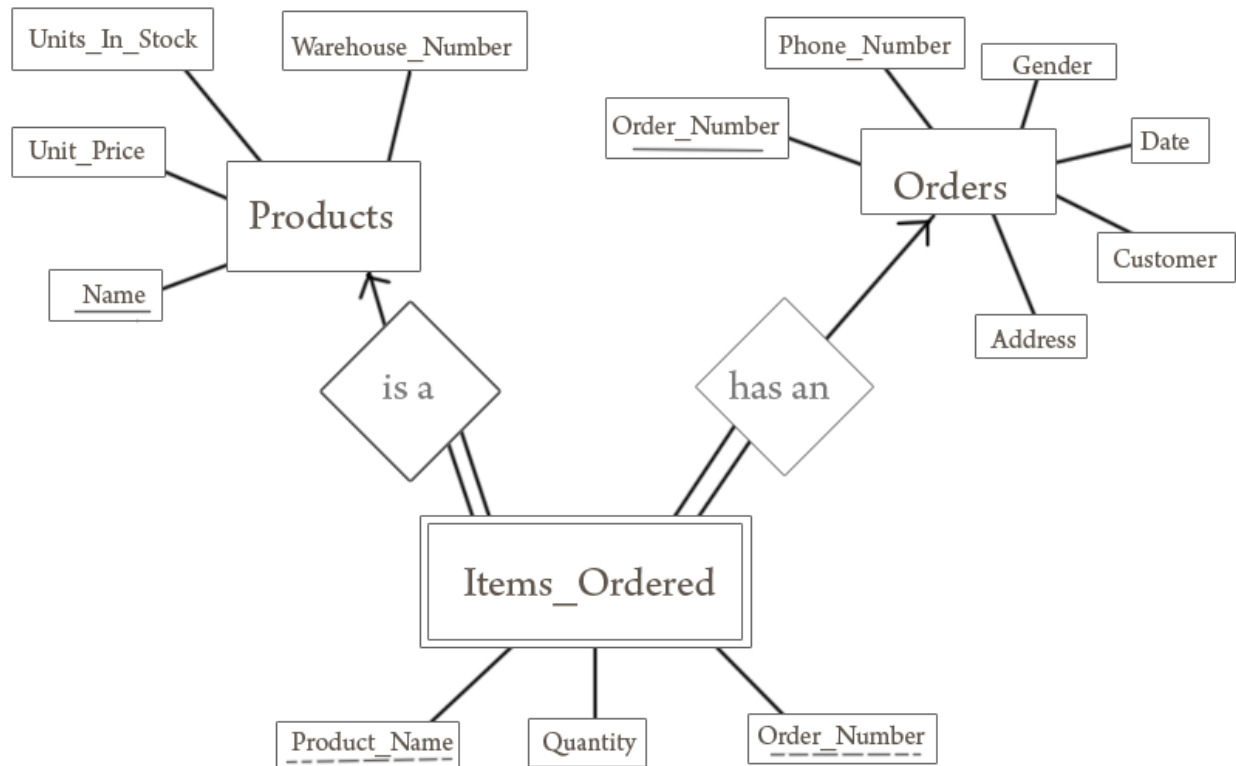


Bradford Knowlton
Assignment #1
September 20, 2010
Database

Company ABC



Products(Product Name, Unit_Price, Units_In_Stock, Warehouse_Number)

Items_Ordered(Product_name, quantity, Order_Number) ** Product_Name & Order_Number are foreign keys linked to Products and Orders respectively**

Orders(Order_Number, Phone_Number, Gender, Date, Customer, Address)

Queries:

Query 1

Retrieve the names and the phone numbers of all the male customers.

```
SELECT customer_name, customer_phone_number FROM Orders Where  
customer_gender = "M"
```

customer_name	customer_phone_number
Sally Adams	123-456-7891
Don Charles	222-333-4444
Don Martin	
Sally Adams	123-456-7891
Don Martin	

Query 2

Retrieve the names and the addresses of all the customers who have placed the order(s), ordered alphabetically by last name, first name.

```
SELECT CASE when SUBSTR(customer_name,4,1) = " " then substr(customer_name, 4) || ", " || substr(customer_name,0,4) else case when SUBSTR(customer_name,5,1) = " " then substr(customer_name,5) || ", " || substr(customer_name,0,5) else substr(customer_name,6) || ", " || substr(customer_name,0,6) end end as full_name, customer_address from Orders WHERE order_number in (SELECT order_number from ordered_items) GROUP BY full_name ORDER BY full_name
```

full_name	customer_address
Adams, Sally	481 Oak, Lan, MI
Charles, Don	
Martin, Don	419 Chip, Grant, NY
Nelson, Mary	108 Pine, Ada, Ca

Query 3

Make a list of the orders that contain more than 2 skates and were placed on 11/04/2009.

```
SELECT order_number from Orders WHERE order_date = "11/04/2009" AND order_number in (select order_number from Ordered_items where product_name = "Skates" AND product_quantity > 2 GROUP BY order_number)
```

order_number
1004
1006

Query 4

Retrieve the names of the customers whose addresses are in Ada, Ca.

SELECT customer_name from orders where customer_address like "%Ada, Ca%"

customer_name
Mary Nelson
Judy Roberts

Query 5

Retrieve all products whose prices are between \$200 AND \$400.

SELECT product_name FROM products WHERE unit_price > 200 AND unit_price < 400

product_name
Iron
Washer
Skates

Query 6

Retrieve a list of products and the total quantities that have been ordered, sorted in the ascending order by their quantities.

SELECT ordered_items.product_name, sum(product_quantity) as total_ordered from ordered_items GROUP BY ordered_items.product_name ORDER BY total_ordered

product_name	total_ordered
Washer	4
Skates	12
Baseball	13
Iron	37

Query 7

Retrieve the names of the customers who have no phone number.

SELECT customer_name from Orders Where customer_phone_number == "" group by customer_name

customer_name
Don Martin
Judy Roberts

Query 8

List the total amount for each order.

```
SELECT Ordered_Items.order_number, sum(product_quantity * unit_price) as
order_total from Products, Ordered_Items WHERE ordered_items.product_name =
Products.product_name GROUP BY Ordered_Items.order_number
```

order_number	order_total
1001	3497.45
1002	3491.45
1003	2065.6499999999996
1004	3189.45
1005	1013.2499999999999
1006	2710.45

Query 9

For each warehouse, retrieve the number of product types in it, and their average price. Show the warehouses in the descending order by the average price.

```
SELECT warehouse_number, avg(unit_price) as avg_price, count(product_name) as
count_of_products FROM Products WHERE 1 GROUP BY warehouse_number
ORDER BY avg_price DESC
```

warehouse_number	avg_price	count_of_products
3	402.99	1
1	314.95	2
2	122.94999999999999	2

Query 10.

For each warehouse that has more than one type of product, count the total number of product types whose prices exceed \$200.

```
SELECT warehouse_number, count(product_name) FROM Products WHERE
warehouse_number in (SELECT warehouse_number FROM products WHERE 1
```

GROUP BY warehouse_number HAVING count(product_name) >1) AND unit_price > 200 GROUP BY warehouse_number

warehouse_number	count(product_name)
1	2
2	1

Raw SQL:

```

DROP TABLE IF EXISTS "Ordered_Items";
CREATE TABLE "Ordered_Items" ("order_number" INTEGER,"product_name" Varchar,"product_quantity"
INTEGER);
INSERT INTO "Ordered_Items" VALUES(1001,'Iron',11);
INSERT INTO "Ordered_Items" VALUES(1002,'Iron',10);
INSERT INTO "Ordered_Items" VALUES(1002,'Washer',1);
INSERT INTO "Ordered_Items" VALUES(1003,'Skates',2);
INSERT INTO "Ordered_Items" VALUES(1003,'Iron',4);
INSERT INTO "Ordered_Items" VALUES(1003,'Washer',1);
INSERT INTO "Ordered_Items" VALUES(1004,'Skates',4);
INSERT INTO "Ordered_Items" VALUES(1004,'Iron',7);
INSERT INTO "Ordered_Items" VALUES(1005,'Iron',3);
INSERT INTO "Ordered_Items" VALUES(1005,'Baseball',12);
INSERT INTO "Ordered_Items" VALUES(1006,'Iron',2);
INSERT INTO "Ordered_Items" VALUES(1006,'Washer',2);
INSERT INTO "Ordered_Items" VALUES(1006,'Baseball','11');
INSERT INTO "Ordered_Items" VALUES(1006,'Skates',6);
DROP TABLE IF EXISTS "Orders";
CREATE TABLE "Orders" ("order_number" INTEGER PRIMARY KEY NOT NULL ,"order_date"
DATETIME,"customer_name" VARCHAR,"customer_address" VARCHAR,"customer_gender"
CHAR,"customer_phone_number" VARCHAR);
INSERT INTO "Orders" VALUES(1001,'11/03/2009','Sally Adams','481 Oak, Lan, MI','M','123-456-7891');
INSERT INTO "Orders" VALUES(1002,'11/03/2009','Don Charles',' ','M','222-333-4444');
INSERT INTO "Orders" VALUES(1003,'11/03/2009','Don Martin','419 Chip, Grant, NY','M','');
INSERT INTO "Orders" VALUES(1004,'11/04/2009','Sally Adams','481 Oak, Lan, MI','M','123-456-7891');
INSERT INTO "Orders" VALUES(1005,'11/04/2009','Don Martin','419 Chip, Grant, NY','M','');
INSERT INTO "Orders" VALUES(1006,'11/04/2009','Mary Nelson','108 Pine, Ada, Ca','F','555-310-0000');
INSERT INTO "Orders" VALUES(1007,'11/05/2009','Judy Roberts','512 Pine, Ada, Ca','F','');
DROP TABLE IF EXISTS "Products";
CREATE TABLE "Products" ("product_name" VARCHAR PRIMARY KEY NOT NULL , "units_in_stock"
INTEGER, "warehouse_number" INTEGER, "unit_price" DOUBLE);
INSERT INTO "Products" VALUES('Iron',104,1,317.95);
INSERT INTO "Products" VALUES('Stove',11,3,402.99);
INSERT INTO "Products" VALUES('Washer',52,1,311.95);
INSERT INTO "Products" VALUES('Skates',20,2,240.95);
INSERT INTO "Products" VALUES('Baseball',40,2,4.95);

```