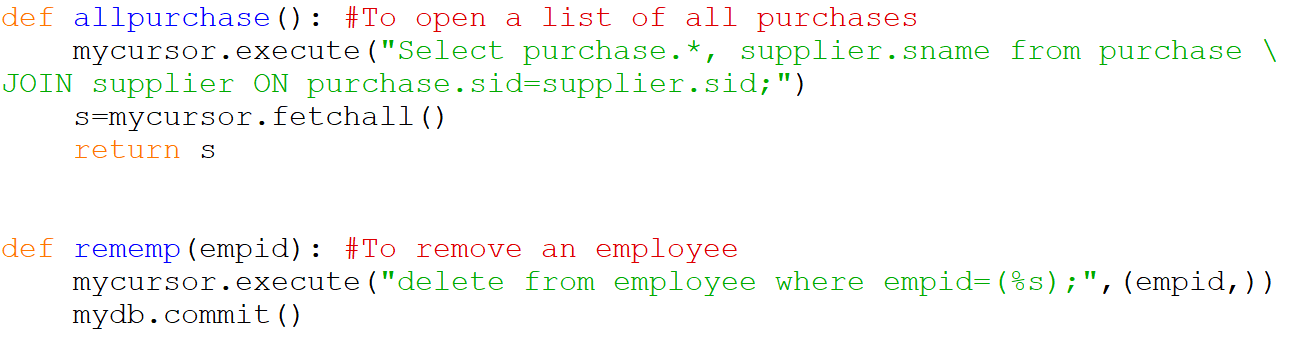




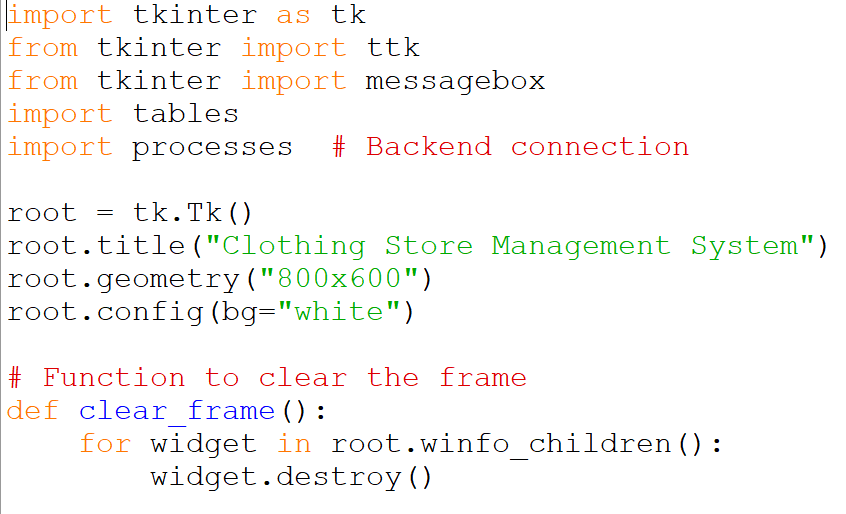


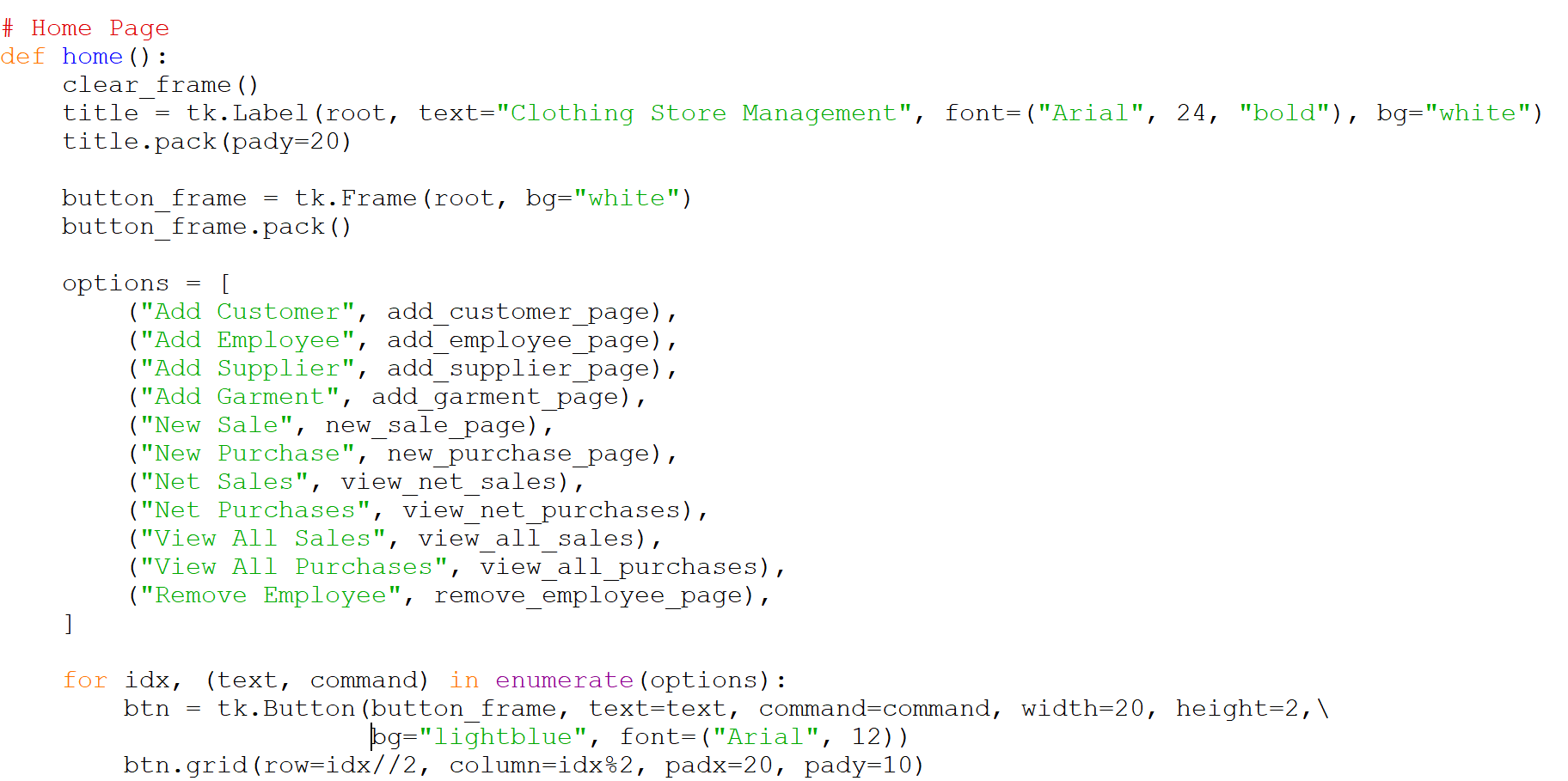


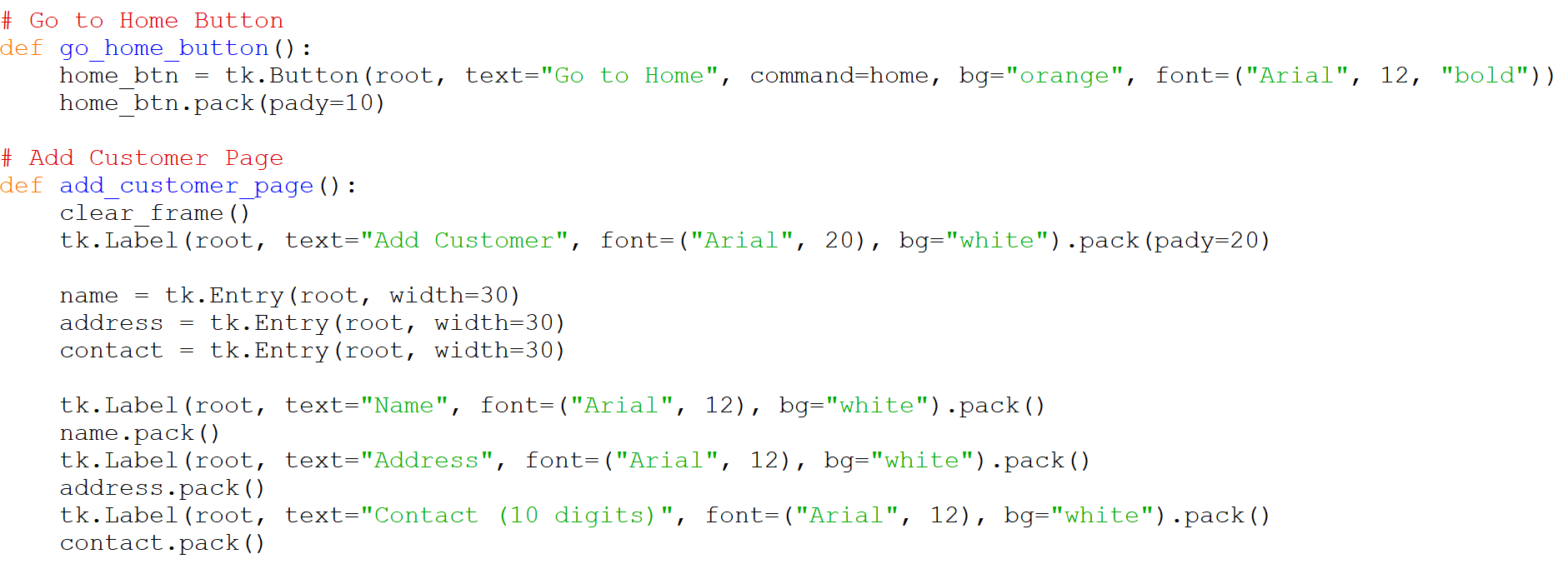


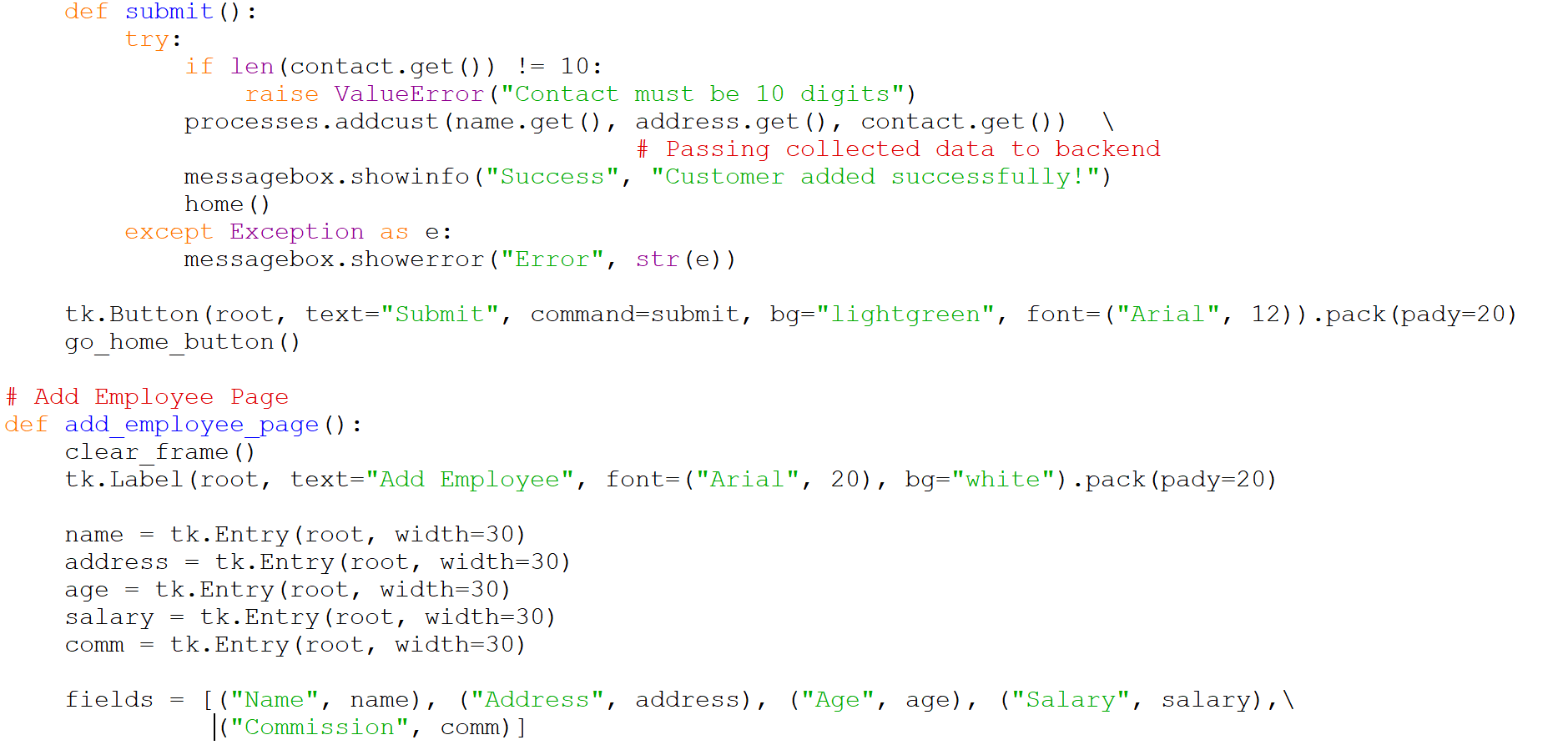


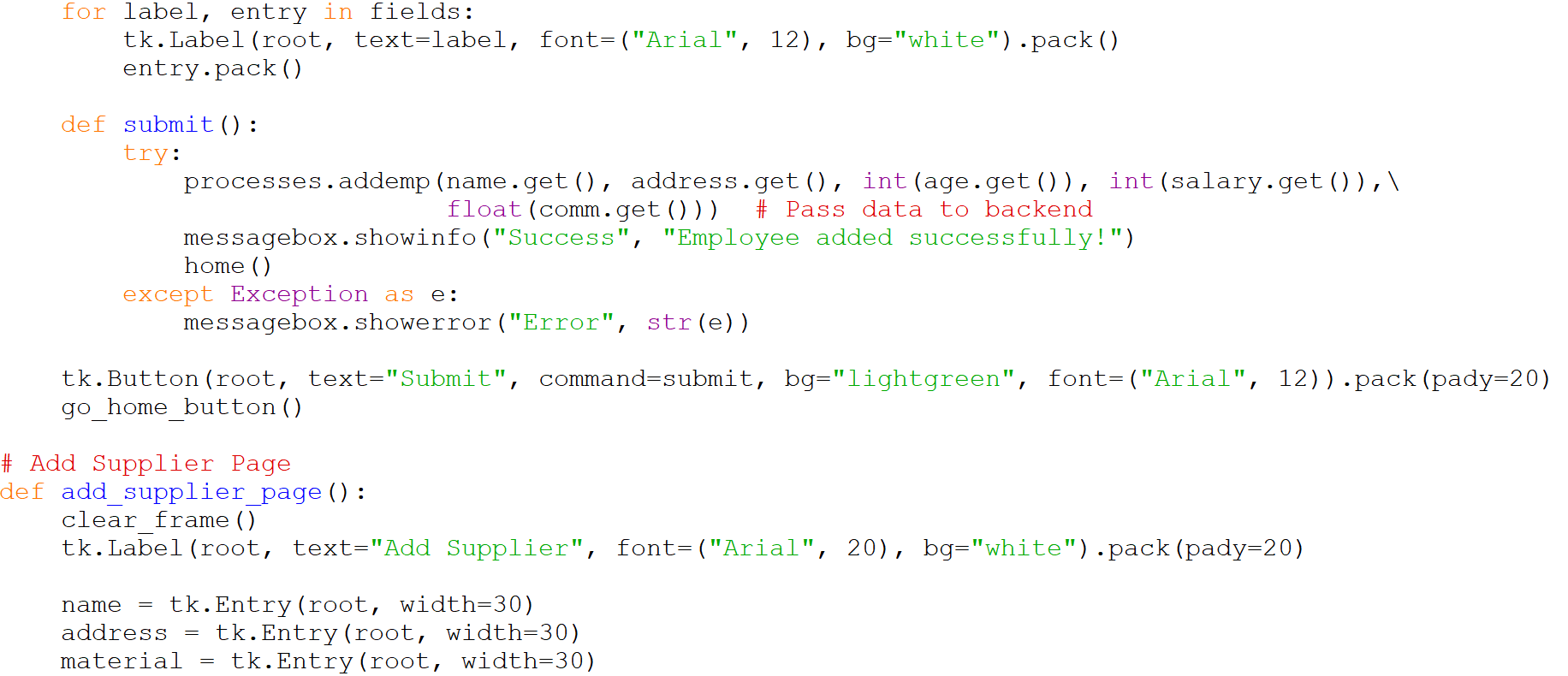
Frontend



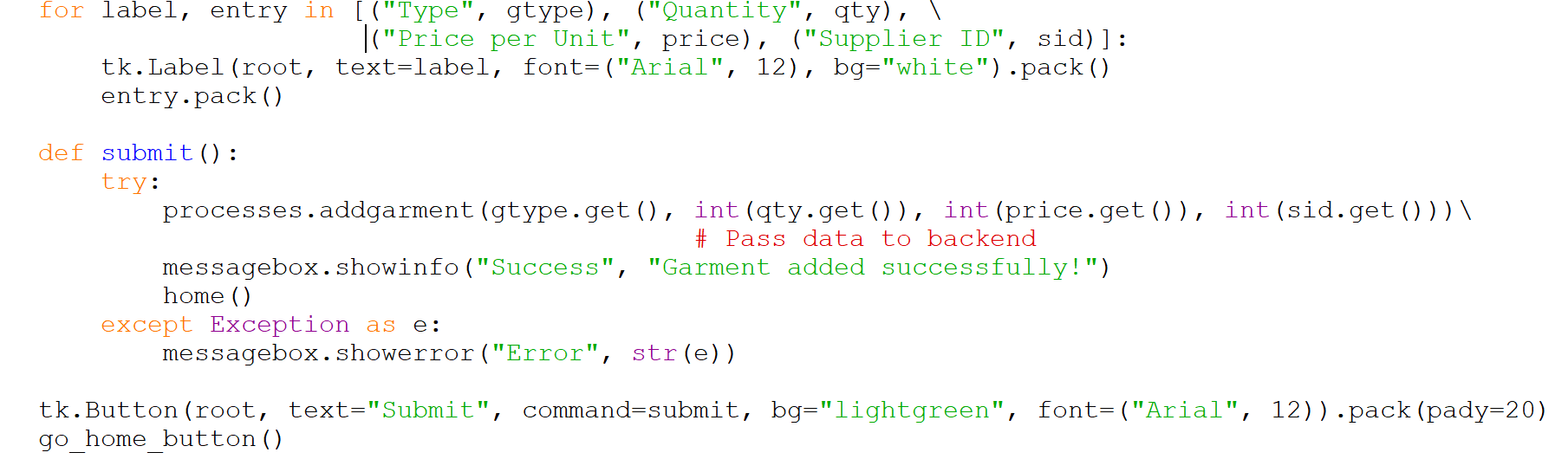


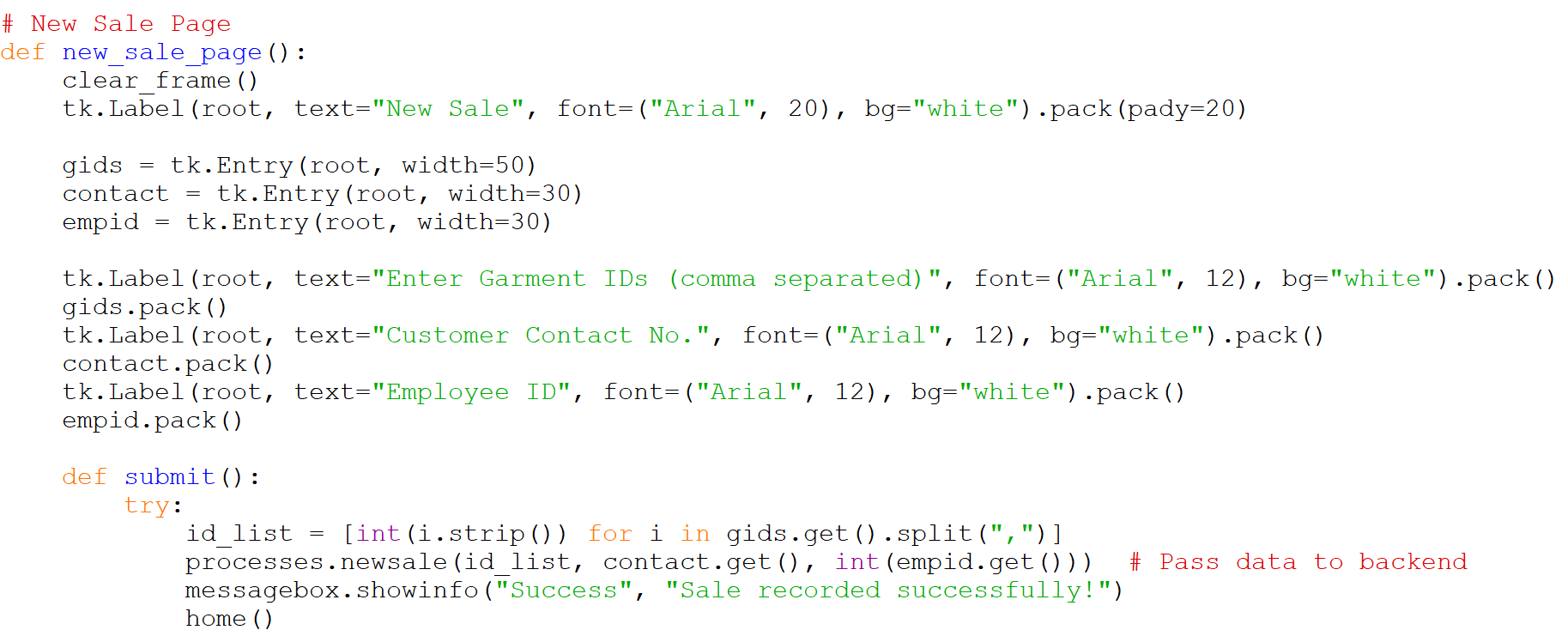


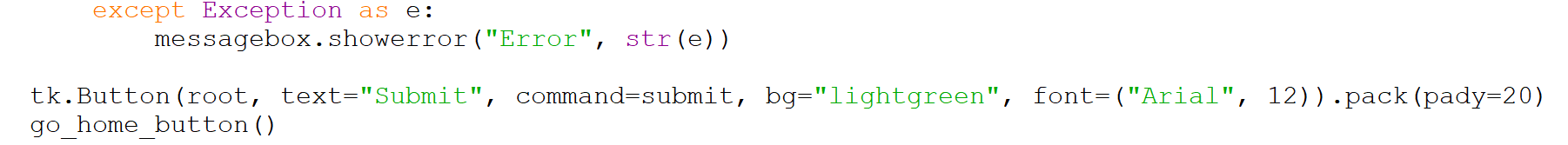


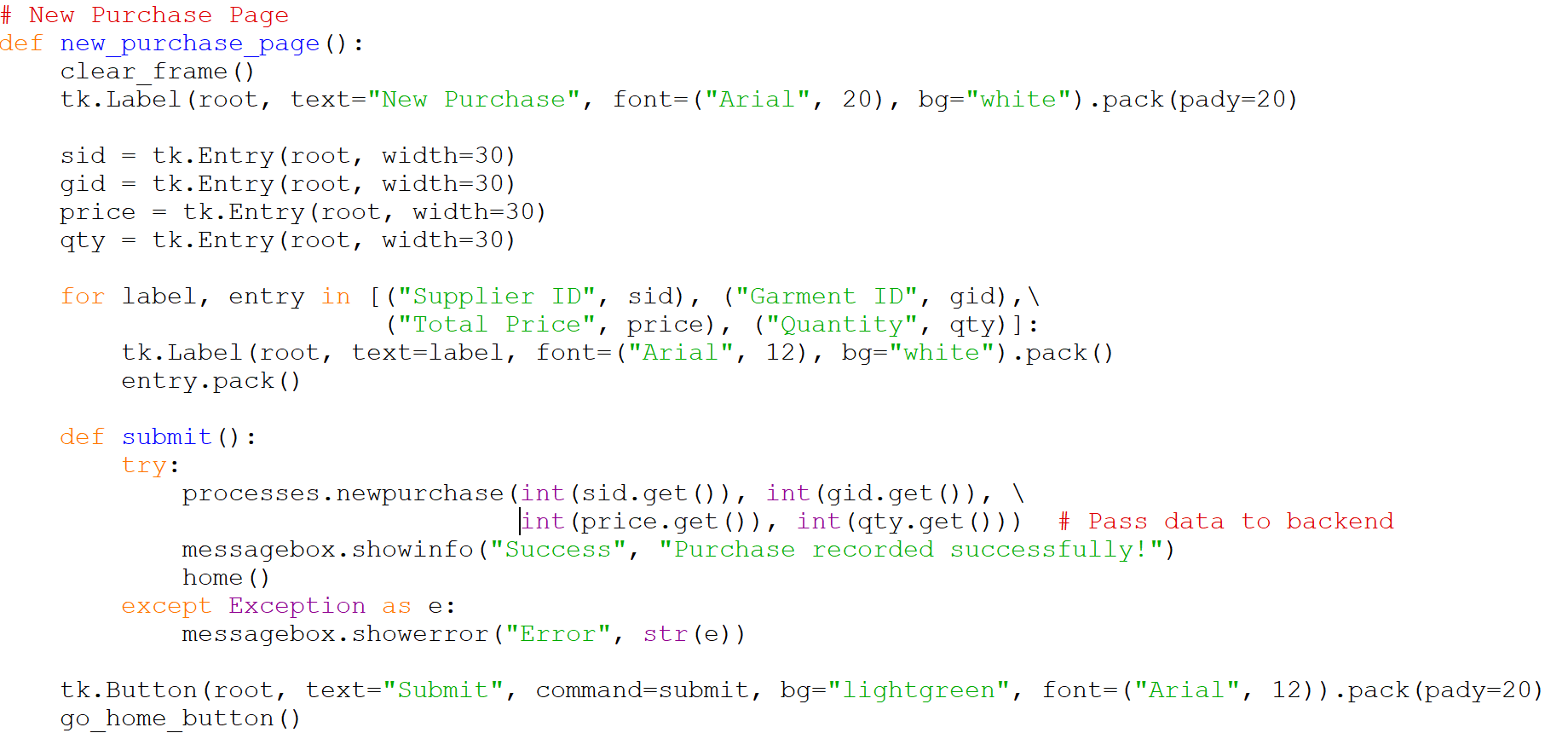




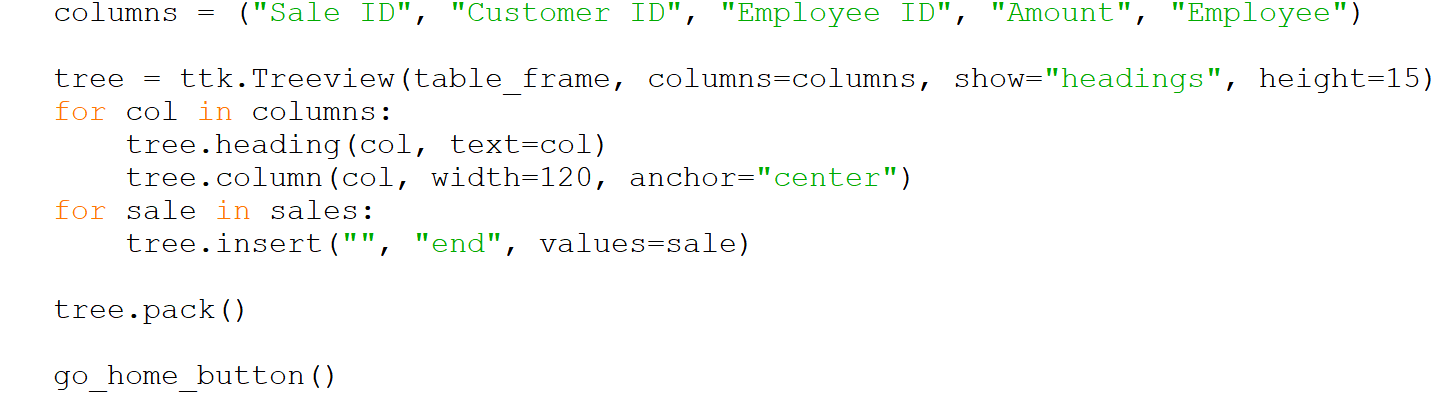


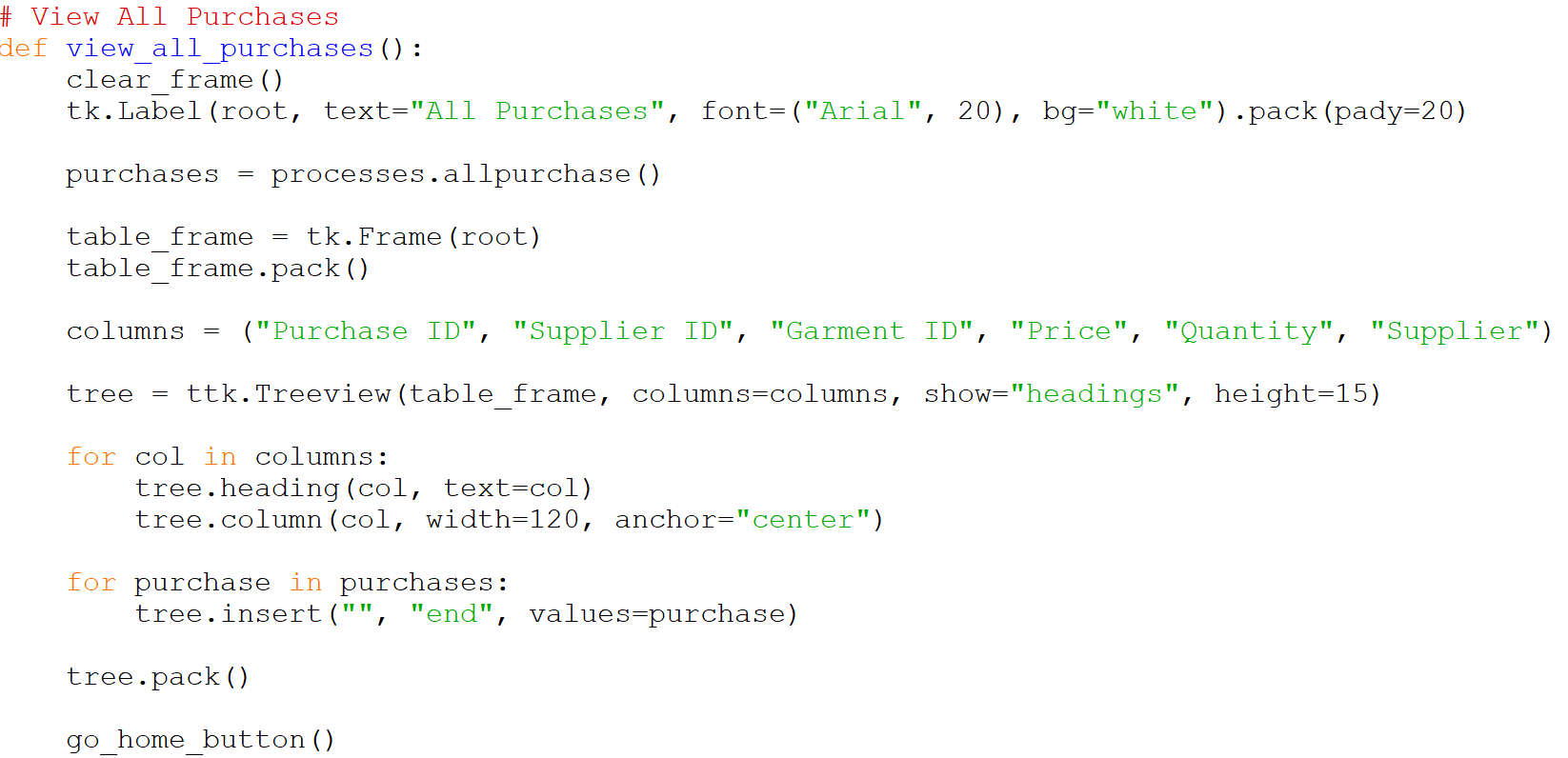


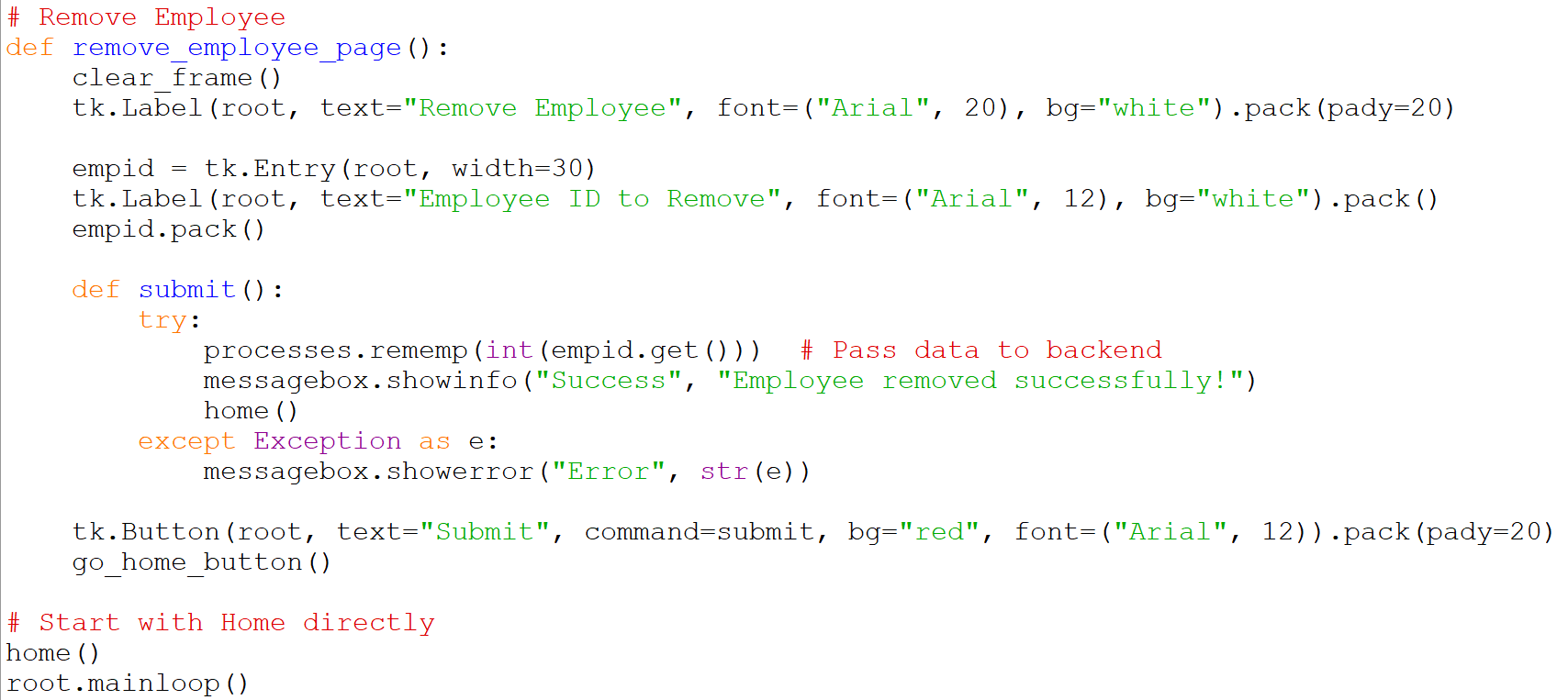






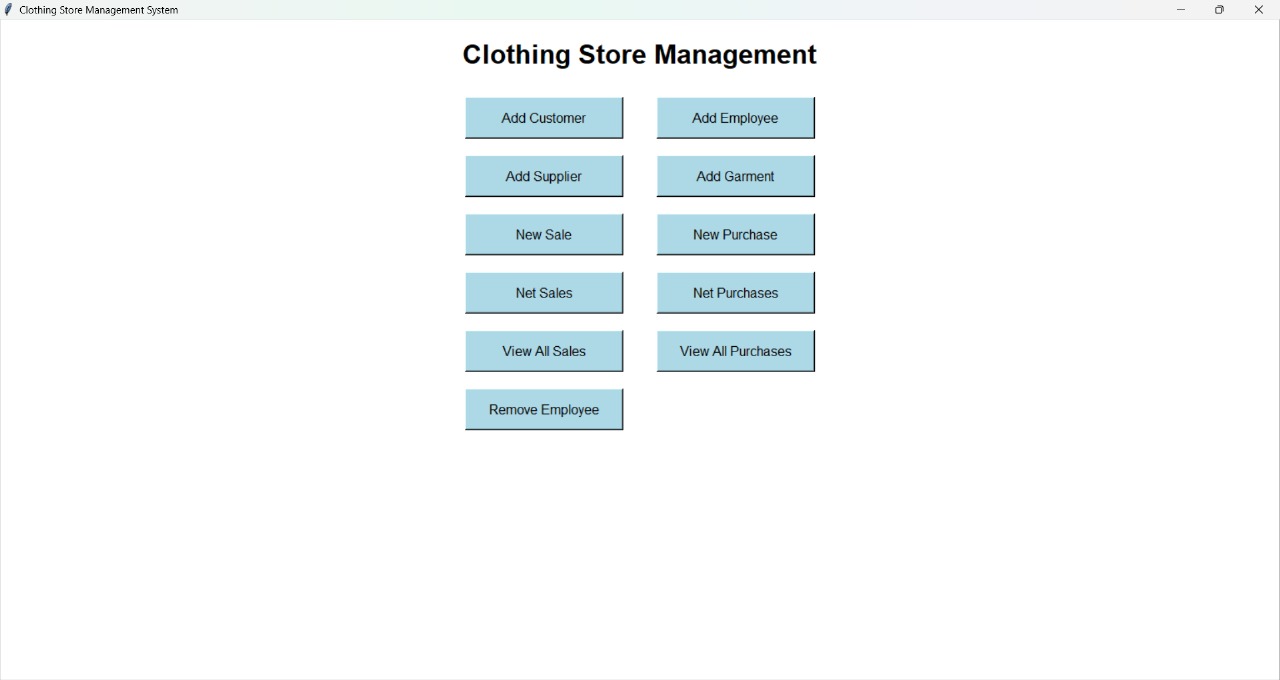


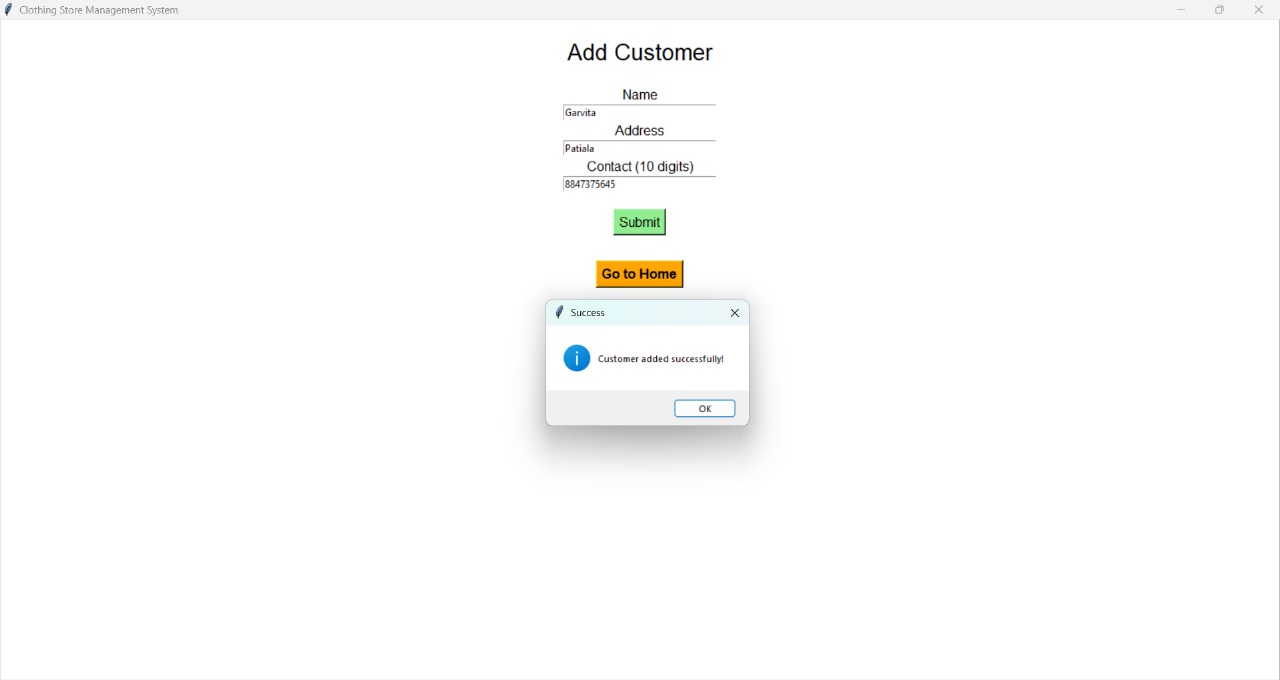


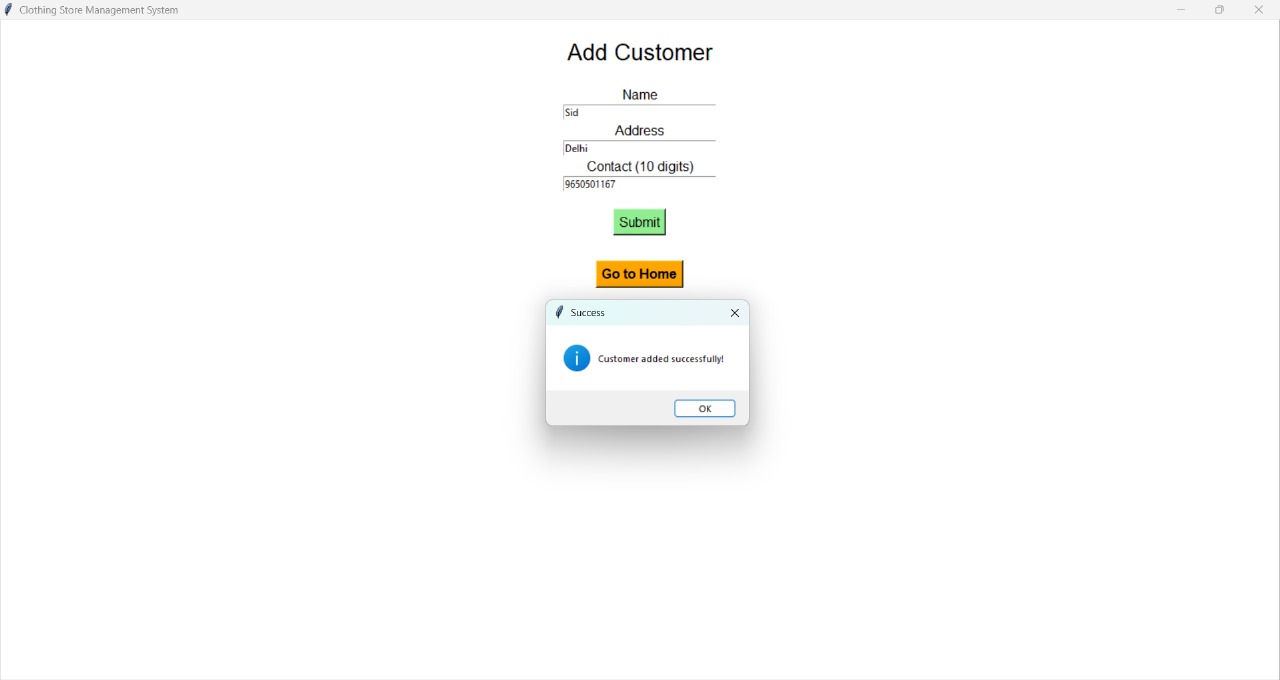


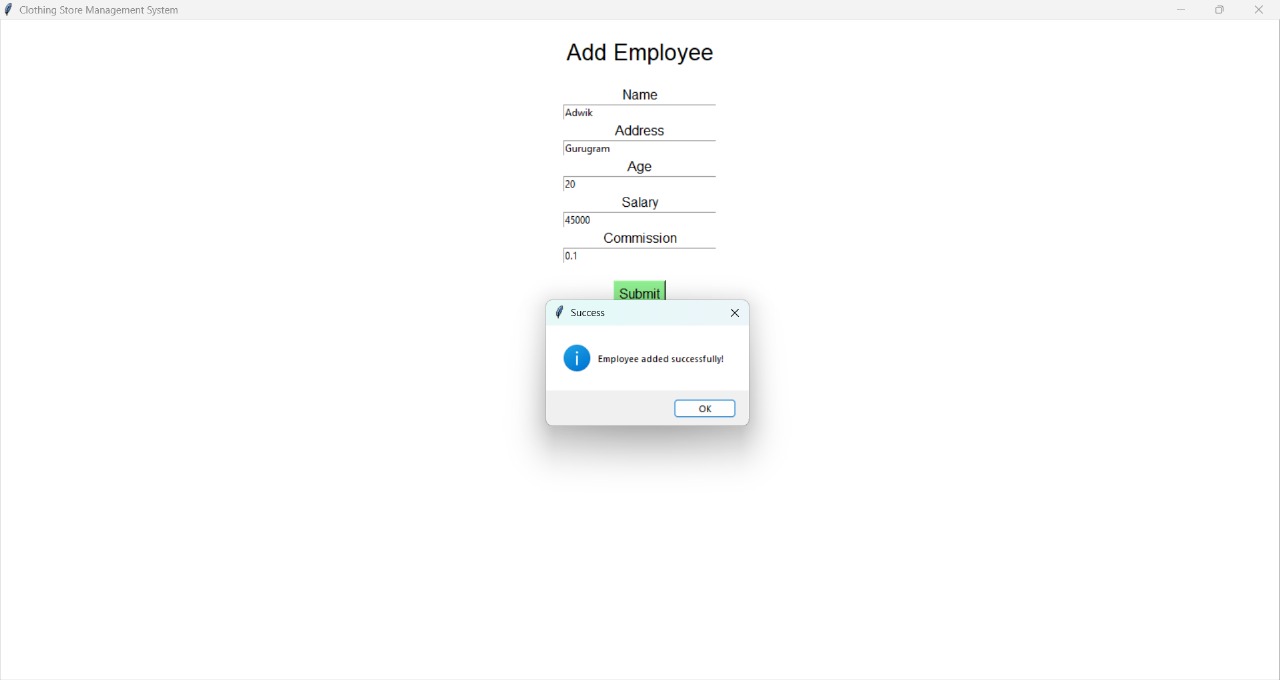
Chapter 7 SQL/PL/SQL Outputs and Discussion

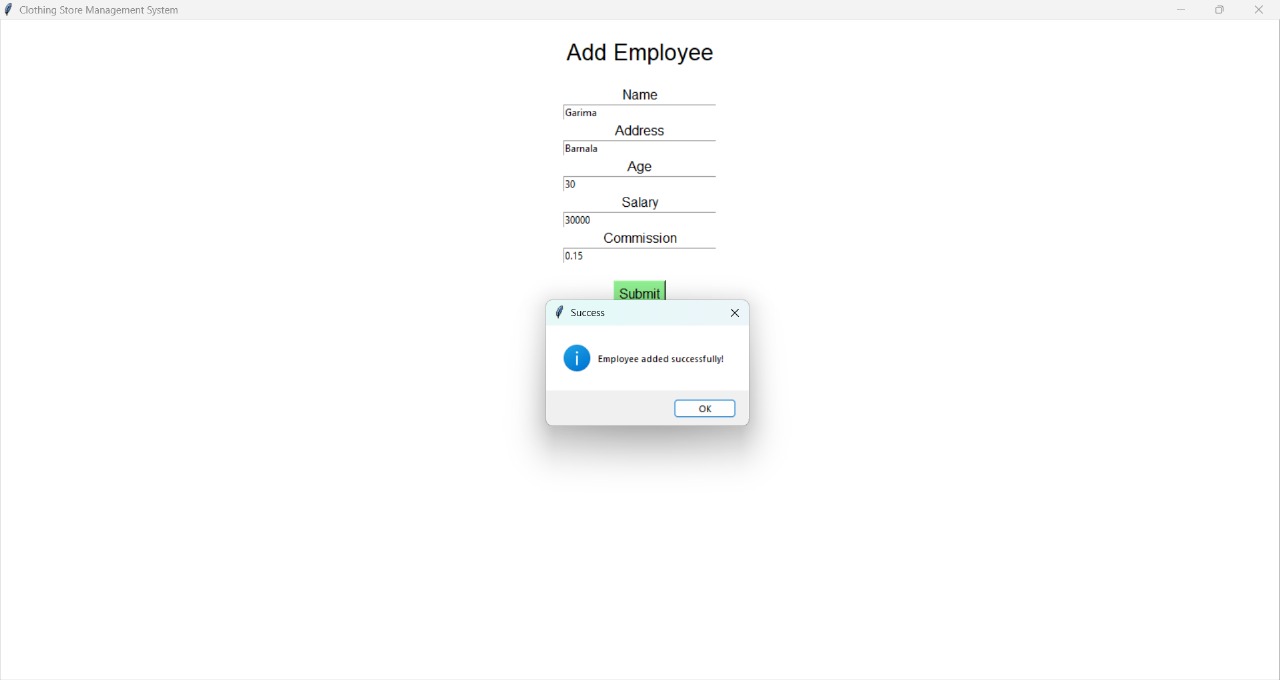
(Insert output results, observations, and discussion.)

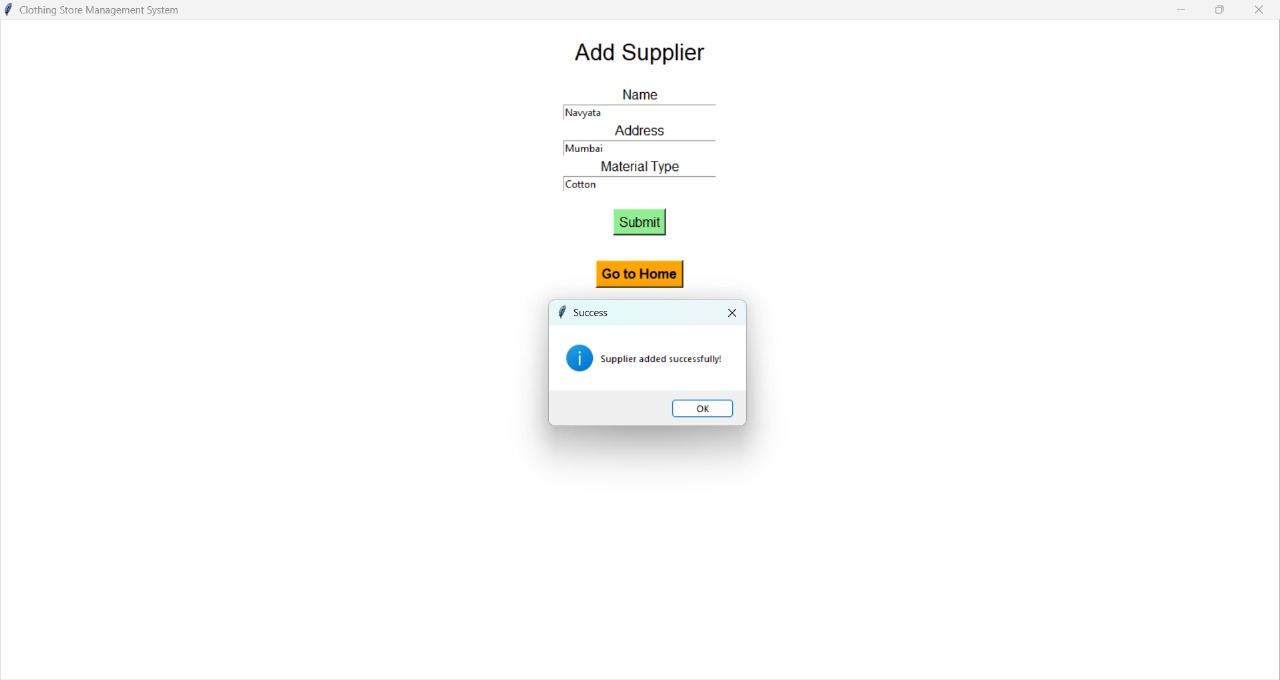


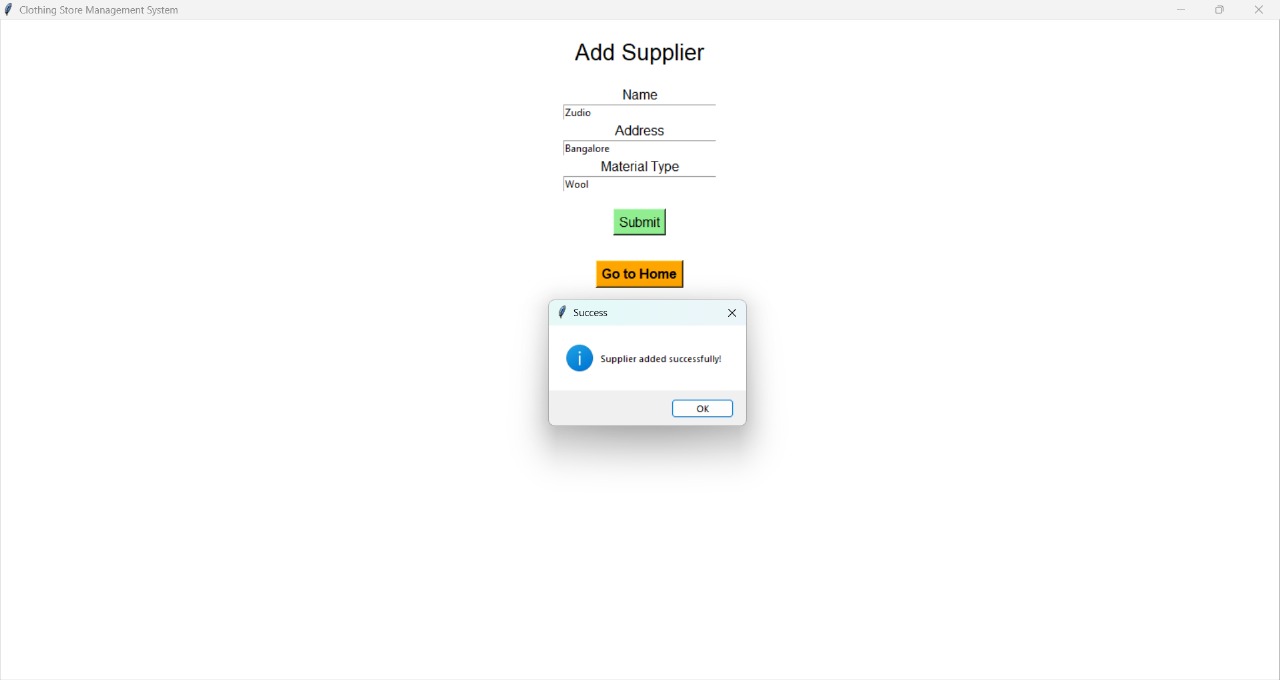
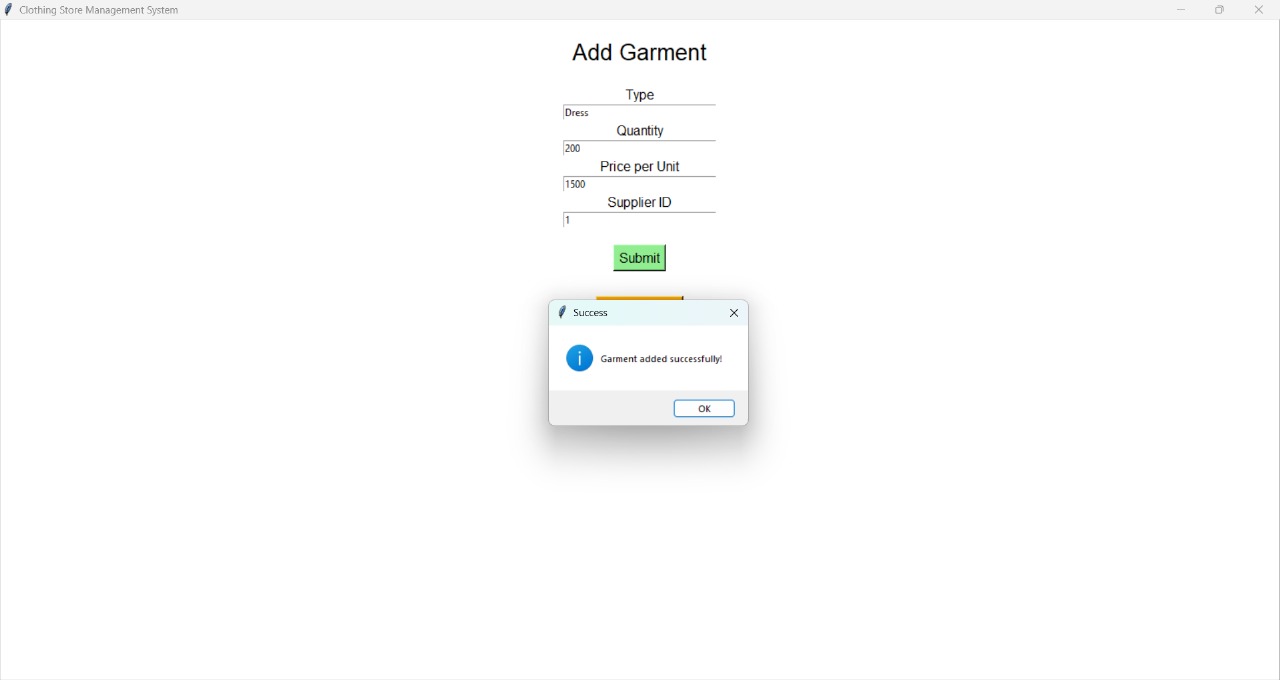


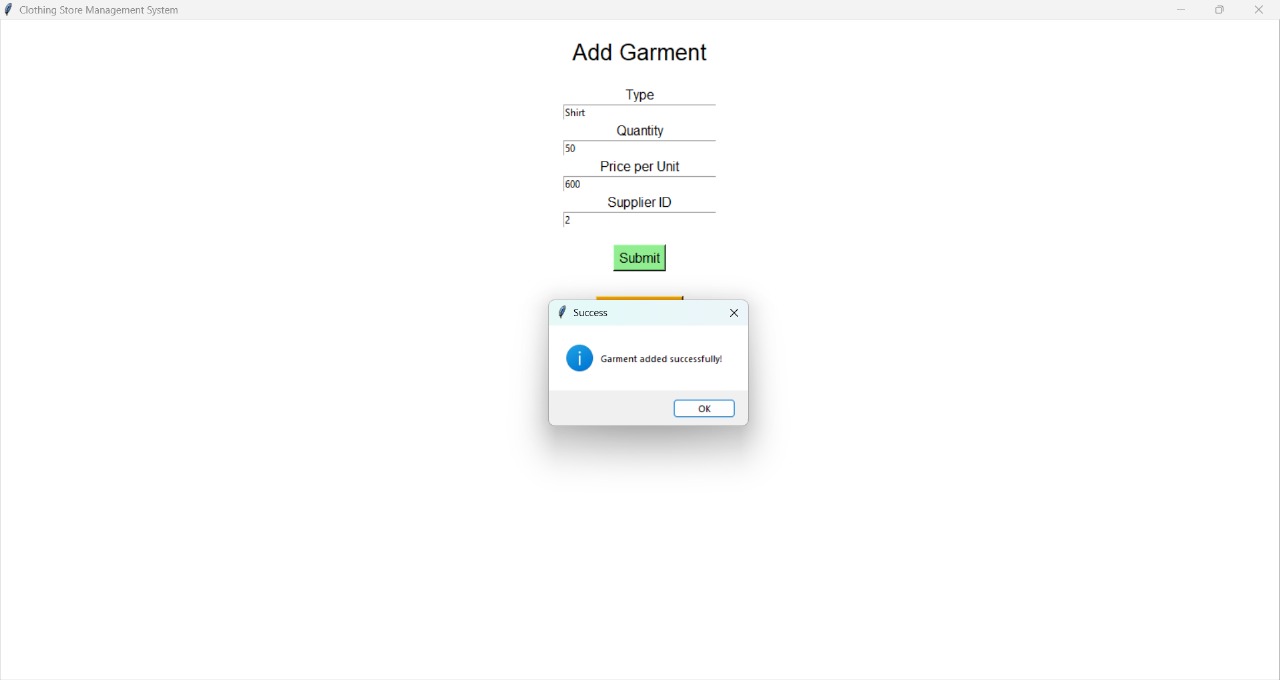


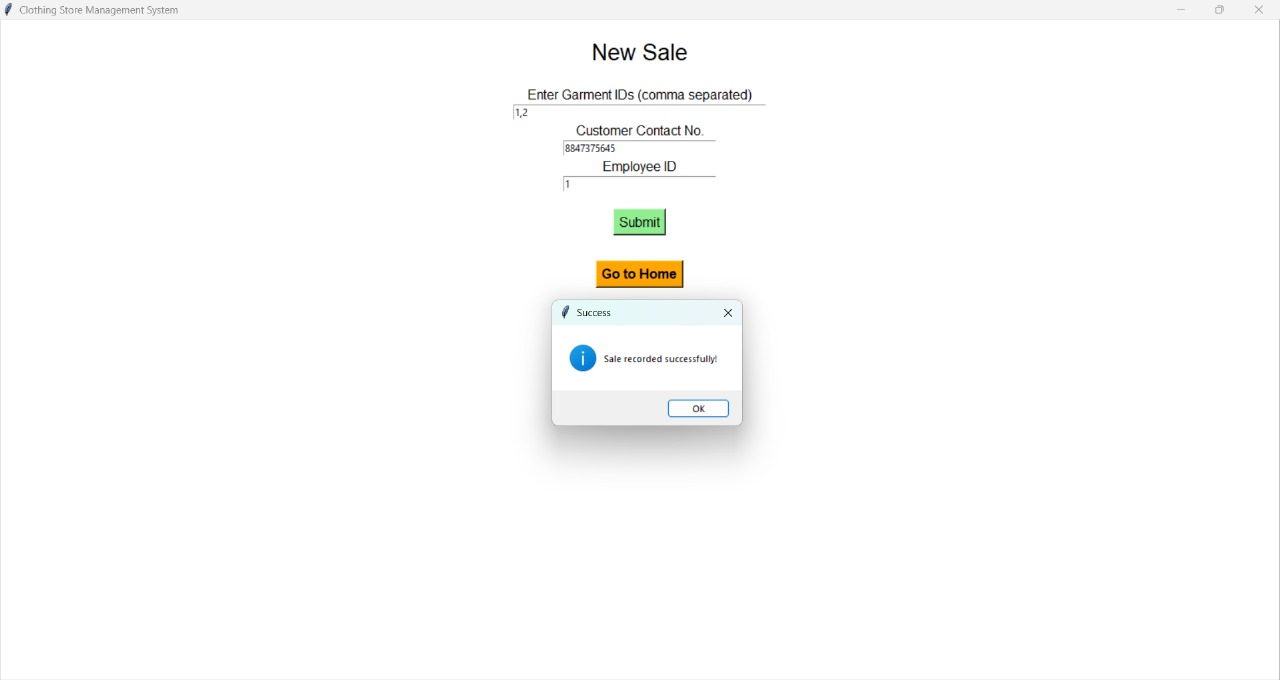


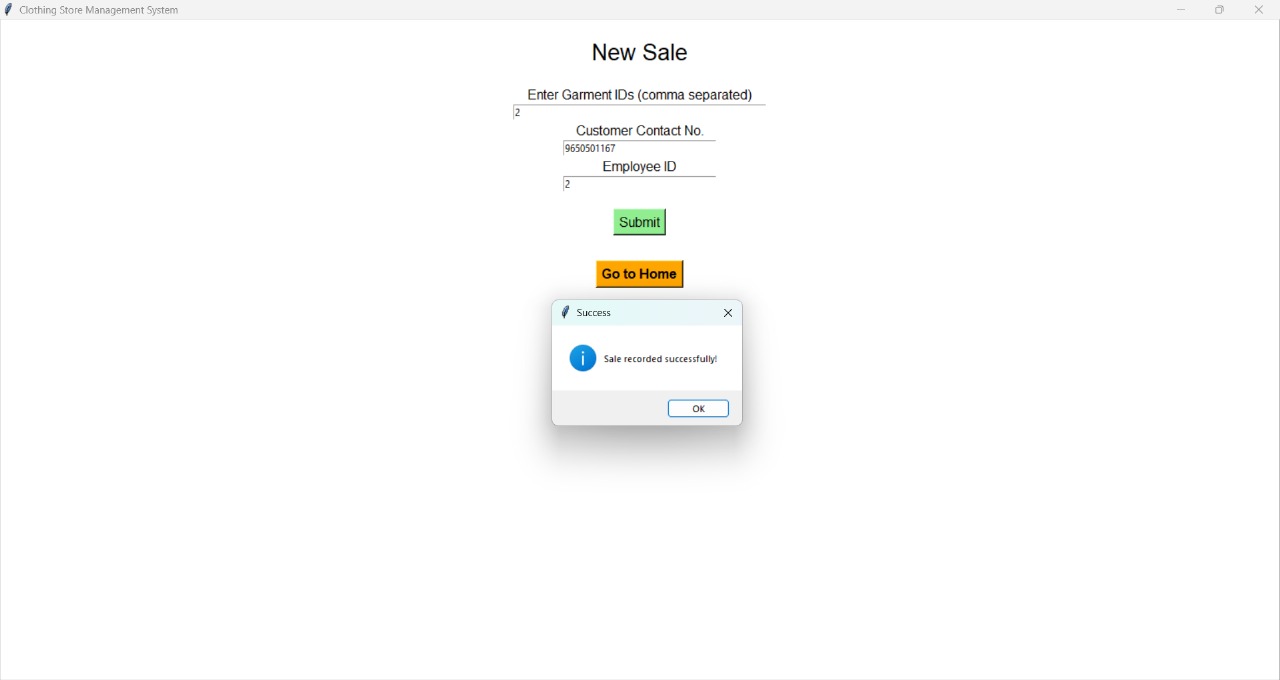


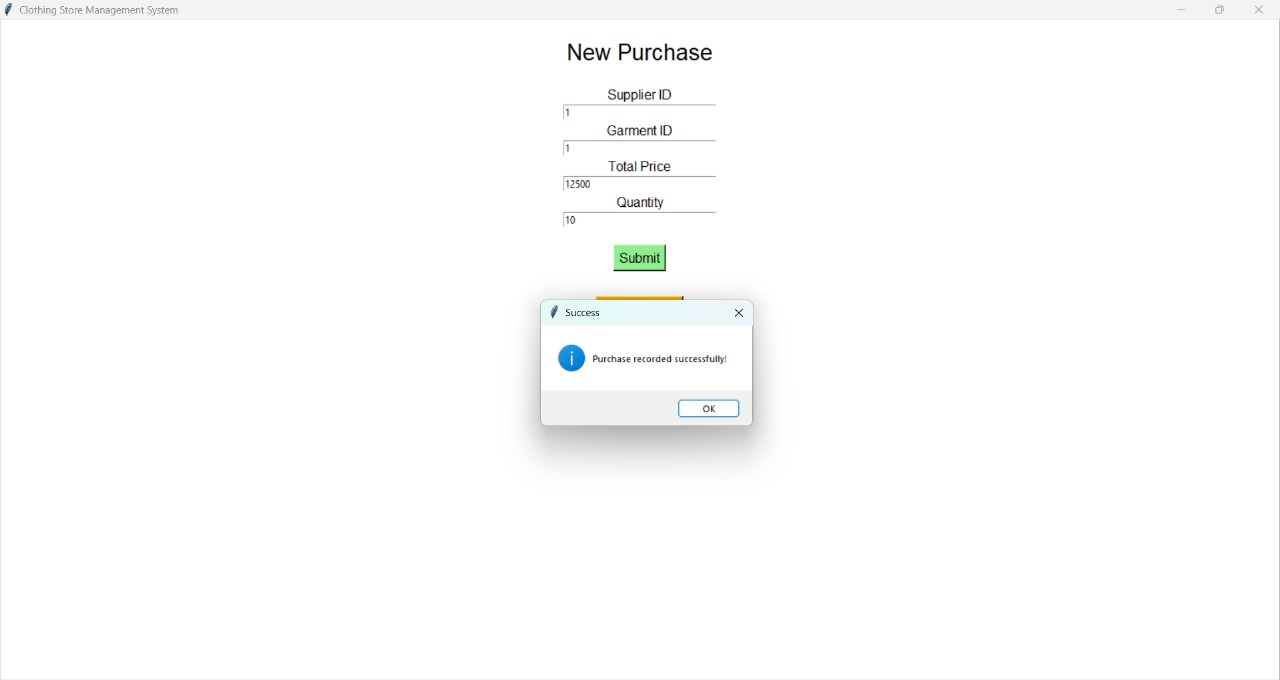


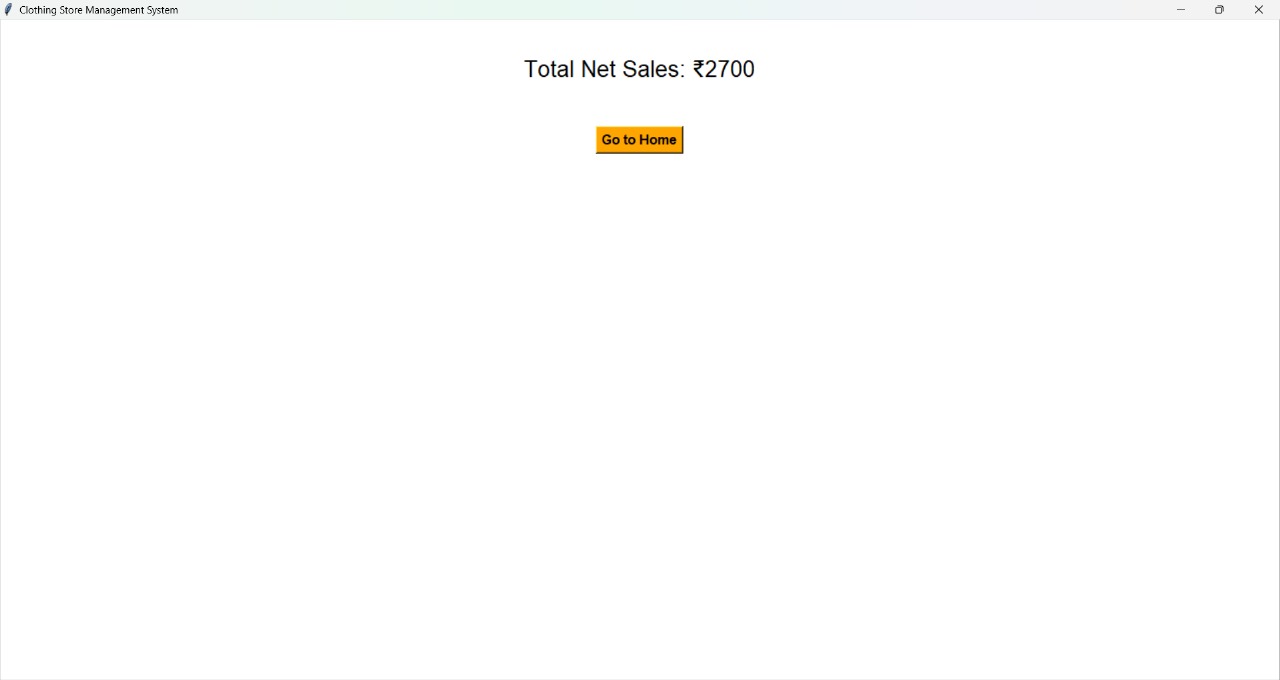
 

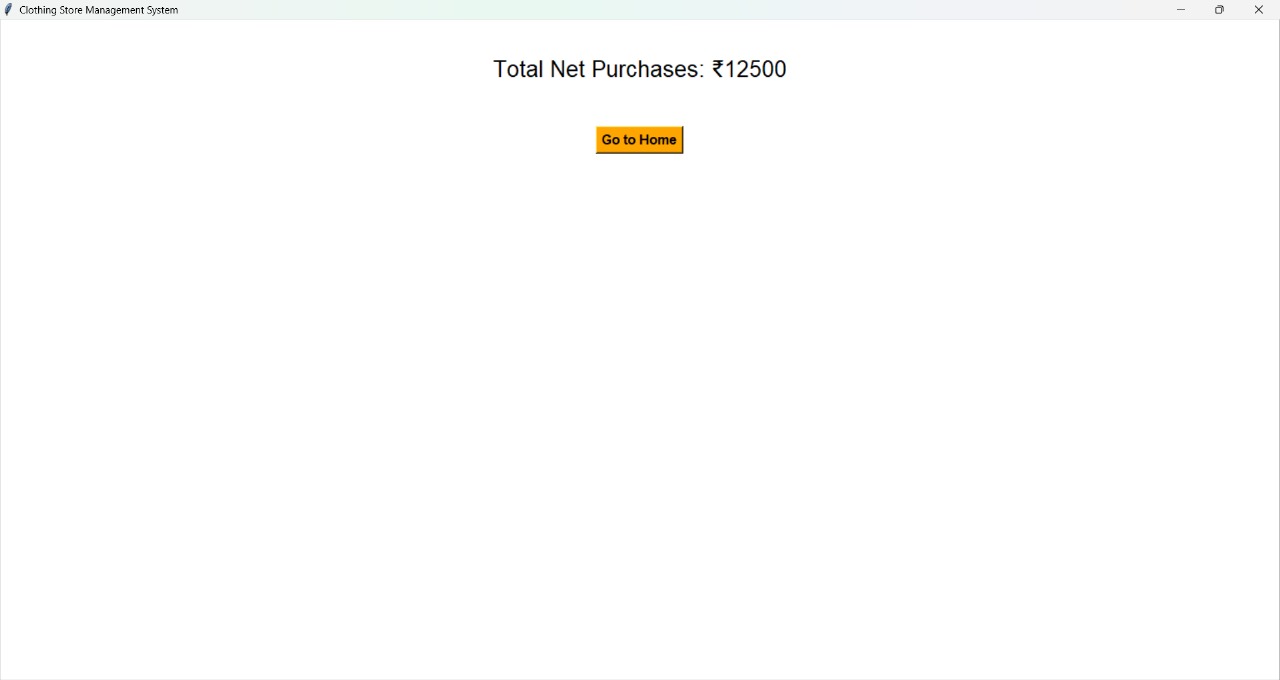


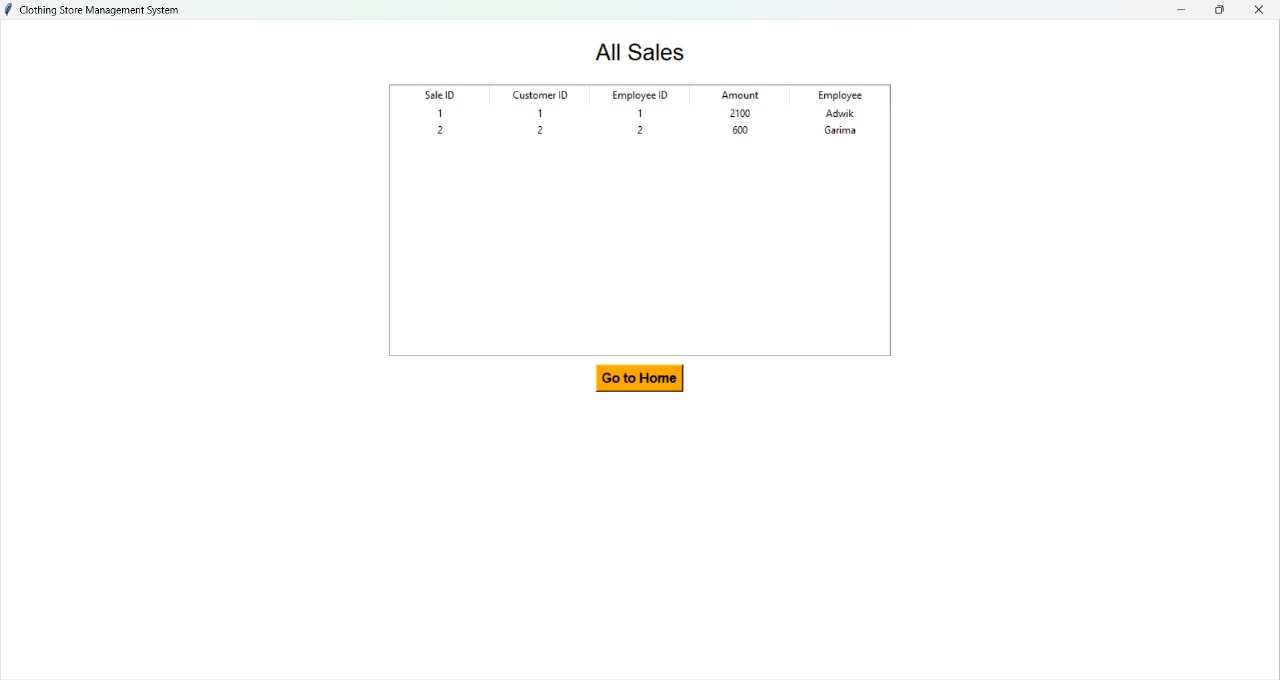


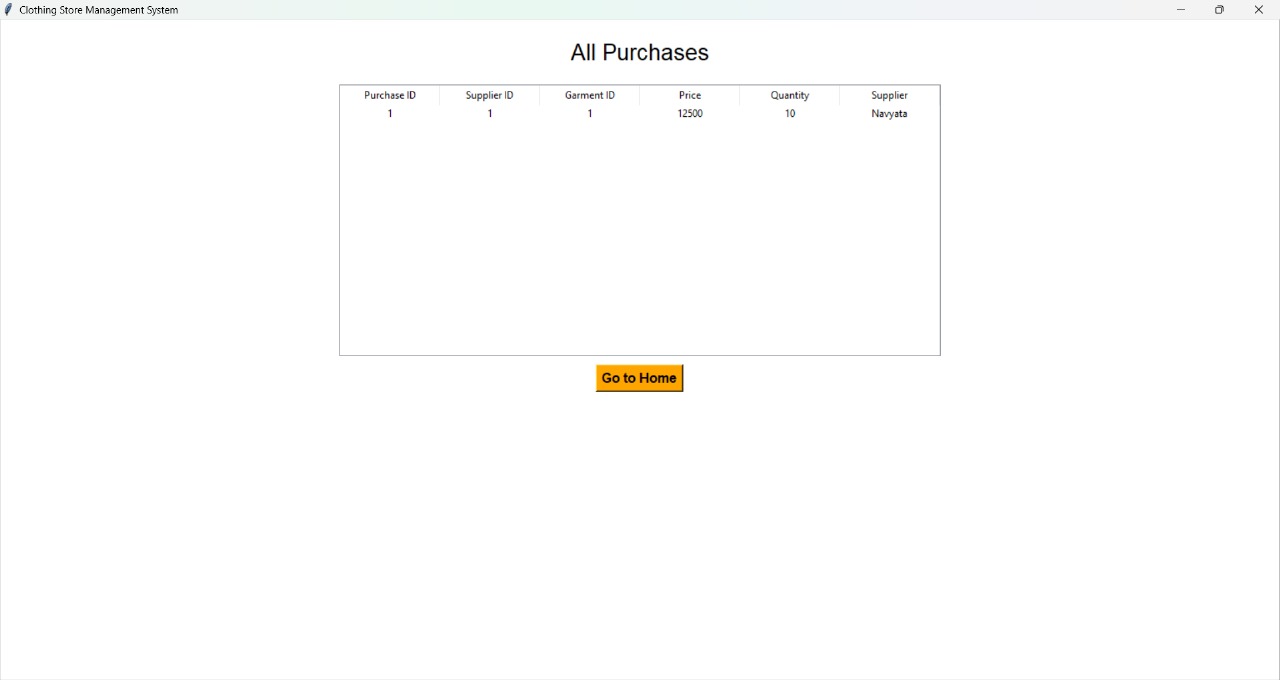


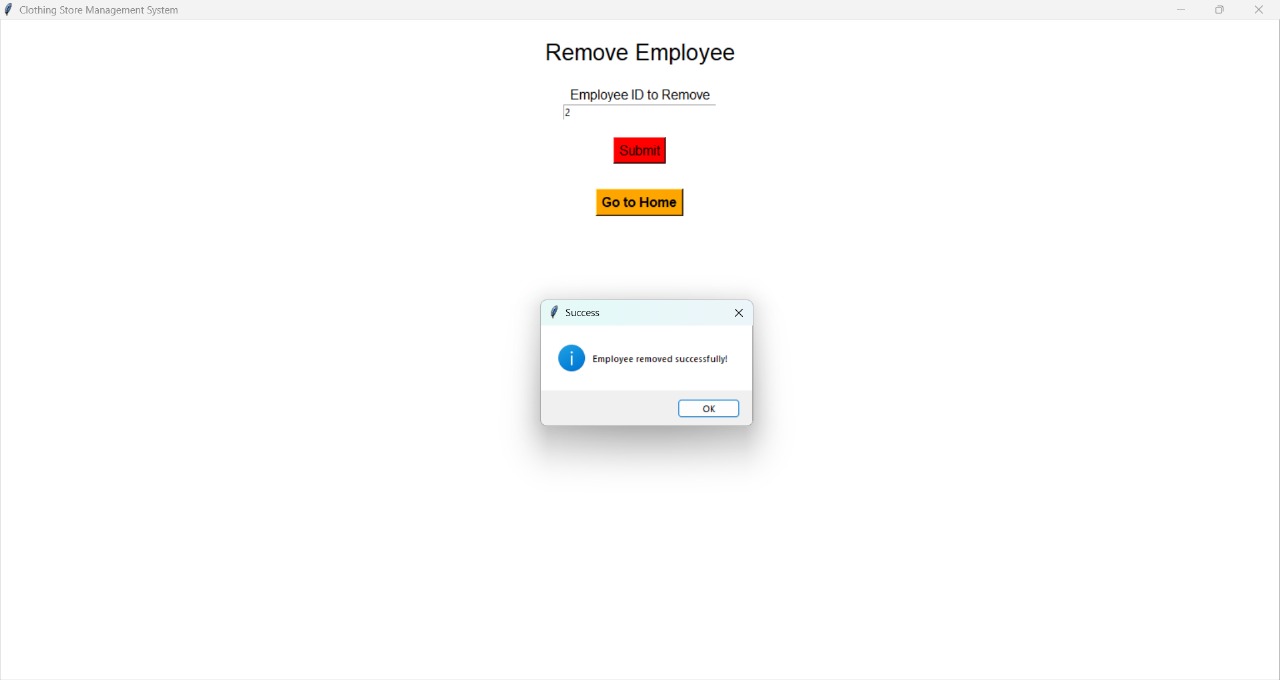












# References

* MySQL Documentation: https://dev.mysql.com/doc/
* Python MySQL Connector Documentation: https://pypi.org/project/ mysql-connector-python/
* Database System Concepts by Abraham Silberschatz, Henry F. Korth, and S. Sudarshan
* Fundamentals of Database Systems by Ramez Elmasri and Shamkant B. Navathe
* W3Schools - SQL and MySQL Tutorials: https://www.w3schools. com/sql/
* GeeksforGeeks - Python MySQL: https://www.geeksforgeeks.org/ python-database-connection-mysql/
* Class notes and instructor lectures
* Personal understanding and practical implementation