CS:532 Database Systems
Project 2
Report

By

Prathamesh Lonkar
Bnumber: B00811727

and

Shreya N. Desai
Bnumber: B00813851

Objective

Our objective for this project is to create a Retail Management System which will allow us to search, update, display the various details of various entities involved in the Management Systems. We will be able to do the following operations in our project:

- Display the data of each table in our Database.
- Find the Monthly sale of a particular product.
- Insert the data into Purchase and Products Table.
- Search data of Employees, Customers and the details of a Purchases.

Procedures, Triggers, Sequences

In our Project we have used the following procedures, Triggers, Sequences:

1.)Insert into Purchase table:
Objective: Objective of this Procedure is to properly insert the data into the purchase table after validatng all the field which are required.

In the below procedure we are first validating whether the given cid,pid,eid are valid or not if they are valid and the purchase quantity is less than the quantity on hand(qoh) then only we are proceeding to insert the purchase. After the insertion is done we are checking if the updated qoh is less than the threshold, if it is less then we are automatically printing the message and ordering from supply, and displaying the new quantity in hand.

- procedure insert_in_purchase(p_id in products.pid%type,e_id in employees.eid%type,c_id in customers.cid%type,qty_no in purchases.qty%type, message out varchar2,message1 out varchar2,message2 out varchar2) is
  price products.original_price%type;
  quantity products.qoh%type;
  quant_threshold products.qoh_threshold%type;
  quantity1 products.qoh%type;
  quant_threshold1 products.qoh_threshold%type;
  new_sid suppliers.sid%type default 's1';
  M number default 10;
  new_supply supply.quantity%type;
  n_pid products.pid%type;
  n_cid customers.cid%type;
  n_eid employees.eid%type;
  new_supply1 supply.quantity%type;
  sup_pid number;
  begin
  select count(employees.eid) into n_eid from employees where employees.eid=e_id;
  select count(customers.cid) into n_cid from customers where customers.cid=c_id;

```
select count(products.pid) into n_pid from products where products.pid=p_id;
select original_price, qoh,qoh_threshold into price, quantity, quant_threshold
from products where pid=p_id;
if(qty_no > quantity) then
message:='invalid size of quantity';
else
if(n_pid<1) then
message:='Invalid Pid';
else
if(n_eid<1) then
message:='Invalid Eid';
else
if(n_cid<1) then
message:='Invalid Cid';
else

insert into purchases values (pur.nextval, e_id, p_id, c_id, qty_no,
sysdate,(qty_no * price));
update products set products.qoh=products.qoh-qty_no where
products.pid=p_id;
select qoh,qoh_threshold into quantity1, quant_threshold1 from products
where products.pid=p_id;
if(quantity1<quant_threshold1) then
new_supply := (M+quantity1+5);
message2:= 'quantity is less than the threshold';
select supply.sid into new_sid from supply where supply.pid=p_id order by
supply.sid asc fetch first 1 rows only;
insert into supply values (sup.nextval,p_id,new_sid,sysdate,new_supply);
select products.qoh into new_supply1 from products where products.pid=p_id;
message1:= 'The new quantity ordered is:' ||new_supply1;


end if;
end if;
end if;
end if;
end if;
end insert_in_purchase;
```

2.)Inserting into the product table.

Objective: Objective of this query is to insert the product details into the product table.

In the procedure bellow we are validating that the new product to be added dose not have the same pid as of the products already in the table. If the pid is unique then and only then we are inserting into the table.

- procedure insert_in_products(p_id in products.pid%type,p_name in products.pname%type,qoh_no in products.qoh%type,qoh_thre in products.qoh_threshold%type, original_p in products.original_price%type,discount in products.discnt_rate%type,message out varchar2) is
  cnt_pid number;
  begin
  select count(products.pid) into cnt_pid from products where products.pid=p_id;
  if(cnt_pid>0)then
  message:= 'Product Id already Exists';
  else
  insert into products values (p_id, p_name, qoh_no, qoh_thre, original_p, discount);
  end if;
  end insert_in_products;

3.)Displaying all the tables.
Objective: Objective of these procedures are to display the table data using a sys_refcursor.

We are simply fetching the data and sending it to the java code using the sys_refcursor.

- procedure show_products(rc out sys_refcursor);
  begin
  open rc for
  select * from products;
  end show_products;

- procedure show_purchases(rc out sys_refcursor);
  begin
  open rc for
  select * from purchases;
  end show_purchases;

- procedure show_customers(rc out sys_refcursor);
  begin
  open rc for
  select * from customers;
  end show_customers;

- procedure show_employees(rc out sys_refcursor);
  begin
  open rc for
  select * from employees;
  end show_employees;

- procedure show_supply(rc out sys_refcursor);
  begin
  open rc for
  select * from supply;
  end show_supply;

- procedure show_suppliers(rc out sys_refcursor);
  begin
  open rc for
  select * from suppliers;
  end show_suppliers;

- procedure show_logs(rc out sys_refcursor);
  begin
  open rc for
  select * from logs;
  end show_logs;

4.) Displaying the Average Monthly Sale of a product.
Objective: the objective of this procedure is to report the average monthly sale.

In this procedure we are firstly checking if the pid entered is valid or not. If it is valid then we are finding the monthly sale of that product by grouping the date ,pid,pname.

- procedure show_monthly_sale(rc out sys_refcursor, id in products.pid%type,message out varchar2);
  valid_pid products.pid%type;
  begin
  select count(products.pid) into valid_pid from products where products.pid = id;
  if(valid_pid<1)then
  message:='Invalid Pid';
  else
  open rc for
  select products.pid, products.pname, to_char(purchases.ptime, 'Mon-YYYY') as time, sum(purchases.qty) as qty, sum(purchases.total_price) as total_price,(sum(purchases.total_price)/sum(purchases.qty)) as avg from products, purchases where products.pid=purchases.pid AND products.pid=id group by to_char(purchases.ptime,'Mon-YYYY'),products.pid, products.pname;
  end if;
  end show_monthly_sale;

5.) For displaying the data requested to be searched.
Objective: The objective of this procedure is to display the data which is requested to be searched.

In all of the procedures we are first checking wheter the data is valid or not i.e. we are validating the pid, cid, in the following procedure.

- procedure show_EmployeeInfo(rc out sys_refcursor, id in employees.eid%type,message out varchar2);
  cnt_emp number;
  begin
  select count(employees.eid) into cnt_emp from employees where employees.eid=id;
  if(cnt_emp<1) then
  message:= 'Invalid Eid';
  else
  open rc for
  select * from employees where employees.eid=id;
  end if;
  end show_EmployeeInfo;

- procedure show_CustomerInfo(rc out sys_refcursor, id in customers.cid%type,message out varchar2);
  cnt_cus number;
  begin
  select count(customers.cid) into cnt_cus from customers where customers.cid=id;
  if(cnt_cus<1) then
  message:= 'Invalid Cid';
  else
  open rc for
  select * from customers where customers.cid=id;

  end if;
  end show_CustomerInfo;

- procedure show_PurchaseInfo(rc out sys_refcursor, id in purchases.cid%type,message out varchar2);
  cnt_pur number;
  begin
  select count(purchases.cid) into cnt_pur from purchases where purchases.cid=id;

```
if(cnt_pur<1) then
message:= 'Invalid Purchase Info';
        else
   open rc for
   select * from purchases where purchases.cid=id;

end if;
end show_PurchaseInfo;
```

6.) Triggers:

Objective: The objective of these triggers are to insert into the log table and make changes to other table when some value is changed or added in a particular table.

- ```
  create or replace trigger trigger1
  after
  insert
  on purchases
  for each row
  declare
  n_cid char(4);
  begin
  n_cid := :new.cid;
  insert into logs values (logf.nextval,user, sysdate, 'purchases','insert',
  :NEW.pur#);
  update customers set customers.visits_made=customers.visits_made+1 ,
  customers.last_visit_date=sysdate where customers.cid=n_cid;
  end trigger1;
  /
  show errors
  /
  ```
- ```
  create or replace trigger trigger4
  after
  insert
  on supply
  for each row
  declare
  n_pid char(4);
  n_qty number(5);
  begin
  n_pid := :new.pid;
  n_qty := :new.quantity;
  insert into logs  values (logf.nextval,user, sysdate, 'supply','insert', :NEW.sup#);
  ```

```
update products set products.qoh=products.qoh+n_qty where
products.pid=n_pid;

end trigger4;
/
show errors


create or replace trigger trigger2
after
update
of qoh on  products
for each row
begin
insert into logs values (logf.nextval,user, sysdate, 'products','update',
:NEW.pid);
end trigger2;
/
show errors
/
create or replace trigger trigger3
after
update
of visits_made on customers
for each row
begin
insert into logs values (logf.nextval,user, sysdate, 'customers','update',
:NEW.cid);
end trigger3;
```

7.) Sequences

Objective: The objective of the sequences is to isert a value dynamically every time a data is been inserted into the tables.

The following sequences are been used to generate a particular pattern of data to be inserted. We are using sequence to insert pur#, sup# and log# values.

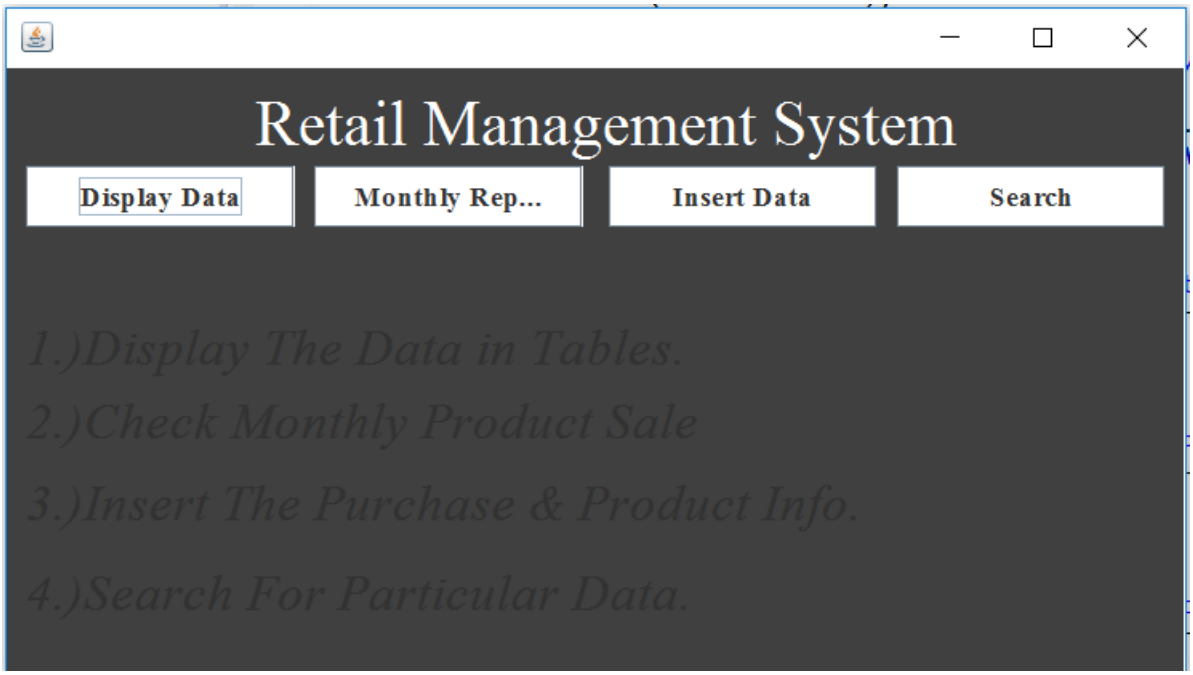
- create sequence pur
  increment by 1
  start with 100011
  minvalue 000000
  maxvalue 1000000
  nocycle
  cache 20;

- create sequence sup
  increment by 1
  start with 11
  minvalue 0000
  maxvalue 1000
  nocycle
  cache 20;

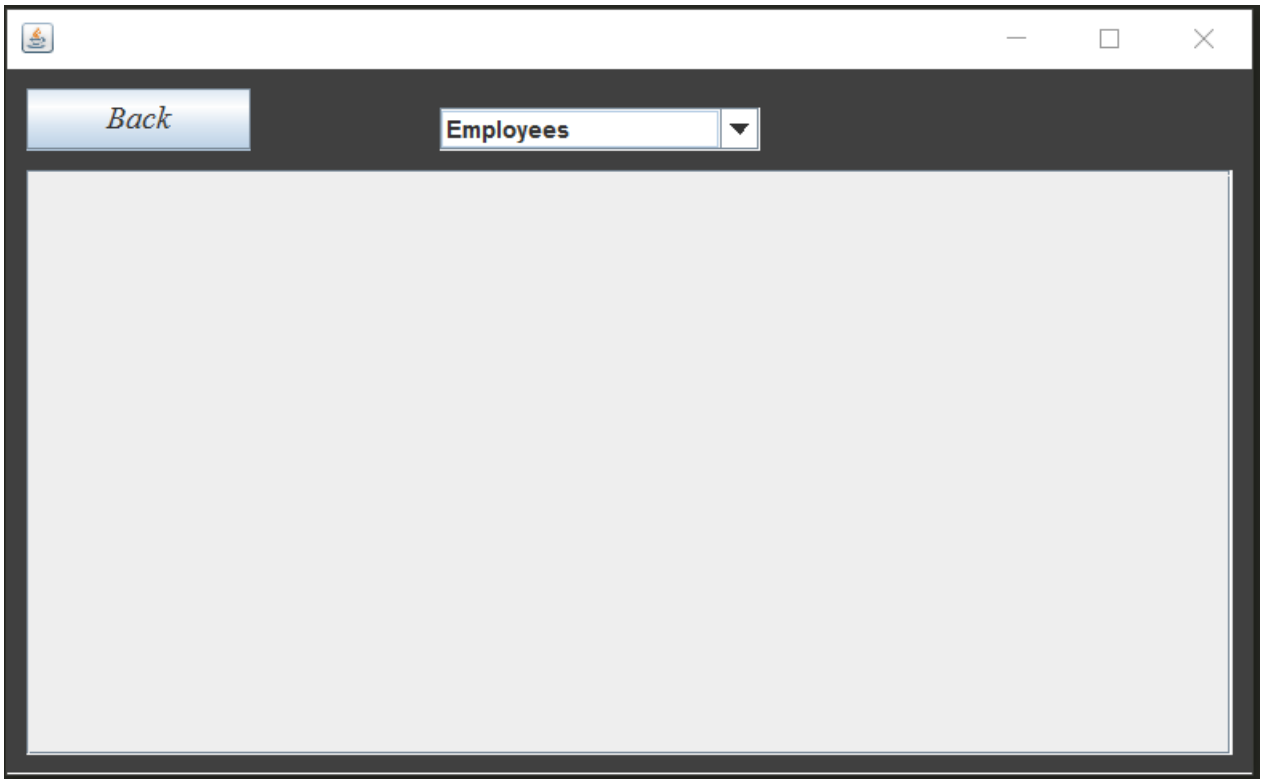
- create sequence logf
  increment by 1
  start with 0
  minvalue 00000
  maxvalue 10000
  nocycle
  cache 20;

Interface of the Project

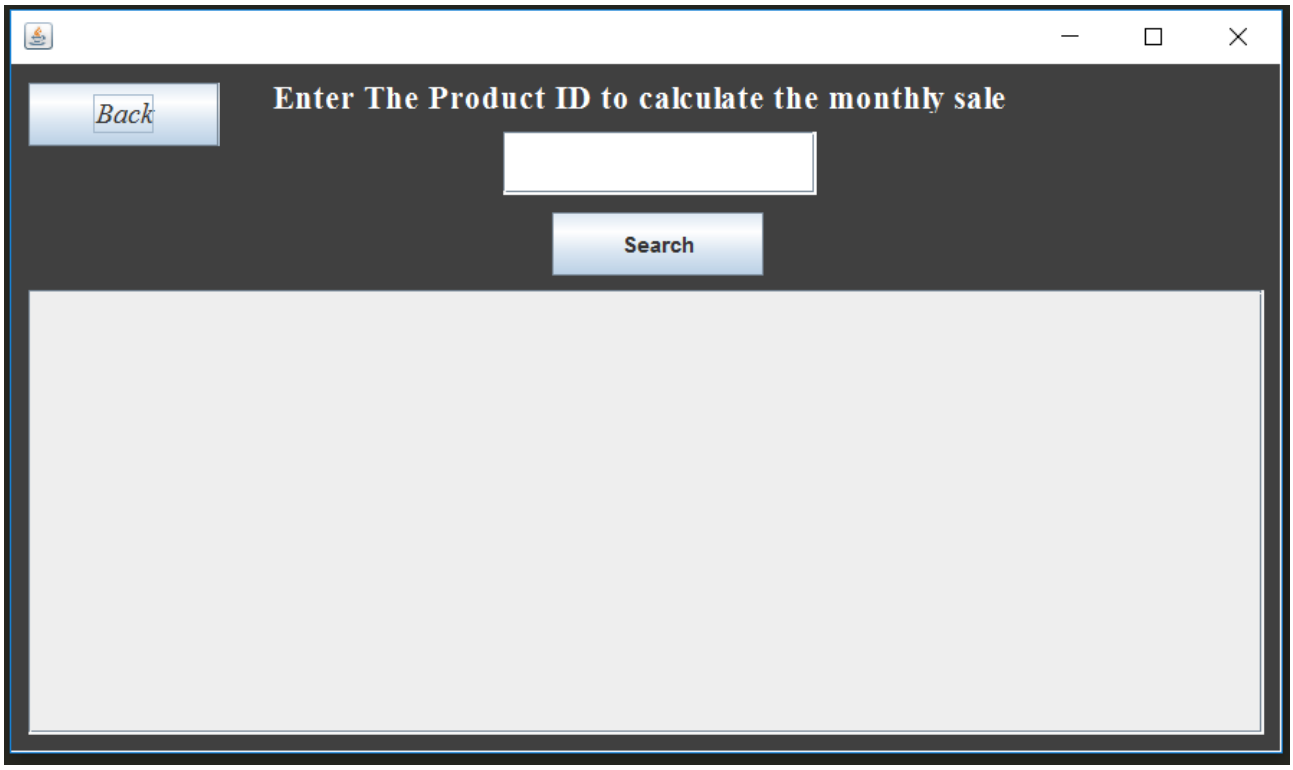
1.)Start Page



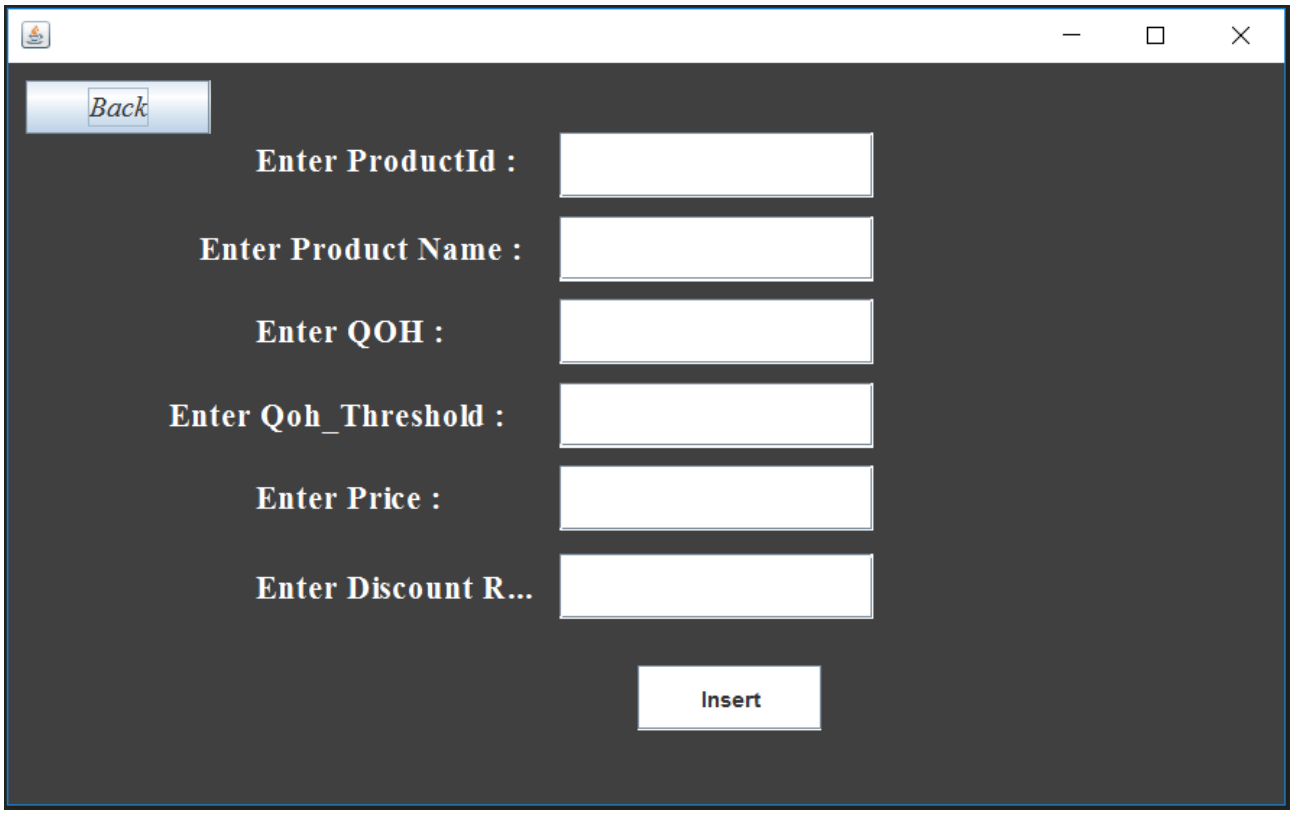
2)Display Page



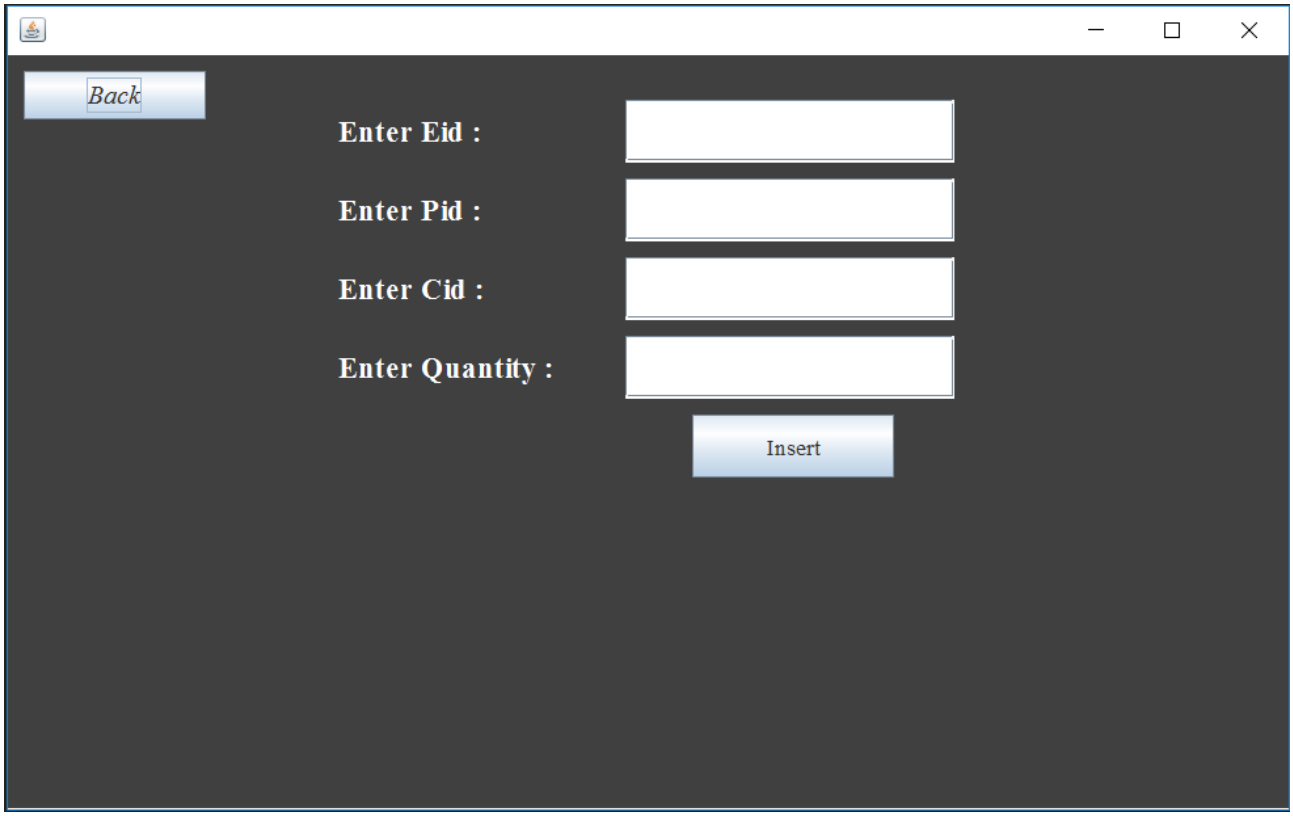
3.) Show Monthly Report



4.) Add Product



5.)Add Purchases



6.)Search

Thank You