



4/15/2021


**BTech CSE – Sem 4 DBMS**  
**E-Commerce Application**

**Group Members:**

AU1940109 Jap Purohit

AU1940118 Raj Gariwala

AU1940282 Poojan Sheth



# Description:

**The motto of this application is to provide everything to your door steps through an online platform in such a hard time of COVID19 as it's not safe to go out and buy goods, electronics, clothes, etc.**

E-commerce (electronic commerce) is the buying and selling of goods and services, or the transmitting of funds or data, over an electronic network, primarily the internet. E-commerce is powered by the internet, where customers can access an online store to browse through, and place orders for products or services via their own devices

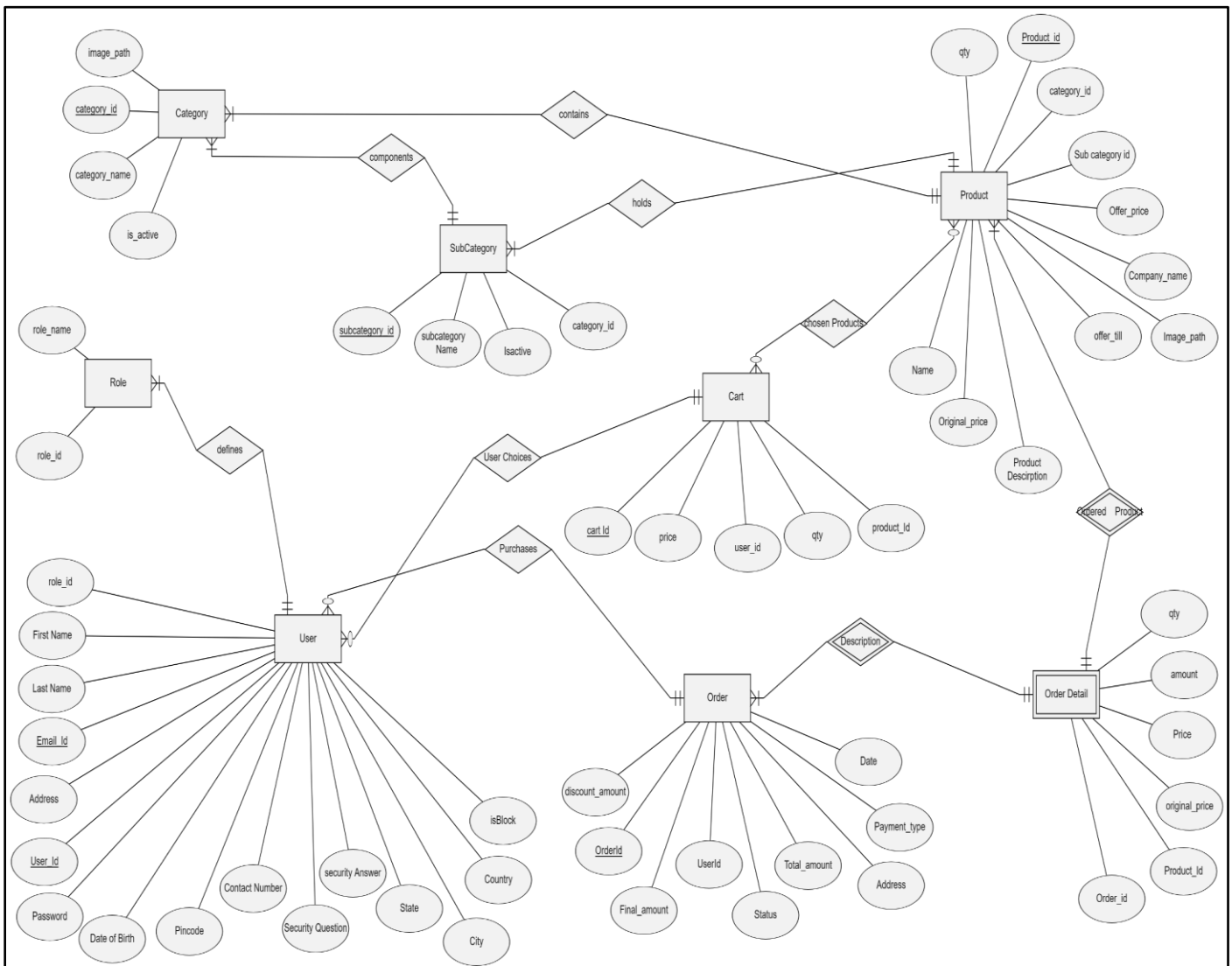
We had tried to model an E-commerce web application. For doing this we used Oracle 19c for designing databases, implementing various queries, functions, triggers and various back – end activity. We had used Java language as our medium (Eclipse IDE), JSP Servlet, Bootstrap, CSS, JavaScript, AJAX, HTML web pages for front – end , JDBC for connecting front – end with Oracle 19c and Tomcat (Version 8.5) for connecting front – end with Java.

The user can search the products, add the quantities for buying it later, purchasing extraordinary as well as products used in day to day life. The best part of this is users get amazing discounts and offers on the day to day needs and this also helps the admin that users get used to with the site. Users also get an option to suggest the admin to add new products. The site also gives an option to see users the most trending products sold from the site. With the cash on delivery option without subscriptions and no shipping cost which helps in developing trust of the user. The admin can have records of all top and least selling monthwise and annual reports of products, categorical reports and user reports.

## Benefits of ecommerce

- increase control over brand messaging and customer engagement
- more opportunities to innovate
- direct access to customer and their data
- gain higher margins
- strong brand loyalty
- expanding market opportunities

# Entity – Relational Diagram:



# Table Design (Data - Dependency)

- **role Table (Entity)**

- **Attributes:**

- **roleID number(1):** Is an assigned **primary key** to uniquely identify any role and its details.
    - **role name varchar2(10):** Name of any specific role.

- **userDetails Table (Entity)**

- **Attributes:**

- **firstName varchar2(30):** First name of any specific user.
    - **lastName varchar2(30):** Last name of any specific user.
    - **emailID varchar2(40):** Email of any specific user.
    - **passWord varchar2(20):** Password of any specific user.
    - **address varchar2(150):** Address of any specific user.
    - **userID number(20):** Is an assigned **primary key** to uniquely identify any specific user and its details.
    - **dateOfBirth date:** Date of birth of any specific user which is kept **not null**.
    - **city varchar2(30):** City of any specific user.
    - **state varchar2(30):** State of any specific user.
    - **pincode varchar2(30):** Pincode of any specific user.
    - **country varchar2(30):** Country of any specific user.
    - **contactNumber number(10):** Contact Number of any specific user.
    - **roleID number(1):** Is a **foreign key** that is referenced from the role Table to specify the role of any specific user.
    - **isBlock number(1):** its value should be in ('0' , '1') and **not null** to check whether any specific user is blocked or not.
    - **securityQuestion varchar2(50):** Security question for any specific user.
    - **securityAnswer varchar2(50):** Answer of the securityQuestion for any specific user.

- **categoryDetails Table (Entity)**

- **Attributes:**

- **categoryID number(3):** Is an assigned **primary key** to uniquely identify any specific category and its details.
    - **categoryName varchar2(40):** Name of any specific category.
    - **imagePath varchar2(50):** Image Path of any specific category.
    - **isActive number(1):** its value should be in ('0' , '1') and **not null** to check whether any specific category is active or not.

- **subCategoryDetails Table (Entity)**

- **Attributes:**

- **subCategoryID number(3):** Is an assigned **primary key** to uniquely identify any sub - category and its details.
    - **subCategoryName varchar2(40):** Name of any specific sub - category.
    - **categoryID number(3):** Is a **foreign key** that is referenced from the categoryDetails Table to specify the category of any specific sub – category.
    - **isActive number(1):** its value should be in ('0' , '1') and **not null** to check whether any specific sub - category is active or not.

- **productDetails Table (Entity)**

- **Attributes:**

- **productID number(1):** Is an assigned **primary key** to uniquely identify any specific product and its details.
    - **categoryID number(3):** Is a **foreign key** that is referenced from the categoryDetails Table to specify the category of any specific product.
    - **subCategoryID number(3):** Is a **foreign key** that is referenced from the subCategoryDetails Table to specify the sub - category of any specific product.
    - **originalPrice number(9,2):** Original Price of any specific product.
    - **offerPrice number(9,2):** Offer Price of any specific product.
    - **productName varchar2(50):** Name of any specific product.
    - **companyName varchar2(50):** Name of any specific company.
    - **quantity number(6):** Quantity of any specific product.
    - **offerTill date:** Till what date offer is there for any specific product.
    - **imagePath varchar2(50):** Image Path of any specific product.
    - **productDescription varchar2(200):** Description of any specific product.
    - **isActive number(1):** its value should be in ('0' , '1') and **not null** to check whether any specific product is active or not.

- **cartDetails Table (Entity)**

- **Attributes:**

- **cartID number(10):** Is an assigned **primary key** to uniquely identify any specific cart and its details.
    - **userID number(10):** Is a **foreign key** that is referenced from the userDetails Table to specify the user of any specific cart.
    - **productID number(3):** Is a **foreign key** that is referenced from the productDetails Table to specify the product of any specific cart.
    - **price number(9,2):** Price of any specific cart.
    - **qty number(6):** Quantity of any specific product.

- **orderDetails Table (Entity)**

- **Attributes:**

- **orderID number(10):** Is an assigned **primary key** to uniquely identify any specific order and its details.
    - **userID number(10):** Is a **foreign key** that is referenced from the userDetails Table to specify the user of any specific order.
    - **address varchar2(150):** Address of any specific order.
    - **paymentType number(1):** To select the payment type of any specific order.
    - **finalAmount number(7,2):** Final amount to be paid without applying tax for any specific order.
    - **totalAmount number(7,2):** Total amount to be paid applying tax for any specific order.
    - **discountAmount number(7,2):** Discount amount for any specific order.
    - **quantity number(7):** Quantity of any specific order.
    - **orderDate date:** Order date of any specific order.
    - **deliveryDate date:** Delivery date of any specific order.
    - **deliveryStatus number(1):** To see the delivery status of any specific order.
    - **role name varchar2(10):** Name of any specific role.

- **orderDetailed Table (Entity)**

- **Attributes:**

- **orderID number(10):** Is a **foreign key** that is referenced from the orderDetails Table to specify the order details of any specific order.
    - **userID number(10):** Is a **foreign key** that is referenced from the userDetails Table to specify the order details of any specific user.
    - **productID number(10):** Is a **foreign key** that is referenced from the productDetails Table to specify the order details of any specific product.
    - **Price number(10,2):** Price of any specific order.
    - **qty number(10):** Quantity of any specific order.
    - **amount number(10,2):** Total amount after multiplying price with quantity of any specific order.

# FUNCTIONS:

## 1. Function 1:

It takes random users from the table and sends a mail to the user to notify him that the products available in that cart of that user has some special offer for him as a result he will be visiting the site and can buy that product

```
REM function 1
create or replace function randomUser
return number
is
userid number;
begin
SELECT userid into userid
FROM (SELECT distinct userid
FROM orderdetails ORDER BY DBMS_RANDOM.RANDOM)
WHERE rownum < 2;
return userid;
end;
/
```

## 2. Function 2:

It shows the most selling products of particular categories, categorical wise data. If the admin wants to know which is the top selling category that sells the maximum amount of quantity, least selling category that sells the minimum amount of quantity then that can be inferred from it. It shows the top selling categories whose selling is highest and lowest selling categories whose selling is highest monthly, like which month has the highest selling product and changes the categories monthly.

```
REM function 2
create or replace type topSellingCategoryArray is varray(2) of number;
/

create or replace function topSellingCategory return topSellingCategoryArray as
cursor c_top is select categorydetails.categoryid
from orderdetails join orderdetailed on orderdetailed.orderid=orderdetails.orderid
join productdetails on productdetails.productid= orderdetailed.productid
join categorydetails on categorydetails.categoryid = productdetails.categoryid
where to_char(orderdate,'Month')=to_char(sysdate,'Month')
group by categorydetails.categoryname,categorydetails.categoryid,to_char(orderdate,'Month')
order by sum(qty) desc fetch first 1 row only;
cursor c_least is select categorydetails.categoryid
from orderdetails join orderdetailed on orderdetailed.orderid=orderdetails.orderid
join productdetails on productdetails.productid= orderdetailed.productid
join categorydetails on categorydetails.categoryid = productdetails.categoryid
where to_char(orderdate,'Month')=to_char(sysdate,'Month')
group by categorydetails.categoryname,categorydetails.categoryid,to_char(orderdate,'Month')
order by sum(qty) fetch first 1 row only;
```

```

tcid topSellingCategoryArray;
r_top c_top%rowtype;
r_least c_least%rowtype;
begin
    tcid :=topSellingCategoryArray(-1,-1);
    for r_top in c_top loop
        tcid(1) := r_top.categoryid;
    end loop;
    for r_least in c_least loop
        tcid(2) := r_least.categoryid;
    end loop;
    return tcid;
end;
/

```

```

declare
    tcid topSellingCategoryArray;

begin
    tcid := topSellingCategory();
    dbms_output.put_line(tcid(1)||' '||tcid(2));
end;
/

```

### 3. Function 3:

This function will display an overall selling report of any specific product in which details like Product Description, Quantity Sold, SubCategory Name, Category Name, Total Earning, Quantity Left, Offer Price, Original Price of Top Selling Product(s) and the Least Selling Product(s) is/are there. It will also display details like Product Description, Quantity Sold, SubCategory Name, Category Name, Total earning, Quantity Left, Offer Price, Original Price of Monthly product report in which the Top Selling and Least Selling product(s) of the current Month is/are there.

```

REM function 3
create or replace type productReport is varray(4) of number;
/

create or replace function productReportFunction return productReport as

cursor c_top is select productdetails.categoryid,categoryname,orderdetailed.productid,sum(qty)
                from orderdetailed
                join productdetails on productdetails.productid= orderdetailed.productid
                join categorydetails on categorydetails.categoryid = productdetails.categoryid
                group by orderdetailed.productid,productdetails.categoryid,categoryname
                order by sum(qty) desc fetch first 1 row only;

cursor c_least is select productdetails.categoryid,orderdetailed.productid,sum(qty)
                from orderdetailed
                join productdetails on productdetails.productid= orderdetailed.productid
                join categorydetails on categorydetails.categoryid = productdetails.categoryid
                group by categorydetails.categoryid,orderdetailed.productid,productdetails.categoryid,categoryname
                order by sum(qty) fetch first 1 row only;

```



```

cursor c_topm is select to_char(orderdate,'Month'),productdetails.categoryid,orderdetails.productid,sum(qty)
from orderdetails join orderdetails on orderdetails.orderid=orderdetails.orderid
join productdetails on productdetails.productid= orderdetails.productid
join categorydetails on categorydetails.categoryid = productdetails.categoryid
where to_char(orderdate,'Month')=to_char(sysdate,'Month')
group by categorydetails.categoryid,to_char(orderdate,'Month'),orderdetails.productid,productdetails.categoryid,categoryname
order by sum(qty) desc fetch first 1 row only;

cursor c_leastm is select to_char(orderdate,'Month'),productdetails.categoryid,orderdetails.productid,sum(qty)
from orderdetails join orderdetails on orderdetails.orderid=orderdetails.orderid
join productdetails on productdetails.productid= orderdetails.productid
join categorydetails on categorydetails.categoryid = productdetails.categoryid
where to_char(orderdate,'Month')=to_char(sysdate,'Month')
group by categorydetails.categoryid,to_char(orderdate,'Month'),orderdetails.productid,productdetails.categoryid,categoryname
order by sum(qty) fetch first 1 row only;

tcid productReport;
r_top c_top%rowtype;
r_least c_least%rowtype;
r_topm c_topm%rowtype;
r_leastm c_leastm%rowtype;

begin
    tcid := productReport(-1,-1,-1,-1);
    for r_top in c_top loop
        tcid(1) := r_top.productid;
    end loop;

    for r_least in c_least loop
        tcid(2) := r_least.productid;
    end loop;

    for r_topm in c_topm loop
        tcid(3) := r_topm.productid;
    end loop;

    for r_leastm in c_leastm loop
        tcid(4) := r_leastm.productid;
    end loop;
    dbms_output.put_line(tcid(1)||' '||tcid(2)||' '||tcid(3)||' '||tcid(4));
    return tcid;
end;
/

declare
    tcid productReport;

begin
    tcid := productReportFunction();
    dbms_output.put_line(tcid(1)||' '||tcid(2));
end;
/

```

#### 4. Function 4:

This function will display the User Order Report of any specific user in which details like Name, Email Address, Contact Number and other details of All Time High Buyer and All Time Low Buyer is/are there. Similarly, it will display the Monthly report in which details like Name, Email Address, Contact Number and other details of This Month High Buyer (Current\_Month\_name) and the This Month Low Buyer (Current\_Month\_Name) is/are there. It will also display Overall Order Report in which the value of All Time Total Sell, Total Discount as well as the This Month Total Sell (Current\_Month\_Name), Total Discount (Current\_Month\_Name) will be displayed.

```

create or replace type orderReport is varray(8) of number;
/
create or replace function orderReportFunction return orderReport as
tcid orderReport;
begin
    tcid := orderReport (-1,-1,-1,-1,-1,-1,-1,-1);
    select sum(amount) into tcid(1) from orderdetails
    join orderdetailed on orderdetailed.orderid=orderdetails.orderid
    where to_char(orderdate,'Month')=to_char(sysdate,'Month')
    group by to_char(orderdate,'Month');
    select sum((originalprice-price)*qty) into tcid(2) from orderdetails
    join orderdetailed on orderdetailed.orderid=orderdetails.orderid
    where to_char(orderdate,'Month')=to_char(sysdate,'Month')
    group by to_char(orderdate,'Month');
    select sum(amount) into tcid(3) from orderdetailed;
    select sum((originalprice-price)*qty) into tcid(4) from orderdetailed;
    select userid into tcid(5) from orderdetails group by userid order by count(*) desc fetch first 1 row only;
    select userid into tcid(6) from orderdetails group by userid order by count(*) fetch first 1 row only;
    select userid into tcid(7) from orderdetails
    where to_char(orderdate,'Month')=to_char(sysdate,'Month')
    group by to_char(orderdate,'Month')
    order by count(*) desc fetch first 1 row only;
    select userid into tcid(8) from orderdetails
    where to_char(orderdate,'Month')=to_char(sysdate,'Month')
    group by to_char(orderdate,'Month')
    order by count(*) fetch first 1 row only;
    dbms_output.put_line(tcid(1)||' '||tcid(2)||' '||tcid(3)||' '||tcid(4)||' '||tcid(5)||' '||tcid(6)||' '||tcid(7)||' '||tcid(8));
    return tcid;
end;
/

declare
    tcid orderReport;

begin
    tcid := orderReportFunction ();
    dbms_output.put_line(tcid(1)||' '||tcid(2));
end;
/

```

## 5. Function 5:

This function helps in providing offers on particular products for the admin defined period of time. If there are branded products and companies are interested to get their product sold on discounted price like 500 off or more on the website, on any special days, so this helps admin to add offer price for that given amount of time given by the company, can be updated and the offer prize is set in every Cart table also if that certain product exists in users cart.

```

REM function 6
create or replace procedure offerByPrice(product_id productDetails.productid&type, price number,offertillInput varchar2)as
begin
    update productDetails set offerprice = originalprice - price,offertill = to_date(offertillInput,'yyyymmdd')
    where productid= product_id;
end;
/

```

## 6. Function 6:

This function will display Category Details of any specific category in which details like Product Name, Category Name, SubCategory Name of All Time Top Selling and All Time Least Selling is/are there. It will also display a report in which the names of Sub-Category Names, Product Names and the value of Total Selling Done By This Category, Total Discount Done By This Category as well as the Total Selling Done This month By This Category , Total Discount Done This month By This Category will be displayed

```
create or replace type categoryReport is varray(8) of number;
/

create or replace function categoryReportFunction(id in number) return categoryReport as
tcid categoryReport;
begin
    tcid := categoryReport(-1,-1,-1,-1,-1,-1,-1,-1);
    select sum(amount) into tcid(1) from orderdetails
        join orderdetailed on orderdetailed.orderid=orderdetails.orderid
        join productdetails on productdetails.productid= orderdetailed.productid
        join categorydetails on categorydetails.categoryid = productdetails.categoryid
        where categorydetails.categoryid=id and to_char(orderdate,'Month')=to_char(sysdate,'Month')
        group by categorydetails.categoryid,to_char(orderdate,'Month');
    select sum((orderdetailed.originalprice-orderdetailed.price)*orderdetailed.qty) into tcid(2) from orderdetails
        join orderdetailed on orderdetailed.orderid=orderdetails.orderid
        join productdetails on productdetails.productid= orderdetailed.productid
        join categorydetails on categorydetails.categoryid = productdetails.categoryid
        where categorydetails.categoryid=id and to_char(orderdate,'Month')=to_char(sysdate,'Month')
        group by categorydetails.categoryid,to_char(orderdate,'Month');
    select sum(amount) into tcid(3) from orderdetailed
        join productdetails on productdetails.productid= orderdetailed.productid
        join categorydetails on categorydetails.categoryid = productdetails.categoryid
        where categorydetails.categoryid=id group by categorydetails.categoryid;
    select sum((orderdetailed.originalprice-orderdetailed.price)*orderdetailed.qty) into tcid(4) from orderdetailed
        join productdetails on productdetails.productid= orderdetailed.productid
        join categorydetails on categorydetails.categoryid = productdetails.categoryid
        where categorydetails.categoryid=id group by categorydetails.categoryid;
    select orderdetailed.productid into tcid(5) from orderdetailed
        join productdetails on productdetails.productid= orderdetailed.productid
        join categorydetails on categorydetails.categoryid = productdetails.categoryid
        where productdetails.categoryid = id
        group by orderdetailed.productid,productdetails.categoryid,categoryname
        order by sum(qty) desc fetch first 1 row only;

    select orderdetailed.productid into tcid(6) from orderdetailed
        join productdetails on productdetails.productid= orderdetailed.productid
        join categorydetails on categorydetails.categoryid = productdetails.categoryid
        where productdetails.categoryid =id group by orderdetailed.productid,productdetails.categoryid,categoryname
        order by sum(qty) fetch first 1 row only;
    select orderdetailed.productid into tcid(7) from orderdetails
        join orderdetailed on orderdetailed.orderid=orderdetails.orderid
        join productdetails on productdetails.productid= orderdetailed.productid
        join categorydetails on categorydetails.categoryid = productdetails.categoryid
        where categorydetails.categoryid=id and to_char(orderdate,'Month')=to_char(sysdate,'Month')
        group by categorydetails.categoryid,to_char(orderdate,'Month'),orderdetailed.productid,productdetails.categoryid,categoryname
        order by sum(qty) desc fetch first 1 row only;
    select orderdetailed.productid into tcid(8) from orderdetails
        join orderdetailed on orderdetailed.orderid=orderdetails.orderid
        join productdetails on productdetails.productid= orderdetailed.productid
        join categorydetails on categorydetails.categoryid = productdetails.categoryid
        where categorydetails.categoryid=id and to_char(orderdate,'Month')=to_char(sysdate,'Month')
        group by categorydetails.categoryid,to_char(orderdate,'Month'),orderdetailed.productid,productdetails.categoryid,categoryname
        order by sum(qty) fetch first 1 row only;
    dbms_output.put_line(tcid(1)||' '||tcid(2)||' '||tcid(3)||' '||tcid(4)||' '||tcid(5)||' '||tcid(6)||' '||tcid(7)||' '||tcid(8));
    return tcid;
end;
/

declare
    tcid categoryReport;

begin
    tcid := categoryReportFunction (4);
    dbms_output.put_line(tcid(1)||' '||tcid(2));
end;
/
```

## 7. Function 7:

This function helps in providing a percentage of discount on original price of particular product for the admin defined period of time. If there are many products that are not getting sold less on the website or if there are special events like Diwali offer or Big Billion Days like offer so this helps the admin to add a discount on original price for the particular amount of time, he is interested to keep the offer ON. Also, it allows the admin to provide offers in terms of percentage so that even users can infer from that. The percentage of discount on original prize is set in users Cart table also if that certain product exists in users' cart.

```
REM function 7
create or replace procedure offerByPercentage(product_id productDetails.productid%type, percent number, offertillInput varchar2)as
begin
    update productDetails set offerprice = originalprice*(1-(percent/100)), offertill = to_date(offertillInput, 'yyyymmdd')
    where productid= product_id;
end;
/
```

## 8. Function 8:

This function helps in giving discount in percentage on particular categories if there are really special events like 1<sup>st</sup>, 2<sup>nd</sup> or 10<sup>th</sup> anniversary of that website than in such situation the admin can give flat 30 to 50% discount on the whole category and even such offers attract more and more users towards this website. By this categorical discount the changes are of prizes are updated on all tables like ProductDetails table and in users Cart table if that certain product exists in users' cart.

```
REM function 8
create or replace procedure offerByCategoryID(category_id categoryDetails.categoryid%type, percent number, offertillInput varchar2)as
begin
    update productDetails set offerprice = originalprice*(1-(percent/100)), offertill = to_date(offertillInput, 'yyyymmdd')
    where categoryid = category_id;
end;
/
```

## 9. Function 9:

It shows the most selling product as per the maximum product bought by the total users of that site and that get displayed in the home page of the user so he can see which top 5 products are the most trending products available for now and also that records are written in table format so that admin can access and can have an idea that which of this is their product is been sold the most.

```

REM function 9
create or replace type returningid is varray(5) of int;
/
create or replace function topSellingProduct return returningid as
cursor c_main is select orderdetailed.productid,sum(qty) sum_qty
                   from orderdetailed
                   group by orderdetailed.productid
                   order by sum_qty desc fetch next 5 rows only;
rid returningid;
r_main c_main%rowtype;
begin
    rid := returningid(-1,-1,-1,-1,-1);
    for r_main in c_main loop
        rid(c_main%ROWCOUNT):=r_main.productid;
    end loop;
    return rid;
end;
/

declare
    rid returningid;
begin
    rid:=topSellingProduct();
    dbms_output.put_line(rid(1)||' '||rid(2)||' '||rid(3)||' '||rid(4)||' '||rid(5));
end;
/

```

## 10. Function 10:

This function tells us the total amount of saving done by all users after purchasing products on the discounted/offer prize this gives a record to the admin about the total users active on the site or waiting for more deals

```

REM function 10
create or replace function totalSavingDoneByWebSite return number as
saved number;
begin
    select sum((originalprice-price)*qty) into saved
    from orderdetailed;
    return saved;
end;
/

```

## 11. Function 11:

This function tells us the total amount of saving done by a particular user after purchasing products on the discounted/offer prize that gives a record to the admin about individual user's activeness on the site.

```
REM function 11
create or replace function totalSavingDoneByUser(userNum in number) return number is
saved number;
begin
select sum((originalprice-price)*qty) into saved
      from orderdetailed
      where userid = userNum
      group by userid;
return saved;
end;
/
```

# TRIGGERS:

## 1. Trigger 1

This trigger will be fired when any specific admin tries to set offer price as well as the original price of the productDetails table to be less than “0” i.e. negative before updating the table. And it will generate an error message, “Price going negative”.

```
REM trigger 1
create or replace trigger priceCannotBeNegative
before update of offerprice,originalprice
on productdetails
for each row
begin
    if (:new.offerprice < 0 or :new.originalprice < 0) then
        raise_application_error(-20200, 'Price going negative');
    end if;
end;
/
```

## 2. Trigger 2

This trigger will be fired when any specific user tries to keep the new password same as the old password from the userDetails table before updating. And it will generate an error message, “Password Matching”.

```
REM trigger 2
create or replace trigger changePassword
before update of password
on userDetails
for each row
declare
begin
    if (:new.password = :old.password) then
        raise_application_error(-20100, 'Password Matching');
    end if;
end;
/
```

## 3. Trigger 3

This trigger will be fired when any specific user will insert an email which is already existing in userDetails table before inserting while signing up. And it will generate an error message, “Email ID Already Exist”.



```

REM trigger 3
create or replace trigger emailValidation
before insert
on userdetails
for each row
declare
    vCount NUMBER;
begin
    select count(*) into vCount from userdetails where emailid like :new.emailid;
    if(vCount > 0) then
        raise_application_error(-20000,'Email ID Already Exist');
    end if;
end;
/

```

#### 4. Trigger 4

This trigger will be fired when any specific user will try to purchase an order in which the items of order will be more than the quantity in the productDetails table before inserting quantity. And will generate an error message, “Quantity Excess”

```

REM trigger 4
CREATE OR REPLACE TRIGGER orderDetail
before insert
on orderDetailed
FOR EACH ROW
DECLARE
    v_product_id productdetails.productid%TYPE;
    v_product_qty productdetails.quantity%TYPE;
    new_qty orderDetailed.qty%type := :new.qty;
    new_productid orderDetailed.qty%type := :new.productid;
    new_orderid orderDetailed.orderid%type := :new.orderid;
    new_userid orderDetailed.userid%type := :new.userid;
BEGIN
    SELECT
        productid,
        quantity
    INTO
        v_product_id,v_product_qty
    FROM
        productdetails
    WHERE
        productid = new_productid;

    if(new_qty > v_product_qty) then
        delete from orderDetails where orderid=new_orderid;
        raise_application_error(-20001,'Quantity Excess');

    elsif(new_qty <= v_product_qty ) then
        v_product_qty := v_product_qty - :new.qty ;
        update productdetails set quantity = v_product_qty where productid = new_productid;
        delete from cartdetails where userid= new_userid and productid=new_productid;
    end if;

END;
/

```



## 5. Trigger 5

This trigger will be fired when any specific user will try to add items in cart which is more than the quantity in the productDetails table before inserting or updating in quantity. And will generate an error message, "Quantity Excess"

```
REM trigger 5
CREATE OR REPLACE TRIGGER cartDetail
before insert or update
on cartdetails
FOR EACH ROW
DECLARE
    v_product_id    productdetails.productid%TYPE;
    v_product_qty    productdetails.quantity%TYPE;
    new_qty cartDetails.qty%type := :new.qty;

BEGIN
    SELECT
        productid,
        quantity
    INTO
        v_product_id,v_product_qty
    FROM
        productdetails
    WHERE
        productid = :new.productid;

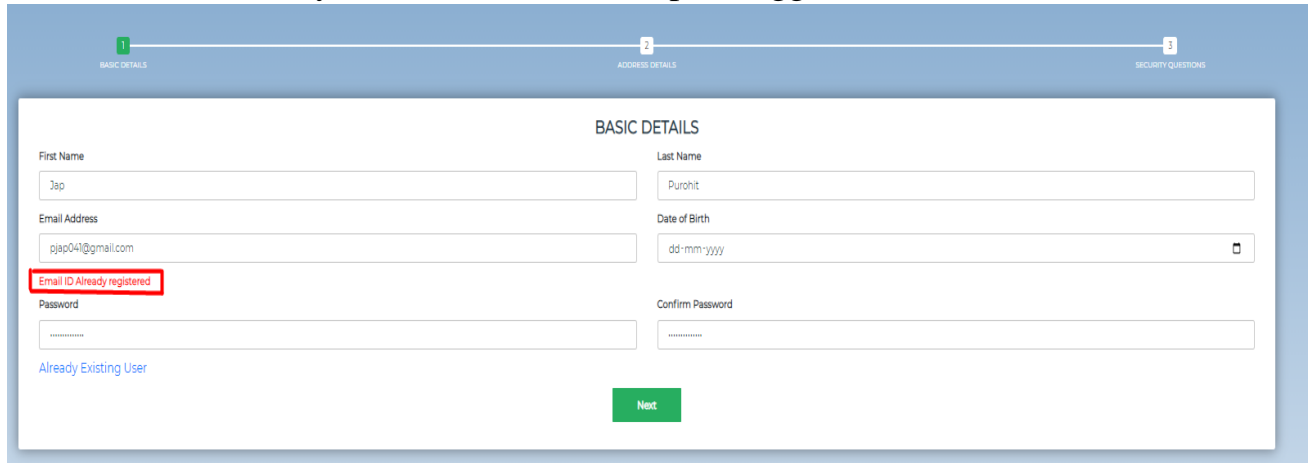
    if(new_qty > v_product_qty) then
        raise_application_error(-20110,'Quantity Excess');
    end if;

END;
/
```

# OUTPUTS:

## 1. Email Validation:

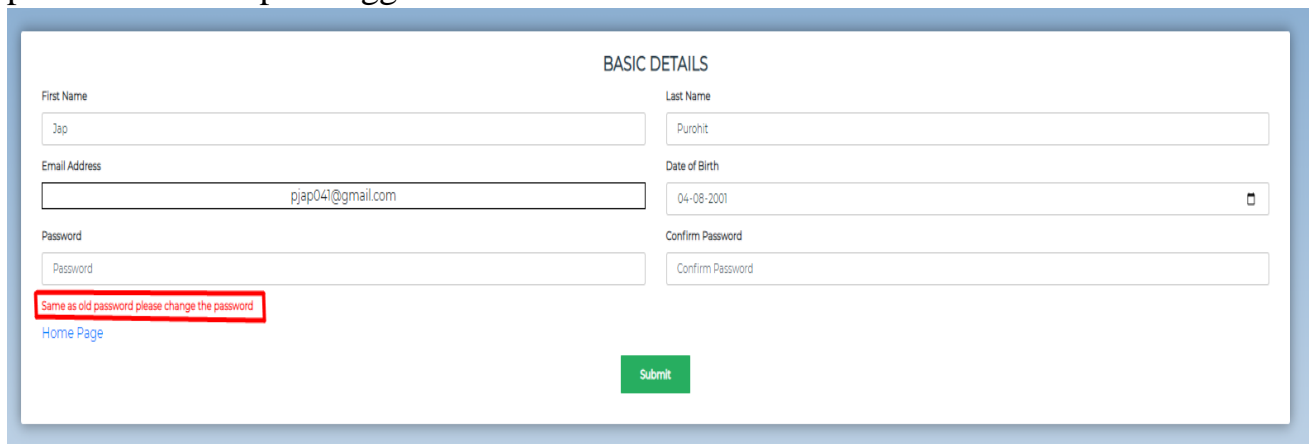
During the signup of new user. We validate whether the entered Email-ID exist or not and if Email-ID exist then it shows error on signup form and doesn't allow to sign up. And this functionality was done with the help of trigger.



The screenshot shows a 'BASIC DETAILS' form with the following fields: First Name (Jap), Last Name (Purohit), Email Address (pjap041@gmail.com), Date of Birth (dd-mm-yyyy), Password, and Confirm Password. A red error message 'Email ID Already registered' is displayed below the Email Address field. A 'Next' button is at the bottom right. The form is part of a three-step process, with '1 BASIC DETAILS' highlighted at the top.

## 2. Same Password Entry:

While updating the password if user enters same old password, then it states error that same as old password and cannot change password. Thereby this functionality was possible with help of trigger.

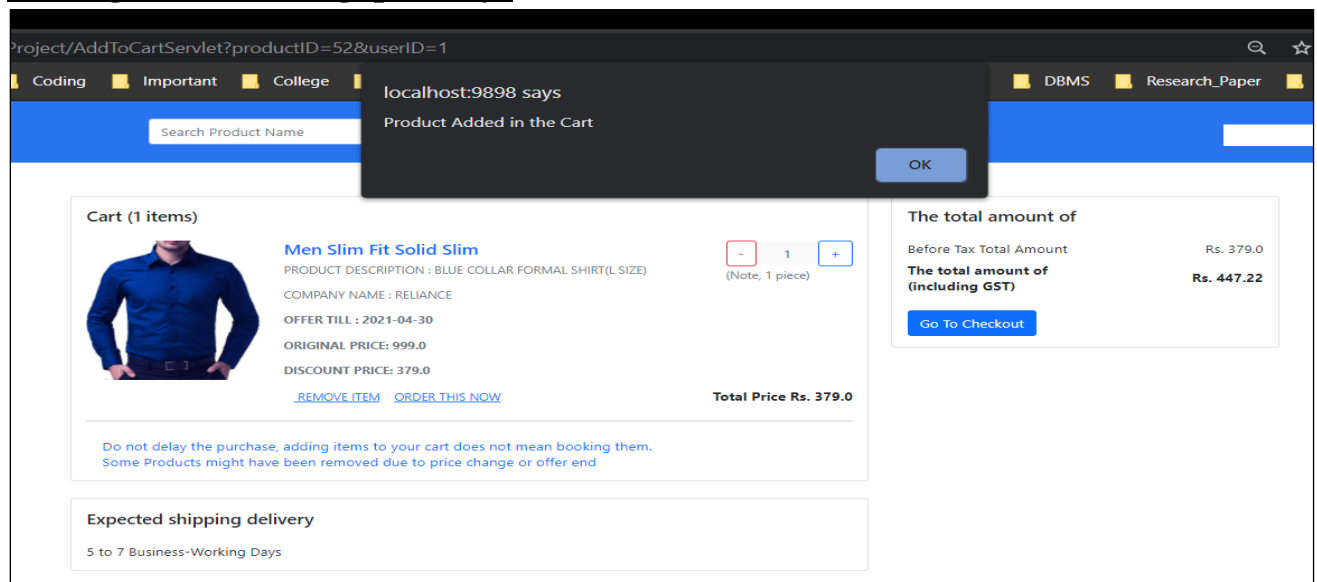


The screenshot shows the same 'BASIC DETAILS' form, but with the Password and Confirm Password fields filled with the same value. A red error message 'Same as old password please change the password' is displayed below the Password field. A 'Submit' button is at the bottom right. The form is part of a three-step process, with '2 ADDRESS DETAILS' highlighted at the top.

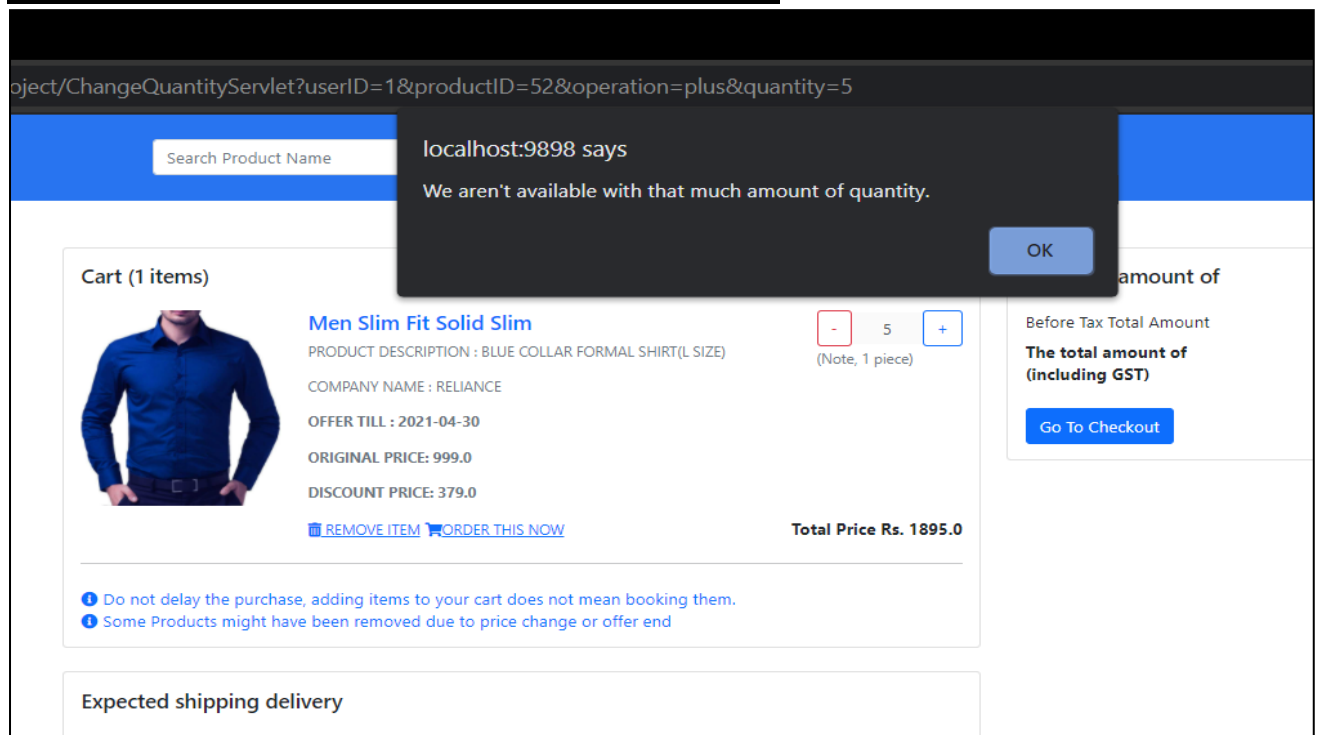
## 3. Check Quantity Before Adding in Cart:

While updating or adding product in the cart, we have checked if adding amount of quantity is available in the product tables for that particular product. If exists then it is added successfully or else it shows error message. This functionality was possible due help of trigger.

## Message while adding quantity:



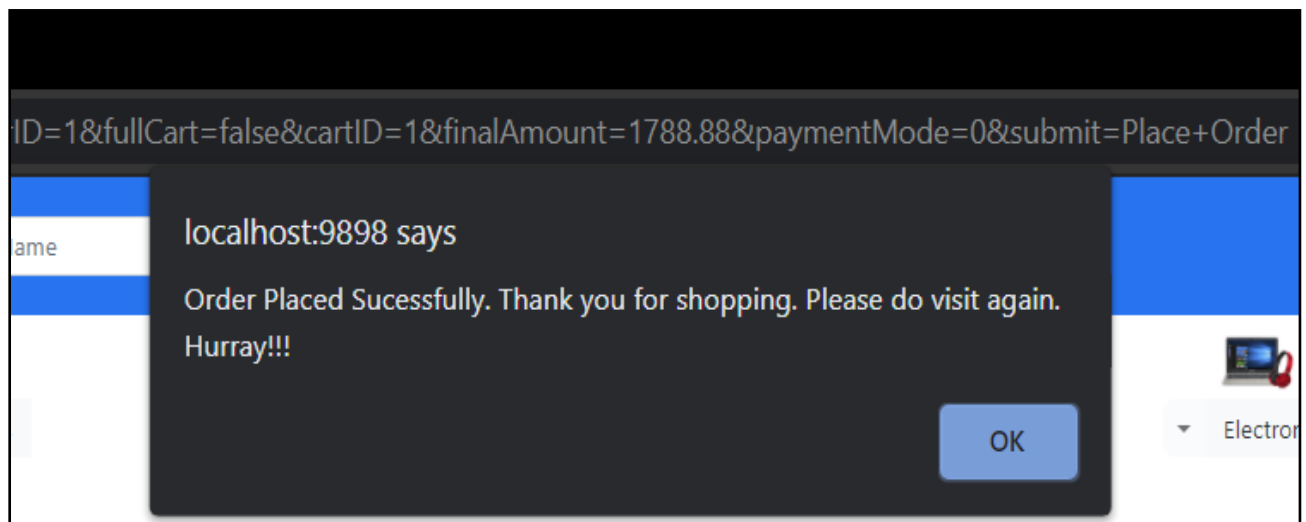
## Error Message If That Quantity is not available:



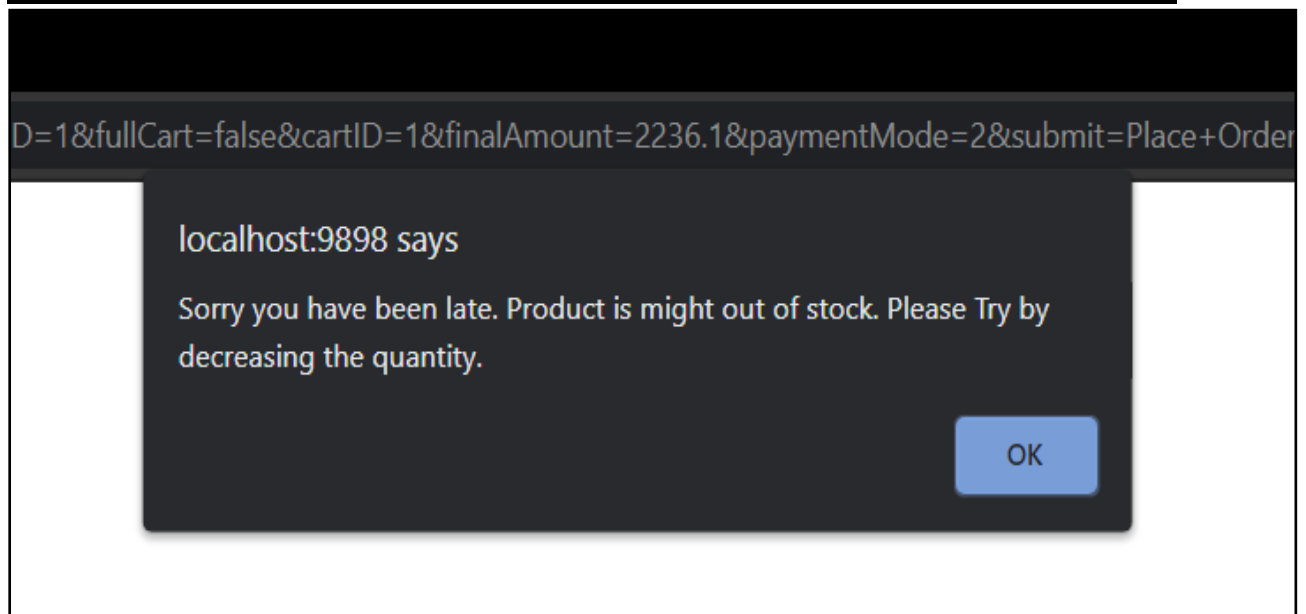
## 4. Checking Quantity Before Placing Order

While placing order, we have checked if placed amount of quantity is available in the product tables for that particular product. If exists then it is ordered is successfully placed or else it shows error message. This functionality was possible due help of trigger.

### Success message if order is placed successfully



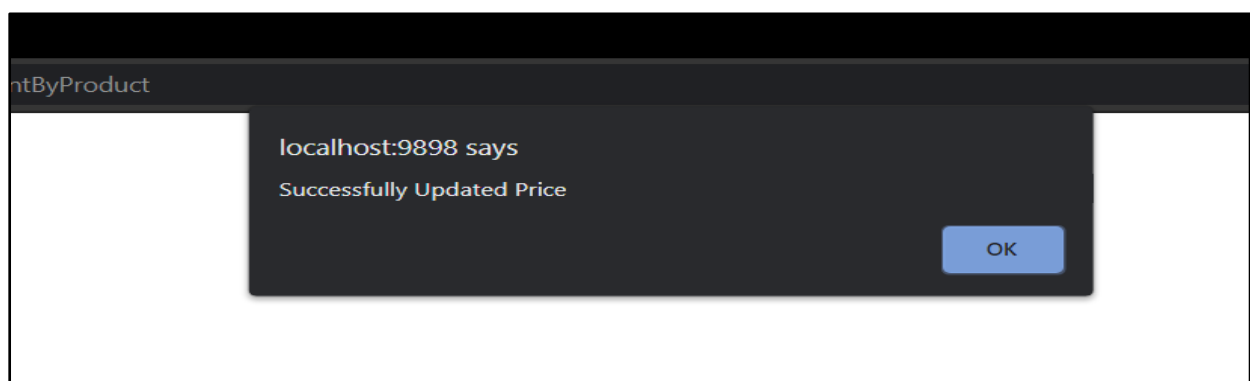
### Error Message If that product is out of stock for that particular quantity:



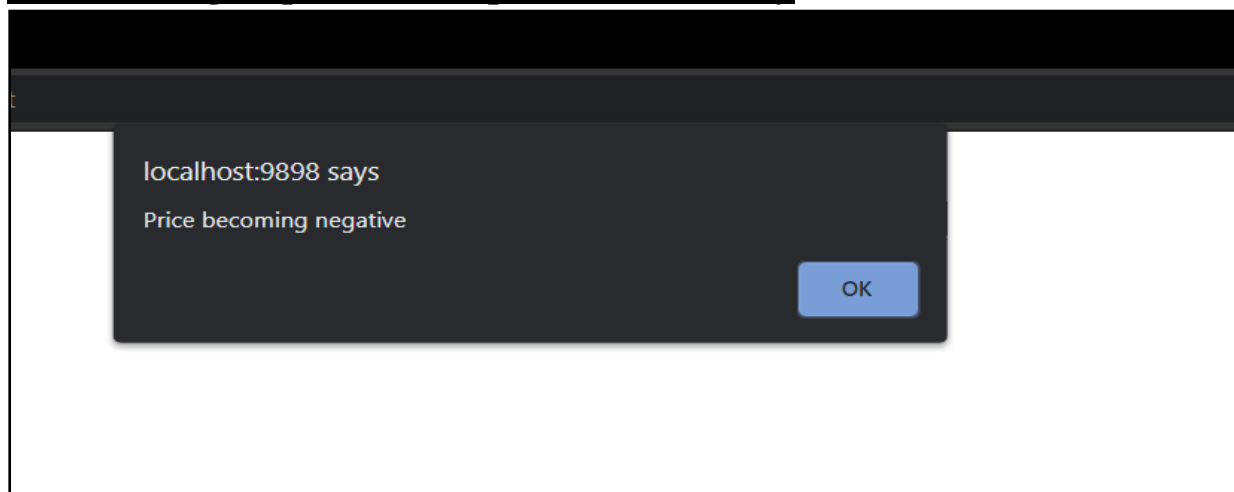
## 5. Checking Product Price Before Updating New Price is negative or not

While updating price or applying offer, we have checked if new price of that product is becoming negative or not. If it is becoming negative then it doesn't not allow to update price.

### Success message if price updated successfully



## Error message if price is not updated successfully

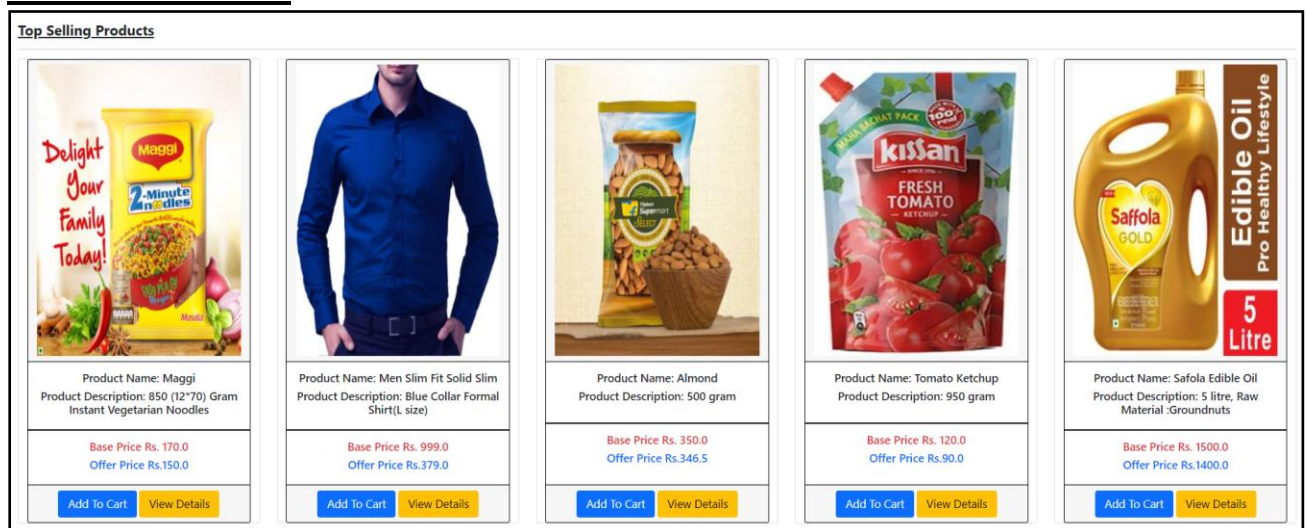


## 6. Top Selling Products Display:

It shows the most selling product as per the maximum product bought by the total users of that site and that get displayed in the home page of the user so he can see which top 5 products are the most trending products available for now and also that records are written in table format so that admin can access and can have an idea that which of this is their product is been sold the most.

### User Side:

### In form of Cards:








### In form of carousel:



## Admin Side:

Top Five Selling Products

Home Page Log out

Product ID	Product Name	Company Name	Category Name	SubCategory Name	Original Price	Offer Price	Offer Till	Quantity Left	Quantity Sold	Total Earning	Image	Product Description	Is Active
7	Maggi	Nestle	Grocery	Packaged Foods	170.0	150.0	2021-04-30	85	5	750.0		850 (12*70) Gram Instant Vegetarian Noodles	Yes
52	Men Slim Fit Solid Slim	Reliance	Fashion	Men	999.0	379.0	2021-04-30	0	4	1516.0		Blue Collar Formal Shirt(L size)	Yes
3	Almond	California	Grocery	Staples	350.0	346.5	2021-04-29	146	4	1386.0		500 gram	Yes
8	Tomato Ketchup	Kissan	Grocery	Packaged Foods	120.0	90.0	2021-04-30	146	4	360.0		950 gram	Yes
2	Safola Edible Oil	Safola	Grocery	Staples	1500.0	1400.0	2021-04-30	97	3	4200.0		5 litre, Raw Material Groundnuts	Yes

## 7. Previous Order History:

This shows previous order of that particular user and shows total amount discount availed by that user.

### Previous Order of Jap

Total Discount Availed : 3014.0

Website Discount Percentage : 1.00%

#### Previous Order (9 items)



#### Sugar

PRODUCT DESCRIPTION : 1 KG SUGAR

ORDER ID : 13

PAYMENT MODE : CASH ON DELIVERY

ORDER DATE : 2021-04-15

DELIVERY DATE : 2021-04-22

OFFER PRICE : RS. 50.0

DISCOUNT AVAILED: RS. 0.0

TOTAL PRICE WITHOUT TAX: RS. 100.0

TOTAL PRICE WITH TAX: RS. 118.0

DELIVERY STATUS : NOT-DELIVERED

FINAL QUANTITY: 2



#### Tomato Ketchup

PRODUCT DESCRIPTION : 950 GRAM

ORDER ID : 12

PAYMENT MODE : NET-BANKING

ORDER DATE : 2021-04-15

DELIVERY DATE : 2021-04-22

OFFER PRICE : RS. 90.0

DISCOUNT AVAILED: RS. 60.0

TOTAL PRICE WITHOUT TAX: RS. 180.0

FINAL QUANTITY: 2

## **8. Update Operation of User Profile:**

This gives user rights to edit his/her basic profile, address profile and security part.

BASIC DETAILS

First Name

Jap

Last Name

Purohit

Email Address

pjap041@gmail.com

Date of Birth

04-08-2001

Password

password

Confirm Password

confirm password

Home Page

Submit

ADDRESS DETAILS

Street Number/ House Number

Street Number/ House Number

Landmark

Landmark

City

Ahmedabad

Pincode

380007

State

Gujarat

Country

India

Phone Number

9726624242

Home Page

Submit

SECURITY QUESTIONS

Please Select Your Security Question

What is your favourite food dish

Please Answer Your Security Question

dosa

Submit

## 9. Reset Price

This function helps to reset offertill date. If the offer is over then it resets offerprice equal to original price and offertill equal to sysdate. This was possible due to help of procedure.



**Reset Price**  
This will reset the price of products whose offer is over

[Reset Offer Price](#)





On success we get:

localhost:9898 says  
Price reseted successfully




## 10. Product Report

This function will display an overall selling report of any specific product in which details like Product Description, Quantity Sold, SubCategory Name, Category Name, Total Earning, Quantity Left, Offer Price, Original Price of Top Selling Product(s) and the Least Selling Product(s) is/are there. It will also display details like Product Description, Quantity Sold, SubCategory Name, Category Name, Total earning, Quantity Left, Offer Price, Original Price of Monthly product report in which the Top Selling and Least Selling product(s) of the current Month is/are there.

Overall Selling Report			
Top Selling Product		Least Selling Product	
	Product Name: Maggi		Product Name: Sugar
	Product Description: 850 (12*70) Gram Instant Vegetarian Noodles		Product Description: 1 kg Sugar
	Quantity Sold: 5		Quantity Sold: 3
	SubCategory Name: Packaged Foods		SubCategory Name: Staples
	Category Name: Grocery		Category Name: Grocery
Total Earning: 750.0		Total Earning: 150.0	
Quantity Left: 85		Quantity Left: 197	
Offer Price: 150.0		Offer Price: 50.0	
Original Price: 170.0		Original Price: 50.0	
Monthly Product Report			
Top Selling Product of This Month		Least Selling Product of This Month	
	Product Name: Maggi		Product Name: Sugar
	Product Description: 850 (12*70) Gram Instant Vegetarian Noodles		Product Description: 1 kg Sugar
	Quantity Sold: 5		Quantity Sold: 3
	SubCategory Name: Packaged Foods		SubCategory Name: Staples
	Category Name: Grocery		Category Name: Grocery
Total Earning: 750.0		Total Earning: 150.0	
Quantity Left: 85		Quantity Left: 197	
Offer Price: 150.0		Offer Price: 50.0	
Original Price: 170.0		Original Price: 50.0	

## 11. Particular Category Report

This function will display Category Details of any specific category in which details like Product Name, Category Name, SubCategory Name of All Time Top Selling and All Time Least Selling is/are there. It will also display a report in which the names of Sub-Category Names, Product Names and the value of Total Selling Done By This Category, Total Discount Done By This Category as well as the Total Selling Done This month By This Category , Total Discount Done This month By This Category will be displayed

Category Name		Grocery	
Image			
Sub-Category Names		Dairy & Eggs Fruit & Vegetables Packaged Foods Breads & Beverages Staples	
Product Names		Almond Amul Chessa Block Amul Partner Amul Taza Milk Apple Capasium Chocosa Cucumber Dark Fantasy Choco Fills Gowardhan Pasteurised Table Salted Butter Jam Maggi Nestle Classic Coffee Nestle Milkmaid Orange Pomegranate Pringles Cream & Onion Redbull Rice Safex Edible Oil Silk Fruit & Nut Sugar Tomato Ketchup Topicana Mixed Fruit Delight Wheat Flour	
Total Selling Done By This Category		9840.0	
Total Discount Done By This Category		534.0	
Total Selling Done This month By This Category		9840.0	
Total Discount Done This month By This Category		534.0	
Top Selling for This Month (April)		Least Selling for This Month (April)	
Product Name: Maggi		Product Name: Sugar	
Category Name: Grocery		Category Name: Grocery	
SubCategory Name: Packaged Foods		SubCategory Name: Staples	
<a href="#">View More Details</a>		<a href="#">View More Details</a>	




## 12. Category Trend

It shows the most selling products of particular categories, categorical wise data. If the admin wants to know which is the top selling category that sells the maximum amount of quantity, least selling category that sells the minimum amount of quantity then that can be inferred from it. It shows the top selling categories whose selling is highest and lowest selling categories whose selling is highest monthly, like which month has the highest selling product and changes the categories monthly.

Category Trend This Month

[Home Page](#) [Log out](#)




**Top Selling Category for April**

**Category Name:**  
Grocery

**SubCategory Name under This Category :**  
Dairy & Eggs  
Fruits & Vegetables  
Packaged Foods  
Snacks & Beverages  
Staples

**Product Name under This Category :**  
Almond  
Amul Cheese Block  
Amul Panner  
Amul Taaza Milk  
Apple  
Capsicum  
Chocos  
Cucumber  
Dark Fantasy Choco Fills  
Gowardhan Pasteurised Table Salted Butter  
Jam  
Maggi  
Nescafe Classic Coffee  
Nestle Milkmaid  
Orange  
Pomegranate  
Pringles Cream & Onion  
Redbull  
Rice  
Safola Edible Oil  
Silk Fruit & Nut  
Sugar  
Tomato Ketchup  
Tropicana Mixed Fruit Delight  
Wheat Flour



**Least Selling Category for April**

**Category Name:**  
Fashion

**SubCategory Name under This Category :**  
Children  
Men  
Women

**Product Name under This Category :**  
Half Sleeve Solid Women Black Top  
Baby Boys & Baby Girls Casual Dress Shorts  
Boys Printed Cotton Blend T Shirt  
Girls Midi/Knee Length Party Dress  
Jacquard Embroidered Kurta & Churidar Material  
Men Black Sports Shorts  
Men Slim Fit Solid Slim  
Regular Boys Blue Jeans  
Slides  
Slim Men Black Jeans  
Slip on Walking Shoes For Boys  
Solid Men Mandarin Collar White  
Walking Shoes For Men (Blue)  
Women Black Heels Sandal  
Women Light Blue Jeans

## 13. Order Report

This function will display the User Order Report of any specific user in which details like Name, Email Address, Contact Number and other details of All Time High Buyer and All Time Low Buyer is/are there. Similarly, it will display the Monthly report in which details like Name, Email Address, Contact Number and other details of This Month High Buyer (Current\_Month\_name) and the This Month Low Buyer (Current\_Month\_Name) is/are there. It will also display Overall Order Report in which the value of All Time Total Sell, Total Discount as well as the This Month Total Sell (Current\_Month\_Name), Total Discount (Current\_Month\_Name) will be displayed.

## Overall Order Report

All Time Total Sell	8412.0
All Time Total Discount	3014.0
This Month Total Sale (April)	8412.0
This Month Total Discount (April)	3014.0

## User Order Report

## All Time High Buyer

Name: Jap Purohit

Email Address : jpap041@gmail.com  
Contact Number : 9726624242[See More Details](#)

## All Time Low Buyer

Name: Poojan Sheth

Email Address : poojansheth76@gmail.com  
Contact Number : 9426264242[See More Details](#)

## This Month High Buyer (April)

Name: Jap Purohit

Email Address : jpap041@gmail.com  
Contact Number : 9726624242[See More Details](#)

## This Month Least Buyer (April)

Name: Poojan Sheth

Email Address : poojansheth76@gmail.com  
Contact Number : 9426264242[See More Details](#)

## 14. Apply Discount By Product:

This function helps in providing offers on particular products for the admin defined period of time. If there are branded products and companies are interested to get their product sold on discounted price like 500 off or more on the website, on any special days, so this helps admin to add offer price for that given amount of time given by the company, can be updated and the offer prize is set in every Cart table also if that certain product exists in user's cart.



### Apply Discount on Product

This will apply discount on particular product by price/percentage

[Apply Discount On Product](#)

#### PRODUCT DETAILS

Product Name

SUGAR

Apply Discount Type

By Price

Value of Discount Percentage/Discount Amount

145

Offer Till

07-05-2021

[Update Price](#)

## 15. Apply Discount By Category:

This function helps in giving discount in percentage on particular categories if there are really special events like 1<sup>st</sup>, 2<sup>nd</sup> or 10<sup>th</sup> anniversary of that website than in such situation the admin can give flat 30 to 50% discount on the whole category and even such offers attract more and more users towards this website. By this categorical discount the changes are of prizes are updated on all tables like ProductDetails table and in users Cart table if that certain product exists in users' cart.

**CATEGORY DETAILS**

Category Name

Cosmetics

Value of Discount Percentage

10

Offer Till

15-04-2021

Update Price

## **16. User Report:**


This show case report of individual user and shows necessary information like total discount, purchase for this particular month or overall. And also displays other necessary information

### User Report

Field	Value
Name	Jap Purohit
Email Address	pjap041@gmail.com
Contact Number	9726624242
Date Of Birth	2001-08-04
Address	C6 New Arpana Flats CNG Pump Ahmedabad Gujarat India Pincode:- 380007
City	Ahmedabad
State	Gujarat
Country	India
Pincode	380007
Role of User	Admin
Total Purchase Done Till Now	8362.0
Total Discount Availed Till Now	3014.0
Total Purchase Done In This Month (April)	8362.0
Total Discount Availed In This Month (April)	3014.0

## 17. Product Report

This show case report of individual products and shows necessary information like total revenue generate, total quantity sold for this particular month or overall. And also displays other necessary information

Product Report	
Field	Value
Product Name	Sugar
Product Description	1 kg Sugar
Image	
Company Name	Madhur
Category Name	Grocery (Category ID : 1)
Sub-Category Name	Staples (Sub-Category ID : 1)
Original Price	50.0
Offer Price	50.0
This Offer Applicable	2021-05-01
Product Active	Yes
Quantity Left	196
Total Quantity	4
Total Quantity Sold In This Month (April)	4
Total Revenue Generate	200.0
Total Revenue Generate In This Month (April)	200.0

## 18. Luck Draw Report

This selects random person from order details and sends mail that they have won a special discount and contact to admin to avail the discount. Random was selected from with the help of function.

BASIC DETAILS

First Name

Poojan

Last Name

Sheth

Address

A201 Bluh Bluh Flats Karkariya Lake Ahmedabad Gujrat India Pincode:- 380007

Email Address

poojansheth76@gmail.com


Date of Birth

31-03-2021

[Admin Dashboard](#)

Send Email

Discount From BigDeals.com



j.p.p@ahduni.edu.in

to poojansheth76

10:40 PM (0 minutes ago)

☆

↶

⋮

Respect Sir/Madam

You have won special discount offer for our website. Please contact our admin to avail the discount.

Regards.....

Congratulations!

## 19. Create, Update Operation for User-Table:

Admin gets facility to edit role of a particular user and block status from admin dashboard and also add user.

Manage Users

[Home Page](#) [Log out](#)

[Add User](#)

User ID	First Name	Last Name	Email ID	Password	Address	Date of Birth	City	State	Country	Pincode	Total Purchase Done	Total Discount Done	Contact Number	Role ID	Is Block	Security Question	Security Answer	Action
1	Jap	Purohit	pjap041@gmail.com	Jappurohit@041	C6 New Arpana Flats CNG Pump Ahmedabad Gujarat India Pincode:- 380007	2001-08-04	Ahmedabad	Gujarat	India	380007	8362.0	3014.0	9726624242	Admin	No	What is your favourite food dish	dosa	<a href="#">Update</a>
2	Mitansh	Gor	gormitansh@gmail.com	Mitanshgor@041	Royal Technosoft CG Road Oppo Citibank Ahmedabad Gujarat India Pincode:- 380007	2000-08-01	Ahmedabad	Gujarat	India	380007	0.0	0.0	9909995363	User	No	Who is your favourite player	virat	<a href="#">Update</a>
3	Poojan	Sheth	poojansheth76@gmail.com	Poojansheth@041	A201 Bluh Bluh Flats Karkariya Lake Ahmedabad Gujarat India Pincode:- 380007	2021-03-31	Ahmedabad	Gujarat	India	380007	50.0	0.0	9426264242	User	No	What is name of your pet	ladies	<a href="#">Update</a>

## 20. CRUD Operation for Product-Table:

Admin gets facility to perform CRUD Operation (Create, Update and Delete) on product table.

Manage Products

[Home Page](#) [Log out](#)

[Add Product](#)

[Deleted Product List](#)

Product ID	Product Name	Company Name	Category Name	SubCategory Name	Original Price	Offer Price	Offer Till	Quantity	Image	Product Description	Is Active	Action
1	Jam	Kissan	Grocery	Packaged Foods	199.0	180.0	2021-04-30	50		430 gram, Flavour : Mixed Fruit	Yes	<a href="#">Update</a> <a href="#">Delete</a>
2	Safola Edible Oil	Safola	Grocery	Staples	1500.0	1400.0	2021-04-30	97		5 litre, Raw Material :Groundnuts	Yes	<a href="#">Update</a> <a href="#">Delete</a>
3	Almond	California	Grocery	Staples	350.0	346.5	2021-04-29	146		500 gram	Yes	<a href="#">Update</a> <a href="#">Delete</a>
4	Sugar	Madhur	Grocery	Staples	50.0	50.0	2021-05-01	196		1 kg Sugar	Yes	<a href="#">Update</a> <a href="#">Delete</a>

## 21. CRUD Operation for Category-Table:



Admin gets facility to perform CRUD Operation (Create, Update and Delete) on category table.

Manage Category

Home Page Log out

Add Category

Deleted Category List

Category ID	Category Name	IsActive	Category Display Image	Action
1	Grocery	Yes		<a href="#">Update</a> <a href="#">Delete</a>
2	Toys	Yes		<a href="#">Update</a> <a href="#">Delete</a>

## 22. CRUD Operation for Sub Category-Table:

Admin gets facility to perform CRUD Operation (Create, Update and Delete) on sub-category table.

Add Sub-Category

Deleted Sub-Category List

Sub-Category ID	Sub-Category Name	Category Name	IsActive	Action
1	Staples	Grocery	Yes	<a href="#">Update</a> <a href="#">Delete</a>
2	Snacks & Beverages	Grocery	Yes	<a href="#">Update</a> <a href="#">Delete</a>
3	Packaged Foods	Grocery	Yes	<a href="#">Update</a> <a href="#">Delete</a>
4	Dairy & Eggs	Grocery	Yes	<a href="#">Update</a> <a href="#">Delete</a>
5	Fruits & Vegetables	Grocery	Yes	<a href="#">Update</a> <a href="#">Delete</a>
6	Puzzles	Toys	Yes	<a href="#">Update</a> <a href="#">Delete</a>
7	Soft-Toys	Toys	Yes	<a href="#">Update</a> <a href="#">Delete</a>

### 23. Display Order-Table:

Admin gets facility to see the order table and see whether order is delivered or not.

#### Order Detail

Order ID	User ID	Address	Payment Type	ProductID	Product Name	Quantity	Offer Price	Total Amount	Discount Amount	Order Date	Delivery Date	Delivery Status
14	3	A201 Bluh Bluh Flats Karkariya Lake Ahmedabad Gujarat India Pincode:- 380007	COD	4	Sugar	1	50.0	50.0	0.0	2021-04-15	2021-04-22	Not-Delievered
13	1	C6 New Arpana Flats CNG Pump Ahmedabad Gujarat India Pincode:- 380007	COD	4	Sugar	2	50.0	100.0	0.0	2021-04-15	2021-04-22	Not-Delievered
12	1	C6 New Arpana Flats CNG Pump Ahmedabad Gujarat India Pincode:- 380007	Net-Banking	8	Tomato Ketchup	2	90.0	180.0	60.0	2021-04-15	2021-04-22	Not-Delievered
11	1	C6 New Arpana Flats CNG Pump Ahmedabad Gujarat India Pincode:- 380007	COD	3	Almond	3	346.5	1039.5	10.5	2021-04-15	2021-04-22	Not-Delievered
10	1	C6 New Arpana Flats CNG Pump Ahmedabad Gujarat India Pincode:- 380007	Credit-Card	7	Maggi	5	150.0	750.0	100.0	2021-04-15	2021-04-22	Not-Delievered

### 24. Category Display User Side:

This is display of category display on user side and in each dropdown shows subcategory for that category and it displays category and subcategory which is active. If it is not active then it doesn't display on user display.

