**E-COMMERCE DATABASE FILE**

create database ecommercedb ;

use ecommercedb ;

create table product

(

pid int primary key auto\_increment,

pname varchar(100),

pdesc varchar(200),

price decimal(10,2),

qty int

);

create table user

(

uid int primary key auto\_increment,

username varchar(50),

email varchar(100) unique,

contact varchar(15),

password varchar(255),

address varchar(1000),

status varchar(20)

);

create table orderhistory

(

orderid int primary key auto\_increment,

orderedat datetime,

pid int,

pname varchar(100),

price decimal(10,2),

qty int,

totalprice decimal(10,2),

deliverydate date,

uid int,

foreign key (pid) references product(pid),

foreign key (uid) references user(uid)

);

select \* from orderhistory ;

create table cart

(

uid int,

pid int,

pname varchar(100),

price decimal(10,2),

qty int,

totalprice decimal(10,2),

added\_at datetime default current\_timestamp,

primary key (uid, pid),

foreign key (uid) references user(uid),

foreign key (pid) references product(pid)

);

delimiter //

create procedure adminlogin

(

in iemail varchar(100),

in ipassword varchar(100)

)

begin

declare aemail varchar(100) default 'admin@ecom.com';

declare apassword varchar(100) default 'root';

if iemail = aemail and ipassword = apassword then

select 'Login successful. Welcome, Admin!' as message;

else

select 'Invalid credentials. Access denied.' as message;

end if;

end //

delimiter ;

drop procedure adminlogin ;

call adminlogin('admin@eco.com', 'root');

delimiter //

create procedure getProductList()

begin

select \* from product where qty > 0 ;

end //

delimiter ;

drop procedure getProductList ;

delimiter //

create procedure addProduct

(

in tname varchar(100),

in tdesc varchar(200),

in tprice decimal(10,2),

in tqty int

)

begin

insert into product(pname, pdesc, price, qty)

values(tname, tdesc, tprice, tqty);

select concat('Product "', tname, '" added successfully with quantity ', tqty, '.') as message;

end //

delimiter ;

drop procedure addProduct ;

delimiter //

create procedure updateProduct

(

in tpid int,

in tpname varchar(100),

in tpdesc varchar(200),

in tprice decimal(10,2),

in tpqty int

)

begin

update product

set

pname = ifnull(tpname, pname),

pdesc = ifnull(tpdesc, pdesc),

price = ifnull(tprice, price),

qty = ifnull(tpqty, qty)

where pid = tpid;

if row\_count() > 0 then

select concat('Product ID ', tpid, ' updated successfully.') as message;

else

select concat('No product found with ID ', tpid, '. No changes made.') as message;

end if;

end //

delimiter ;

drop procedure updateProduct ;

delimiter //

create procedure deleteProduct

(

in tpid int

)

begin

delete from product

where pid = tpid;

if row\_count() > 0 then

select concat('Product ID ', tpid, ' deleted successfully.') as message;

else

select concat('No product found with ID ', tpid, '. Deletion failed.') as message;

end if;

end //

delimiter ;

delimiter //

create procedure registerUser

(

in tusername varchar(50),

in temail varchar(100),

in tcontact varchar(15),

in tpassword varchar(255),

in taddress varchar(1000)

)

begin

declare emailexists int default 0;

-- Check if email already exists

select count(\*) into emailexists

from user

where email = temail;

if emailexists > 0 then

select 'Email already exists. Please use a different one.' as message;

else

insert into user (username, email, contact, password, address)

values (tusername, temail, tcontact, tpassword, taddress);

select 'User registered successfully.' as message;

end if;

end //

delimiter ;

delimiter //

create procedure getUserList()

begin

select uid, username, email, contact, address, status

from user;

end //

delimiter ;

drop procedure getUserList ;

delimiter //

create trigger set\_user\_status\_active

before insert on user

for each row

begin

set new.status = 'active';

end //

delimiter ;

delimiter //

create procedure userlogin

(

in identifier varchar(100),

in input\_password varchar(255)

)

begin

declare userid int;

declare uname varchar(50);

select uid, username

into userid, uname

from user

where (username = identifier or email = identifier)

and password = input\_password

and status = 'active';

if userid is not null then

select userid,

concat('welcome ', uname) as message;

else

select null as userid,

'invalid credentials or inactive account.' as message;

end if;

end //

delimiter ;

drop procedure userlogin ;

delimiter //

create procedure addtocart

(

in puid int,

in ppid int,

in pqty int

)

begin

declare vexists int default 0;

declare vprice decimal(10,2);

declare vpname varchar(100);

declare vtotal decimal(10,2);

declare vstock int;

-- fetch product details and current stock

select price, pname, qty into vprice, vpname, vstock

from product

where pid = ppid;

-- check for sufficient stock

if vstock < pqty then

select concat('insufficient stock for ', vpname) as message;

else

set vtotal = vprice \* pqty;

-- check if item already exists in cart

select count(\*) into vexists

from cart

where uid = puid and pid = ppid;

if vexists = 1 then

update cart

set qty = qty + pqty,

totalprice = totalprice + vtotal,

added\_at = current\_timestamp

where uid = puid and pid = ppid;

select concat('cart updated: ', vpname, ' quantity increased') as message;

else

insert into cart (uid, pid, pname, price, qty, totalprice)

values (puid, ppid, vpname, vprice, pqty, vtotal);

select concat('item added to cart: ', vpname) as message;

end if;

end if;

end //

delimiter ;

drop procedure addtocart ;

delimiter //

create procedure viewcart

(

in puid int

)

begin

select

c.pid,

c.pname,

c.price,

c.qty,

c.totalprice,

c.added\_at

from cart c

where c.uid = puid

order by c.added\_at desc;

end; //

delimiter ;

drop procedure viewcart ;

delimiter //

create procedure removefromcart

(

in puid int,

in ppid int

)

begin

declare vexists int;

-- Step 1: Check if item exists in cart

select count(\*) into vexists

from cart

where uid = puid and pid = ppid;

if vexists = 0 then

select 'item not found in cart.' as message;

else

-- Step 2: Remove item

delete from cart

where uid = puid and pid = ppid;

-- Step 3: Return success message

select 'item removed from cart successfully.' as message;

end if;

end //

delimiter ;

create table bank

(

bankid int primary key auto\_increment,

accountno varchar(20) unique,

bankname varchar(100),

ifsccode varchar(20),

balance decimal(12,2),

uid int,

foreign key (uid) references user(uid)

);

delimiter //

create procedure placeorder(in p\_uid int)

begin

-- Step 1: Insert each cart item into orderhistory

insert into orderhistory (

orderedat, pid, pname, price, qty, totalprice, deliverydate, uid

)

select

current\_timestamp,

pid,

pname,

price,

qty,

totalprice,

date\_add(current\_date, interval 5 day),

uid

from cart

where uid = p\_uid;

-- Step 2: Deduct ordered quantity from product stock

update product p

join cart c on p.pid = c.pid

set p.qty = p.qty - c.qty

where c.uid = p\_uid;

-- Step 3: Clear cart

delete from cart where uid = p\_uid;

-- Step 4: Confirm order placement

select 'Order placed successfully from cart.' as message;

end //

delimiter ;

delimiter //

create procedure viewAllOrderHistory()

begin

select

orderid,

orderedat,

pid,

pname,

price,

qty,

totalprice,

deliverydate,

uid

from orderhistory

order by orderedat desc;

end //

delimiter ;

delimiter //

create procedure viewMyOrderHistory

(

in p\_uid int

)

begin

select

orderid,

orderedat,

pid,

pname,

price,

qty,

totalprice,

deliverydate

from orderhistory

where uid = p\_uid

order by orderedat desc;

end //

delimiter ;

-- bank insert queries

insert into bank (accountno, bankname, ifsccode, balance, uid)

values ('123456', 'Axis Bank', 'AXIS1234', 125000.00, 5);

insert into bank (accountno, bankname, ifsccode, balance, uid)

values ('987654', 'HDFC Bank', 'HDFC5678', 118000.50, 2);

insert into bank (accountno, bankname, ifsccode, balance, uid)

values ('112233', 'ICICI Bank', 'ICICI4321', 130500.75, 3);

insert into bank (accountno, bankname, ifsccode, balance, uid)

values ('556677', 'SBI Bank', 'SBIN2468', 142000.00, 4);

-- Procedure Call Statements(Product)

call getProductList();

call addProduct('Shirt', 'Tailored cotton shirt with breathable comfort and all-day versatility.', 999.99, 15);

call addProduct('Jeans', 'Slim-fit denim jeans with stretch fabric for all-day comfort and style.', 1499.50, 20);

call addProduct('Sneakers', 'Lightweight sneakers with cushioned sole and breathable mesh for everyday wear.', 1999.00, 25);

call addProduct('Backpack', 'Durable backpack with padded straps, laptop compartment, and water-resistant fabric.', 1299.75, 30);

call addProduct('Watch', 'Minimalist analog watch with leather strap and scratch-resistant glass.', 2499.99, 10);

call addProduct('Jacket', 'Windproof jacket with fleece lining and adjustable hood for cold weather comfort.', 3499.50, 12);

call updateProduct(1,null,null,1999.50,null);

call deleteProduct(6);

-- Procedure Call Statements(User)

call getUserList();

call registerUser('alina', 'alina@gmail.com','9988776655','alina123', 'Flat 302, Green Residency, Pune');

call registerUser('devraj', 'devraj@gmail.com','9123456780','devraj123', 'B-12, Sector 5, Navi Mumbai');

call registerUser('meera', 'meera@gmail.com','7894561230','meera123', 'Plot 45, Shanti Nagar, Nagpur');

call registerUser('arjun', 'arjun@gmail.com','9001122334','arjun123', 'House No. 7, MG Road, Nashik');

call registerUser('tara', 'tara@gmail.com','9876543210','tara123', 'Villa 9, Palm Meadows, Aurangabad');

call userlogin('alina@gmail.com','alina123');

-- Procedure Call Statements(Cart)

call addtocart(3,1,2);

call viewcart(3);

call vieworderhistory();

update user set status = 'inactive' where email = 'alina@gmail.com';

-- Select Statements

select \* from product ;

select \* from user ;

select \* from cart ;

select \* from bank ;

select \* from orderhistory ;

-- Alter Statements

alter table user

add column action varchar(20);