# PROJECT REPORT

**SUBMITTED FOR UCS310 DATABASE MANAGEMENT SYSTEM**

**Project Evaluation**

## **Submitted By:**

|  |  |
| --- | --- |
| **NAME OF THE STUDENT:** | **ROLL NO.** |
| Parth Vohra  Govind Singla  Bipasha Gupta  Aarushi Gupta | 102016044  102196004  102196005  102016060 |

**Batch:** 2CS10

**Submitted To:**

Dr. Manisha Kaushal

Assistant Professor



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**THAPAR INSTITUTE OF ENGINEERING AND TECHNOLOGY**

# ACKNOWLEDGEMENT

We would like to express our profound gratitude and special thanks to our mentor Dr. Manisha Kaushal Assistant Professor of Computer Science and Engineering Department, Thapar Institute Patiala, for their contributions to the completion of my project titled ShoppinglyX. Your useful advice and suggestions were really helpful to us during the project’s completion. In this aspect, we are eternally grateful to you.

# INDEX

|  |  |  |
| --- | --- | --- |
| **S.NO** | **TOPIC** | **PAGE NO.** |
| 1. | INTRODUCTION-SQL,PL/SQL | 4 |
| 2. | Problem Statement | 5 |
| 3. | ER-Diagram | 6 |
| 4. | ER Diagram to Tables | 7 |
| 5. | Normalization | 8 |
| 6. | TABLES | 9,10 |
| 7. | SQL/PLSQL | 11,12,13,14,15,16 |
| 9. | Output screenshots | 16,17,18 |

# INTRODUCTION: SQL & PL/SQL

SQL: Structured Query Language is a domain-specific language used in programming and designed for managing data held in a relational database management system (RDBMS), or for stream processing in a relational data stream management system (RDSMS). It is particularly useful in handling structured data where there are relations between different entities/variables of the data. First, it introduced the concept of accessing many records with one single command; and second, it eliminates the need to specify how to reach a record, e.g. with or without an index.

Originally based upon relational algebra. SQL consists of many types of statements, which may be informally classed as sublanguages, commonly: a data query language (DQL), a data definition language (DDL), a data control language (DCL), and a data manipulation language (DML). The scope of SQL includes data query, data manipulation (insert, update and delete), data definition (schema creation and modification), and data access control.

SQL was one of the first commercial languages for Edgar F. Codd's relational model. It became the most widely used database language.

PL/SQL: PL/SQL (Procedural Language for SQL) is Oracle Corporation's procedural extension for SQL and the Oracle relational database. PL/SQL is available in Oracle Database. Oracle Corporation usually extends PL/SQL functionality with each successive release of the Oracle Database.

PL/SQL includes procedural language elements such as conditions and loops. It allows declaration of constants and variables, procedures and functions, types and variables of those types, and triggers. It can handle exceptions (runtime errors). Arrays are supported involving the use of PL/SQL collections. One can create PL/SQL units such as procedures, functions, packages, types, and triggers, which are stored in the database for reuse by applications that use any of the Oracle Database programmatic interface

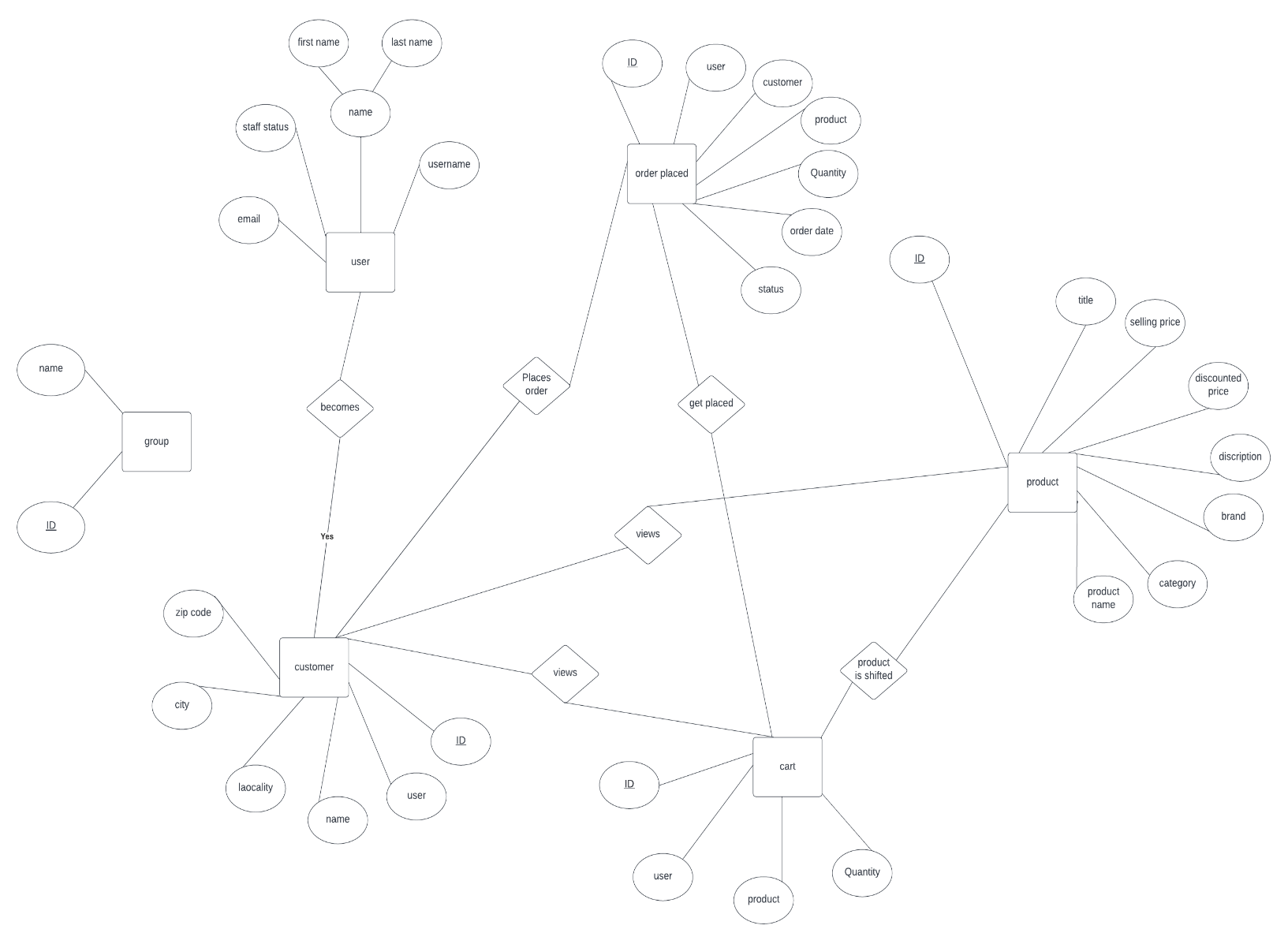
**Problem Statement**

Enhancement of Online Shopping Systems with features to suit modern needs.

The project is about building an **Online Shopping Platform(e-commerce)** to suit modern needs. The proposed website will be **highly integrated** with the sellers and will be **easy to use** for the customer. The current implementation of the e-commerce website portal is outdated and has various pitfalls. It has a **poor User Interface**, **User Experience**, and **inconsistent** **design**. They don’t integrate well with the sellers inline and lack essential features like a issue forum. Their mobile application suffers from the same problems. A lot of features remain unused because of poor design and implementation. Users must look for different sites to go through the resources required to buy the product. For the same product, students check different resources and websites and get stuck in ensuring which one they should look to buy a cheaper and good product.

Even users can upload photos of the ordered product in one place. Our goal is to save customers time in looking for hours as to which resource to use to buy a good and cheaper product. Our goal is measured by how good the feedback we get from our users is.

**ER-DIAGRAM**

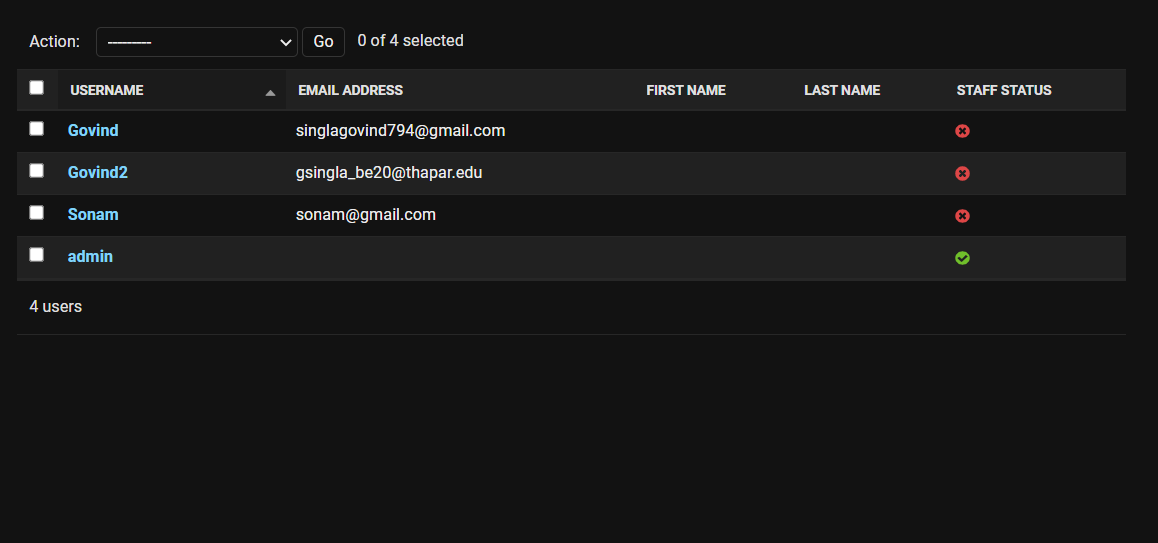
****

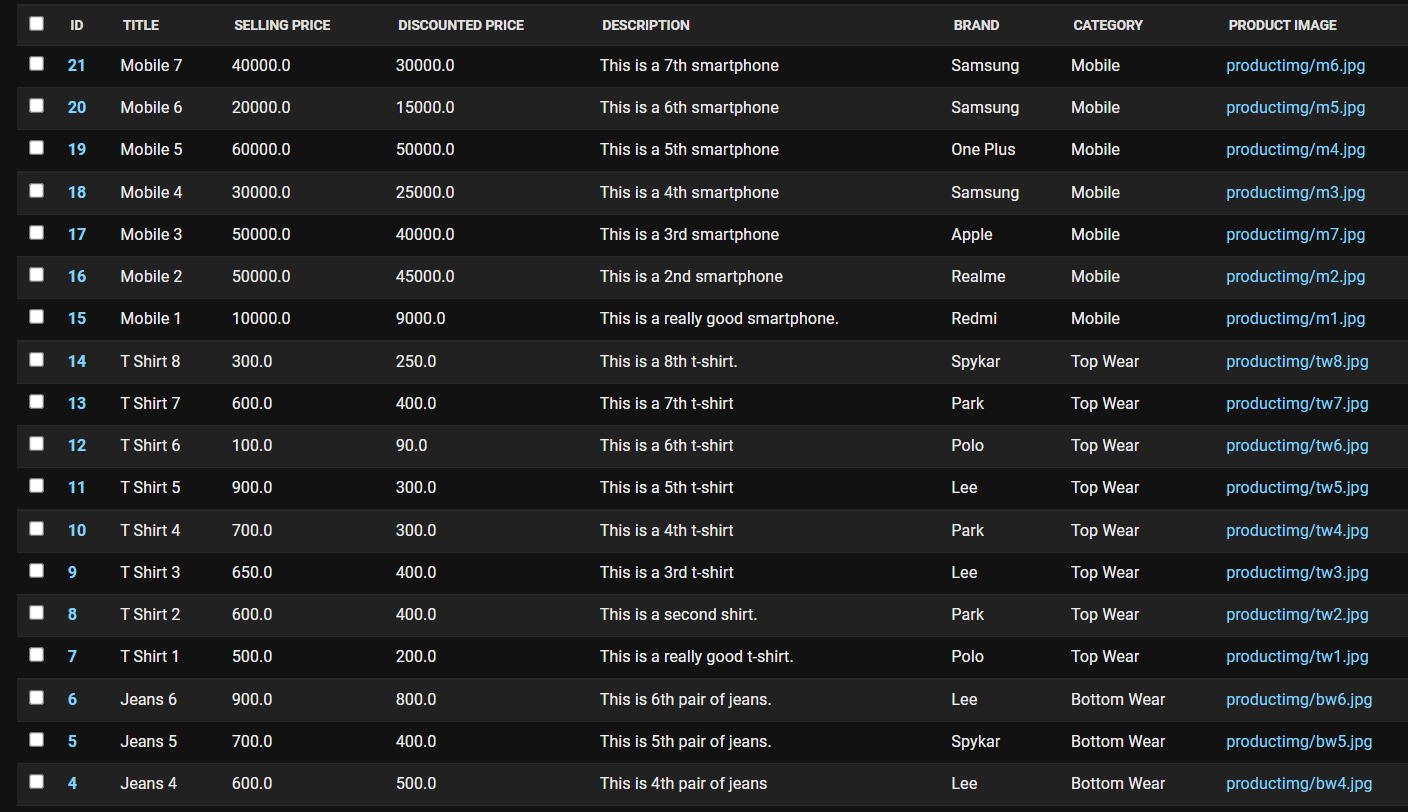
**ER-DIAGRAM TO TABLES**

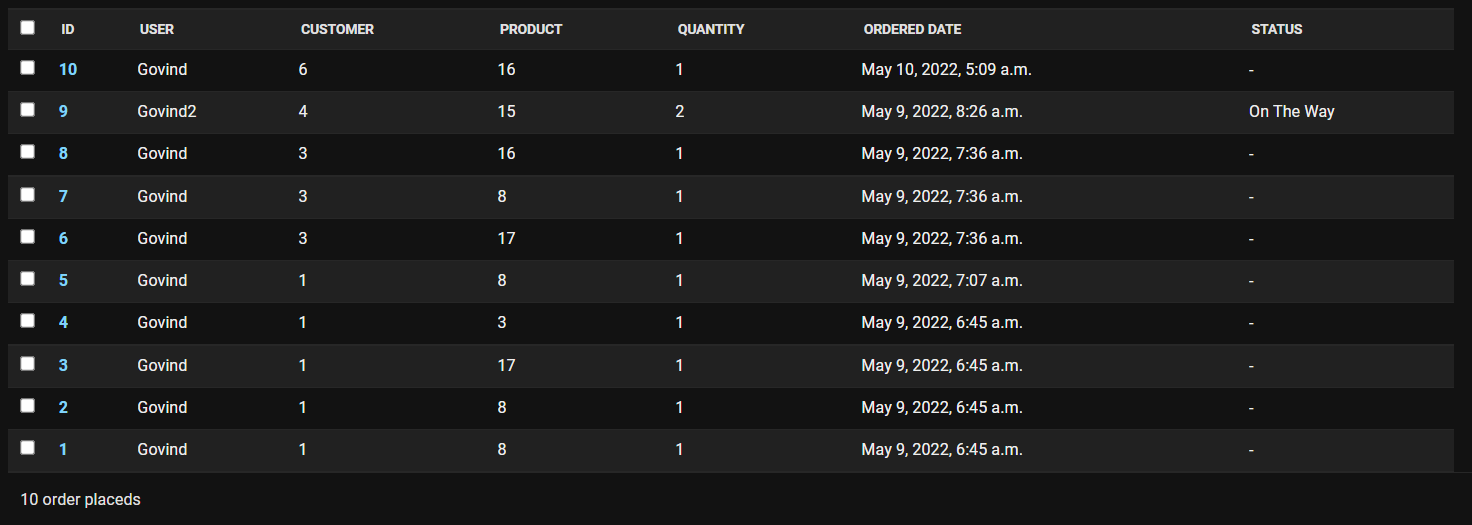
|  |  |
| --- | --- |
| TABLE | ATTRIBUTES |
| cart | ID, user, product, Quantity |
| user | email, staff status, name, firstname, lastname, username |
| customer | zipcode, city, locality, name, user, ID |
| Order placed | Id, user, customer, product, quantity, ordered date, status |
| product | Id, title, selling price, discounted price, description, brand, category, product name |

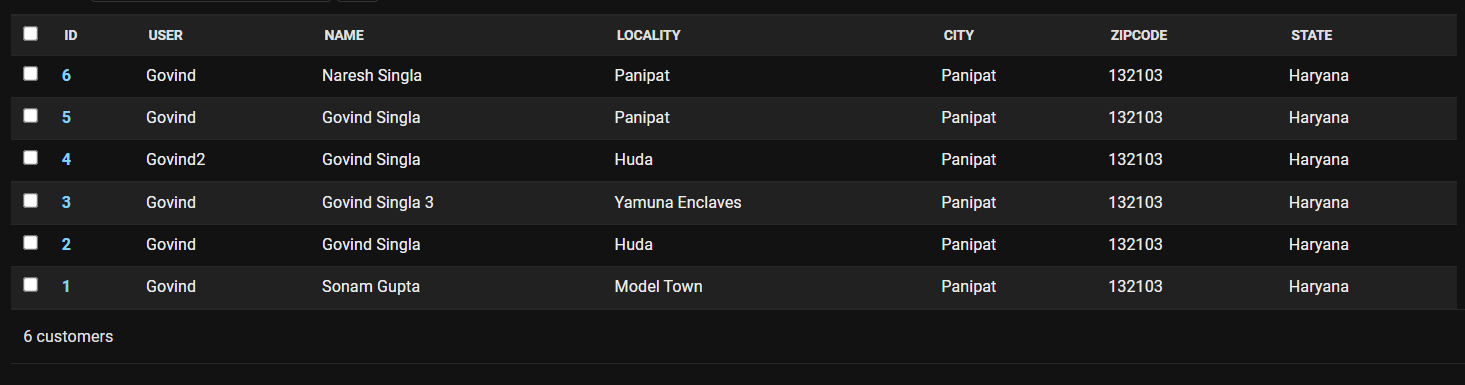
**Normalization**

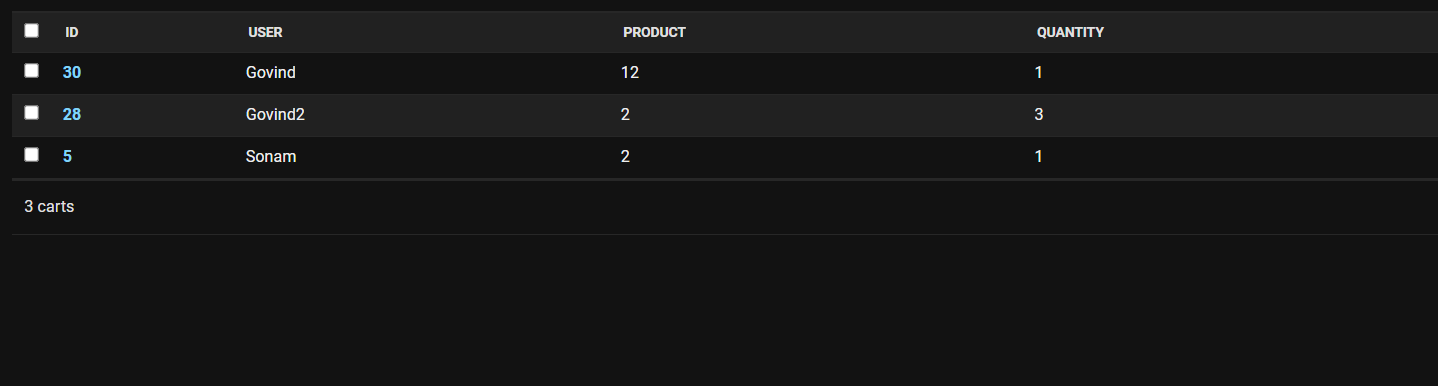
# TABLES











**SQL/PLSQL Code**

Customer Table

customer\_table = "CREATE TABLE app\_customers (id integer NOT NULL PRIMARY KEY AUTOINCREMENT, user varchar(200) NOT NULL REFERENCES auth\_user ('id') DEFERRABLE INITIALLY DEFERRED, name varchar(200) NOT NULL, locality varchar(200) NOT NULL, city varchar(50) NOT NULL, zipcode integer NOT NULL, state varchar(50) NOT NULL)"

Order Placed Table

order\_placed\_table = "CREATE TABLE app\_orderplaceds (id integer NOT NULL PRIMARY KEY AUTOINCREMENT, user varchar(200) NOT NULL REFERENCES auth\_user ('id') DEFERRABLE INITIALLY DEFERRED, customer bigint NOT NULL REFERENCES app\_customer ('id') DEFERRABLE INITIALLY DEFERRED, product bigint NOT NULL REFERENCES app\_product ('id') DEFERRABLE INITIALLY DEFERRED, quantity integer unsigned NOT NULL CHECK ('quantity' >= 0), ordered\_date datetime NOT NULL, status varchar(50) NOT NULL)"

Product Table

product\_table = "CREATE TABLE app\_products (id integer NOT NULL PRIMARY KEY AUTOINCREMENT, title varchar(100) NOT NULL, selling\_price real NOT NULL, discounted\_price real NOT NULL, description text NOT NULL, brand varchar(100) NOT NULL, category varchar(2) NOT NULL, product\_image varchar(100) NOT NULL)"

Group Table

group\_table = "CREATE TABLE auth\_groups (id integer NOT NULL PRIMARY KEY AUTOINCREMENT, name varchar(150) NOT NULL UNIQUE)"

User Table

user\_table = "CREATE TABLE auth\_users (id integer NOT NULL PRIMARY KEY AUTOINCREMENT, password varchar(128) NOT NULL, last\_login datetime NULL, is\_superuser bool NOT NULL, username varchar(150) NOT NULL UNIQUE, last\_name varchar(150) NOT NULL, email varchar(254) NOT NULL, is\_staff bool NOT NULL, is\_active bool NOT NULL, date\_joined datetime NOT NULL, first\_name varchar(150) NOT NULL)"

Using Procedures

plus\_cart = "create or REPLACE procedure cart\_plus(user\_name in varchar) AS begin Update app\_orderplaceds Set quantity = quantity + 1 Where user = user\_name; end;"  ### will be a procedure

minus\_cart = "create or REPLACE procedure cart\_minus(user\_name in varchar) AS begin Update app\_orderplaceds Set quantity = quantity - 1 Where user = user\_name; end;"  ### will be a procedure

remove\_cart = "create or REPLACE procedure cart\_remove(user\_name in varchar) AS begin delete from app\_orderplaceds Where user = user\_name; end;"  ### will be a procedure

cart\_insert = "insert into app\_carts values(?,?,?,?)"

address\_view = "select \* from app\_customers where user=?"

orders\_view = "select \* from app\_orderplaceds where user=?"

customer\_registration = "insert into app\_customers values(?,?)"

order\_place\_insert = "insert into app\_orderplaceds values(?,?,?,?,?,?,?)"

Using Cursors

filter\_mobile = "DECLARE c\_id app\_product.id%type; c\_title app\_product.title%type; c\_sellingprice app\_product.selling\_price%type; c\_discountedprice app\_product.discounted\_price%type; c\_description app\_product.description%type; c\_brand app\_product.brand%type; c\_category app\_product.category%type; c\_productimage app\_product.product\_image%type; CURSOR filter\_result\_mobile is SELECT id, title, selling\_price,discounted\_price ,description,brand,category,product\_image FROM app\_products; BEGIN OPEN filter\_result\_mobile; LOOP FETCH filter\_result\_mobile into c\_id, c\_title,c\_sellingprice,c\_discountedprice ,c\_description ,c\_brand ,c\_category ,c\_productimage; EXIT WHEN filter\_result\_mobile%notfound; END LOOP; CLOSE filter\_result\_mobile; END;"

filter\_mobile\_redmi = "DECLARE c\_id app\_product.id%type; c\_title app\_product.title%type; c\_sellingprice app\_product.selling\_price%type; c\_discountedprice app\_product.discounted\_price%type; c\_description app\_product.description%type; c\_brand app\_product.brand%type; c\_category app\_product.category%type; c\_productimage app\_product.product\_image%type; CURSOR filter\_result\_mobile is SELECT id, title, selling\_price,discounted\_price ,description,brand,category,product\_image FROM app\_products where category=’Redmi’; BEGIN OPEN filter\_result\_mobile; LOOP FETCH filter\_result\_mobile into c\_id, c\_title,c\_sellingprice,c\_discountedprice ,c\_description ,c\_brand ,c\_category ,c\_productimage; EXIT WHEN filter\_result\_mobile%notfound; END LOOP; CLOSE filter\_result\_mobile; END;"

filter\_mobile\_samsung = "DECLARE c\_id app\_product.id%type; c\_title app\_product.title%type; c\_sellingprice app\_product.selling\_price%type; c\_discountedprice app\_product.discounted\_price%type; c\_description app\_product.description%type; c\_brand app\_product.brand%type; c\_category app\_product.category%type; c\_productimage app\_product.product\_image%type; CURSOR filter\_result\_mobile is SELECT id, title, selling\_price,discounted\_price ,description,brand,category,product\_image FROM app\_products where category=’Samsung’; BEGIN OPEN filter\_result\_mobile; LOOP FETCH filter\_result\_mobile into c\_id, c\_title,c\_sellingprice,c\_discountedprice ,c\_description ,c\_brand ,c\_category ,c\_productimage; EXIT WHEN filter\_result\_mobile%notfound; END LOOP; CLOSE filter\_result\_mobile; END;"

filter\_mobile\_below = "DECLARE c\_id app\_product.id%type; c\_title app\_product.title%type; c\_sellingprice app\_product.selling\_price%type; c\_discountedprice app\_product.discounted\_price%type; c\_description app\_product.description%type; c\_brand app\_product.brand%type; c\_category app\_product.category%type; c\_productimage app\_product.product\_image%type; CURSOR filter\_result\_mobile is SELECT id, title, selling\_price,discounted\_price ,description,brand,category,product\_image FROM app\_products where discounted\_price<20000; BEGIN OPEN filter\_result\_mobile; LOOP FETCH filter\_result\_mobile into c\_id, c\_title,c\_sellingprice,c\_discountedprice ,c\_description ,c\_brand ,c\_category ,c\_productimage; EXIT WHEN filter\_result\_mobile%notfound; END LOOP; CLOSE filter\_result\_mobile; END;"

filter\_mobile\_above = "DECLARE c\_id app\_product.id%type; c\_title app\_product.title%type; c\_sellingprice app\_product.selling\_price%type; c\_discountedprice app\_product.discounted\_price%type; c\_description app\_product.description%type; c\_brand app\_product.brand%type; c\_category app\_product.category%type; c\_productimage app\_product.product\_image%type; CURSOR filter\_result\_mobile is SELECT id, title, selling\_price,discounted\_price ,description,brand,category,product\_image FROM app\_products where discounted\_price>20000; BEGIN OPEN filter\_result\_mobile; LOOP FETCH filter\_result\_mobile into c\_id, c\_title,c\_sellingprice,c\_discountedprice ,c\_description ,c\_brand ,c\_category ,c\_productimage; EXIT WHEN filter\_result\_mobile%notfound; END LOOP; CLOSE filter\_result\_mobile; END;"

filter\_topwear = "DECLARE c\_id app\_product.id%type; c\_title app\_product.title%type; c\_sellingprice app\_product.selling\_price%type; c\_discountedprice app\_product.discounted\_price%type; c\_description app\_product.description%type; c\_brand app\_product.brand%type; c\_category app\_product.category%type; c\_productimage app\_product.product\_image%type; CURSOR filter\_result\_topwear is SELECT id, title, selling\_price,discounted\_price ,description,brand,category,product\_image FROM app\_products; BEGIN OPEN filter\_result\_topwear; LOOP FETCH filter\_result\_topwear into c\_id, c\_title,c\_sellingprice,c\_discountedprice ,c\_description ,c\_brand ,c\_category ,c\_productimage; EXIT WHEN filter\_result\_topwear%notfound; END LOOP; CLOSE filter\_result\_topwear; END;"

filter\_topwear\_lee = "DECLARE c\_id app\_product.id%type; c\_title app\_product.title%type; c\_sellingprice app\_product.selling\_price%type; c\_discountedprice app\_product.discounted\_price%type; c\_description app\_product.description%type; c\_brand app\_product.brand%type; c\_category app\_product.category%type; c\_productimage app\_product.product\_image%type; CURSOR filter\_result\_topwear is SELECT id, title, selling\_price,discounted\_price ,description,brand,category,product\_image FROM app\_products where category=’Lee’; BEGIN OPEN filter\_result\_topwear; LOOP FETCH filter\_result\_topwear into c\_id, c\_title,c\_sellingprice,c\_discountedprice ,c\_description ,c\_brand ,c\_category ,c\_productimage; EXIT WHEN filter\_result\_topwear%notfound; END LOOP; CLOSE filter\_result\_topwear; END;"  ## will be a cursor

filter\_topwear\_spykar = "DECLARE c\_id app\_product.id%type; c\_title app\_product.title%type; c\_sellingprice app\_product.selling\_price%type; c\_discountedprice app\_product.discounted\_price%type; c\_description app\_product.description%type; c\_brand app\_product.brand%type; c\_category app\_product.category%type; c\_productimage app\_product.product\_image%type; CURSOR filter\_result\_topwear is SELECT id, title, selling\_price,discounted\_price ,description,brand,category,product\_image FROM app\_products where category=’SpyKar’; BEGIN OPEN filter\_result\_topwear; LOOP FETCH filter\_result\_topwear into c\_id, c\_title,c\_sellingprice,c\_discountedprice ,c\_description ,c\_brand ,c\_category ,c\_productimage; EXIT WHEN filter\_result\_topwear%notfound; END LOOP; CLOSE filter\_result\_topwear; END;"

filter\_topwear\_above = "DECLARE c\_id app\_product.id%type; c\_title app\_product.title%type; c\_sellingprice app\_product.selling\_price%type; c\_discountedprice app\_product.discounted\_price%type; c\_description app\_product.description%type; c\_brand app\_product.brand%type; c\_category app\_product.category%type; c\_productimage app\_product.product\_image%type; CURSOR filter\_result\_topwear is SELECT id, title, selling\_price,discounted\_price ,description,brand,category,product\_image FROM app\_products where discounted\_price>250; BEGIN OPEN filter\_result\_topwear; LOOP FETCH filter\_result\_topwear into c\_id, c\_title,c\_sellingprice,c\_discountedprice ,c\_description ,c\_brand ,c\_category ,c\_productimage; EXIT WHEN filter\_result\_topwear%notfound; END LOOP; CLOSE filter\_result\_topwear; END;"

filter\_topwear\_below = "DECLARE c\_id app\_product.id%type; c\_title app\_product.title%type; c\_sellingprice app\_product.selling\_price%type; c\_discountedprice app\_product.discounted\_price%type; c\_description app\_product.description%type; c\_brand app\_product.brand%type; c\_category app\_product.category%type; c\_productimage app\_product.product\_image%type; CURSOR filter\_result\_topwear is SELECT id, title, selling\_price,discounted\_price ,description,brand,category,product\_image FROM app\_products where discounted\_price<250; BEGIN OPEN filter\_result\_topwear; LOOP FETCH filter\_result\_topwear into c\_id, c\_title,c\_sellingprice,c\_discountedprice ,c\_description ,c\_brand ,c\_category ,c\_productimage; EXIT WHEN filter\_result\_topwear%notfound; END LOOP; CLOSE filter\_result\_topwear; END;"

filter\_bottom\_wear = "DECLARE c\_id app\_product.id%type; c\_title app\_product.title%type; c\_sellingprice app\_product.selling\_price%type; c\_discountedprice app\_product.discounted\_price%type; c\_description app\_product.description%type; c\_brand app\_product.brand%type; c\_category app\_product.category%type; c\_productimage app\_product.product\_image%type; CURSOR filter\_result\_bottomwear is SELECT id, title, selling\_price,discounted\_price ,description,brand,category,product\_image FROM app\_products; BEGIN OPEN filter\_result\_bottomwear; LOOP FETCH filter\_result\_bottomwear into c\_id, c\_title,c\_sellingprice,c\_discountedprice ,c\_description ,c\_brand ,c\_category ,c\_productimage; EXIT WHEN filter\_result\_bottomwear%notfound; END LOOP; CLOSE filter\_result\_bottomwear; END;"

filter\_bottom\_wear\_lee = "DECLARE c\_id app\_product.id%type; c\_title app\_product.title%type; c\_sellingprice app\_product.selling\_price%type; c\_discountedprice app\_product.discounted\_price%type; c\_description app\_product.description%type; c\_brand app\_product.brand%type; c\_category app\_product.category%type; c\_productimage app\_product.product\_image%type; CURSOR filter\_result\_bottomwear is SELECT id, title, selling\_price,discounted\_price ,description,brand,category,product\_image FROM app\_products where category=’Lee’; BEGIN OPEN filter\_result\_bottomwear; LOOP FETCH filter\_result\_bottomwear into c\_id, c\_title,c\_sellingprice,c\_discountedprice ,c\_description ,c\_brand ,c\_category ,c\_productimage; EXIT WHEN filter\_result\_bottomwear%notfound; END LOOP; CLOSE filter\_result\_bottomwear; END;"

filter\_bottom\_wear\_spykar = "DECLARE c\_id app\_product.id%type; c\_title app\_product.title%type; c\_sellingprice app\_product.selling\_price%type; c\_discountedprice app\_product.discounted\_price%type; c\_description app\_product.description%type; c\_brand app\_product.brand%type; c\_category app\_product.category%type; c\_productimage app\_product.product\_image%type; CURSOR filter\_result\_bottomwear is SELECT id, title, selling\_price,discounted\_price ,description,brand,category,product\_image FROM app\_products where category=’SpyKar’; BEGIN OPEN filter\_result\_bottomwear; LOOP FETCH filter\_result\_bottomwear into c\_id, c\_title,c\_sellingprice,c\_discountedprice ,c\_description ,c\_brand ,c\_category ,c\_productimage; EXIT WHEN filter\_result\_bottomwear%notfound; END LOOP; CLOSE filter\_result\_bottomwear; END;"

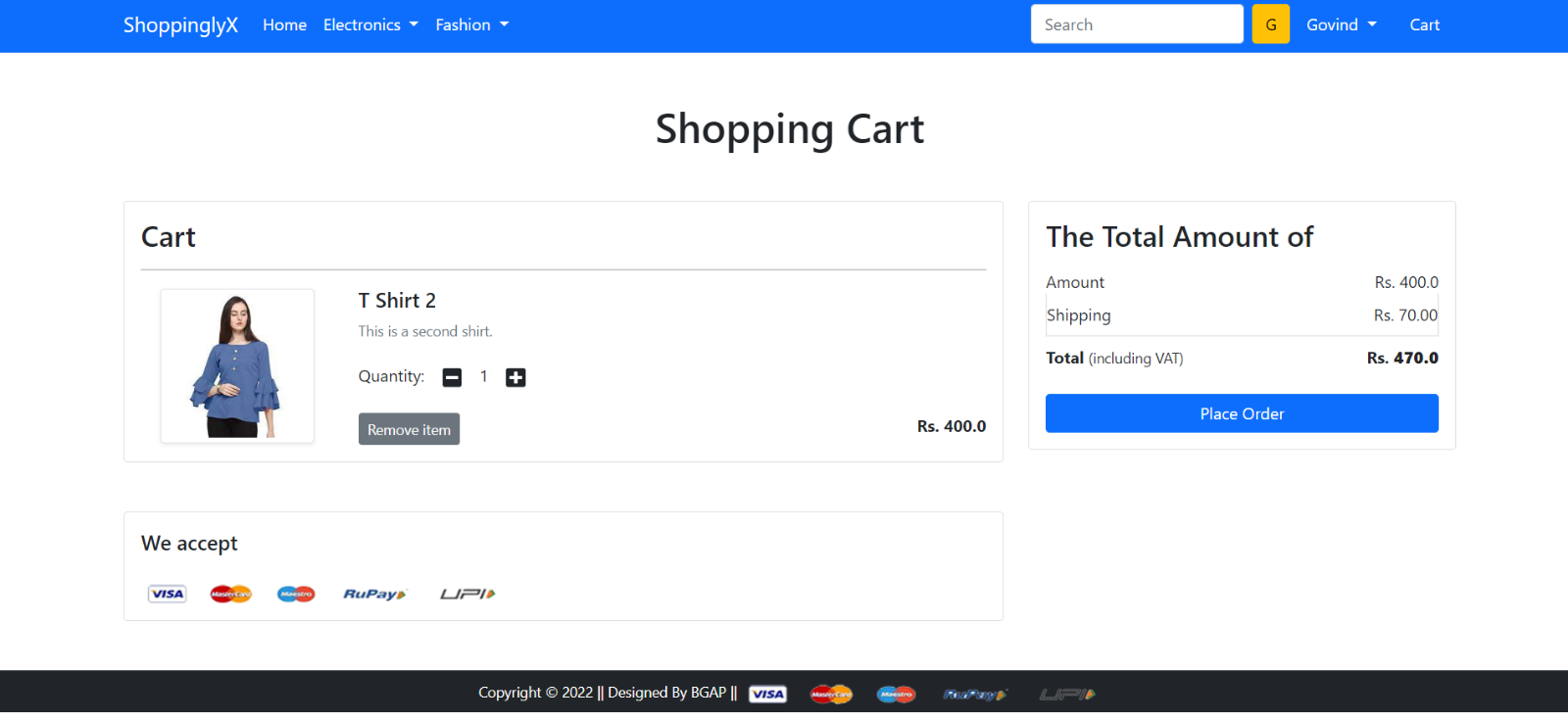
filter\_bottom\_wear\_above = "DECLARE c\_id app\_product.id%type; c\_title app\_product.title%type; c\_sellingprice app\_product.selling\_price%type; c\_discountedprice app\_product.discounted\_price%type; c\_description app\_product.description%type; c\_brand app\_product.brand%type; c\_category app\_product.category%type; c\_productimage app\_product.product\_image%type; CURSOR filter\_result\_bottomwear is SELECT id, title, selling\_price,discounted\_price ,description,brand,category,product\_image FROM app\_products where discounted\_price>250; BEGIN OPEN filter\_result\_bottomwear; LOOP FETCH filter\_result\_bottomwear into c\_id, c\_title,c\_sellingprice,c\_discountedprice ,c\_description ,c\_brand ,c\_category ,c\_productimage; EXIT WHEN filter\_result\_bottomwear%notfound; END LOOP; CLOSE filter\_result\_bottomwear;END;"

filter\_bottom\_wear\_below = "DECLARE c\_id app\_product.id%type; c\_title app\_product.title%type; c\_sellingprice app\_product.selling\_price%type; c\_discountedprice app\_product.discounted\_price%type; c\_description app\_product.description%type; c\_brand app\_product.brand%type; c\_category app\_product.category%type; c\_productimage app\_product.product\_image%type; CURSOR filter\_result\_bottomwear is SELECT id, title, selling\_price,discounted\_price ,description,brand,category,product\_image FROM app\_products where discounted\_price<250; BEGIN OPEN filter\_result\_bottomwear; LOOP FETCH filter\_result\_bottomwear into c\_id, c\_title,c\_sellingprice,c\_discountedprice ,c\_description ,c\_brand ,c\_category ,c\_productimage; EXIT WHEN filter\_result\_bottomwear%notfound; END LOOP; CLOSE filter\_result\_bottomwear; END;"

Using Functions

count\_orders = "CREATE OR REPLACE FUNCTION count\_orders(user\_name in varchar) RETURN number IS total\_orders number(2); BEGIN SELECT count(\*) into total\_orders FROM app\_orderplaceds where user = user\_name; RETURN total\_orders; END;"

**Output/Screenshots**

****