

Final Project Report

CSC572 Advanced Database Concepts

Camron Khan

Time Log:

1. 10/3/17, 3 hours, Started and completed the Entities, Attributes, Functional Dependencies, Primary Keys, and Special Restrictions sections
2. 10/5/17, 2 hours, Started and completed the DBDL 3NF and Data Structure Diagram sections
3. 10/7/17, 2 hours, Made changes to DBDL based on user feedback
4. 12/2/17, 5 hours, Create queries, forms, reports, and switchboard for Part 2 of the project
5. 12/3/17, 5 hours, Compiled developed ID rational and security plan in additional to compiling assets into the final report

Entities:

1. Order
2. OrderLine
3. Customer
4. SalesRep
5. ItemClass
6. Part
7. Warehouse
8. Inventory
9. Manufacturer

Attributes:

1. OrderID
2. OrderDate
3. OrdererType
4. OrderLineQuantity
5. OrderLineQuotedPrice
6. CustomerID
7. CustomerName
8. CustomerStreet
9. CustomerCity
10. CustomerState
11. CustomerZip
12. CustomerBalance
13. CustomerCreditLimit
14. SalesRepID
15. SalesRepFirstName
16. SalesRepLastName
17. SalesRepStreet
18. SalesRepCity
19. SalesRepState

20. SalesRepZip
21. SalesRepCommissionRate
22. ItemClassID
23. ItemClassDescription
24. PartID
25. PartDescription
26. PartPrice
27. PartCost
28. WarehouseID
29. WarehouseDescription
30. InventoryQuantity
31. ManufacturerID
32. ManufacturerName

Functional Dependencies:

1. OrderID → OrderDate, OrdererType, CustomerID
2. (OrderID, PartID) → OrderLineQuantity, OrderLineQuotedPrice
3. CustomerID → CustomerName, CustomerStreet, CustomerCity, CustomerState, CustomerZip, CustomerBalance, CustomerCreditLimit, SalesRepID
4. SalesRepID → SalesRepFirstName, SalesRepLastName, SalesRepStreet, SalesRepCity, SalesRepState, SalesRepZip, SalesRepCommissionTotal, SalesRepCommissionRate
5. ItemClassID → ItemClassDescription
6. PartID → PartDescription, PartPrice, PartCost, ManufacturerID
7. WarehouseID → WarehouseDescription
8. (WarehouseID, PartID) → InventoryQuantity
9. ManufacturerID → ManufacturerName

Primary Keys:

1. OrderID
2. (OrderID, PartID)
3. CustomerID
4. SalesRepID
5. ItemClassID
6. PartID
7. WarehouseID
8. (WarehouseID, PartID)
9. ManufacturerID

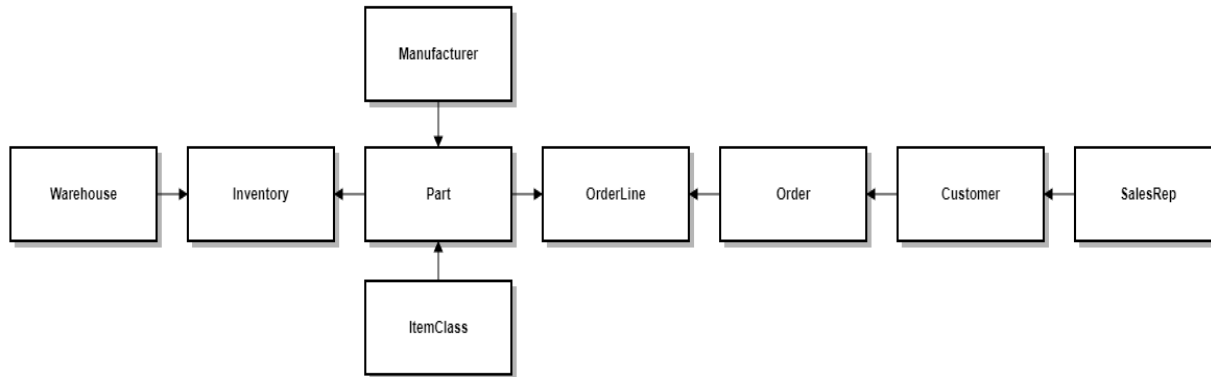
Special Restrictions:

1. Legal values for the OrderedByType column in the Order table are 'customer' and 'sales rep'
2. Legal values for the ItemClassDescription column in the ItemClass table are 'hand tools', 'power tools', 'safety equipment', and 'miscellaneous equipment'
3. The total number of units on hand for a given part is calculated by summing the quantity of that part at each warehouse
4. The total on-hand value for a given part is calculated by multiplying that part's total number of units on hand by that part's cost
5. The SalesRepCommissionRate must be between 0 and 100 inclusive

DBDL 3NF:

1. Order(OrderID, OrderDate, OrdererType, CustomerID)
FK CustomerID → Customer
2. OrderLine(OrderID, PartID, OrderLineQuantity, OrderLineQuotedPrice)
FK OrderID → Order
FK PartID → Part
3. Customer(CustomerID, CustomerName, CustomerStreet, CustomerCity, CustomerState, CustomerZip, CustomerBalance, CustomerCreditLimit, SalesRepID)
FK SalesRepID → SalesRep
4. SalesRep(SalesRepID, SalesRepFirstName, SalesRepLastName, SalesRepStreet, SalesRepCity, SalesRepState, SalesRepZip, SalesRepCommissionTotal, SalesRepCommissionRate)
5. ItemClass(ItemClassID, ItemClassDescription)
6. Part(PartID, PartDescription, PartPrice, PartCost, ItemClassID, ManufacturerID)
FK ItemClassID → ItemClass
FK ManufacturerID → Manufacturer
7. Warehouse(WarehouseID, WarehouseDescription)
8. Inventory(WarehouseID, PartID, InventoryQuantity)
FK WarehouseID → Warehouse
FK PartID → Part
9. Manufacturer(ManufacturerID, ManufacturerName)

Data Structure Diagram



Rationales for Assigning IDs:

I used incrementing alphanumeric (i.e., A-Z, 0-9) case insensitive character arrays of varying lengths for ManufacturerID, ItemClassID, PartID, WarehouseID, SalesRepID, CustomerID, and OrderID as it is an easy solution to comprehend and implement. In practice, I would prefer to use a case sensitive alphanumeric hash to ensure uniqueness and performance.

Security Plan

I would take three steps to implement a security plan. The first step I would take to secure my MS Access database would be to encrypt it with a password. Encrypting the database makes the data unreadable by any programs, tools, or users without the password.

The second step I would take would be to migrate the data to MS SQL Server and use MS Access as a client (if MS Access was still desired to be used at all). This would allow users to continue building queries, forms, and reports in MS Access using MS SQL Server as the backend database server. Doing so would allow the implementation of role-based permissions in which a role, or group, is created and assigned permissions. Then, users can be assigned roles to gain access to only the data they need – i.e., principle of least privilege.

Given the above move to a MS SQL Server backend, I would implement stored procedures with strongly typed parameters to protect against SQL injection attacks as my third step. Doing so would allow SQL Server to create an execution plan before a query is executed. As a result, there is a very limited set of values that would be considered acceptable values as opposed to appending partial strings, which would allow malicious users to inject SQL scripts.

SQL Implementation

```
/* TABLES *****/
CREATE TABLE Manufacturers (
    ManufacturerID CHAR(3) PRIMARY KEY,
    ManufacturerName VARCHAR(25) NOT NULL
);

CREATE TABLE ItemClasses (
    ItemClassID CHAR(2) PRIMARY KEY,
    ItemClassDescription VARCHAR(25) NOT NULL,
    CONSTRAINT CHK_ItemClasses_ItemClassDescription CHECK (ItemClassDescription IN ('hand tools', 'power tools', 'safety equipment', 'miscellaneous'))
);

CREATE TABLE Parts (
    PartID CHAR(8) PRIMARY KEY,
    PartDescription VARCHAR(15) NOT NULL,
    PartPrice DECIMAL(6,2) NOT NULL DEFAULT 0,
    PartCost DECIMAL(6,2) NOT NULL DEFAULT 0,
    ItemClassID CHAR(2) NOT NULL,
    ManufacturerID CHAR(3) NOT NULL,
    CONSTRAINT FK_Parts_ItemClasses FOREIGN KEY (ItemClassID) REFERENCES ItemClasses(ItemClassID),
    CONSTRAINT FK_Parts_Manufacturers FOREIGN KEY (ManufacturerID) REFERENCES Manufacturers(ManufacturerID)
);

CREATE TABLE Warehouses (
    WarehouseID CHAR(3) PRIMARY KEY,
    WarehouseDescription VARCHAR(50) NOT NULL
);

CREATE TABLE Inventories (
    WarehouseID CHAR(3) NOT NULL,
    PartID CHAR(8) NOT NULL,
    InventoryQuantity DECIMAL(4) NOT NULL DEFAULT 0,
    CONSTRAINT PK_Inventories PRIMARY KEY (WarehouseID,PartID),
    CONSTRAINT FK_Inventories_Warehouses FOREIGN KEY (WarehouseID) REFERENCES Warehouses(WarehouseID),
```

```

CONSTRAINT FK_Inventories_Parts FOREIGN KEY (PartID) REFERENCES Parts(PartID)
);

CREATE TABLE SalesReps (
    SalesRepID CHAR(2) PRIMARY KEY,
    SalesRepFirstName VARCHAR(15) NOT NULL,
    SalesRepLastName VARCHAR(15) NOT NULL,
    SalesRepStreet VARCHAR(100) NOT NULL,
    SalesRepCity VARCHAR(30) NOT NULL,
    SalesRepState CHAR(2) NOT NULL,
    SalesRepZip CHAR(5) NOT NULL,
    SalesRepCommissionTotal DECIMAL(7,2) NOT NULL DEFAULT 0,
    SalesRepCommissionRate INTEGER NOT NULL DEFAULT 0,
    CONSTRAINT CHK_SalesReps_SalesRepCommissionRate CHECK (SalesRepCommissionRate >= 0 AND SalesRepCommissionRate <= 100)
);

CREATE TABLE Customers (
    CustomerID CHAR(4) PRIMARY KEY,
    CustomerName VARCHAR(15) NOT NULL,
    CustomerStreet VARCHAR(100) NOT NULL,
    CustomerCity VARCHAR(30) NOT NULL,
    CustomerState CHAR(2) NOT NULL,
    CustomerZip CHAR(5) NOT NULL,
    CustomerBalance DECIMAL(6,2) NOT NULL DEFAULT 0,
    CustomerCreditLimit DECIMAL(4) NOT NULL DEFAULT 0,
    SalesRepID CHAR(2) NOT NULL,
    CONSTRAINT FK_Customers_SalesReps FOREIGN KEY (SalesRepID) REFERENCES SalesReps(SalesRepID)
);

CREATE TABLE Orders (
    OrderID CHAR(5) PRIMARY KEY,
    OrderDate DATE NOT NULL DEFAULT DATE(),
    OrdererType VARCHAR(10) NOT NULL,
    CustomerID CHAR(4) NOT NULL,
    CONSTRAINT FK_Orders_Customers FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID),
    CONSTRAINT CHK_Orders_OrdererType CHECK (OrdererType IN ('customer', 'sales rep'))
);

CREATE TABLE OrderLines (
    OrderID CHAR(5) NOT NULL,
    PartID CHAR(8) NOT NULL,
    OrderLineQuantity DECIMAL(4) NOT NULL DEFAULT 0,
    OrderLineQuotedPrice DECIMAL(6,2) NOT NULL DEFAULT 0,
    CONSTRAINT PK_OrderLines PRIMARY KEY (OrderID,PartID),
    CONSTRAINT FK_OrderLines_Orders FOREIGN KEY (OrderID) REFERENCES Orders(OrderID),
    CONSTRAINT FK_OrderLines_Parts FOREIGN KEY (PartID) REFERENCES Parts(PartID)
);

```

```

);

/* INSERT DATA *****/

INSERT INTO Manufacturers (ManufacturerID, ManufacturerName) VALUES ('001', 'Alabama Mfg. ');
INSERT INTO Manufacturers (ManufacturerID, ManufacturerName) VALUES ('002', 'Alaska Mfg. ');
INSERT INTO Manufacturers (ManufacturerID, ManufacturerName) VALUES ('003', 'Arizona Mfg. ');
INSERT INTO Manufacturers (ManufacturerID, ManufacturerName) VALUES ('004', 'Arkansas Mfg. ');
INSERT INTO Manufacturers (ManufacturerID, ManufacturerName) VALUES ('005', 'California Mfg. ');
INSERT INTO Manufacturers (ManufacturerID, ManufacturerName) VALUES ('006', 'Colorado Mfg. ');
INSERT INTO Manufacturers (ManufacturerID, ManufacturerName) VALUES ('007', 'Connecticut Mfg. ');
INSERT INTO Manufacturers (ManufacturerID, ManufacturerName) VALUES ('008', 'Delaware Mfg. ');

INSERT INTO ItemClasses (ItemClassID, ItemClassDescription) VALUES ('01', 'hand tools');
INSERT INTO ItemClasses (ItemClassID, ItemClassDescription) VALUES ('02', 'power tools');
INSERT INTO ItemClasses (ItemClassID, ItemClassDescription) VALUES ('03', 'safety equipment');
INSERT INTO ItemClasses (ItemClassID, ItemClassDescription) VALUES ('04', 'miscellaneous');

INSERT INTO Parts (PartID, PartDescription, PartPrice, PartCost, ItemClassID, ManufacturerID) VALUES ('0000001', 'Angle Drill', 9
9.99, 26.33, '02', '001');
INSERT INTO Parts (PartID, PartDescription, PartPrice, PartCost, ItemClassID, ManufacturerID) VALUES ('0000002', 'Circular Saw',
84.49, 18.84, '02', '002');
INSERT INTO Parts (PartID, PartDescription, PartPrice, PartCost, ItemClassID, ManufacturerID) VALUES ('0000003', 'Claw Hammer', 2
0.49, 4.23, '01', '003');
INSERT INTO Parts (PartID, PartDescription, PartPrice, PartCost, ItemClassID, ManufacturerID) VALUES ('0000004', 'Brace Drill Aug
er', 15.99, 4.40, '01', '004');
INSERT INTO Parts (PartID, PartDescription, PartPrice, PartCost, ItemClassID, ManufacturerID) VALUES ('0000005', 'Spring Clamp',
3.49, 0.65, '04', '005');
INSERT INTO Parts (PartID, PartDescription, PartPrice, PartCost, ItemClassID, ManufacturerID) VALUES ('0000006', 'Fish Tape', 7.9
9, 1.35, '04', '006');
INSERT INTO Parts (PartID, PartDescription, PartPrice, PartCost, ItemClassID, ManufacturerID) VALUES ('0000007', 'Safety Goggles'
, 5.49, 0.47, '03', '007');
INSERT INTO Parts (PartID, PartDescription, PartPrice, PartCost, ItemClassID, ManufacturerID) VALUES ('0000008', 'Gloves', 7.49,
2.33, '03', '008');
INSERT INTO Parts (PartID, PartDescription, PartPrice, PartCost, ItemClassID, ManufacturerID) VALUES ('0000009', 'Pliers', 5.49,
1.87, '01', '001');
INSERT INTO Parts (PartID, PartDescription, PartPrice, PartCost, ItemClassID, ManufacturerID) VALUES ('000000A', 'Thermometer', 1
1.49, 3.56, '04', '002');
INSERT INTO Parts (PartID, PartDescription, PartPrice, PartCost, ItemClassID, ManufacturerID) VALUES ('000000B', 'Shovel Type 1',
10.99, 3.99, '01', '005');
INSERT INTO Parts (PartID, PartDescription, PartPrice, PartCost, ItemClassID, ManufacturerID) VALUES ('000000C', 'Shovel Type 2',
10.99, 3.99, '01', '005');
INSERT INTO Parts (PartID, PartDescription, PartPrice, PartCost, ItemClassID, ManufacturerID) VALUES ('000000D', 'Shovel Type 3',
10.99, 3.99, '01', '005');
INSERT INTO Parts (PartID, PartDescription, PartPrice, PartCost, ItemClassID, ManufacturerID) VALUES ('000000E', 'Shovel Type 4',
10.99, 3.99, '01', '005');

```

[illegible]

```

INSERT INTO Parts (PartID, PartDescription, PartPrice, PartCost, ItemClassID, ManufacturerID) VALUES ('00000012', 'Clips Type 5',
9.99, 1.99, '04', '001');
INSERT INTO Parts (PartID, PartDescription, PartPrice, PartCost, ItemClassID, ManufacturerID) VALUES ('00000013', 'Clips Type 6',
9.99, 1.99, '04', '001');
INSERT INTO Parts (PartID, PartDescription, PartPrice, PartCost, ItemClassID, ManufacturerID) VALUES ('00000014', 'Clips Type 7',
9.99, 1.99, '04', '001');

INSERT INTO Warehouses (WarehouseID, WarehouseDescription) VALUES ('001', 'Northern Warehouse');
INSERT INTO Warehouses (WarehouseID, WarehouseDescription) VALUES ('002', 'Eastern Warehouse');
INSERT INTO Warehouses (WarehouseID, WarehouseDescription) VALUES ('003', 'Southern Warehouse');
INSERT INTO Warehouses (WarehouseID, WarehouseDescription) VALUES ('004', 'Western Warehouse');
INSERT INTO Warehouses (WarehouseID, WarehouseDescription) VALUES ('005', 'Northeastern Warehouse');
INSERT INTO Warehouses (WarehouseID, WarehouseDescription) VALUES ('006', 'Southeastern Warehouse');
INSERT INTO Warehouses (WarehouseID, WarehouseDescription) VALUES ('007', 'Northwestern Warehouse');
INSERT INTO Warehouses (WarehouseID, WarehouseDescription) VALUES ('008', 'Southwestern Warehouse');

INSERT INTO Inventories (WarehouseID, PartID, InventoryQuantity) VALUES ('002', '00000002', 534);
INSERT INTO Inventories (WarehouseID, PartID, InventoryQuantity) VALUES ('003', '00000003', 878);
INSERT INTO Inventories (WarehouseID, PartID, InventoryQuantity) VALUES ('005', '00000005', 122);
INSERT INTO Inventories (WarehouseID, PartID, InventoryQuantity) VALUES ('001', '00000006', 19);
INSERT INTO Inventories (WarehouseID, PartID, InventoryQuantity) VALUES ('004', '0000000A', 4957);
INSERT INTO Inventories (WarehouseID, PartID, InventoryQuantity) VALUES ('008', '00000001', 331);
INSERT INTO Inventories (WarehouseID, PartID, InventoryQuantity) VALUES ('008', '00000001', 0);
INSERT INTO Inventories (WarehouseID, PartID, InventoryQuantity) VALUES ('005', '00000003', 1010);
INSERT INTO Inventories (WarehouseID, PartID, InventoryQuantity) VALUES ('006', '00000004', 3);
INSERT INTO Inventories (WarehouseID, PartID, InventoryQuantity) VALUES ('007', '00000008', 333);
INSERT INTO Inventories (WarehouseID, PartID, InventoryQuantity) VALUES ('003', '00000009', 323);

INSERT INTO SalesReps (SalesRepID, SalesRepFirstName, SalesRepLastName, SalesRepStreet, SalesRepCity, SalesRepState, SalesRepZip,
SalesRepCommissionTotal, SalesRepCommissionRate) VALUES ('01', 'Camron', 'Khan', '1755 Golf Rd', 'Schaumburg', 'IL', '60173', 1034
8.33, 33);
INSERT INTO SalesReps (SalesRepID, SalesRepFirstName, SalesRepLastName, SalesRepStreet, SalesRepCity, SalesRepState, SalesRepZip,
SalesRepCommissionTotal, SalesRepCommissionRate) VALUES ('02', 'Antoinette', 'Pestillo', '1755 Golf Rd', 'Schaumburg', 'IL', '6017
3', 23434.44, 50);
INSERT INTO SalesReps (SalesRepID, SalesRepFirstName, SalesRepLastName, SalesRepStreet, SalesRepCity, SalesRepState, SalesRepZip,
SalesRepCommissionTotal, SalesRepCommissionRate) VALUES ('03', 'Frank', 'Policht', '1755 Golf Rd', 'Schaumburg', 'IL', '60173', 90
044.22, 10);
INSERT INTO SalesReps (SalesRepID, SalesRepFirstName, SalesRepLastName, SalesRepStreet, SalesRepCity, SalesRepState, SalesRepZip,
SalesRepCommissionTotal, SalesRepCommissionRate) VALUES ('04', 'Stephanie', 'Hutcheson', '1755 Golf Rd', 'Schaumburg', 'IL', '6017
3', 12324.33, 40);
INSERT INTO SalesReps (SalesRepID, SalesRepFirstName, SalesRepLastName, SalesRepStreet, SalesRepCity, SalesRepState, SalesRepZip,
SalesRepCommissionTotal, SalesRepCommissionRate) VALUES ('05', 'Mike', 'Kravtsov', '1755 Golf Rd', 'Schaumburg', 'IL', '60173', 60
433.21, 5);
INSERT INTO SalesReps (SalesRepID, SalesRepFirstName, SalesRepLastName, SalesRepStreet, SalesRepCity, SalesRepState, SalesRepZip,
SalesRepCommissionTotal, SalesRepCommissionRate) VALUES ('06', 'Floyd', 'Dexter', '1755 Golf Rd', 'Schaumburg', 'IL', '60173', 975
7.57, 10);

```



```

INSERT INTO SalesReps (SalesRepID, SalesRepFirstName, SalesRepLastName, SalesRepStreet, SalesRepCity, SalesRepState, SalesRepZip,
SalesRepCommissionTotal, SalesRepCommissionRate) VALUES ('07', 'Bob', 'Garcia', '1755 Golf Rd', 'Schaumburg', 'IL', '60173', 23449
.98, 5);
INSERT INTO SalesReps (SalesRepID, SalesRepFirstName, SalesRepLastName, SalesRepStreet, SalesRepCity, SalesRepState, SalesRepZip,
SalesRepCommissionTotal, SalesRepCommissionRate) VALUES ('08', 'Jenna', 'Mazeikis', '1755 Golf Rd', 'Schaumburg', 'IL', '60173', 7
5303.22, 25);

INSERT INTO Customers (CustomerID, CustomerName, CustomerStreet, CustomerCity, CustomerState, CustomerZip, CustomerBalance, Custom
erCreditLimit, SalesRepID) VALUES ('0001', 'IL Tools', '123 Road Rd', 'Chicago', 'IL', '60657', 8546.44, 9000, '01');
INSERT INTO Customers (CustomerID, CustomerName, CustomerStreet, CustomerCity, CustomerState, CustomerZip, CustomerBalance, Custom
erCreditLimit, SalesRepID) VALUES ('0002', 'NY Tools', '123 Street Rd', 'New York', 'NY', '10029', 200.45, 5000, '02');
INSERT INTO Customers (CustomerID, CustomerName, CustomerStreet, CustomerCity, CustomerState, CustomerZip, CustomerBalance, Custom
erCreditLimit, SalesRepID) VALUES ('0003', 'TX Tools', '123 Avenue Rd', 'Houston', 'TX', '77001', 1745.04, 3000, '03');
INSERT INTO Customers (CustomerID, CustomerName, CustomerStreet, CustomerCity, CustomerState, CustomerZip, CustomerBalance, Custom
erCreditLimit, SalesRepID) VALUES ('0004', 'LA Tools', '123 Boulevard Rd', 'Los Angeles', 'CA', '90016', 3478.21, 8000, '04');
INSERT INTO Customers (CustomerID, CustomerName, CustomerStreet, CustomerCity, CustomerState, CustomerZip, CustomerBalance, Custom
erCreditLimit, SalesRepID) VALUES ('0005', 'FL Tools', '123 Highway Rd', 'Miami', 'FL', '33114', 8766.99, 9000, '05');
INSERT INTO Customers (CustomerID, CustomerName, CustomerStreet, CustomerCity, CustomerState, CustomerZip, CustomerBalance, Custom
erCreditLimit, SalesRepID) VALUES ('0006', 'SF Tools', '123 Route Rd', 'San Francisco', 'CA', '94114', 234.11, 2000, '06');
INSERT INTO Customers (CustomerID, CustomerName, CustomerStreet, CustomerCity, CustomerState, CustomerZip, CustomerBalance, Custom
erCreditLimit, SalesRepID) VALUES ('0007', 'WA Tools', '123 Frontage Rd', 'Seattle', 'WA', '98113', 774.34, 1000, '07');
INSERT INTO Customers (CustomerID, CustomerName, CustomerStreet, CustomerCity, CustomerState, CustomerZip, CustomerBalance, Custom
erCreditLimit, SalesRepID) VALUES ('0008', 'CO Tools', '123 Lane Rd', 'Denver', 'CO', '80220', 5578.39, 7000, '08');

INSERT INTO Orders (OrderID, OrderDate, OrdererType, CustomerID) VALUES ('00001', '2017-03-01', 'customer', '0001');
INSERT INTO Orders (OrderID, OrderDate, OrdererType, CustomerID) VALUES ('00002', '2017-04-01', 'sales rep', '0002');
INSERT INTO Orders (OrderID, OrderDate, OrdererType, CustomerID) VALUES ('00003', '2017-05-01', 'customer', '0003');
INSERT INTO Orders (OrderID, OrderDate, OrdererType, CustomerID) VALUES ('00004', '2017-06-01', 'sales rep', '0004');
INSERT INTO Orders (OrderID, OrderDate, OrdererType, CustomerID) VALUES ('00005', '2017-07-01', 'customer', '0005');
INSERT INTO Orders (OrderID, OrderDate, OrdererType, CustomerID) VALUES ('00006', '2017-08-01', 'sales rep', '0006');
INSERT INTO Orders (OrderID, OrderDate, OrdererType, CustomerID) VALUES ('00007', '2017-09-01', 'customer', '0007');
INSERT INTO Orders (OrderID, OrderDate, OrdererType, CustomerID) VALUES ('00008', '2017-10-01', 'sales rep', '0008');
INSERT INTO Orders (OrderID, OrderDate, OrdererType, CustomerID) VALUES ('00009', '2017-11-01', 'customer', '0002');
INSERT INTO Orders (OrderID, OrderDate, OrdererType, CustomerID) VALUES ('0000A', '2017-11-01', 'sales rep', '0007');

INSERT INTO OrderLines (OrderID, PartID, OrderLineQuantity, OrderLineQuotedPrice) VALUES ('00001', '00000001', 10, 90);
INSERT INTO OrderLines (OrderID, PartID, OrderLineQuantity, OrderLineQuotedPrice) VALUES ('00001', '00000002', 40, 80);
INSERT INTO OrderLines (OrderID, PartID, OrderLineQuantity, OrderLineQuotedPrice) VALUES ('00002', '00000003', 50, 19);
INSERT INTO OrderLines (OrderID, PartID, OrderLineQuantity, OrderLineQuotedPrice) VALUES ('00003', '00000004', 60, 15);
INSERT INTO OrderLines (OrderID, PartID, OrderLineQuantity, OrderLineQuotedPrice) VALUES ('00004', '00000005', 70, 3);
INSERT INTO OrderLines (OrderID, PartID, OrderLineQuantity, OrderLineQuotedPrice) VALUES ('00004', '00000006', 80, 7);
INSERT INTO OrderLines (OrderID, PartID, OrderLineQuantity, OrderLineQuotedPrice) VALUES ('00004', '00000007', 90, 5);
INSERT INTO OrderLines (OrderID, PartID, OrderLineQuantity, OrderLineQuotedPrice) VALUES ('00005', '00000008', 20, 7);
INSERT INTO OrderLines (OrderID, PartID, OrderLineQuantity, OrderLineQuotedPrice) VALUES ('00006', '00000009', 10, 5);
INSERT INTO OrderLines (OrderID, PartID, OrderLineQuantity, OrderLineQuotedPrice) VALUES ('00007', '0000000A', 80, 10);
INSERT INTO OrderLines (OrderID, PartID, OrderLineQuantity, OrderLineQuotedPrice) VALUES ('00007', '00000001', 40, 80);

```

```

INSERT INTO OrderLines (OrderID, PartID, OrderLineQuantity, OrderLineQuotedPrice) VALUES ('00007', '00000002', 30, 70);
INSERT INTO OrderLines (OrderID, PartID, OrderLineQuantity, OrderLineQuotedPrice) VALUES ('00008', '00000003', 40, 15);
INSERT INTO OrderLines (OrderID, PartID, OrderLineQuantity, OrderLineQuotedPrice) VALUES ('00008', '00000004', 30, 14);
INSERT INTO OrderLines (OrderID, PartID, OrderLineQuantity, OrderLineQuotedPrice) VALUES ('00008', '00000005', 60, 2);
INSERT INTO OrderLines (OrderID, PartID, OrderLineQuantity, OrderLineQuotedPrice) VALUES ('00009', '00000006', 20, 6);
INSERT INTO OrderLines (OrderID, PartID, OrderLineQuantity, OrderLineQuotedPrice) VALUES ('0000A', '00000007', 90, 4);
INSERT INTO OrderLines (OrderID, PartID, OrderLineQuantity, OrderLineQuotedPrice) VALUES ('0000A', '00000008', 20, 6);
INSERT INTO OrderLines (OrderID, PartID, OrderLineQuantity, OrderLineQuotedPrice) VALUES ('0000A', '00000009', 30, 4);
INSERT INTO OrderLines (OrderID, PartID, OrderLineQuantity, OrderLineQuotedPrice) VALUES ('0000A', '0000000A', 10, 9);

```

```

/* QUERIES *****/

```

```

/* For a given sales rep, list the number (C2), the name (broken out first name
and last name and each should be C15), the address (broken out into street,
city, state and zip with each being a character variable that you can specify
the valid length), the total commission (D7,2), and the commission rate. */

```

```

SELECT  SalesRepID,
        SalesRepFirstName,
        SalesRepLastName,
        SalesRepStreet,
        SalesRepCity,
        SalesRepState,
        SalesRepZip,
        SalesRepCommissionTotal,
        SalesRepCommissionRate
FROM    SalesReps
WHERE   SalesRepID = [Enter Sales Rep ID];

```

```

/* For a given customer, list the number (C4), the name (C15), the address
(broken out as specified above), the current balance (D6,2), and the credit
limit (D4). Also list the number and name of the sales rep who represents
the customer. */

```

```

SELECT  Customers.CustomerID,
        Customers.CustomerName,
        Customers.CustomerStreet,
        Customers.CustomerCity,
        Customers.CustomerState,
        Customers.CustomerZip,
        Customers.CustomerBalance,
        Customers.CustomerCreditLimit,
        SalesReps.SalesRepID,
        SalesReps.SalesRepFirstName,
        SalesReps.SalesRepLastName,
FROM    Customers

```

```

INNER JOIN SalesReps ON Customers.SalesRepID = SalesReps.SalesRepID
WHERE      Customers.CustomerID = [Enter Customer ID];

/* For a given part, list the part number (C8), the description (C15), the price
(D6,2), the cost (D6,2), the total number of units on hand (summed across
warehouses) (D4), and the on-hand value (cost times total units on hand). */

SELECT      Parts.PartID,
            Parts.PartDescription,
            Parts.PartPrice,
            Parts.PartCost,
            SUM(Inventories.InventoryQuantity) AS UnitsOnHand,
            (Parts.PartCost * UnitsOnHand) AS OnHandValue
FROM        Inventories
INNER JOIN  Parts ON Inventories.PartID = Parts.PartID
WHERE      Inventories.PartID = [Enter Part ID]
GROUP BY   Parts.PartID,
            Parts.PartDescription,
            Parts.PartPrice,
            Parts.PartCost,
            OnHandValue;

/* REPORTS *****/

/* For each warehouse, list the warehouse number and the warehouse
description. In addition, for each part currently stored in the warehouse, list
the tool number, tool description, and the number of units of the tool currently
stored in the warehouse. */

SELECT      Warehouses.WarehouseID,
            Warehouses.WarehouseDescription,
            Parts.PartID,
            Parts.PartDescription,
            Inventories.InventoryQuantity
FROM        ((Warehouses
INNER JOIN  Inventories ON Warehouses.WarehouseID = Inventories.WarehouseID)
INNER JOIN  Parts ON Inventories.PartID = Parts.PartID)
ORDER BY   Warehouses.WarehouseID,
            Parts.PartID;

/* For each item class, list the class number and description, as well as the tool
numbers and tool descriptions of all tools in the item class. Note that the
legal values for item class are hand tools (ex, shovels, saws, rakes,
hammers, etc.), power tools (ex, sanders, drills, saws, etc.), safety
equipment (ex, hazard cones, signs, barricades, etc.), and miscellaneous
equipment. */

```

```

SELECT      ItemClasses.ItemClassID,
            ItemClasses.ItemClassDescription,
            Parts.PartID,
            Parts.PartDescription
FROM        ItemClasses
INNER JOIN  Parts ON ItemClasses.ItemClassID = Parts.ItemClassID
ORDER BY    ItemClass.ItemClassID,
            Parts.PartID;

```

/* For each order, list the order number (C5), the order date, and the number and name of the customer who placed the order and an indicator of whether the customer or sales rep placed the order. In addition, for each order line for the order, list the part number, description, number (D4), and the quoted price. */

```

SELECT      Orders.OrderID,
            Orders.OrderDate,
            Customers.CustomerID,
            Customers.CustomerName,
            Orders.OrdererType,
            OrderLines.PartID,
            Parts.PartDescription,
            OrderLines.OrderLineQuantity,
            OrderLines.OrderLineQuotedPrice
FROM        ((Orders
INNER JOIN  Customers ON Orders.CustomerID = Customers.CustomerID)
INNER JOIN  OrderLines ON Orders.OrderID = OrderLines.OrderID)
INNER JOIN  Parts ON OrderLines.PartID = Parts.PartID)
ORDER BY    OrderLines.OrderID,
            OrderLines.PartID;

```

/* For each sales rep, list the number and name. In addition, list the number, name, and address (broken out as specified above) for each customer represented by the sales rep. */

```

SELECT      SalesReps.SalesRepID,
            SalesReps.SalesRepFirstName,
            SalesReps.SalesRepLastName,
            Customers.CustomerID,
            Customers.CustomerName,
            Customers.CustomerStreet,
            Customers.CustomerCity,
            Customers.CustomerState,
            Customers.CustomerZip
FROM        SalesReps

```

```

INNER JOIN Customers ON SalesReps.SalesRepID = Customers.SalesRepID
ORDER BY SalesReps.SalesRepID,
         Customers.SalesRepID;

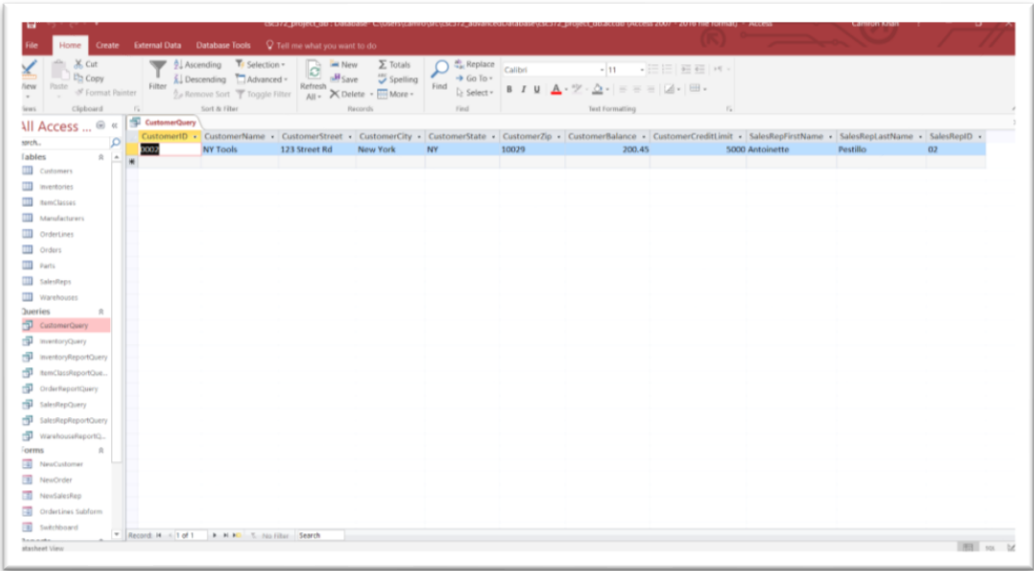
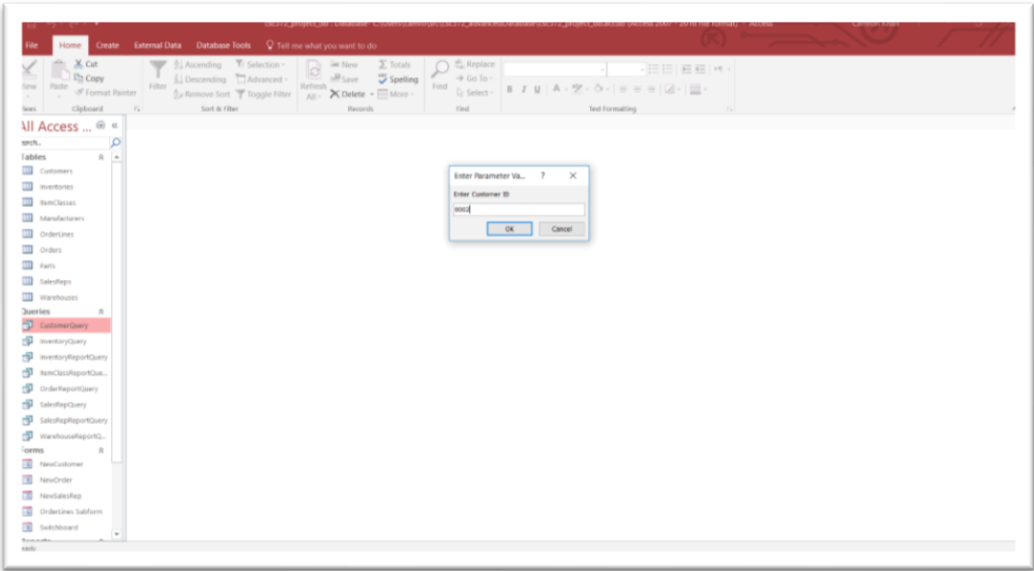
/* For each part that has zero for number on hand, list the part number,
description, warehouse number, warehouse description, manufacturer ID
(C3), manufacturer name (C25). In addition, list all other warehouses that
have the part in stock, including the warehouse number, warehouse
description and number on hand. The report should be in part number order
followed by ascending number on hand ordering within the part number. */

SELECT Parts.PartID,
       Parts.PartDescription,
       Inventories.InventoryQuantity,
       Warehouses.WarehouseID,
       Warehouses.WarehouseDescription,
       Manufacturers.ManufacturerID,
       Manufacturers.ManufacturerName
FROM   (((Parts
INNER JOIN Inventories AS InventoriesMin ON Parts.PartID = InventoriesMin.PartID)
INNER JOIN Inventories ON Parts.PartID = Inventories.PartID)
INNER JOIN Warehouses ON Inventories.WarehouseID = Warehouses.WarehouseID)
INNER JOIN Manufacturers ON Parts.ManufacturerID = Manufacturers.ManufacturerID)
GROUP BY Parts.PartID,
         Parts.PartDescription,
         Inventories.InventoryQuantity,
         Warehouses.WarehouseID,
         Warehouses.WarehouseDescription,
         Manufacturers.ManufacturerID,
         Manufacturers.ManufacturerName
HAVING MIN(InventoriesMin.InventoryQuantity) = 0
ORDER BY Parts.PartID,
         Inventories.InventoryQuantity;

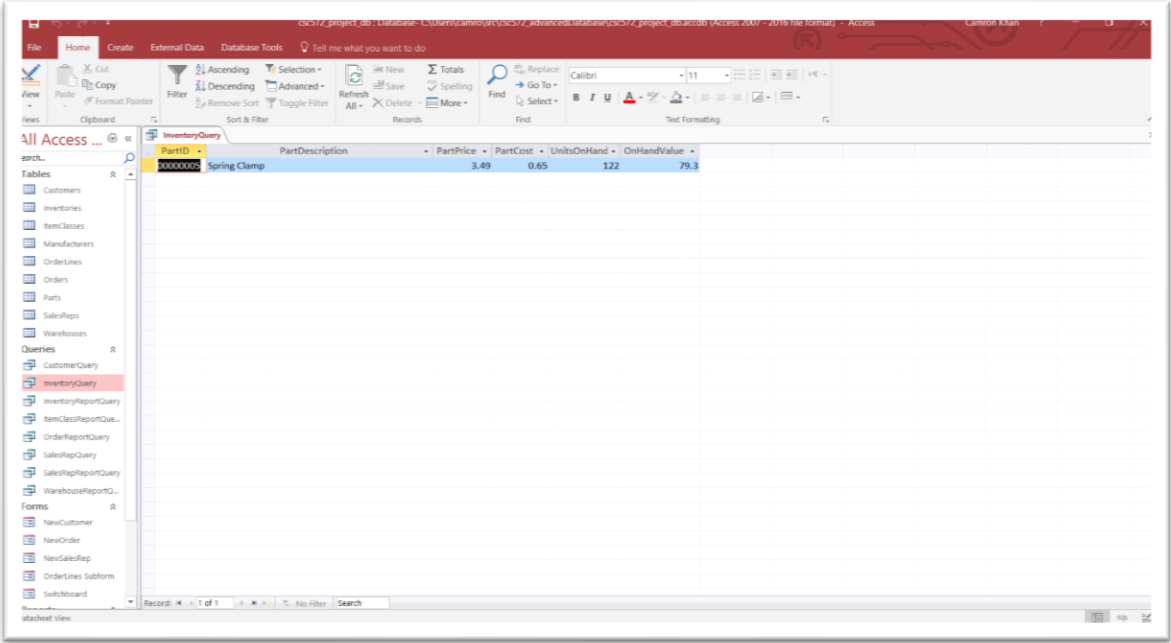
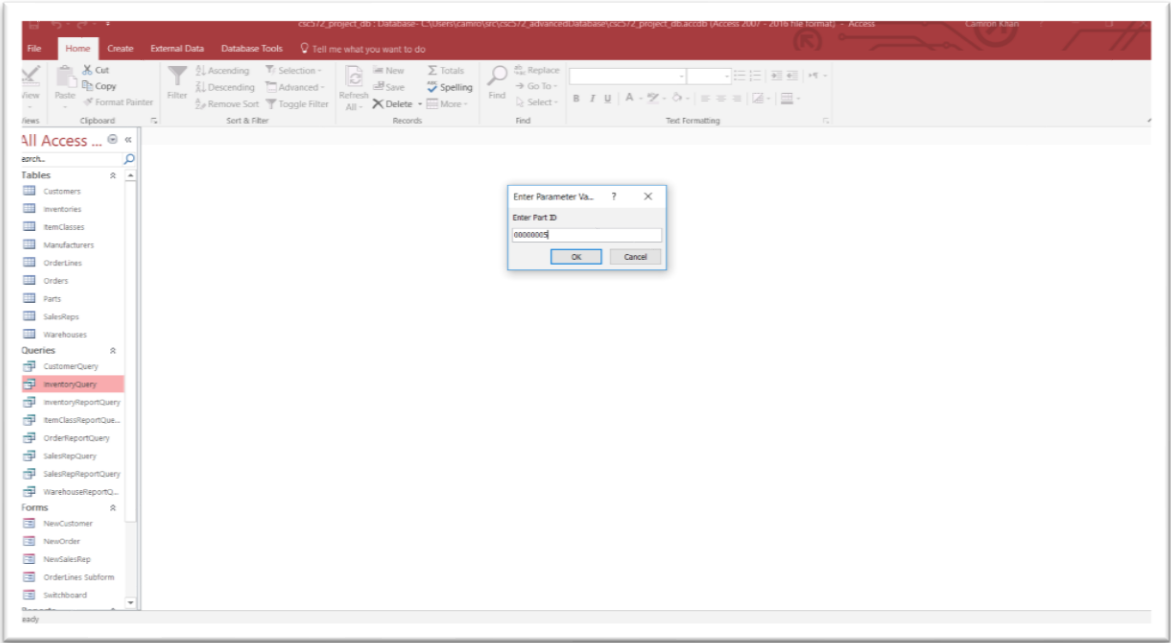
```

Screen Shots

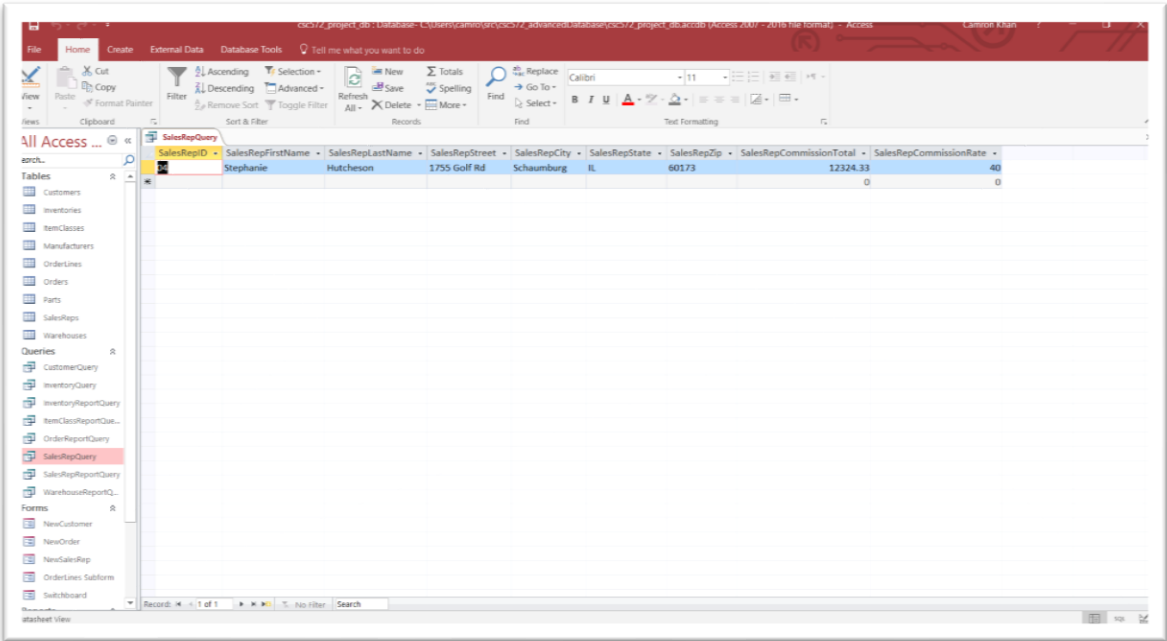
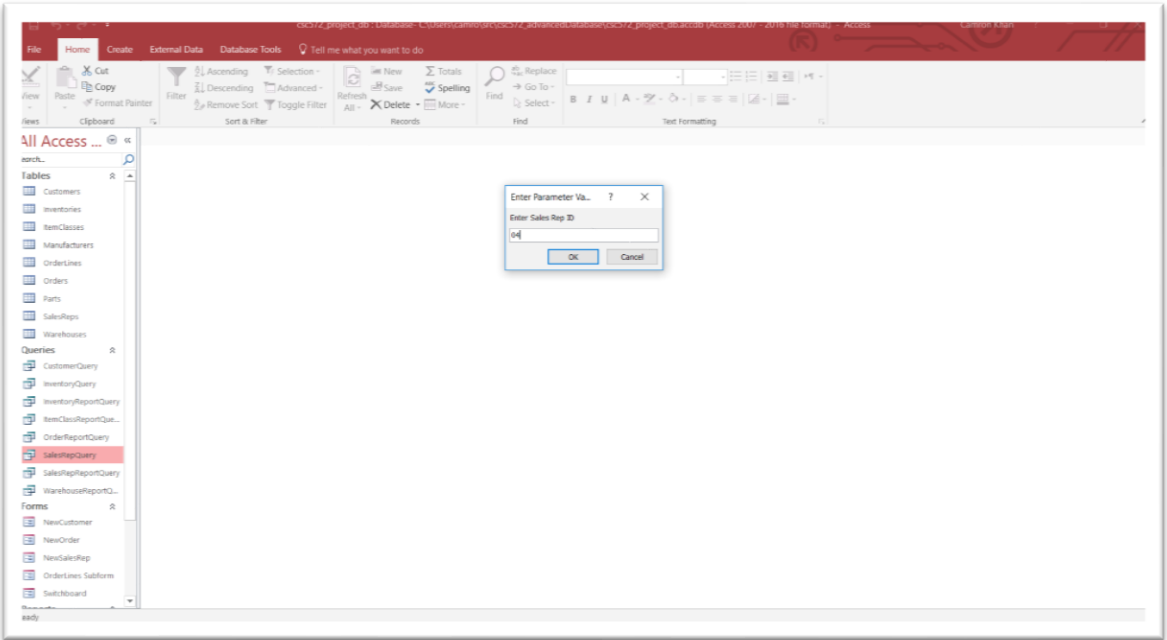
Query: Customer



Query: Inventory



Query: Sales Rep



Form: New Customer

Microsoft Access - New Customer Form

Customers

CustomerID:

CustomerName:

CustomerStreet:

CustomerCity:

CustomerState:

CustomerZip:

CustomerBalance:

CustomerCreditLimit:

SalesRep:

Customers	Adam
Dexter	Floyd
Garcia	Bob
Hutcherson	Stephanie
Rhian	Carson
Kravisov	Mike
Maxwellis	Jenna
Proffitts	Autumnette
Pulchit	Frank

Microsoft Access - New Customer Form

Customers

CustomerID: 0004

CustomerName: IN Tools

CustomerStreet: 123 Circle Rd

CustomerCity: Indianapolis

CustomerState: IN

CustomerZip: 46077

CustomerBalance: 0

CustomerCreditLimit: 5000

SalesRep:

Customers	Adam
Dexter	Floyd
Garcia	Bob
Hutcherson	Stephanie
Rhian	Carson
Kravisov	Mike
Maxwellis	Jenna
Proffitts	Autumnette
Pulchit	Frank

Microsoft Access - Customers Table

CustomerID	CustomerName	CustomerStreet	CustomerCity	CustomerState	CustomerZip	CustomerBalance	CustomerCreditLimit	SalesRepID
0004	IN Tools	123 Circle Rd	Indianapolis	IN	46077	0	5000	
0002	NY Tools	123 Street Rd	New York	NY	10009	200.45	3000.02	
0003	TX Tools	123 Avenue Rd	Houston	TX	77001	1745.04	3000.03	
0004	LA Tools	123 Boulevard Rd	Los Angeles	CA	90008	3478.21	3000.04	
0005	FL Tools	123 Highway Rd	Miami	FL	13114	8768.99	3000.05	
0006	SF Tools	123 Route Rd	San Francisco	CA	94134	234.11	2000.06	
0007	WA Tools	123 Frontage Rd	Seattle	WA	98113	774.84	1000.07	
0008	CO Tools	123 Lane Rd	Denver	CO	80220	5578.39	7000.08	
0009	WI Tools	123 Court Rd	Madison	WI	53708	500	1000.02	
0004	IN Tools	123 Circle Rd	Indianapolis	IN	46077	0	5000.03	

Form: New Order

The screenshot shows the 'NewOrder' form in Microsoft Access. The form is in 'Form View' and displays the following fields:

- OrderID: [Empty]
- OrderDate: 12/4/2017
- OrderType: Sales rep
- Customer: [Empty]
- OrderLines: A table with columns OrderID, Part, OrderLineQuantity, and OrderLineQuantityPrice. The table is currently empty.

The left-hand pane shows the 'NewOrder' form selected under the 'Forms' tab. The status bar at the bottom indicates 'Records: 1 of 1'.

The screenshot shows the 'NewOrder' form in Microsoft Access. The form is in 'Form View' and displays the following fields:

- OrderID: 0000C
- OrderDate: 12/4/2017
- OrderType: Sales rep
- Customer: JF Tools
- OrderLines: A table with columns OrderID, Part, OrderLineQuantity, and OrderLineQuantityPrice. The table contains two rows of data:

OrderID	Part	OrderLineQuantity	OrderLineQuantityPrice
0000C	Clip Type 4	2	55.00
0000C		0	0

The left-hand pane shows the 'NewOrder' form selected under the 'Forms' tab. The status bar at the bottom indicates 'Records: 1 of 1'.

The screenshot shows the 'NewOrder' form in Microsoft Access. The form is in 'Table View' and displays a table with the following columns: OrderID, OrderDate, OrderType, and CustomerID. The table contains 12 rows of data:

OrderID	OrderDate	OrderType	CustomerID
00001	3/1/2017	Sales rep	0001
00002	4/1/2017	Sales rep	0002
00003	5/1/2017	customer	0003
00004	6/1/2017	Sales rep	0004
00005	7/1/2017	customer	0005
00006	8/1/2017	Sales rep	0006
00007	9/1/2017	customer	0007
00008	10/1/2017	Sales rep	0008
00009	11/1/2017	customer	0009
00010	12/1/2017	Sales rep	0010
00011	12/4/2017	customer	0011
00012	12/4/2017	Sales rep	0012

The left-hand pane shows the 'NewOrder' form selected under the 'Forms' tab. The status bar at the bottom indicates 'Records: 12 of 12'.

Form: New Sales Rep

The screenshot displays the Microsoft Access application window. The title bar indicates the file path is 'C:\Users\user\Desktop\Access\Access2016\Access2016.accdb'. The ribbon at the top includes 'File', 'Home', 'Create', 'External Data', 'Database Tools', and 'View'. The 'View' tab is active, showing options like 'Find', 'Go To', and 'Select'. The main area shows a form titled 'NewSalesRep' with the following fields and their values:

Field Name	Value
SalesRepID	
SalesRepFirstName	
SalesRepLastName	
SalesRepAddress	
SalesRepCity	
SalesRepState	
SalesRepZip	
CommissionTotal	0
CommissionRate	0

The left sidebar shows the database structure with the following tables and queries:

- Customers
- Invoices
- ItemClasses
- Manufacturers
- OrderLines
- Orders
- Parts
- ShipIngs
- Shipments
- Invoices
- CustomerQuery
- InventoryQuery
- InventoryReportQuery
- ItemClassQuery
- OrderReportQuery
- ShipIngsQuery
- ShipIngsReportQuery
- ShipmentsReportQuery
- Tables
- NewCustomer
- NewOrder
- NewSalesRep
- OrderLinesTable
- ShipIngsTable

The bottom status bar shows 'Record: 1 of 1' and 'No Filter'.

The screenshot shows the Microsoft Access application window. The title bar reads 'Microsoft Access - Customer.accdb'. The ribbon is set to 'Formulas', with the 'Formulas' tab selected. The 'NewSalesRep' form is open, displaying a list of fields on the left and a form view on the right. The 'SalesRepAddress' field is highlighted with a red box. The form view shows the following data:

Field Name	Value
SalesRepID	DA
SalesRepFirstName	Dave
SalesRepLastName	Carlson
SalesRepAddress	1755 Golf Rd
SalesRepCity	Schaumburg
SalesRepState	IL
SalesRepZip	60179
CommissionPct	6
CommissionRate	14

The screenshot displays the Microsoft Access application window with the 'SalesRep' table open in Datasheet View. The table has 10 records. The 'SalesRepCommissionTotal' field is highlighted in blue. The 'SalesRepCommission' field is also highlighted in blue. The 'SalesRepCommissionTotal' field is highlighted in blue.

	SalesRepID	SalesRepFirstName	SalesRepLastName	SalesRepStreet	SalesRepCity	SalesRepState	SalesRepCommissionTotal	SalesRepCommission
01	01	Camron	Rhan	1755 Golf Rd	Schaumburg	IL	60173	10348.33
02	02	Andriacette	Pastillo	1755 Golf Rd	Schaumburg	IL	60173	23434.44
03	03	Frank	Publico	1755 Golf Rd	Schaumburg	IL	60173	90044.22
04	04	Stephanie	Hutcherson	1755 Golf Rd	Schaumburg	IL	60173	12324.33
05	05	Mike	Kravtsov	1755 Golf Rd	Schaumburg	IL	60173	40433.21
06	06	Floyd	Steiner	1755 Golf Rd	Schaumburg	IL	60173	9757.57
07	07	Bob	Garcia	1755 Golf Rd	Schaumburg	IL	60173	23445.98
08	08	Jenna	Mazzeika	1755 Golf Rd	Schaumburg	IL	60173	75303.22
09	09	Adam	Commins	1755 Golf Rd	Schaumburg	IL	60173	0
10	10	Dave	Carlson	1755 Golf Rd	Schaumburg	IL	60173	0

Form: Switchboard

The screenshot shows the Microsoft Access application window with the title bar "csc572_project_db : Database- C:\Users\camro\src\csc572_advancedDatabase\csc572_project_db.accdb (Access 2007 - 2016 file format) - Access". The ribbon includes "File", "Home", "Create", "External Data", and "Database Tools". The "Home" ribbon is active, showing options for "View", "Clipboard", "Sort & Filter", "Records", "Find", and "Text Formatting".

The main window displays the "Switchboard" form in "Form View". The form has a title bar "Switchboard" and a search bar. The left-hand navigation pane lists the following objects:

- Orders
- Parts
- SalesReps
- Warehouses
- Queries
 - CustomerQuery
 - InventoryQuery
 - InventoryReportQuery
 - ItemClassReportQue...
 - OrderReportQuery
 - SalesRepQuery
 - SalesRepReportQuery
 - WarehouseReportQ...
- Forms
 - NewCustomer
 - NewOrder
 - NewSalesRep
 - OrderLines Subform
 - Switchboard (highlighted)
- Reports
 - InventoryReport
 - ItemClassReport
 - OrderReport
 - SalesRepReport
 - WarehouseReport

The main area of the Switchboard form is divided into three columns: "Queries", "Forms", and "Reports". Each column contains a list of buttons that link to the corresponding objects in the database:

- Queries:** Customer, Inventory, Sales Rep
- Forms:** New Customer, New Order, New Sales Rep
- Reports:** Inventory, Item Class, Order, Sales Rep, Warehouse

The status bar at the bottom indicates "Record: 1 of 1" and "No Filter".

Report: Inventory

FileHomeCreateExternal DataDatabase ToolsTell me what you want to do

View

CutCopyPasteFormat Painter

Filter

AscendingDescendingRemove SortAdvancedToggle Filter

Sort & Filter

RefreshAllNewSaveDeleteMore

Records

Find

ReplaceGo ToSelect

Find

B I U A - ab - - - - -

Text Formatting

All Access ...

Search...

OrdersPartsSalesRepsWarehouses

Queries

CustomerQueryInventoryQueryInventoryReportQueryItemClassReportQue...OrderReportQuerySalesRepQuerySalesRepReportQueryWarehouseReportQ...

Forms

NewCustomerNewOrderNewSalesRepOrderLines SubformSwitchboard

Reports

InventoryReportItemClassReportOrderReportSalesRepReportWarehouseReport

InventoryReport

PartID00000001

PartDescriptionAngle Drill

ManufacturerID001

ManufacturerNameAlabama Mfg.

InventoryQuantity	WarehouseID	WarehouseDescription
0	001	Northern Warehouse
331	008	Southwestern Warehouse

Print

Monday, December 4, 2017

Page 1 of 1

Report View

Report: Item Class

FileHomeCreateExternal DataDatabase ToolsTell me what you want to do

View

CutCopyPasteFormat Painter

Filter

AscendingDescendingRemove Sort

SelectionAdvancedToggle Filter

RefreshAll

NewSaveDelete

TotalsSpellingMore

Find

ReplaceGo ToSelect

Text Formatting

All Access ...

Search...

OrdersPartsSalesRepsWarehouses

Queries

CustomerQueryInventoryQueryInventoryReportQueryItemClassReportQue...OrderReportQuerySalesRepQuerySalesRepReportQueryWarehouseReportQ...

Forms

NewCustomerNewOrderNewSalesRepOrderLines SubformSwitchboard

Reports

InventoryReportItemClassReportOrderReportSalesRepReportWarehouseReport

ItemClassReport

ItemClassID01

ItemClassDescriptionhand tools

PartID	PartDescription
00000003	Claw Hammer
00000004	Brace Drill Aug
00000009	Pliers
00000008	Shovel Type 1
0000000C	Shovel Type 2
0000000D	Shovel Type 3
0000000E	Shovel Type 4
0000000F	Shovel Type 5
0000000G	Shovel Type 6
0000000H	Shovel Type 7

ItemClassID02

ItemClassDescriptionpower tools

PartID	PartDescription
00000001	Angle Drill

Report View

Report: Order

csc572_project_db : Database- C:\Users\camro\src\csc572_advancedDatabase\csc572_project_db.accdb (Access 2007 - 2016 file format) - Access Camron Khan

File Home Create External Data Database Tools Tell me what you want to do

View Paste Cut Copy Format Painter Filter Sort & Filter Advanced Refresh Save Delete Records Find Replace Go To Select Text Formatting

All Access ... Orders

Search...

Orders

Parts

SalesReps

Warehouses

Queries

CustomerQuery

InventoryQuery

InventoryReportQuery

ItemClassReportQue...

OrderReportQuery

SalesRepQuery

SalesRepReportQuery

WarehouseReportQ...

Forms

NewCustomer

NewOrder

NewSalesRep

OrderLines Subform

Switchboard

Reports

InventoryReport

ItemClassReport

OrderReport

SalesRepReport

WarehouseReport

Report View

Orders

OrderID	00001	OrdererType	sales rep	
OrderDate	3/1/2017			
CustomerID	0001			
CustomerName	IL Tools			
PartID	PartDescription	OrderLineQuantity	OrderLineQuotedPrice	
00000002	Circular Saw	40	80	
0000000A	Thermometer	10	90	

OrderID	00002	OrdererType	sales rep	
OrderDate	4/1/2017			
CustomerID	0002			
CustomerName	NY Tools			
PartID	PartDescription	OrderLineQuantity	OrderLineQuotedPrice	
00000003	Claw Hammer	50	19	

OrderID	00003	OrdererType	customer	
OrderDate	5/1/2017			
CustomerID	0003			

Report: Sales Rep

Access 2016 interface showing the SalesRepReport.

Database: csc572_project_db : Database- C:\Users\camro\src\csc572_advancedDatabase\csc572_project_db.accdb (Access 2007 - 2016 file format) - Access

Camron Khan

File Home Create External Data Database Tools Tell me what you want to do

Views Clipboard Sort & Filter Records Find Text Formatting

Search...

Queries

- CustomerQuery
- InventoryQuery
- InventoryReportQuery
- ItemClassReportQue...
- OrderReportQuery
- SalesRepQuery
- SalesRepReportQuery
- WarehouseReportQ...

Forms

- NewCustomer
- NewOrder
- NewSalesRep
- OrderLines Subform
- Switchboard

Reports

- InventoryReport
- ItemClassReport
- OrderReport
- SalesRepReport**
- WarehouseReport

SalesRepReport

SalesRepID: 02
SalesRepFirstName: Antoinette
SalesRepLastName: Pestillo

CustomerID	CustomerName	CustomerStreet	CustomerCity	CustomerStat	CustomerZip
0002	NY Tools	123 Street Rd	New York	NY	10029
0009	WI Tools	123 Court Rd	Madison	WI	53558

SalesRepID: 03
SalesRepFirstName: Frank
SalesRepLastName: Policht

CustomerID	CustomerName	CustomerStreet	CustomerCity	CustomerStat	CustomerZip
0003	TX Tools	123 Avenue Rd	Houston	TX	77001
000A	IN Tools	123 Circle Rd	Indianapolis	IN	46077

SalesRepID: 04
SalesRepFirstName: Stephanie
SalesRepLastName: Hutcheson

CustomerID	CustomerName	CustomerStreet	CustomerCity	CustomerStat	CustomerZip
------------	--------------	----------------	--------------	--------------	-------------

Report View

Report: Warehouse

FileHomeCreateExternal DataDatabase ToolsTell me what you want to do

View

CutCopyPasteFormat Painter

Filter

AscendingDescendingRemove Sort

SelectionAdvancedToggle Filter

RefreshAll

NewSaveDelete

TotalsABC SpellingMore

Find

ReplaceGo ToSelect

B I U A - ab - - - - -

Text Formatting

All Access ...

Search...

OrdersPartsSalesRepsWarehouses

QueriesCustomerQueryInventoryQueryInventoryReportQueryItemClassReportQue...OrderReportQuerySalesRepQuerySalesRepReportQueryWarehouseReportQ...

FormsNewCustomerNewOrderNewSalesRepOrderLines SubformSwitchboard

ReportsInventoryReportItemClassReportOrderReportSalesRepReportWarehouseReport

WarehouseReport

WarehouseID001

WarehouseDescriptionNorthern Warehouse

PartID	PartDescription	InventoryQuantity
00000001	Angle Drill	0
00000006	Fish Tape	19

WarehouseID002

WarehouseDescriptionEastern Warehouse

PartID	PartDescription	InventoryQuantity
00000002	Circular Saw	534

WarehouseID003

WarehouseDescriptionSouthern Warehouse

PartID	PartDescription	InventoryQuantity
00000003	Claw Hammer	878
00000009	Pliers	323

WarehouseID004

WarehouseDescriptionWestern Warehouse

Report View