**HW A - Will Eddy**

**Part 1: Single Table Queries**

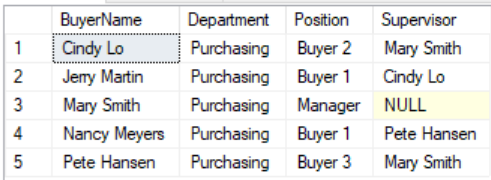
1. Write an SQL statement to show all the fields in the BUYER table. Sort by the Buyer Name and use the \* wildcard character.

**Code:**



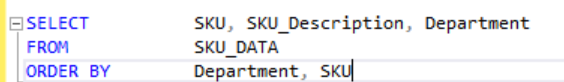
;

**Data:**

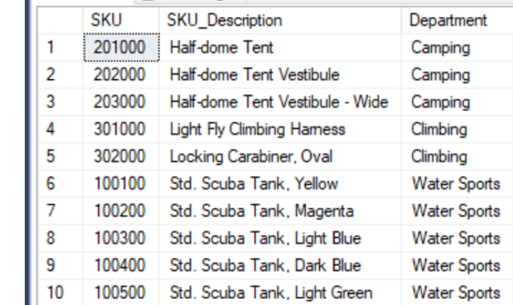


1. Write an SQL statement to show all the SKU, SKU\_Description, and Department from the SKU\_DATA table and sort by Department and then by SKU.

**Code:**

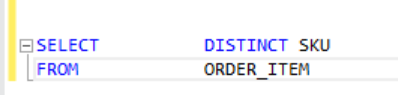
 ;

**Data:**

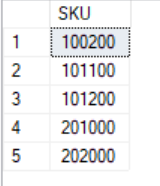


1. Write an SQL statement to show all the unique SKUs from the ORDER\_ITEM table.

**Code:**

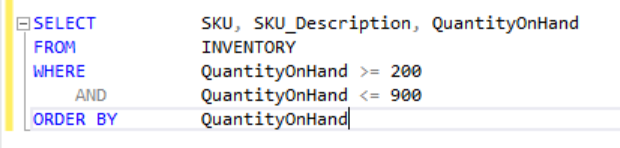
;

**Data:**

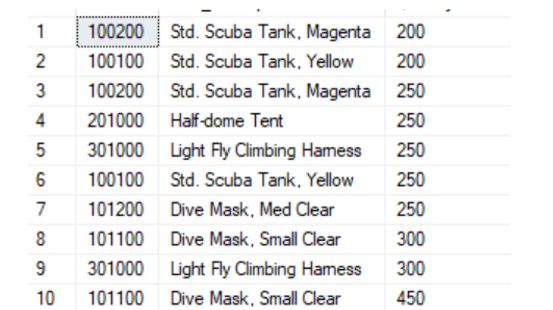


1. Write an SQL statement to show the SKU, SKU\_Description, and QuantityOnHand from the INVENTORY table for all QuantityOnHand greater than or equal to 200 and less than or equal to 900. Sort by quantity on hand and remember to show only the first ten records returned.

**Code:**

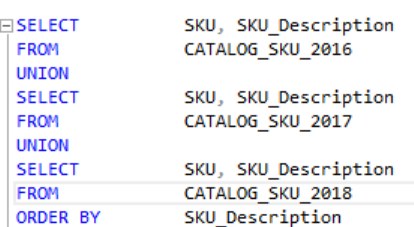
;

**Data:**



1. Write an SQL statement to show all items that were included in the 2016, 2017, and 2018 catalog. Show only the SKU and SKU-Description fields and sort by the SKU\_Description.

**Code:**

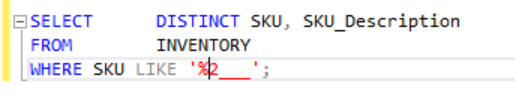
;

**Data:**

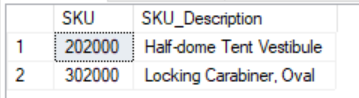


1. Write an SQL statement to show a unique SKU and SKU\_Description for all products having a ‘2’ in the fourth position from the right in SKU. Use the INVENTORY table.

**Code:**

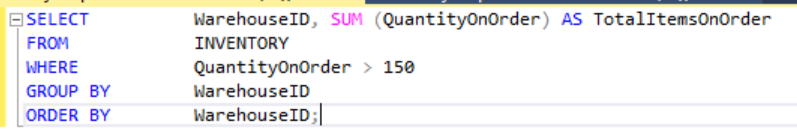


**Data:**

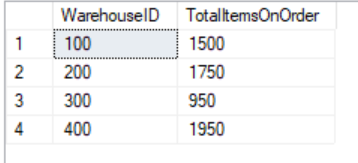


1. Write an SQL statement to display the WarehouseID and the sum of QuantityOnOrder, grouped by WarehouseID. Omit all SKU items that have less than 150 items on order, and name the sum TotalItemsOnOrder and sort by the Warehouse ID. Use the INVENTORY table.

**Code:**

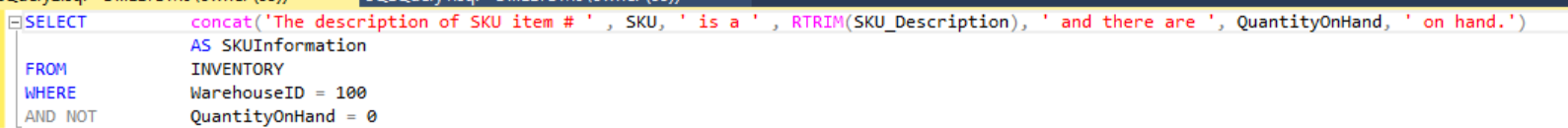


**Data:**

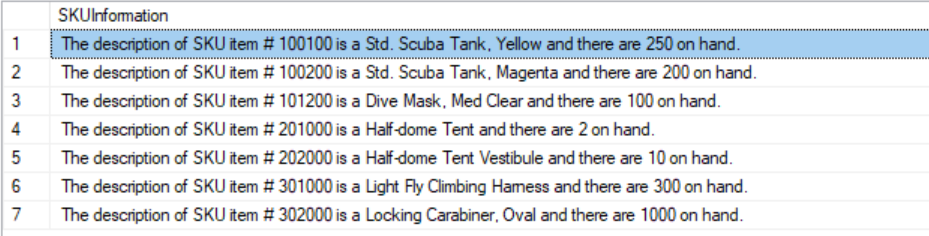


1. Write an SQL statement to produce a single column called SKUInformation that combines the SKU, SKU\_Description, and the QuantityOnHand for the Atlanta warehouse (WarehouseID = 100) together so that each row will look like “The description of SKU item # 201000 is a Half-dome Tent and there are 24 on hand. ” Do not show any quantities that are zero. Use the INVENTORY table and be sure to remove any unneeded spaces. Also make sure your results show all of the finished statement.

**Code:**

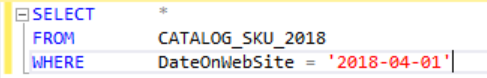


**Data:**

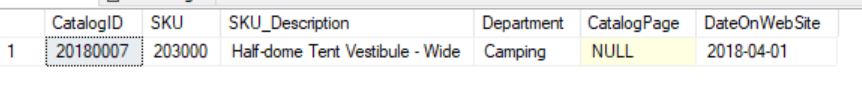


1. Write an SQL statement to show any items put on the catalog web site on April 1st of 2018. Include all fields.

**Code:**

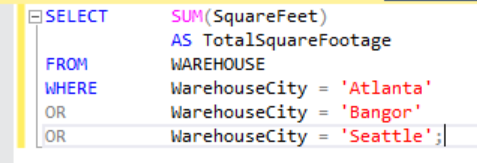


**Data:**

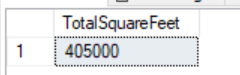


1. Write an SQL statement to display the total square footage for all the warehouses in Atlanta, Bangor, and Seattle. Name the field TotalSqFootage. Use the WAREHOUSE table.

**Code:**



**Data:**

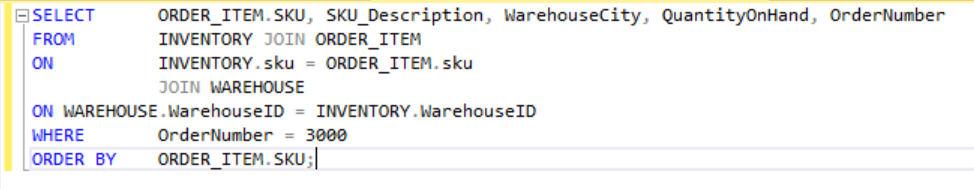


**Part 2: Multiple Table Queries**

**The following queries are based on the same Cape Codd Tables shown in Part 1:**

1. Write an SQL query to show the quantity on hand for all inventory items associated with Order Number 3000. Include the SKU, SKU\_Description, WarehouseCity, QuantityOnHand and OrderNumber. Sort by SKU. Use a Join.

**Code:**



**Data:**

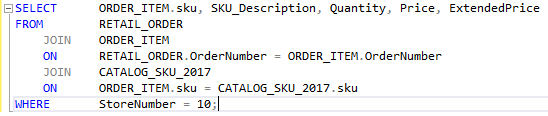


1. Explain why you cannot use a subquery in the query above:

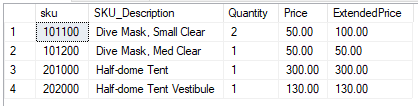
We need fields from tables other than the top-level table.

1. What product(s) did Store Number 10 order during December of 2017? Include SKU, SKU\_Description, Quantity, Price, and ExtendedPrice and do not show duplicates.

**Code:**

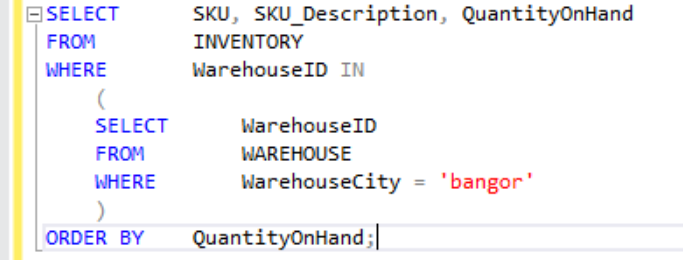


**Data:**



1. Show all the inventory available in the Bangor warehouse. Include the SKU, SKU\_Description, and QuantityOnHand. Use a subquery and sort by QuantityOnHand in descending order.

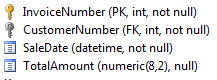
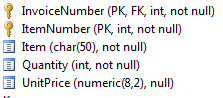
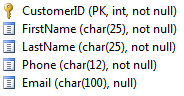
**Code:**



**Data:**



**The following queries are based on the Pirate Toy Store Tables:**

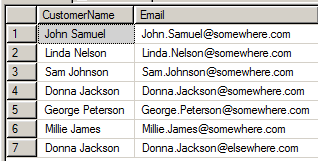
**CUSTOMER INVOICE INVOICE\_ITEM**

1. List the name (first and last strung together without extra spaces) & Email of all customers. Name the new field CustomerName.

**Code:**



**Data:**

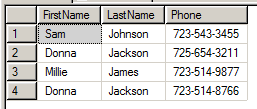


1. List the Phone, First Name, and Last Name for all customers whose last name starts with a 'J'.

**Code:**

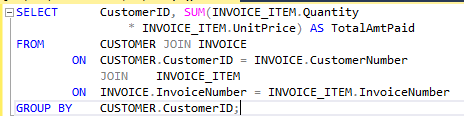


**Data:**

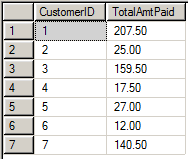


1. Show the Customer ID and the sum of the Extended Price (Quantity \* Unit Price) of all customers who have purchased more than 6 items of any specific product. Name the new field TotalAmtPaid

**Code:**

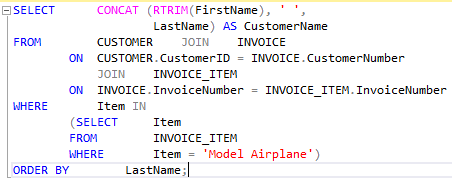


**Data:**

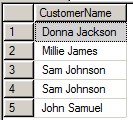


1. Show the First Name and Last Name (strung together in one field with no unneeded spaces and name it CustomerName of all customers who have had an order with an item named “Model Airplane”. Use a subquery. Present the results sorted by Last Name.

**Code:**

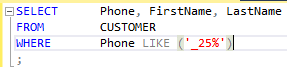


**Data:**



1. List the Phone, FirstName, and LastName for all customers whose second and third characters of the phone number area code is 25.

**Code:**



**Data:**

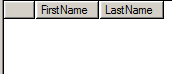


1. List only those customers who have the same name (First and Last).

**Code:**



**Data:**



1. Determine the average and highest TotalAmounts. Name these fields AvgTotalAmount and MaxTotalAmount respectively.

**Code:**



**Data:**



1. Count the total number of invoice items. Name this field TotalInvItems.

**Code:**



**Data:**



1. What is the total dollar amount of all invoice items sold? Name the field TotalSales.

**Code:**

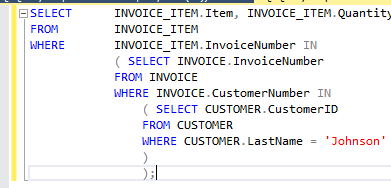


**Data:**

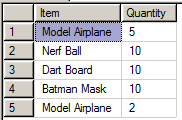


1. List all purchases by Sam Johnson to include the item and quantity purchased. Use a subquery.

**Code:**

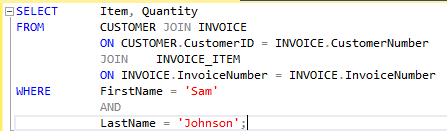


**Data:**



1. Create the same query above in #24 using a JOIN.

**Code:**



**Data:**

