**Lab 3 2 %**

**Due October 12, 2017 midnight via Blearn**

# Using ORACLE server 12c SQL\*Developer (client)

Use your Oracle USER ID on BTACS database.

**Introduction**

**Oracle supports two main types of built-in functions:**

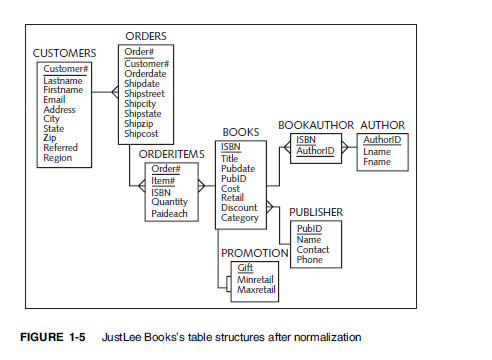
1. Single-row functions ( character functions, numeric functions, conversion functions, date functions, regular expressions functions)
2. Aggregate functions (**MIN, MAX, SUM, AVG, COUNT**, MEDIAN, STDDEV, VARIANCE .. and some more statistical functions)

Note: MIN, MAX, COUNT can be used with numbers, strings, and datetimes. The other functions are used with numbers. NULL values are ignored. DISTINCT keyword operates on distinct values.

We will use the Oracle 11g book’s script for the JustLee books company (see Lab 2). The following is the database model.

Materials:

* Database Systems text book **Chapter 6 on ISO SQL** (note the differences between ISO SQL and Oracle SQL)
* **Aggregate functions** and **GROUP BY… HAVING** clause are covered in Chapter 11 (Oracle 11g SQL book) \*\*\* skip GROUPING SETS, CUBE, and ROLLUP.



**In-class exercise**

* Count the number of orders with shipping state Washington (WA).

**SELECT COUNT (order#) "Number of Orders" FROM ORDERS WHERE UPPER (SHIPSTATE) = 'WA';**

* Count the total number of orders waiting for the shipment (NULL as shipping date).

**SELECT COUNT (order#) "Number of Orders Not shipped" FROM ORDERS WHERE SHIPDATE IS NULL;**

* List each shipping state and the total number of orders in each state. GROUP BY….

**SELECT SHIPSTATE, COUNT (\*) “ORDER COUNT” FROM ORDERS GROUP BY SHIPSTATE ORDER BY SHIPSTATE;**

* How many different states are listed in the orders table? DISTINCT

**SELECT COUNT (DISTINCT SHIPSTATE) "States" FROM ORDERS;**

**Individual work**

1. Calculate the total revenue from all ordered books. Note: ORDERITEMS table holds data about each Order Item (book ISBN), quantity, and the price paid for each book.

**SELECT SUM (QUANTITY\*PAIDEACH)"Total Revenue" FROM ORDERITEMS;**



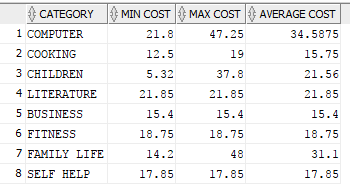
1. What is the average retail price (from BOOKS table) for the “computer” (category) books?

**SELECT AVG (RETAIL)"Average Retail Price" FROM BOOKS WHERE UPPER (CATEGORY) = 'COMPUTER';**



1. List all book categories from the books table. For each category list min, max, and average price (cost column in BOOKS).

**SELECT CATEGORY, MIN (COST)"MIN COST", MAX (COST)"MAX COST", AVG (COST)"AVERAGE COST" FROM BOOKS GROUP BY CATEGORY;**



1. List the publishing years (from BOOKS) and the number of books published in each year.

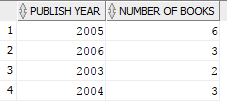
**SELECT**

**EXTRACT (YEAR FROM PUBDATE)"PUBLISH YEAR",**

**COUNT (ISBN)"NUMBER OF BOOKS"**

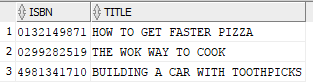
**FROM books**

**GROUP BY EXTRACT (YEAR FROM PUBDATE);**



1. How many different books have not been ordered?

**SELECT ISBN, TITLE FROM BOOKS WHERE ISBN NOT IN (SELECT DISTINCT ISBN FROM ORDERITEMS);**



**SELECT COUNT (ISBN) FROM BOOKS WHERE ISBN NOT IN (SELECT DISTINCT ISBN FROM ORDERITEMS);**



1. How many books (in total) authored by Jones Janice have been ordered?

**SELECT SUM (QUANTITY)"TOTAL BOOKS"**

**FROM BOOKS**

**JOIN BOOKAUTHOR ON BOOKS.ISBN = BOOKAUTHOR.ISBN**

**JOIN ORDERITEMS ON BOOKS.ISBN = BOOKAUTHOR.ISBN**

**WHERE AUTHORID = (SELECT authorid**

**FROM author**

**WHERE UPPER (LNAME) = 'JONES' AND UPPER (FNAME) = 'JANICE')**

**AND ORDERITEMS.ISBN = BOOKAUTHOR.ISBN;**



The above statement returns the sum of the total number of books ordered by Janice Jones including multiple quantities for the same book

The statement below returns the count of total titles that have been published by Janice Jones disregarding the quantity of each book that has been bought.

**SELECT COUNT (\*)"TOTAL BOOKS"**

**FROM BOOKS**

**JOIN BOOKAUTHOR ON BOOKS.ISBN = BOOKAUTHOR.ISBN**

**JOIN ORDERITEMS ON BOOKS.ISBN = BOOKAUTHOR.ISBN**

**JOIN AUTHOR ON BOOKAUTHOR.AUTHORID = AUTHOR.AUTHORID**

**WHERE UPPER (LNAME) = 'JONES' AND UPPER (FNAME) = 'JANICE'**

**AND ORDERITEMS.ISBN = BOOKAUTHOR.ISBN;**



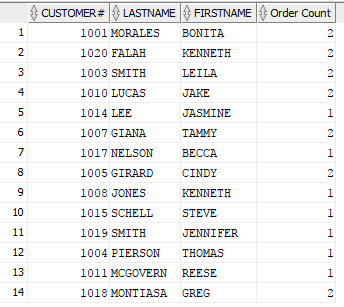
1. List the customers (customer id, last name, first name) and their total number of orders (include only customers who have at least one order).

**SELECT customers.customer#, lastname,firstname,COUNT(\*)"Order Count"**

**FROM**

**ORDERS JOIN CUSTOMERS ON ORDERS.CUSTOMER# = CUSTOMERS.CUSTOMER#**

**GROUP BY customers.customer#, customers.lastname,customers.firstname;**



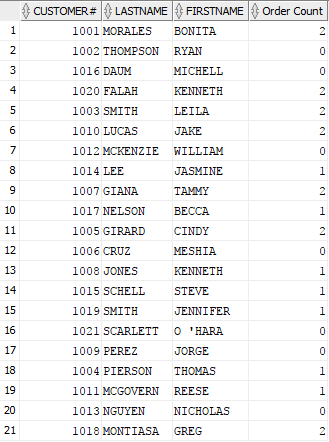
1. List the customer (customer id, last name, first name) and their total number of orders (list all customers even if they do not have orders).

**SELECT customers.customer#, lastname,firstname,COUNT(order#)"Order Count"**

**FROM**

**ORDERS RIGHT OUTER JOIN CUSTOMERS ON ORDERS.CUSTOMER# = CUSTOMERS.CUSTOMER#**

**GROUP BY CUSTOMERS.CUSTOMER#,customers.lastname,customers.firstname;**



1. List the customers (customer id, last name, first name) who have more than 1 order.

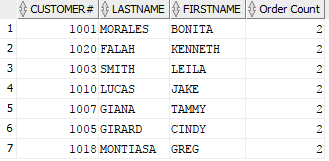
**SELECT customers.customer#, lastname,firstname,COUNT (order#)"Order Count"**

**FROM**

**ORDERS JOIN CUSTOMERS ON ORDERS.CUSTOMER# = CUSTOMERS.CUSTOMER#**

**GROUP BY CUSTOMERS.CUSTOMER#, customers.lastname,customers.firstname**

**HAVING COUNT (order#)>1;**



1. List the customers (customer id, last name, first name) who have ordered at least one “computer” book.

**SELECT CUSTOMERS.CUSTOMER#, CUSTOMERS.LASTNAME, CUSTOMERS.FIRSTNAME**

**FROM ORDERITEMS JOIN BOOKS ON ORDERITEMS.ISBN = BOOKS.ISBN**

**JOIN ORDERS ON ORDERS.ORDER# = ORDERITEMS.ORDER#**

**JOIN CUSTOMERS ON ORDERS.CUSTOMER# = CUSTOMERS.CUSTOMER#**

**WHERE UPPER (BOOKS.CATEGORY) = 'COMPUTER'**

**GROUP BY CUSTOMERS.CUSTOMER#, CUSTOMERS.LASTNAME, CUSTOMERS.FIRSTNAME**;

