----( PROJECT REPORT )----

|  |  |
| --- | --- |
| Subject | IICT |
| Teacher Name | Mohsin Khan |
| Student Names | Abdullah Taufiq |
| Section | BSCS (1B) |
| SAP I’d | 29121 |

**Gigabyte Store Report**

**Introduction:**

In the modern world, technology plays a vital role in every field of life. Many business organizations trying to enhance the customer experience or trying to utilize technology for the management of the organization.

Store Management System is a system that can be used to manage small businesses. Some of the functions of this system are Customer Form, Navigation Form, Vendor Form, Product Form, Customer List, Product List, Vendor List, Invoice list and 7Day Report functionalities are included.

In this way you can say Store Management System can provide you all the information of the business without wasting anytime. Whatever the information you want about your customer, vendor or about your product you just need to click on the selected field or category and it will provide you the desired information.

For this purpose, we use database technology to make it fully functional and the implementation is done using Microsoft Access.

**Scope:**

This application can be used in different businesses:

Any Store can use this to manage the store sales and other stuff.

**Benefits:**

Some of the benefits of this system are:

Providing High security

Easy business solution

Sales History

**Required Application to use this system:**

To run this system or to make changes to this system according to your needs you need to have Microsoft Access installed on your computer.

**Development Platform:**

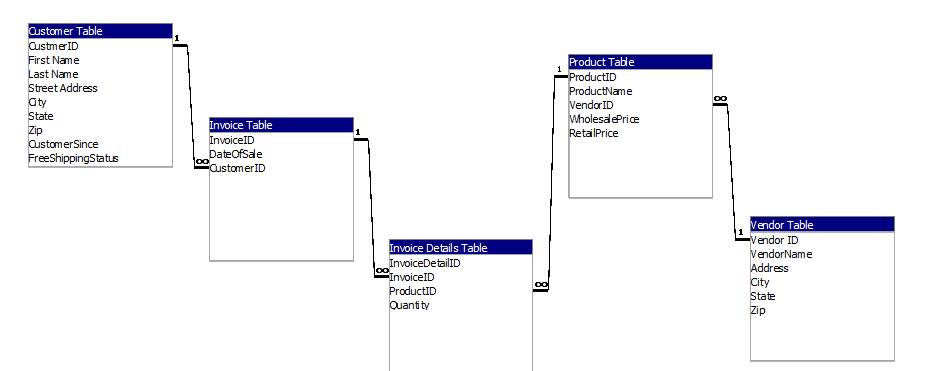
We use Microsoft Access to implement this system.

Microsoft Access is a database management system that combine the Relational Microsoft Jet Database Engine with a graphical user interface and software development tools. It stores data in its own format based on the access Jet database Engine. It can also import or link directly to data stored in other applications and databases. Access is supported by Visual Basic for Applications (VBA), an object-based programming language that can reference a variety of objects including the legacy DAO (Data Access Objects), ActiveX Data Objects, and many other ActiveX components. Visual objects used in forms and reports expose their methods and properties in the VBA programming environment, and VBA code modules may declare and call Windows Operating system operations.

**Features of MS Access:**

Users can create tables, queries, forms and reports, and connect them together with macros. Advanced users can use VBA to write rich solutions with advanced data manipulation and user control. Access also has report creation features that can work with any data source that Access can access.

**Relational Diagram:**

****

**Tables, Forms and Their Functions:**

Some of the tables we created in this database are:

**1: Customer Table**

**2: Invoice Data Table**

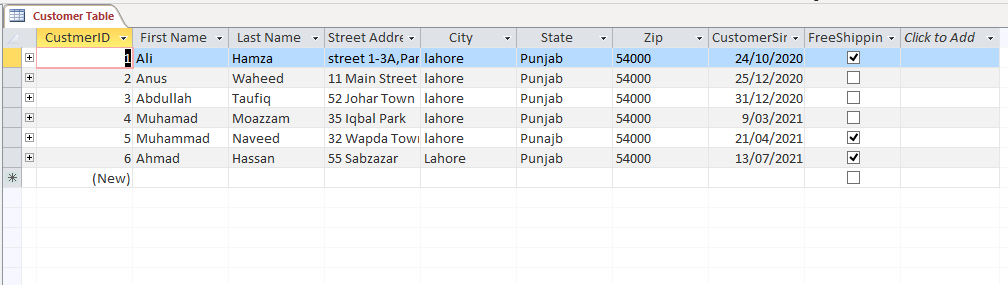
**3: Invoice Table**

**4: Product Table**

**5: Vendor Table**

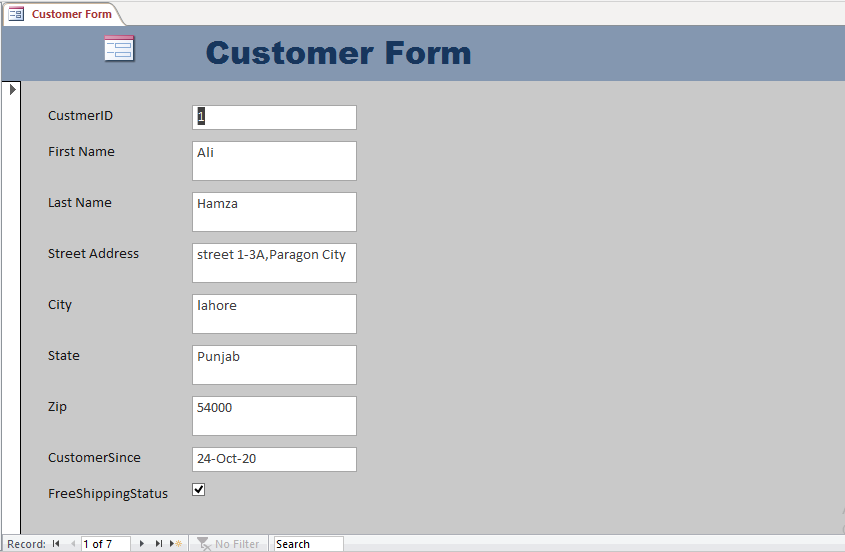
**Customer Table:**

First of all, we created a table and named the table as customer table. After that we filled the required fields. Basically, we create this table to store customers information. This table contains information like customer name, customer address and customer ID etc. The figure below shows the customer table.



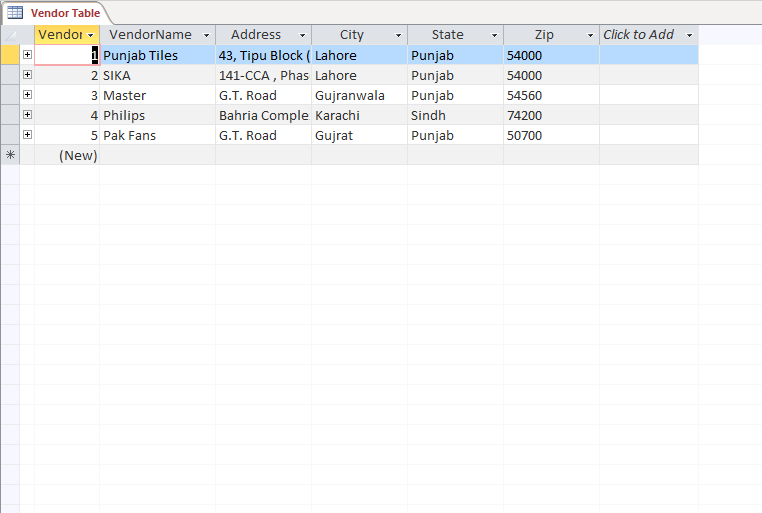
**Customer Form:**

After creating customer table, we create a customer form. This customer form is created to enter customer details. This form is linked with Customer table. So, whatever we enter in customer form automatically saves in customer table. The main purpose to create this form is that one can easily enter or modify data. A graphical representation of customer form is shown below:



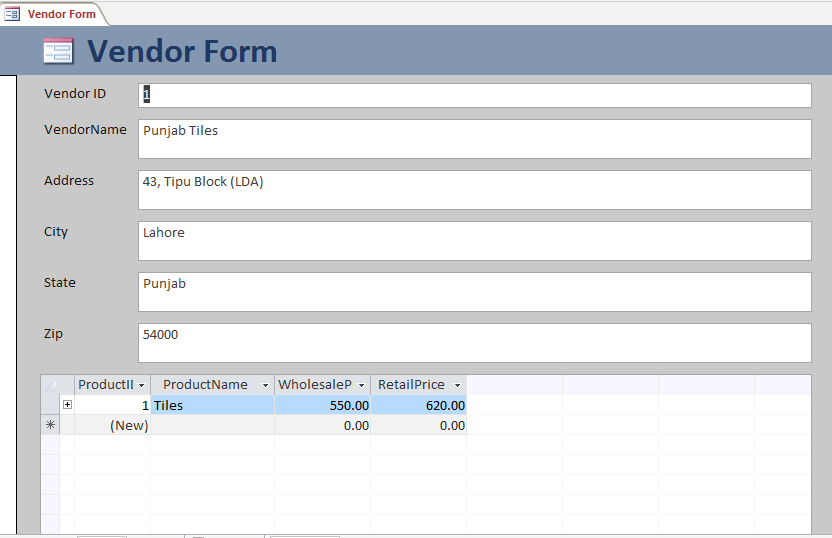
**Vendor Table:**

Now after creating customer table and customer form we need to store and display information about vendors. For that purpose, we created another table and named as vendor table. We named the required fields for storing vendor information like (vendor Id, vendor name and address) etc. Vendor table looks like figure below:



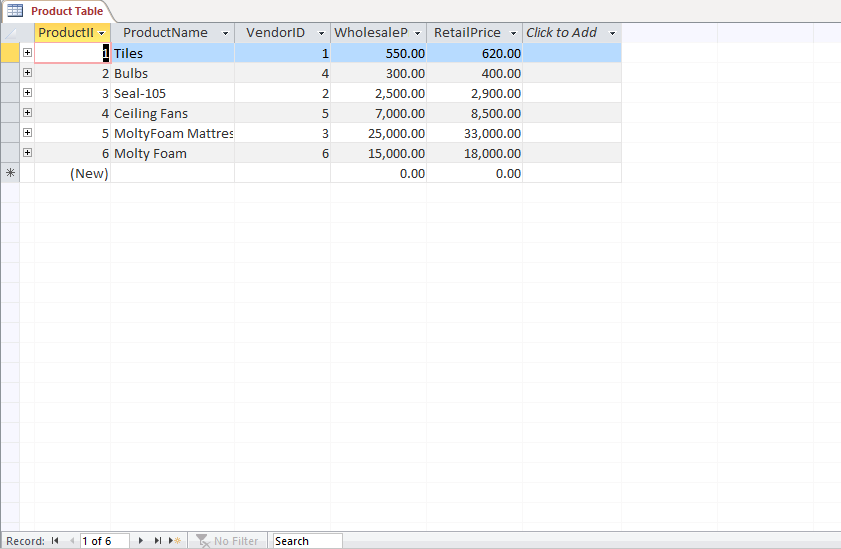
**Vendor Form:**

Now we create another form for entering and storing vendor data by selecting vendor table and named as vendor form. After creating a vendor form, we create another form (vendor sub form) by selecting vendor table. By doing this we can add or modify vendor and product details. Figure below shows vender form:



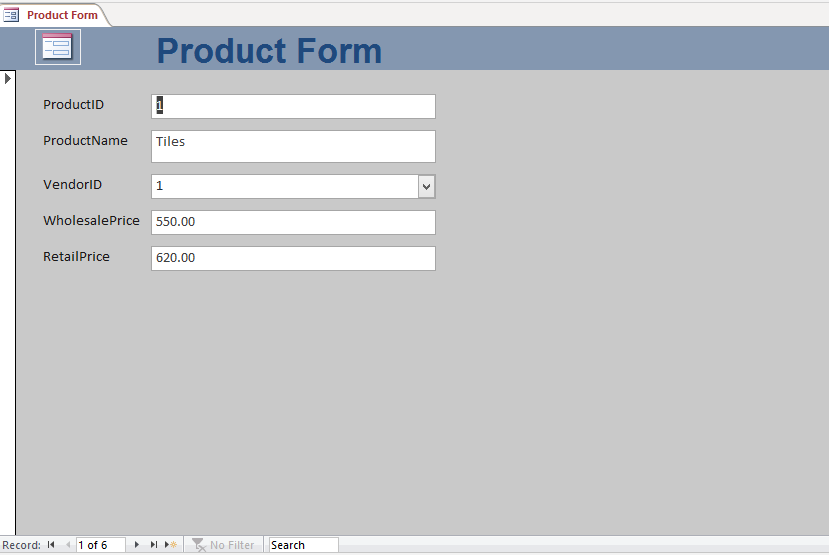
**Product Table:**

Now we are going to display product data and for that we create a table and named as product table. This table contains data like (product ID, product name, vendor ID, and product prices) etc. The vendor ID in product table is linked with vendor ID in vendor table with the help of lookup wizard. The figure below shows the product table:



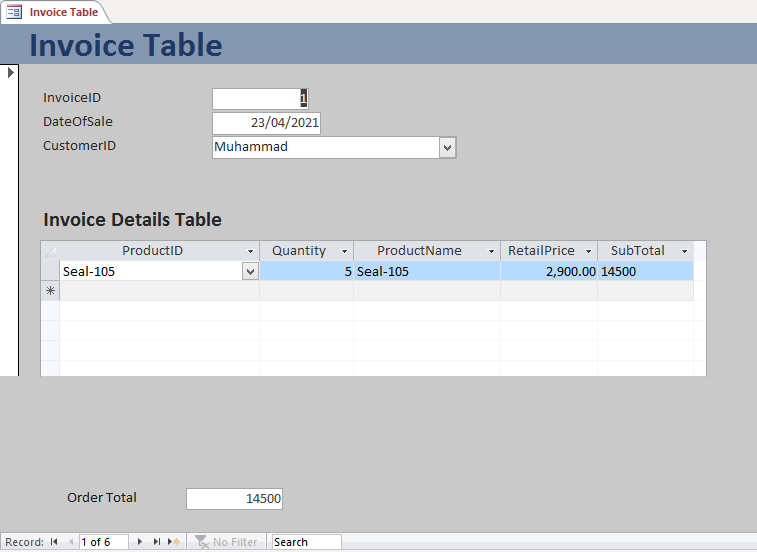
**Product Form:**

As we created a product table now, we need to have a product form to enter and save data to product table. For this purpose, we created a form named as product form and filled the required fields to enter data. The product form looks like the figure below:



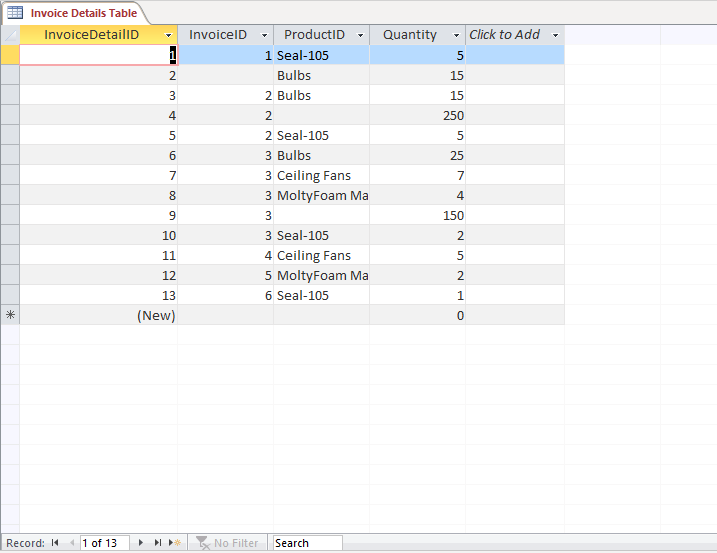
**Invoice Table:**

Now we will create an invoice table to provide customer the bill for his transactions. For that purpose, we created a table named as Invoice table and filled the required fields. Then we create a relationship of customer ID in invoice table with selected fields of customer table. Graphical representation of Invoice table:



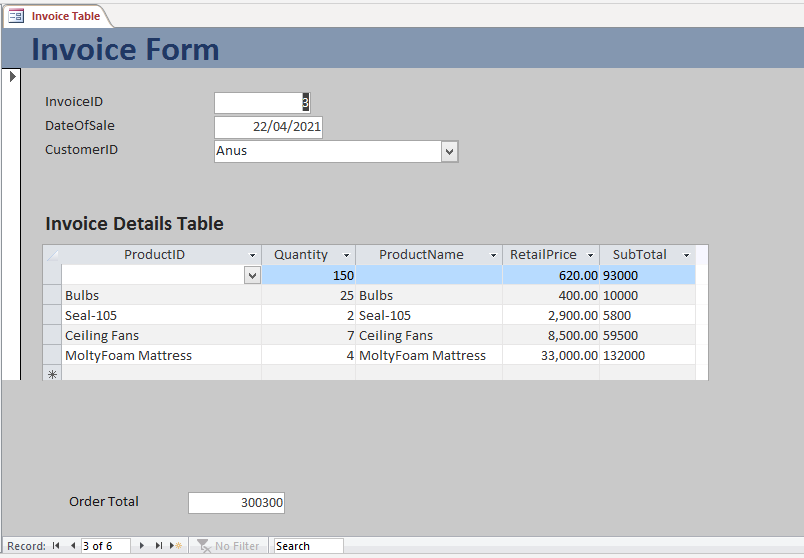
**Invoice Details Table:**

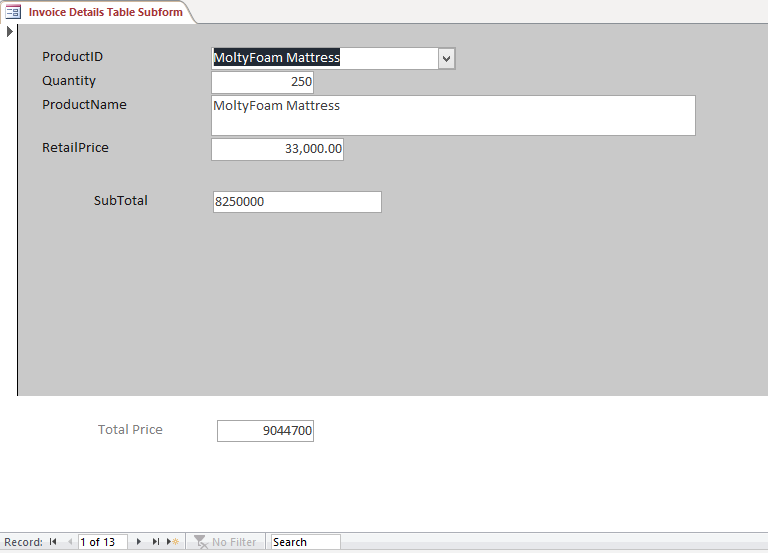
After creating Invoice table, we created another table named as Invoice details table. This table contains the details of invoice table. Invoice ID of this table is linked with Invoice table and product ID is linked with product table. Figure below shows the Invoice details table:



**Invoice Form And Invoice Details Subform:**

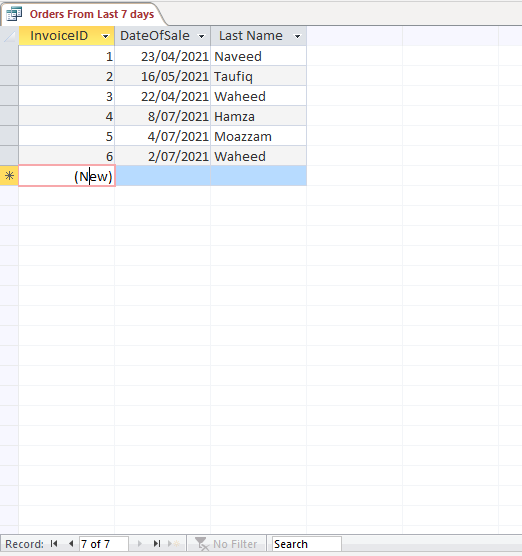
After creating Invoice details table, we are going to create invoice form and invoice details subform with selected fields from invoice table, invoice details table and product table with the help of form wizard. Then we create a sum on the subform and copy the field value to the main form. Figure below shows invoice form and invoice details subform:

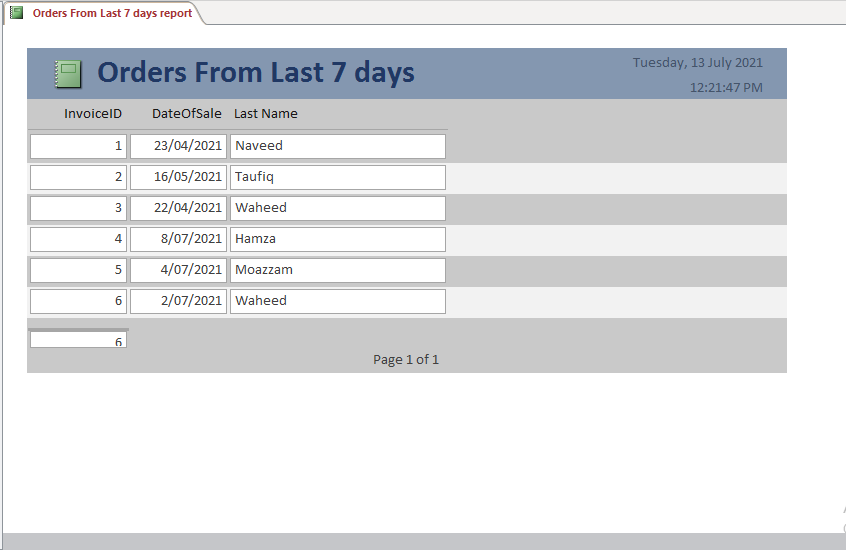


****

**Query and Report:**

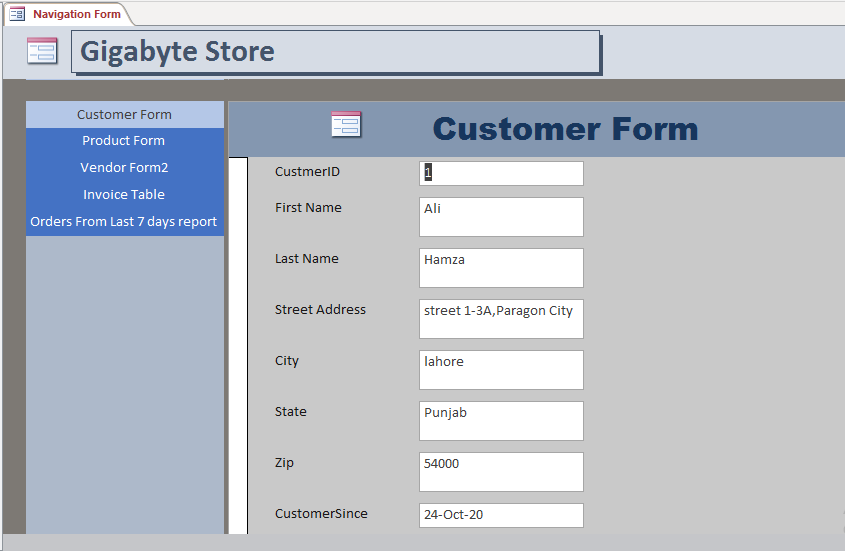
After invoice form, we create query. The purpose to create query is to check orders from last 7 days. And for that we create query with invoice table and customer table and named the query as Order from last 7 days. After creating query, we create Report by selecting query (order from last 7 days). Figure below shows the query and report graphical representation:





**Navigation Form:**

At the end we create a navigation form. The purpose to make navigation form is that one can easily access the required form and enter the required data or modify the data. And for that we added the required forms in the navigation form. Graphical representation of navigation form is shown below:



**The End**