جـــامعة ٦ أكـــتوبر
OCTOBER 6 UNIVERSITY

خالد محمود سيد	212102512
سيف الدين حسين مصطفي	212103918
مارينا صفوت عبدالتواب	212103754
ديفيد ايليا دانيال	212103466
خالد احمد إبراهيم	212104459
خالد وحيد محمد	212104434
حازم محمد فاروق	212102309

Introduction

Our Project is a store system app designed to store data in an Oracle database using a simple layout with buttons (JButtons) and a table (JTable) to display the results and test updates in the database. We worked on this project in two different phases:

First, we used Eclipse IDE to design the app's graphical user interface (GUI) and write the necessary functions and button listeners Code to perform specific actions. We created a user-friendly layout that allows users to input product details such as the product name, price, and quantity. After adding the product information, it is stored in the Oracle database.

In the second phase, we set up an Oracle database to serve as a(Warehouse) for storing the inputted data. We established a connection between Eclipse and the Oracle database using JDBC (Java Database Connectivity) to facilitate data retrieval and updates. This connection enables us to display the stored data in the app's table (JTable) and perform search operations based on the product's ID using a (JComboBox).

Our store system app provides a convenient and efficient way for stores and markets to manage their inventory by adding product details and storing them in an Oracle database. Users can easily update and retrieve the stored information, allowing for effective inventory management and organization.

JDBC

JDBC makes it possible to do establish a connection with a data source, send queries and update statements, and process the results.

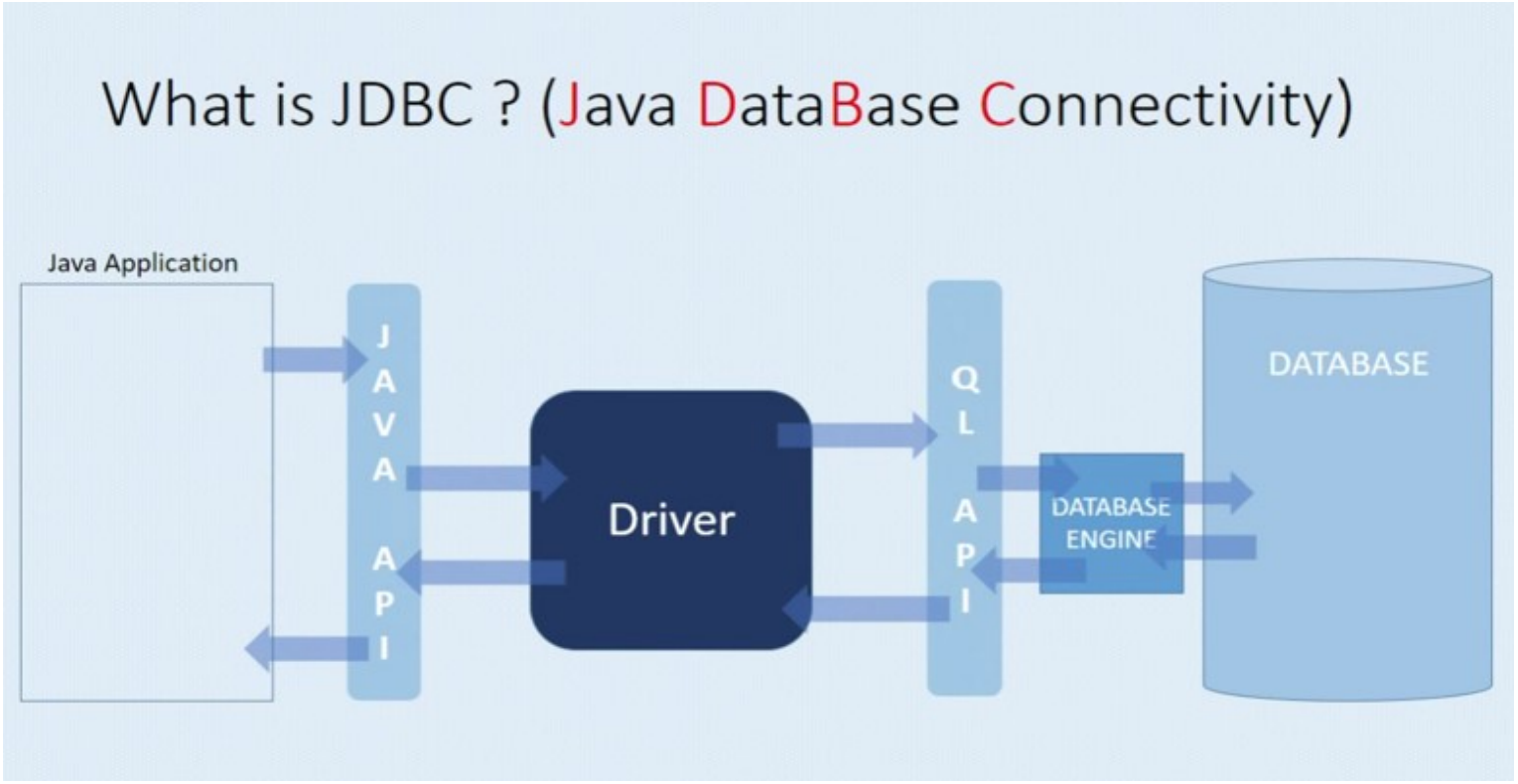
Simply, JDBC makes it possible to do the following things within a Java application:

Establish a connection with a data source.

Send queries and update statements to the data source.

Process the results.

The following figure shows the components of the JDBC model.



1- Add Listener

The mechanism:

1)checks for the existence of the product in the database by using passing the product name inserted by the user into a Boolean function called "CheckDataIsExist", if the user had already inserted the same product before this function returns true and we display a message to the user saying this element was inserted before can't be inserted again.

Database

EDIT	PRODUCT_ID	PRODUCT_NAME	PRICE	QUANTITY
	49	APPLE	1000	2
	25	Iphone	50	70
	50	Mac	5	100
	51	OPPO	28	132
	45	SAMSUNG	5000	50

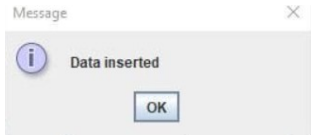
row(s) 1 - 5 of 5

Application

ProductID	Product Name	Price	Quantity
25	Iphone	50	70
45	SAMSUNG	5000	50
49	APPLE	1000	2
50	Mac	5	100
51	OPPO	28	132

Application

OnClick



2) If the function "CheckDataIsExist" returns false (means the product isn't found in the database) then the insert function takes the product name , price and quantity and add them into an SQL statement to be inserted into the database.

Database

EDIT	PRODUCT_ID	PRODUCT_NAME	PRICE	QUANTITY
	61	POCO	5000	20
	49	APPLE	1000	2
	25	Iphone	50	70
	50	Mac	5	100
	51	OPPO	28	132
	45	SAMSUNG	5000	50

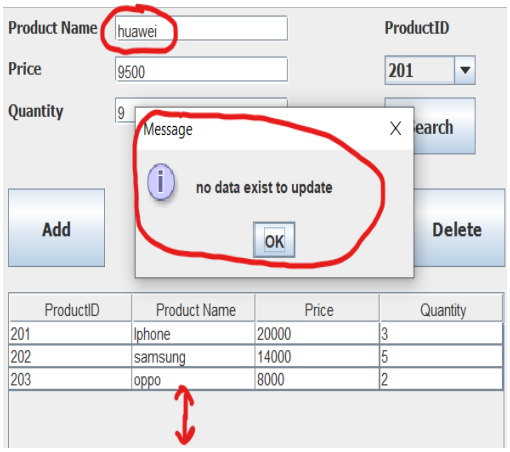
row(s) 1 - 6 of 6

Application

ProductID	Product Name	Price	Quantity
25	Iphone	50	70
45	SAMSUNG	5000	50
49	APPLE	1000	2
50	Mac	5	100
51	OPPO	28	132
61	POCO	5000	20

2- Update Listener

1- There must be a product name in the database when we update.

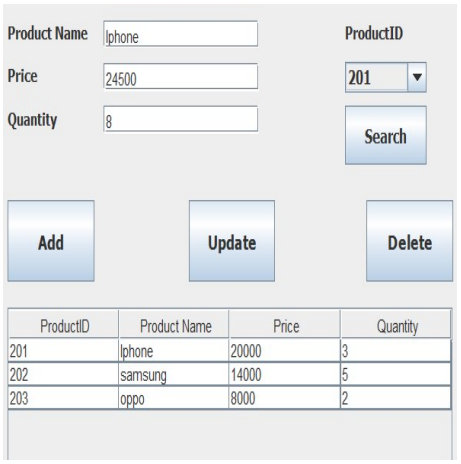


2- if the product name exists in the database:

Database

PRODUCT_ID	PRODUCT_NAME	PRICE	QUANTITY
201	Iphone	24500	8
202	samsung	14000	5
203	oppo	8000	2

Application

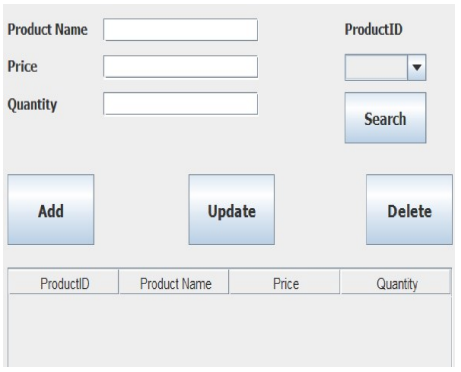


1- Update in the database

Database

PRODUCT_ID	PRODUCT_NAME	PRICE	QUANTITY
201	Iphone	24500	8
202	samsung	14000	5
203	oppo	8000	2

Application

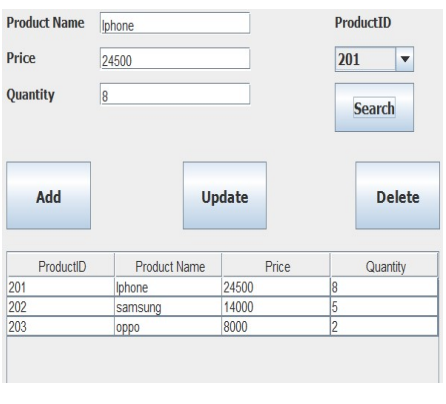


2- Clear all data in the application

Database

PRODUCT_ID	PRODUCT_NAME	PRICE	QUANTITY
201	Iphone	24500	8
202	samsung	14000	5
203	oppo	8000	2

Application



3- Then copy the updated data from the database to the application

3- Delete Listener

Usage(Idea): delete row from the database.

How it work:

1-Check that the user select a row.

2-Perform a Delete function.

What we did in the Delete function:

1- We Take a variable called id as an input that come from the row selected by the user.

2- We make a program connect with the Database.

3- We make a query to perform a delete operation.

4- We execute query in the Database and delete the selected row.

5- We save changes in result set then show result in the table.

4- Search Listener

The search function receives a variable called id, this variable that the user chooses from the Combo Box button. The search is done by id, because each product name has its own id that is not repeated so that I can access the data stored on the data base by making a connection between the application and the database, then a query is made, its function is to perform a search For the id inside the data base, the table that comes out is called result set. A loop is made and I take the data that appeared to me in the table and put it in the text field for each cuolm, which is the name of the product, price and quantity