### **Agha Muqarib**

### **18B-080-SE**

### ------------------------

### **Mansoor Ahmed**

### **18B-129-SE**

Pharmacy Database Application

### **Submitted To: Ma’am Noor-Ul-Huda**

**

***Table of Content:***

* *Introduction*
* *Description*
* *Schema*
* *Script*
* *Entity Relationship Diagram*
* *Functional Dependencies*
* *Data Flow Diagram*
* *Features of Application*
* *Sample Queries*
* *Images of Application*
* *User Manual.*

**Introduction**:

Pharmacy Management system is a database application that will cater the needs of storing, managing, querying and retrieving pharmacy data needs i.e. managing staff, customer, transaction, product and supplier records. The application is generated by a script and is further beautified manually through Alex app builder.

**Description:**

The scenario was opted because of the storage and management requirements of a pharmacy. If a manual storage system is considered, a pharmacy can be maintained through some file or .xlsx sheet but that wouldn’t be the ideal practice. Realizing this, a pharmacy which is a proper Entity made up of several components having their own attributes and operations later to be discussed in the schema, an attempt to make an almost fully-automated database application was brought into action. The application has automated the operation by adding rows or records manually through graphical user interface which is a utility itself, deleting or updating records, generate reports, view charts and performing calculations.

**Schema**:

The schema consists of 6 entities and one bridge table. These tables are as follows:

* **Staff –** The staff table consists of the records of employees or simply staff in the pharmacy. The attributes include Staff ID, which is serving as a Primary key, name, salary, commission, hire date, and their job.
* **Customer –** The customer table consists of records of customers with attributes like customer ID (PK), name and phone number.
* **Supplier –** Supplier contains records of supplier supplying products to pharmacy. The attributes include Supplier ID, Company Name, supplier name, address and phone number.
* **Product** **–** Product table contains all the products the pharmacy sells or simply has as am inventory. It includes Product ID (PK), Product Name, Unit Price, Quantity, Supplier ID (FK) and Category ID (FK).
* **Category** **–** The category table contains the records of the categories the products belong to. Attributes are category ID (PK), category name and the description of category.
* **Transaction –** The transaction tables include records of the transactions made by customers. It’s attributes are Transaction ID (PK), Customer ID (FK), Staff ID (FK) and transaction Date.
* **Transaction Details –** This table serves as a bridge table between transaction and the products on which transactions are made. It’s attributes includes Transaction ID (FK), Product ID (FK), Unit Price, Quantity and Discount.

**Script:**

CREATE TABLE Staff

(

  StaffID NUMBER NOT NULL,

  Name VARCHAR2(20) NOT NULL,

  Job VARCHAR2(30),

  Salary NUMBER(7, 2),

  Commission NUMBER(7, 2),

  HireDate  DATE,

  Address  VARCHAR2(60),

  Phone  VARCHAR2(24),

CONSTRAINT PK\_Staff PRIMARY KEY (StaffID)

)

/

CREATE TABLE Category

(

  CategoryID  NUMBER NOT NULL,

  CategoryName  VARCHAR2(30) NOT NULL,

  Description  VARCHAR2(300),

CONSTRAINT PK\_Category PRIMARY KEY (CategoryID))

/

CREATE TABLE Customer

(

  CustomerID NUMBER NOT NULL,

  CustomerName  VARCHAR2(30),

  Phone  VARCHAR2(24),

CONSTRAINT PK\_Customer PRIMARY KEY (CustomerID)

)

/

CREATE TABLE Supplier

(

  SupplierID  NUMBER NOT NULL,

  CompanyName  VARCHAR2(40) NOT NULL,

  SupplierName  VARCHAR2(30),

  Address  VARCHAR2(60),

  Phone  VARCHAR2(24),

CONSTRAINT PK\_Supplier

  PRIMARY KEY (SupplierID)

)

/

CREATE TABLE Product

(

  ProductID  NUMBER NOT NULL,

  ProductName  VARCHAR2(40) NOT NULL,

  SupplierID  NUMBER,

  CategoryID  NUMBER,

  Quantity  VARCHAR2(20),

  UnitPrice  NUMBER,

CONSTRAINT PK\_Product

  PRIMARY KEY (ProductID),

CONSTRAINT CHK\_Product\_Price   CHECK ((UnitPrice >= 0)),

CONSTRAINT FK\_Product\_Category FOREIGN KEY (CategoryID) REFERENCES Category(CategoryID),

CONSTRAINT FK\_Product\_Supplier FOREIGN KEY (SupplierID) REFERENCES Supplier(SupplierID)

)

/

CREATE TABLE Transaction

(

  TransactionID  NUMBER NOT NULL,

  CustomerID  NUMBER,

  StaffID  NUMBER,

  TransactionDate  DATE,

CONSTRAINT PK\_Transaction

  PRIMARY KEY (TransactionID),

CONSTRAINT FK\_Transaction\_Customer FOREIGN KEY (CustomerID) REFERENCES Customer(CustomerID),

CONSTRAINT FK\_Transaction\_Staff FOREIGN KEY (StaffID) REFERENCES Staff(StaffID)

)

/

CREATE TABLE TransactionDetails

(

  TransactionID  NUMBER NOT NULL,

  ProductID  NUMBER NOT NULL,

  UnitPrice  NUMBER NOT NULL,

  Quantity  NUMBER NOT NULL,

  Discount  NUMBER NOT NULL,

CONSTRAINT PK\_Transaction\_Details

  PRIMARY KEY (TransactionID, ProductID),

CONSTRAINT CHK\_Discount   CHECK ((Discount >= 0 and Discount <= 1)),

CONSTRAINT CHK\_Quantity   CHECK ((Quantity > 0)),

CONSTRAINT CHK\_UnitPrice   CHECK ((UnitPrice >= 0)),

CONSTRAINT FK\_TransactionDetails\_Transaction FOREIGN KEY (TransactionID) REFERENCES Transaction(TransactionID),

CONSTRAINT FK\_TransactionDetails\_Product FOREIGN KEY (ProductID) REFERENCES Product(ProductID)

)

/

begin

INSERT INTO Customer(CustomerID, CustomerName, Phone) VALUES (001, 'Akram','03452004321');

INSERT INTO Customer(CustomerID, CustomerName, Phone) VALUES (002, 'Afzal','03122004321');

INSERT INTO Customer(CustomerID, CustomerName, Phone) VALUES (003, 'Asif','03352004321');

INSERT INTO Customer(CustomerID, CustomerName, Phone) VALUES (004, 'Areeb','03332004321');

INSERT INTO Customer(CustomerID, CustomerName, Phone) VALUES (005, 'Ahsan','03234304321');

INSERT INTO Customer(CustomerID, CustomerName, Phone) VALUES (006, 'Basit','03234204321');

INSERT INTO Customer(CustomerID, CustomerName, Phone) VALUES (007, 'Brian','0338574321');

INSERT INTO Customer(CustomerID, CustomerName, Phone) VALUES (008, 'Catherine','0354554421');

INSERT INTO Customer(CustomerID, CustomerName, Phone) VALUES (009, 'Carla','03335423321');

commit;

end;

/

begin

INSERT INTO Category(CategoryID, CategoryName, Description) VALUES (001, 'Food and beverages', 'Soft drinks, coffees,chips, teas, nimcos');

INSERT INTO Category(CategoryID, CategoryName, Description) VALUES (002, 'Skin care', 'Creams, Tubes,Ointments,Gels,Oil');

INSERT INTO Category(CategoryID, CategoryName, Description) VALUES (003, 'Feminine Products', 'Sanitary Pads, OTC Products,Sanitary Napkins, Hair Removing Creams');

INSERT INTO Category(CategoryID, CategoryName, Description) VALUES (004, 'Medicine', 'Tablets, Syrups,Injections, Drips, Ointments');

INSERT INTO Category(CategoryID, CategoryName, Description) VALUES (005, 'Baby & Mother Care', 'Feeders, Pampers,Nipples, Baby Eatables, Napkins');

commit;

end;

/

begin

INSERT INTO Product(ProductID, ProductName, SupplierID, CategoryID, Quantity, UnitPrice) VALUES (001, 'Johson Baby Lotion', 1, 1, 100, 180);

INSERT INTO Product(ProductID, ProductName, SupplierID, CategoryID, Quantity, UnitPrice) VALUES (002, 'Baby Diaper', 1, 2, 100, 50);

INSERT INTO Product(ProductID, ProductName, SupplierID, CategoryID, Quantity, UnitPrice) VALUES (003, 'Chips', 2, 4, 100, 180);

INSERT INTO Product(ProductID, ProductName, SupplierID, CategoryID, Quantity, UnitPrice) VALUES (004, 'Ascard', 1, 4, 100, 180);

INSERT INTO Product(ProductID, ProductName, SupplierID, CategoryID, Quantity, UnitPrice) VALUES (005, 'Humilin 70/30', 1, 4, 100, 180);

INSERT INTO Product(ProductID, ProductName, SupplierID, CategoryID, Quantity, UnitPrice) VALUES (006, 'Monteka', 1, 4, 100, 180);

commit;

end;

begin

INSERT INTO Staff(StaffID, Name, Job, Salary, Commission, Hiredate, Address, Phone) VALUES (001,'Mansoor Khan','Pharmacy technician',50000,1000,'12/10/2000','Street No XYZ, Block 93, Gulshan, Karachi',   0324789788);

INSERT INTO Staff(StaffID, Name, Job, Salary, Commission, Hiredate, Address, Phone) VALUES (002,'Luiz','Pharmacy assistant.',35000,150,'6/6/1999','House No PQR, Block 3, Sacramento, Karachi', 0321445588);

INSERT INTO Staff(StaffID, Name, Job, Salary, Hiredate, Address, Phone) VALUES (003,'Pique','Pharmacy clerk',25000,'2/3/2010','Street No ABC, Block 35, DHA, Karachi',  0300789788);

INSERT INTO Staff(StaffID, Name, Job, Salary, Commission, Hiredate, Address, Phone) VALUES (004,'Xavi','Pharmacy technician',50000,100,'3/4/2009','House No LMN, Block 90, KDA, Karachi',   0334089788);

INSERT INTO Staff(StaffID, Name, Job, Salary, Commission, Hiredate, Address, Phone) VALUES (005,'Iniesta','Pharmacy technician',200,1000,'6/25/2000','Street No WXY, Block 80, PIB, Karachi', 03747897188);

INSERT INTO Staff(StaffID, Name, Job, Salary, Commission, Hiredate, Address, Phone) VALUES (006,'Bale','Pharmacy assistant',40000,300,'12/31/2000','Street No KLM, Block 93, FB Area, Karachi', 0350897288);

INSERT INTO Staff(StaffID, Name, Job, Salary, Commission, Hiredate, Address, Phone) VALUES (007,'Lionel Messi','Pharmacist',12500,500,'2/28/2000','House No CDE, Block 93, Korangi 2, Karachi', 03217849788);

commit;

end;

/

begin

INSERT INTO Supplier(SUPPLIERID, COMPANYNAME, SUPPLIERNAME, ADDRESS, PHONE) VALUES (001,'Unity Foods Ltd','Nasir Khan Jan','Street No XYZ, Block 93, Gulshan, Karachi',03123325344);

INSERT INTO Supplier(SUPPLIERID, COMPANYNAME, SUPPLIERNAME, ADDRESS, PHONE) VALUES (002,'Hilton Pharma','Akram Bloch','34-C, Block, 6, P.E.C.H.S, Karachi',0324789788);

INSERT INTO Supplier(SUPPLIERID, COMPANYNAME, SUPPLIERNAME, ADDRESS, PHONE) VALUES (003,'RG Pharmaceutica','Dawood Ibrahim','F-95, Off, Hub River Road, S.I.T.E, Karachi',032004234534);

INSERT INTO Supplier(SUPPLIERID, COMPANYNAME, SUPPLIERNAME, ADDRESS, PHONE) VALUES (004,'Shield Corps. Ltd','Sultan Akbar','703, Progressive Square, Block 6, Nursery, Karachi',0321413432);

INSERT INTO Supplier(SUPPLIERID, COMPANYNAME, SUPPLIERNAME, ADDRESS, PHONE) VALUES (005,'Johnson & Johnson','Tughlug','A137, Gulshan 13-B Block 13 B Gulshan-e-Iqbal, Karachi',0308093455233);

INSERT INTO Supplier(SUPPLIERID, COMPANYNAME, SUPPLIERNAME, ADDRESS, PHONE) VALUES (006,'Oriflame','Babar','Suite 813, UNI PLAZA 8th Floor, I.I Chundrigar Rd,Karachi',03325213253);

INSERT INTO Supplier(SUPPLIERID, COMPANYNAME, SUPPLIERNAME, ADDRESS, PHONE) VALUES (007,'Edgewell Personal Care','Humayun','Pak Emp Co-Operative Housing Society Block 6 PECHS, Karachi',0307894324743);

INSERT INTO Supplier(SUPPLIERID, COMPANYNAME, SUPPLIERNAME, ADDRESS, PHONE) VALUES (008,'The Searle Company Ltd','Shahjahan','Mehran Town Sector 23 Korangi, Karachi',034235404523);

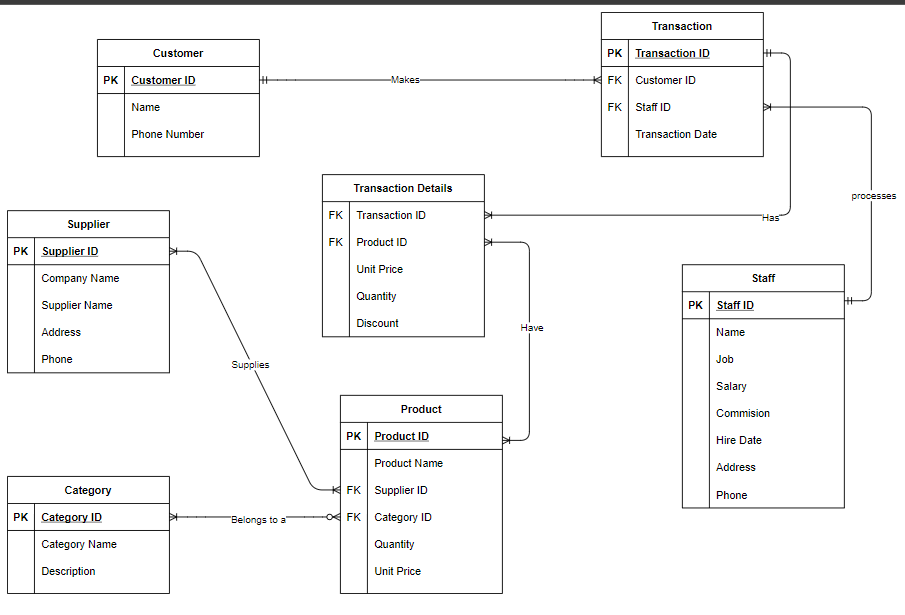
commit;

end;

/

**Entity Relationship Diagram:**

The below diagram depicts the entities (tables) and their relations. The diagram is designed using [app.diagrams.net](https://app.diagrams.net/) , which is an online utility to design diagrams.



**Functional Dependencies:**

Following are the functional dependencies of attributes in each table:

* **Customer** **(Customer ID, Customer Name, Customer Phone number)**

Customer ID🡪 Customer Name

Customer ID🡪 Customer Phone number

* **Supplier (Supplier ID, Supplier Name, Company Name, Phone No)**

Supplier ID🡪 Supplier Name

Supplier ID🡪 Company Name

Company Name🡪 Company Address

Supplier ID🡪 Phone No

* **Category (Category ID, Category Name, Description)**

Category ID 🡪 Category Name

Category Name🡪 Description

* **Product (Product ID, Product Name, Unit Price, Product Quantity (stock))**

Product ID 🡪 Product Name

Product ID 🡪 Unit Price

Product ID 🡪 Product Quantity (stock)

* **Staff (Staff ID, Staff Name, Hire Date, Job, Staff Phone, Address, Salary, Commission)**

Staff ID 🡪 Staff Name

Staff ID 🡪 Hire Date

Staff ID 🡪 Job

Staff Name🡪 Staff Phone

Staff Name🡪 Address

Staff ID🡪 Salary

Staff ID🡪 Commission

* **Transaction (Transaction ID, Transaction Date, Quantity, Discount)**

Transaction ID🡪 Transaction Date

Product ID, Transaction ID🡪 Quantity

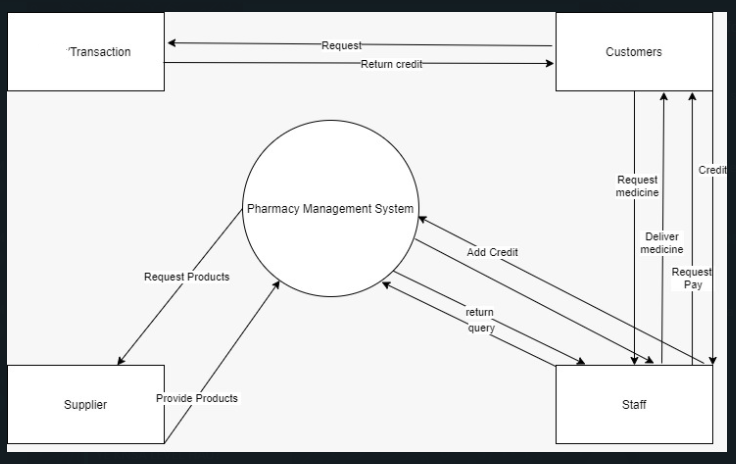
Transaction ID🡪 Discount

Transaction ID🡪 Customer ID

Transaction ID🡪 Staff ID

**Data Flow Diagram:**

The below diagram depicts how the data will flow throughout the database and which entity is doing what operation. It is to be noted that it is a level 0 DFD or simply a context diagram which would describe or give the overview of while application being modeled or analyzed. The diagram is designed using [app.diagrams.net](https://app.diagrams.net/), an online utility to design diagrams of sort and else.



**Features**:

The salient features of the application are:

**Login:**

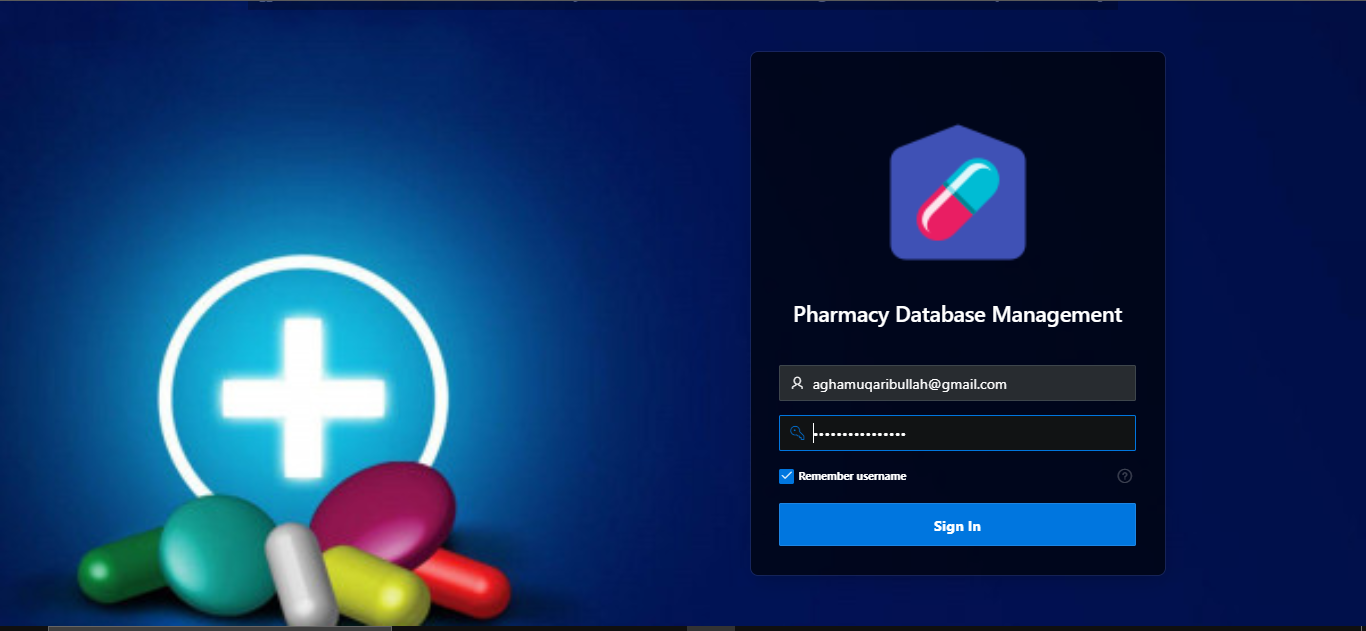
Login page is designed to be for administrator only as a pharmacy has only one person to maintain records. It requires a user name and a password and access to application is only possible if authenticated. The login credentials are as follows:

**URL:** [Pharmacy Database Management - Sign In (oracle.com)](https://apex.oracle.com/pls/apex/tegspace/r/pharmacy-database-management1/login?session=115505939990591)

**Username:** [aghamuqaribullah@gmail.com](mailto:aghamuqaribullah@gmail.com)

**Password:** Amuqarib@best2.

It is to be noted that the login page is self-customized by adding CSS to make it look prettified. The pictorial representation of the login form is as follows:



**Adding a record:**

Adding records as a new row is possible in every table, be it staff, customer, transaction, transaction details, product, supplier or category. Records can be added via a form as well as by writing a query in SQL Commands option.

**Updating a record:**

Editing, updating or modifying records is possible in every table, be it staff, customer, transaction, transaction details, product, supplier or category. Records can be modified via a form as well as by writing a query in SQL Commands option.

**Deleting a record:**

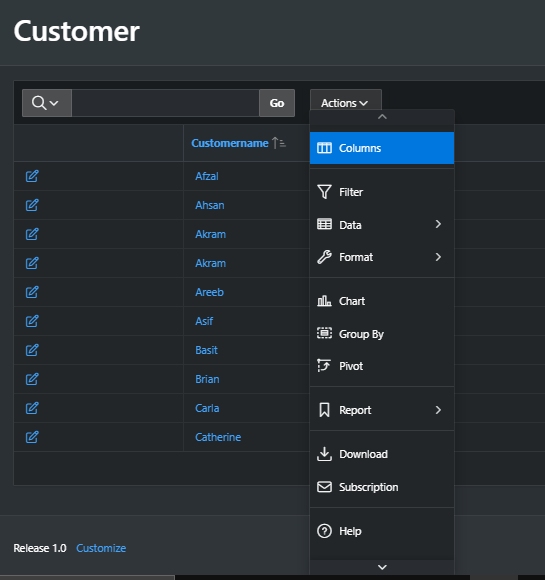
Deleting records as a new row is possible in every table, be it staff, customer, transaction, transaction details, product, supplier or category. Records can be deleted via a form as well as by writing a query in SQL Commands option.

**Search a record:**

Records from any tables can be searched by writing their names as well as they can be filtered as per need. They can also be searched as a Select Query in SQL Command option in every table.

**Perform Operations:**

The application provides the utility of performing multiple mathematical operation likes of which include sum, count, average, aggregate at cetera. The records can also be presented in customized way. The utilities offered are sort, group by, order by at cetera.

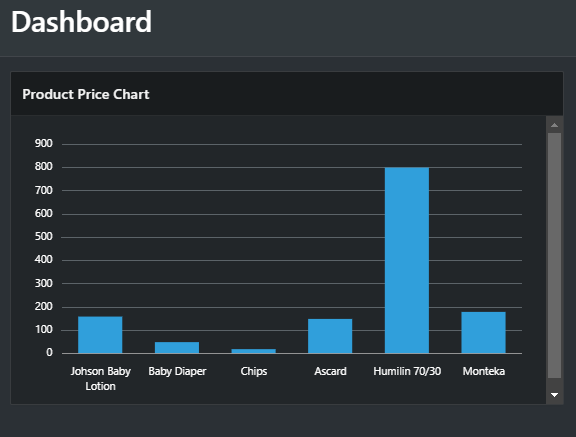


**Dashboard to View Charts:**

There are two charts in the application.

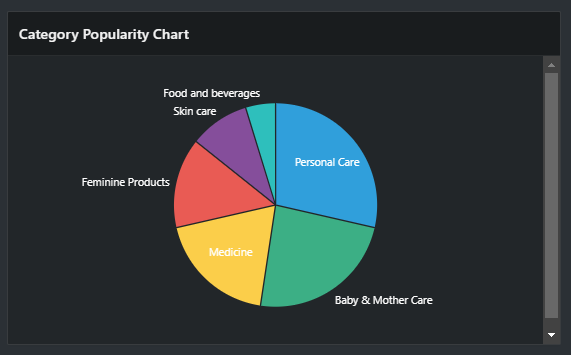
**Product Price Chart:**

The first chart is a bar chart (or more of an interactive report represented graphically) named as Product Price Chart which gives a glance understanding of how each product is priced per unit. It is updated after every addition, modification or deletion of a record in the product table.



**Category Popularity Chart:**

The second chart is a pie chart named as Category Popularity which shows, which is the most popular category in terms of sales. It is updated after every addition, modification or deletion of a record in the transaction table.



**Transactions Report:**

It is a faceted report with an SQL Query behind it. Below is the query at the backend:

SELECT T.TRANSACTIONID,S.NAME AS "STAFF NAME",T.CUSTOMERID,T.TRANSACTIONDATE,

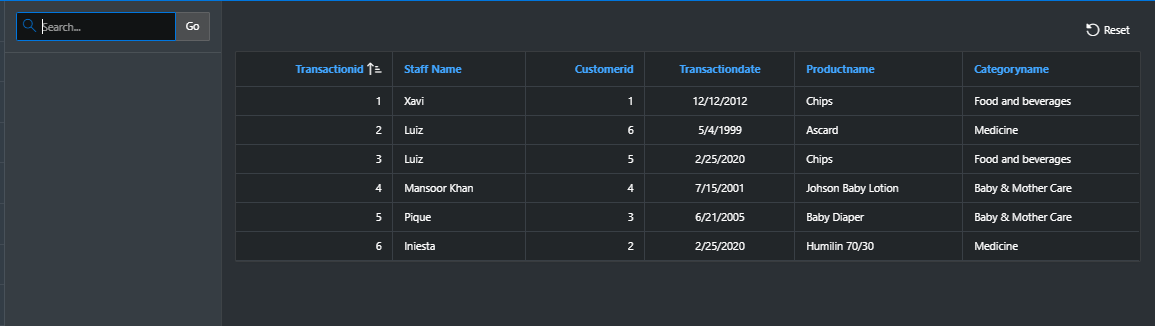
P.PRODUCTNAME,CAT.CATEGORYNAME FROM TRANSACTION T INNER JOIN

TRANSACTIONDETAILS TD ON T.TRANSACTIONID=TD.TRANSACTIONID INNER JOIN

PRODUCT P ON TD.PRODUCTID=P.PRODUCTID INNER JOIN CATEGORY CAT ON

 P.CATEGORYID=CAT.CATEGORYID INNER JOIN STAFF S ON T.STAFFID=S.STAFFID;

The query actually joins all the major attributes to form a report for all the major attributes from multiple tables giving a summary of transactions that took place corresponding to the attributes involved in transaction. The records can also be searched in the search pane.

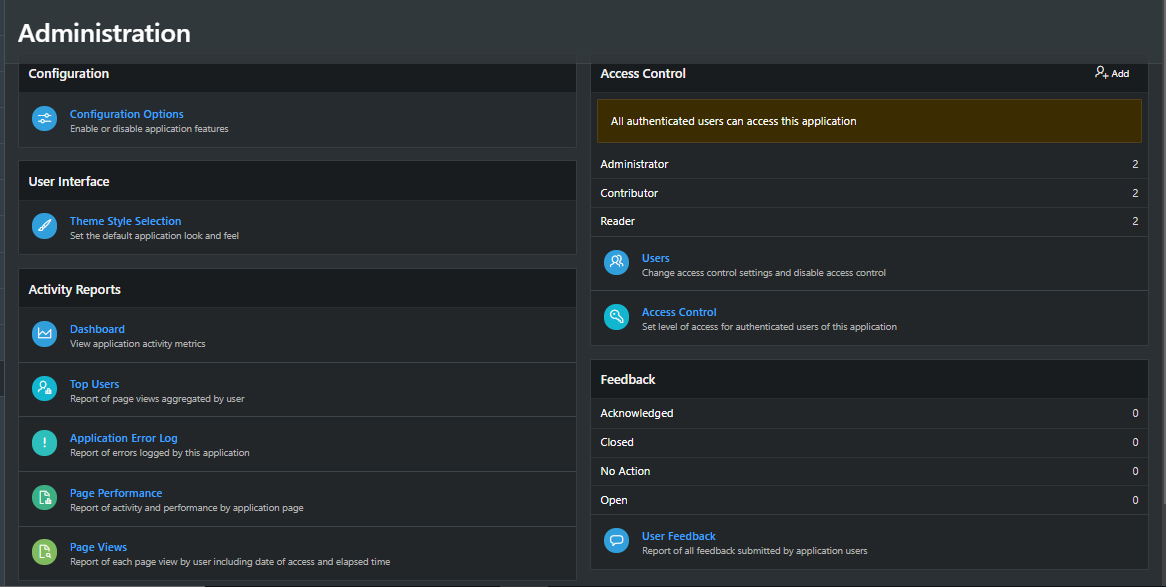


**Administration:**

The administration section provides multiple options.

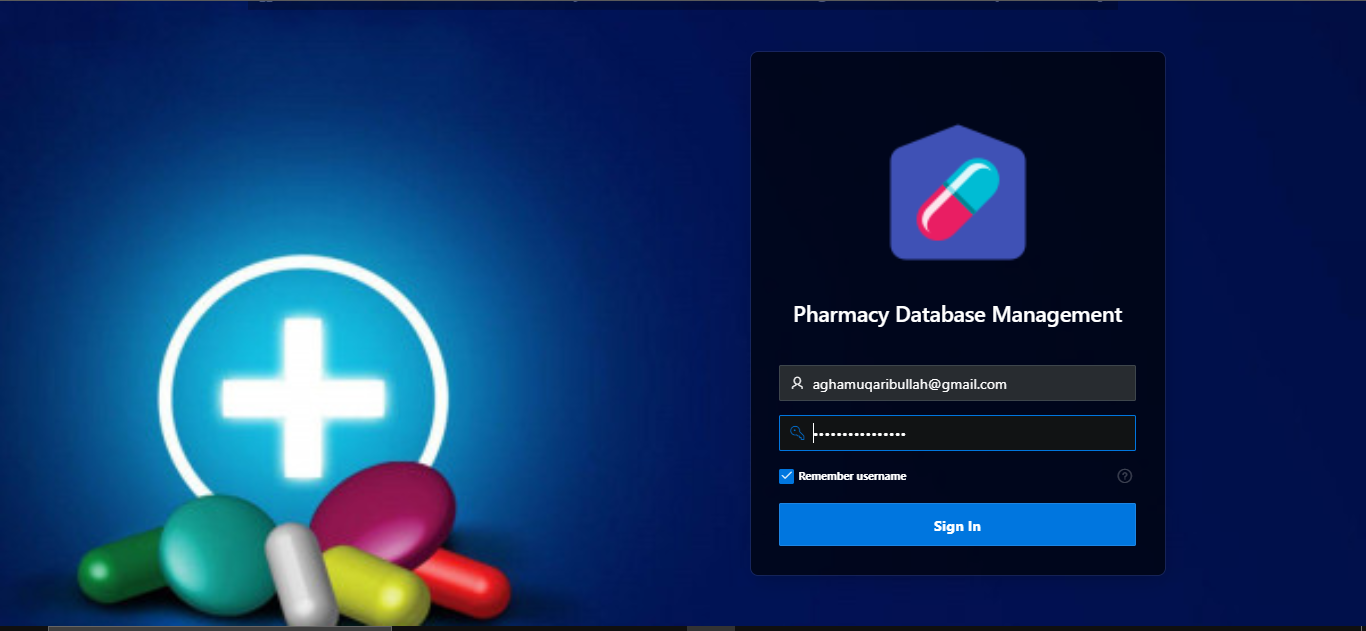
The activity dashboard is a utility, which describes the functional popularity of pages of application. It gives the number of view of a page.

* We can change themes in theme settings.
* Access control can be viewed.
* The most active user can also be viewed.

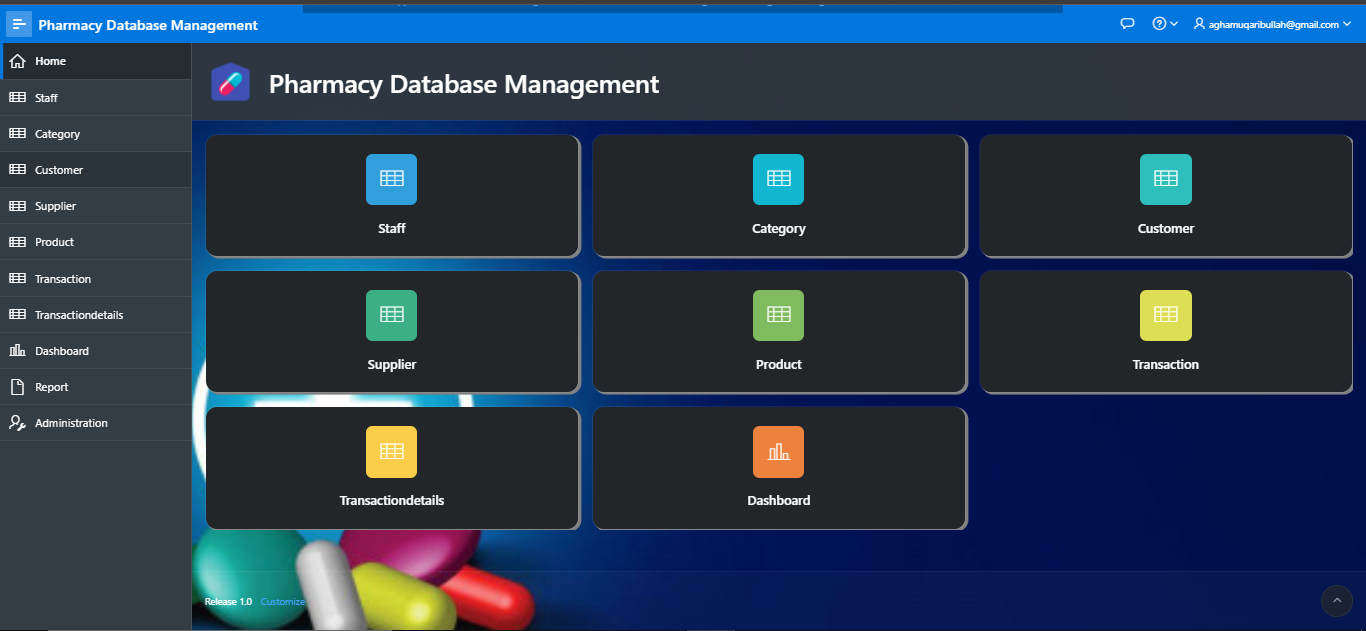


**Images of Application:**

**Login Page:**

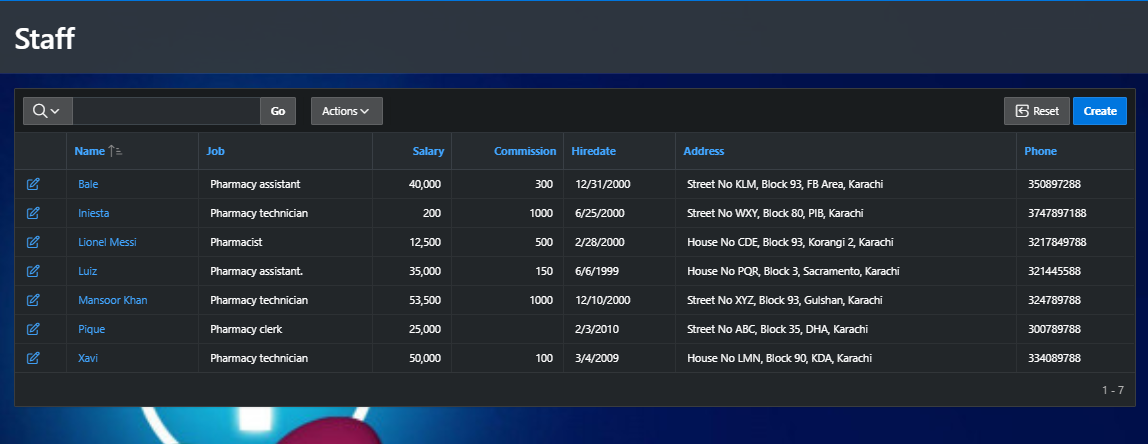


**Home Page:**

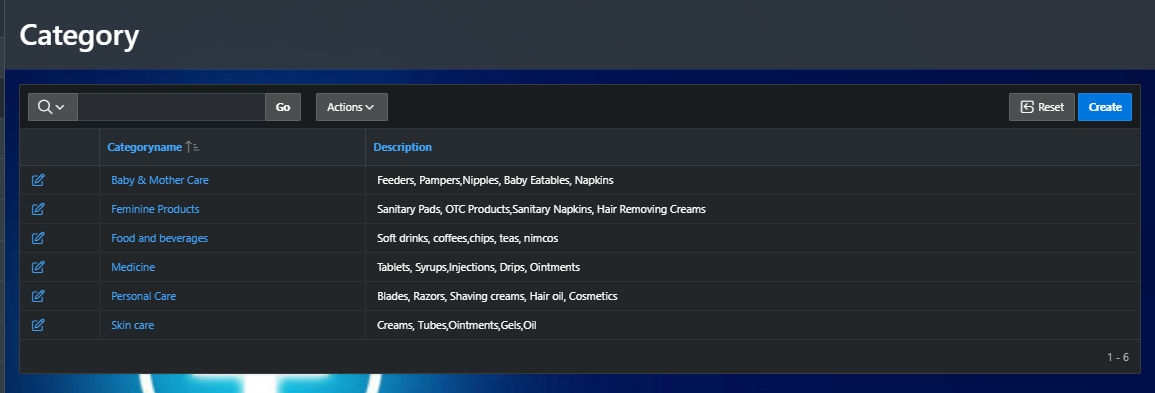


At the left is the navigation bar, and the blocks in middle are tables and a dashboard block.

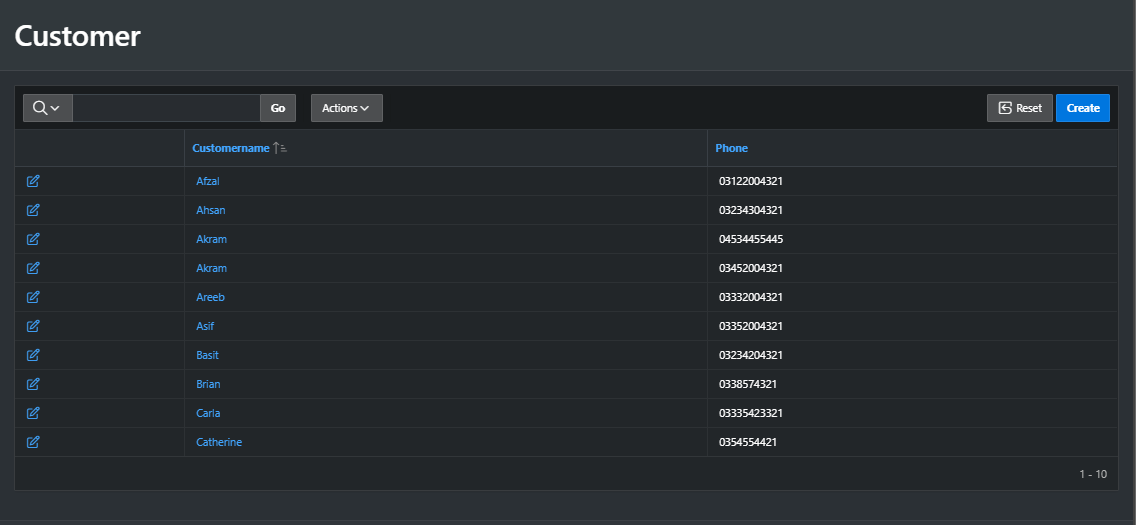
**Staff Table:**



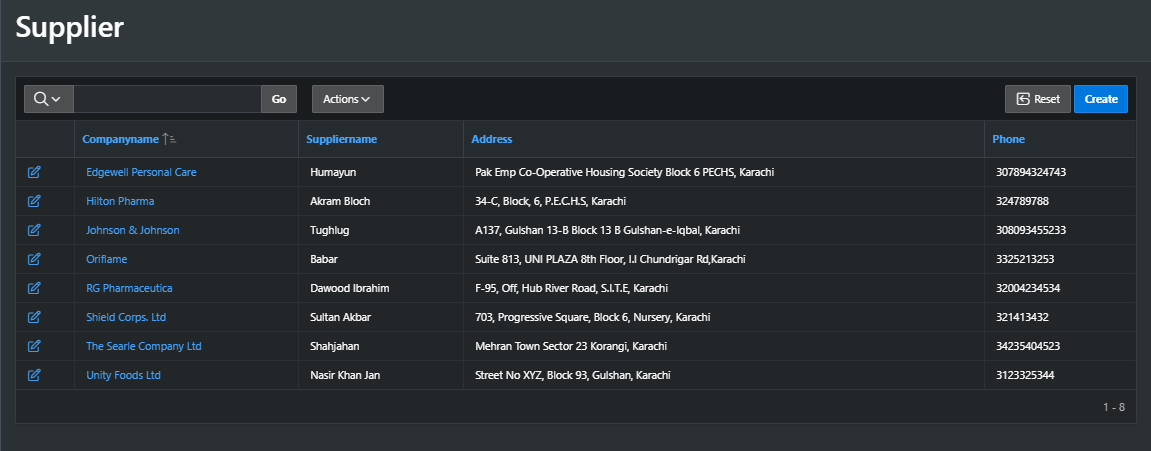
**Category Table:**



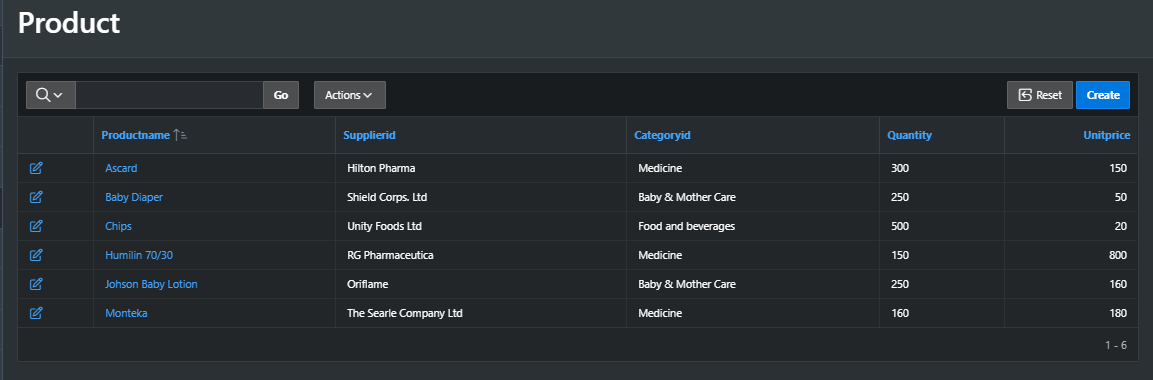
**Customer Table:**



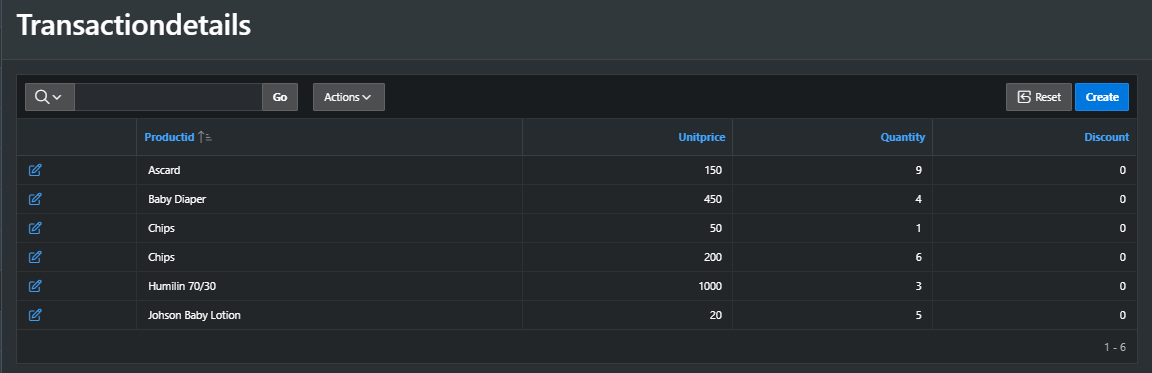
**Supplier Table:**



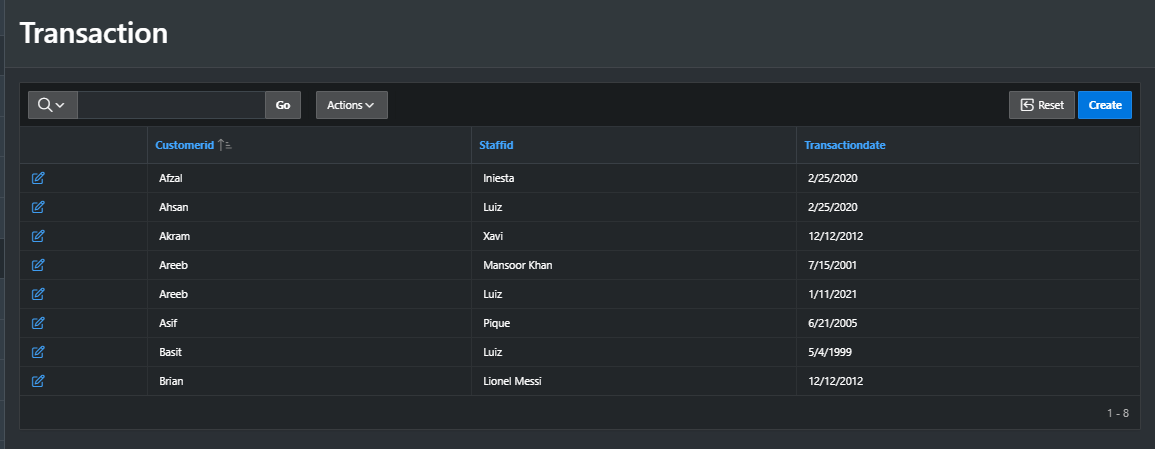
**Product Table:**



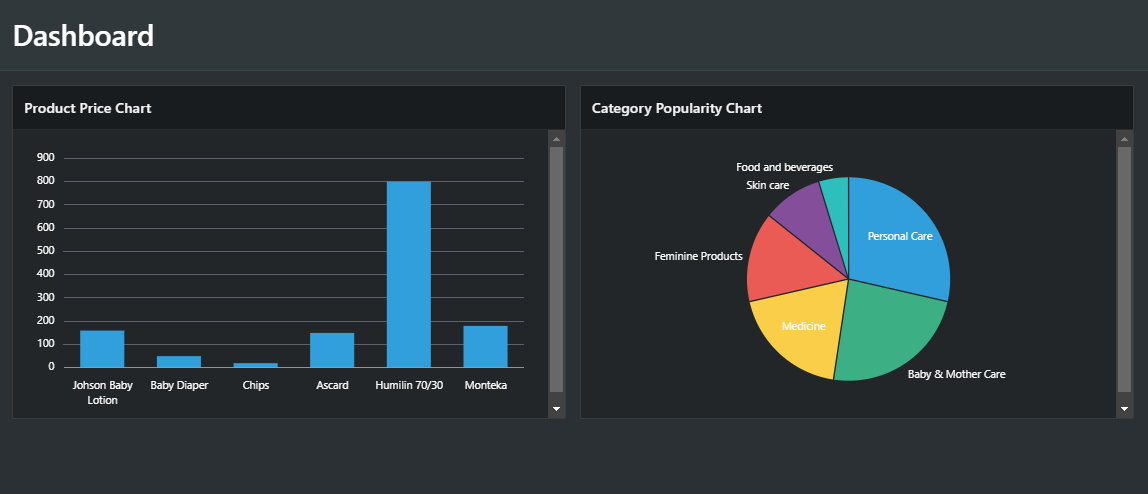
**Transaction Details Table:**



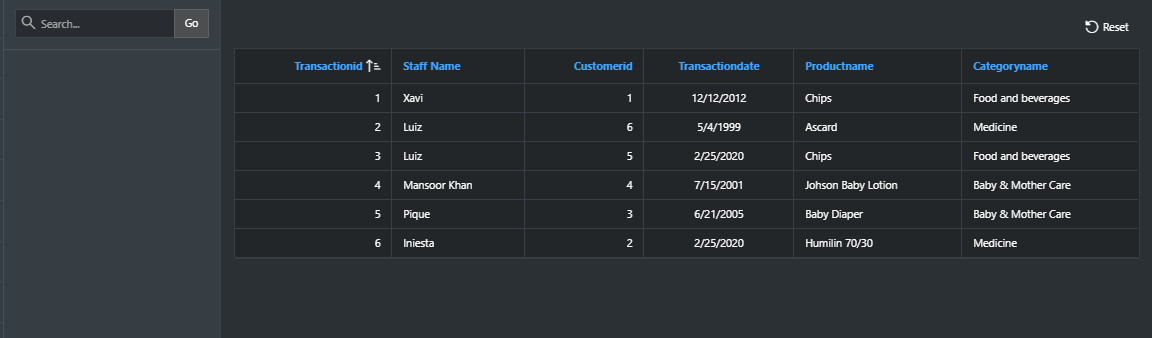
**Transaction Table:**



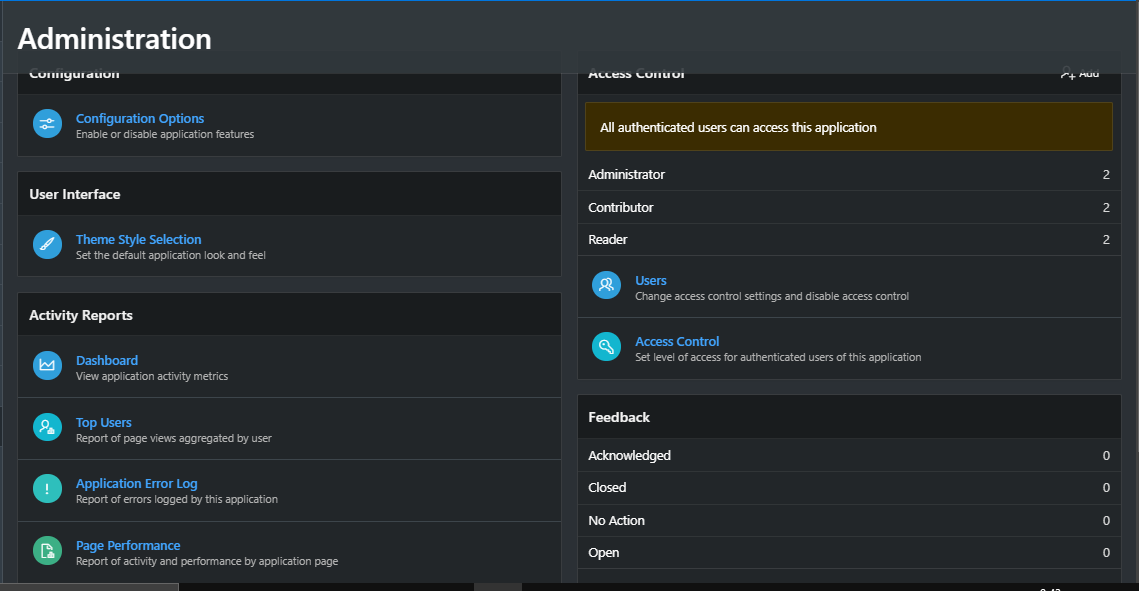
**Dashboard:**



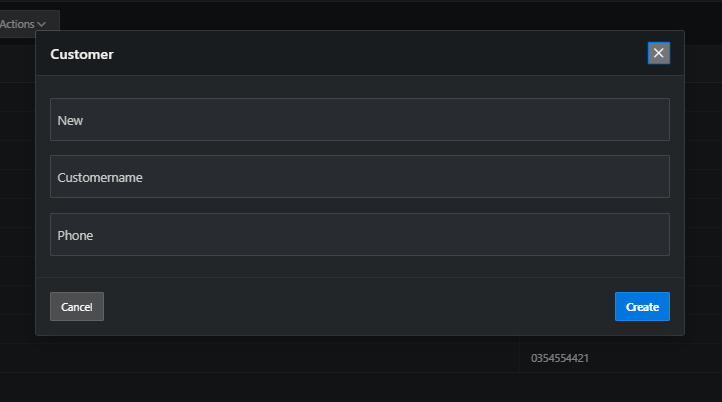
**Report Page:**



**Administration Page:**

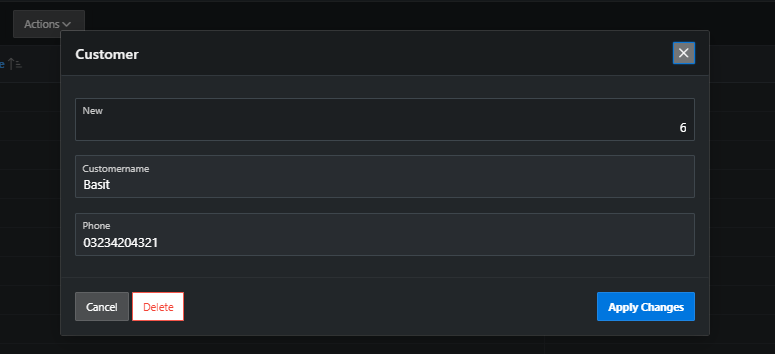


**Add Record Form:**



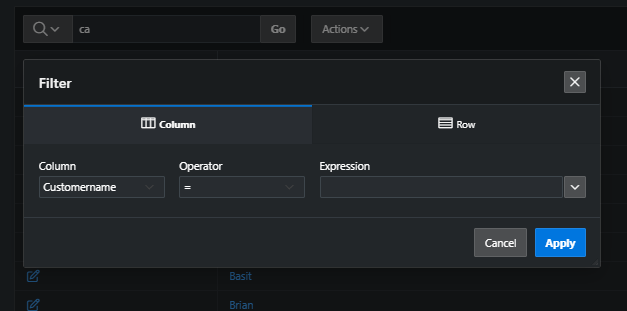
**Note: form is same for all the tables with difference of attributes.**

**Update Record Form:**



**Note: The delete record form is same, just click on that delete button.**

**Search Data Form:**



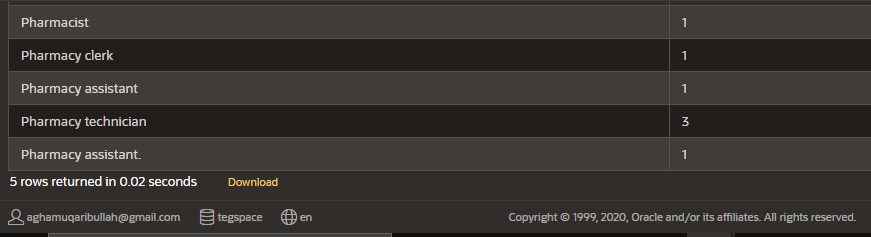
**Queries:**

1. **Create a view to display each designation and number of employees with that particular designation.**

CREATE OR REPLACE VIEW DESIGNATIONS AS SELECT JOB, COUNT(STAFFID)

AS No\_OF\_EMPLOYEES FROM STAFF GROUP BY JOB;

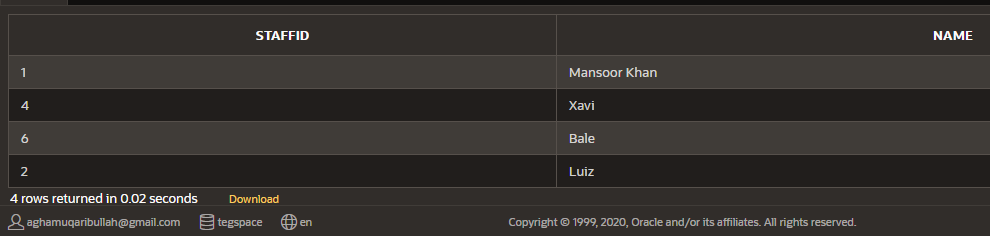
SELECT \* FROM DESIGNATIONS;



1. **To display the employee number and name for all employees who earn more than the average salary. Sort the results in descending order of salary.**

SELECT STAFFID,NAME FROM STAFF WHERE SALARY>(SELECT AVG(SALARY) FROM STAFF)

ORDER BY SALARY DESC;

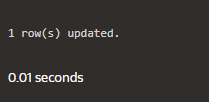


1. **Increase the salary of employee with number 001 by 10% of the salary of employee with number 002.**

UPDATE STAFF SET

SALARY=SALARY+((SELECT SALARY FROM STAFF WHERE STAFFID=002)\*10/100)

WHERE STAFFID=001;



1. **Fetch Order No, Order Date, Product Name, Category Name, Customer Name using joins.**

SELECT T.TRANSACTIONID, P.PRODUCTNAME, T.TRANSACTIONDATE, C.CUSTOMERNAME,

CAT.CATEGORYNAME

FROM TRANSACTION T INNER JOIN TRANSACTIONDETAILS TD ON

T.TRANSACTIONID=TD.TRANSACTIONID INNER JOIN PRODUCT P

ON TD.PRODUCTID=P.PRODUCTID INNER JOIN CUSTOMER C ON

T.CUSTOMERID = C.CUSTOMERID INNER JOIN CATEGORY CAT ON

P.CATEGORYID=CAT.CATEGORYID;



1. **Fetch Order No, Order Date, Product Name, Category Name, Customer Name using Sub Queries.**

SELECT T.TRANSACTIONID,T.TRANSACTIONDATE,(SELECT PRODUCTNAME FROM PRODUCT P

WHERE TD.PRODUCTID=P.PRODUCTID) AS PRODUCT\_NAME,

(SELECT CUSTOMERNAME FROM CUSTOMER C WHERE T.CUSTOMERID=C.CUSTOMERID)

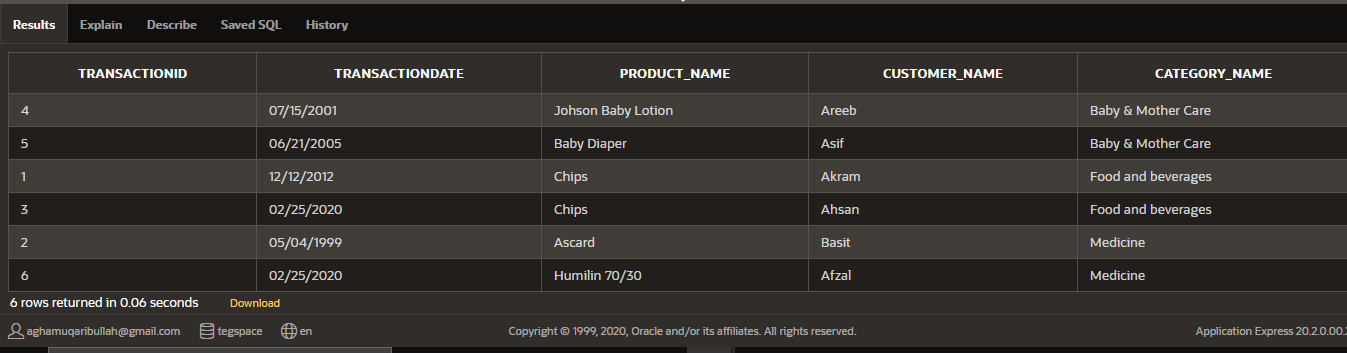
AS CUSTOMER\_NAME,

(SELECT CAT.CATEGORYNAME FROM CATEGORY CAT

WHERE CAT.CATEGORYID=P.CATEGORYID) AS CATEGORY\_NAME

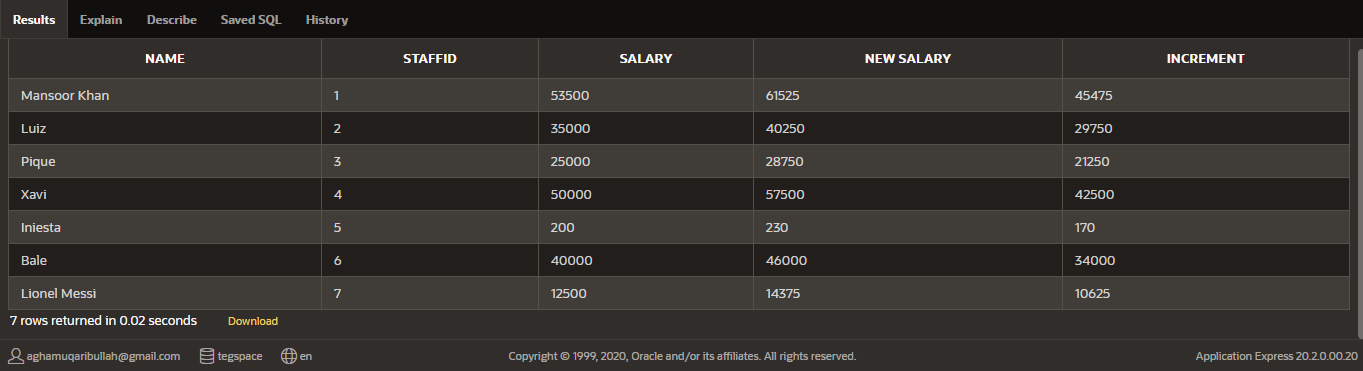
FROM TRANSACTION T, TRANSACTIONDETAILS TD, PRODUCT P WHERE

TD.TRANSACTIONID=T.TRANSACTIONID AND TD.PRODUCTID=P.PRODUCTID;



1. **Display the employee number, name, salary, salary increase by 15% expressed as a whole number (labeled as *New Salary*), the difference between old salary and new salary (labeled as *Increment*).**

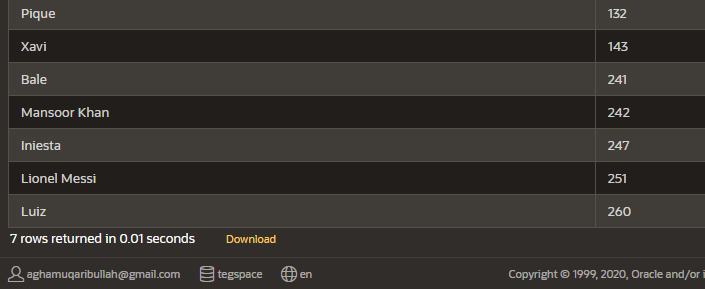
SELECT NAME, STAFFID, SALARY, SALARY+SALARY\*(15/100) AS "NEW SALARY", SALARY-SALARY\*(15/100) AS "INCREMENT" FROM STAFF



1. **Display the employee name and calculate the number of months between today and the date the employee was hired (Labeled as *Months\_Worked*). Order the results by the number of months employed and round the number of months up to the closest whole number.**

SELECT NAME,ROUND(MONTHS\_BETWEEN(SYSDATE,"HIREDATE"),0) AS "MONTHS WORKED"

FROM STAFF ORDER BY "MONTHS WORKED"



1. **Write a query that produces the following for each employee:**

***<employee name>* earns *<salary>* monthly.**

SELECT (INITCAP(NAME) || ' earns PKR ' || SALARY || ' monthly !')

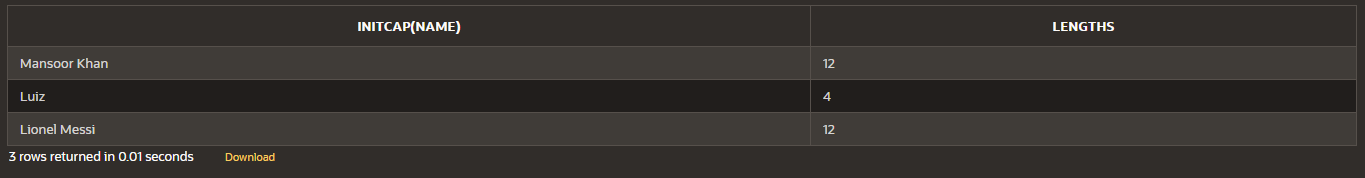
AS "STATEMENT" FROM STAFF



**9. Display the employee’s name (labeled name) with the first letter capitalized and all other letters lowercase and the length of their name (labeled length), for all employees whose name starts with L, S or M.**

SELECT INITCAP(NAME), LENGTH("NAME") AS "LENGTHS" FROM STAFF

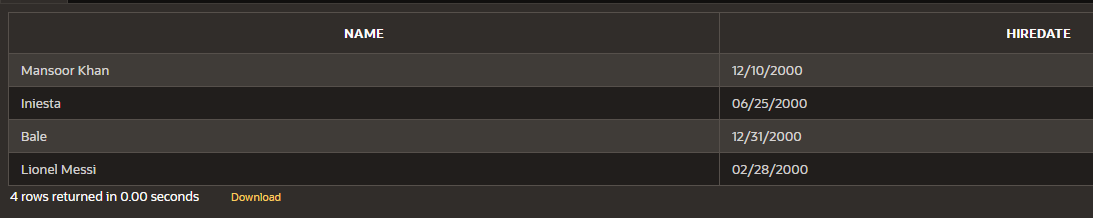
WHERE NAME LIKE('L%') OR NAME LIKE('S%') OR NAME LIKE('M%');



**10. To display the name, department number and hire date of all employees who were hired in *2000*.**

SELECT NAME, HIREDATE FROM STAFF WHERE HIREDATE BETWEEN

TO\_DATE('1-JAN-2000','DD-MM-YYYY') AND TO\_DATE('31-DEC-2000','DD-MM-YYYY');



**User Manual:**

* **Login:**

To login, enter the credentials, i.e. username and password. The username and password are:

**URL:** [Pharmacy Database Management - Sign In (oracle.com)](https://apex.oracle.com/pls/apex/tegspace/r/pharmacy-database-management1/login?session=115505939990591)

**Username:** [aghamuqaribullah@gmail.com](mailto:aghamuqaribullah@gmail.com)

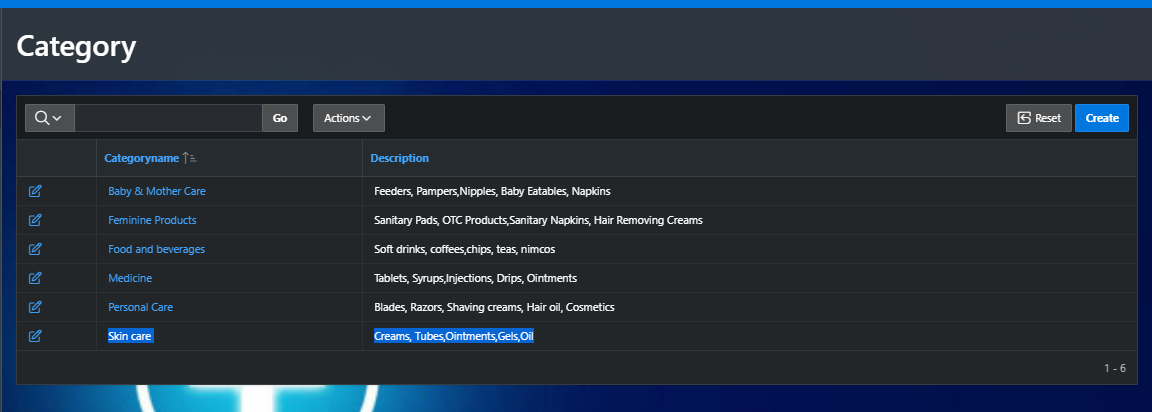
**Password:** Amuqarib@best2.

* **Navigation Bar:**

There is a navigation bar at the left of the home age. Click on the table which you want to open.

* **Add Record:**

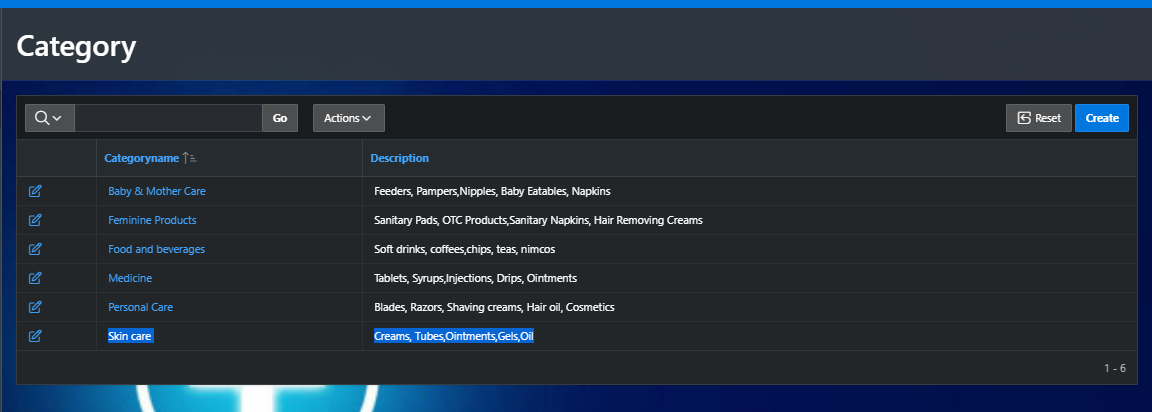
To add a record, click on Create option at top left of the table screen.



A form will open. Enter values, and click on create. The record will be added.

* **Update Record:**

To update a record, click on the pencil or edit icon at the left or start of each record.



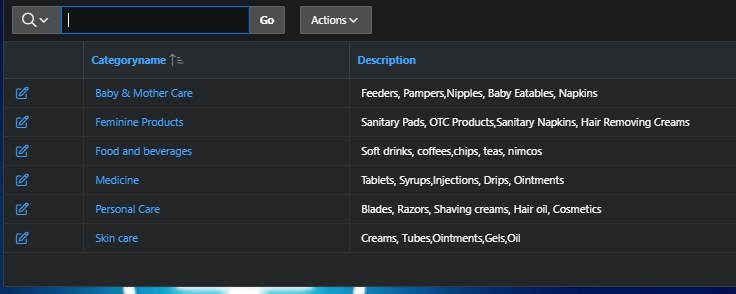
A form will open. Update the record and click on apply changes.

* **Delete Record:**

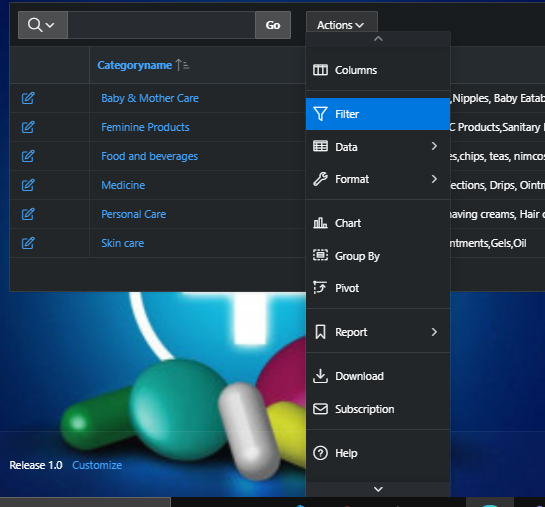
To delete a record, click the pencil or edit icon at the left or start of each record (like in update). A form will pop up. Click on delete at bottom left of the form window. The record will be deleted.

* **Search a Record:**

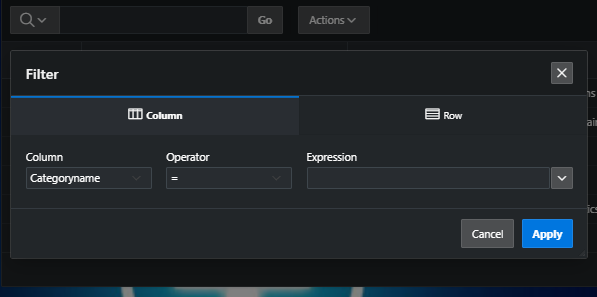
There are 2 ways to search a record. The first one is to simply write what you want to search in the search pane.



The second one is to filter record.



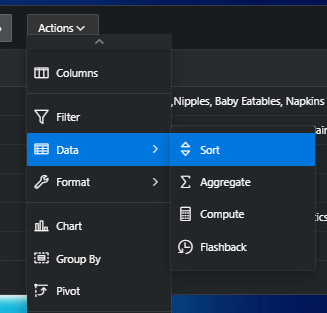
A form window will pop up.



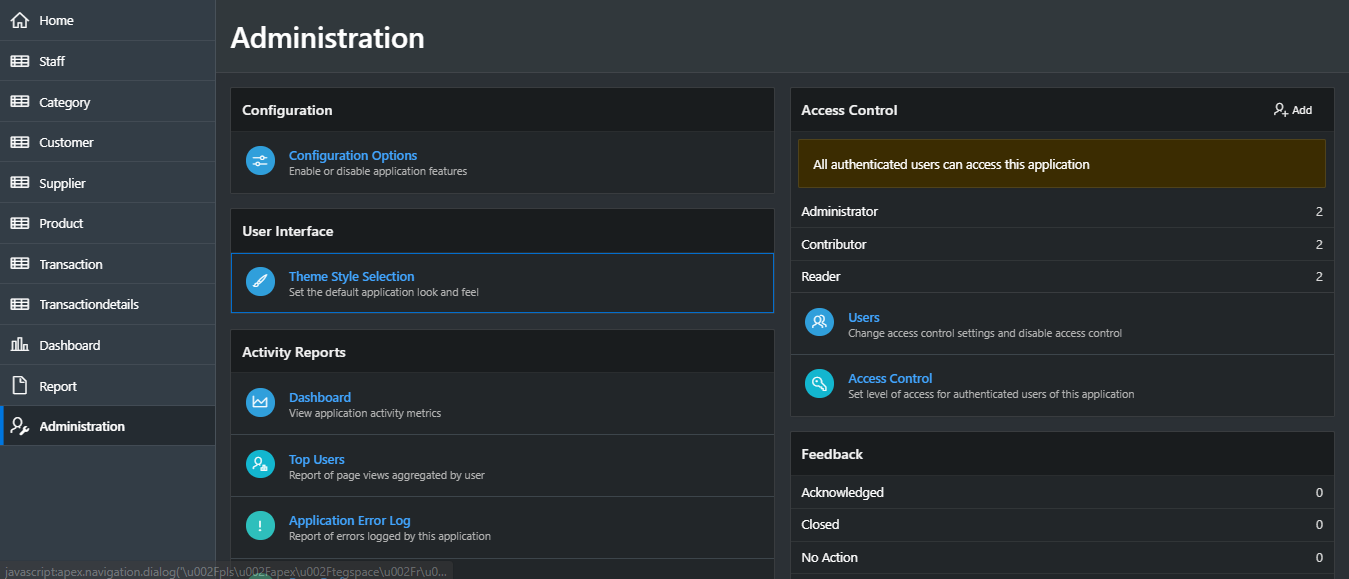
Enter the filter criteria and click apply. The records matching the criteria would be shown.

* **Perform Operations:**

To perform calculations, or sort or present table in customized way, click on actions and perform the desired operation.



* **General Settings:**



For general settings like theme selection or configuration, click on administration in navigation pane and set as per your requirement.