**LECTURE 6** Rules of SQL formatting commands:You do not need to end a SQL command with a semicolon. When you finish entering the command, you can just press [Return]. If you wish, however, you can enter a semicolon at the end of a SQL command.You can continue a long SQL formatting command by typing a **hypen (-)** at the end of the line and pressing [Return]. Oracle SQL uses the column names (defined in a table) or expression names as default columnheadings when displaying query results. Some of the column names may not be descriptive enough and expressions can be cryptic. In addition to using the feature of **column aliases** in the SELECT clause, we can use the **COLUMN … HEADING** command in the following format: COLUMN column\_name HEADING column\_headingTo change a column heading to two or more words, enclose the new heading in single or double quotation marks.COLUMN shipdate HEADING 'Ship Date' To display a column heading on more than one line, use a vertical bar where you want to begin a new line. To change the character used to underline headings to a desired character (such as an equal sign), use the SET UNDERLINE command:SET UNDERLINE =SET UNDERLINE \*SET UNDERLINE ‘-‘The changed format models will stay in effect until we enter a new one, reset the column's format with COLUMN … CLEAR command, or exit from SQL\*Plus.COLUMN retail CLEARHow do you reset all column display attributes?CLEAR COLUMNS How do you remove the current break definition? CLEAR BREAKS How do you remove all COMPUTE definitions and the accompanying BREAK definition? CLEAR COMPUTES or CLEAR COMPUTE How do you temporarily suppress (but do not remove) column display attributes? COLUMN title OFF What command do you use to list all column display attributes? COLUMN What command do you use to list a specific column display attribute, such as 'title'? COLUMN title How do you 'copy' a column display attribute from the one that has been defined previously? COLUMN paideach FORMAT $999,999.99 COLUMN cost LIKE paideach By including the **break column** in your ORDER BY clause, you create meaningful subsets of records in your output. Which of the following is a correct way of establishing deptno as the break column? BREAK ON deptno SKIP 1 How do you insert two lines after every row? BREAK ON ROW SKIP 2 Which of the following statements is false? If you want to use different spacing techniques in one report or insert space after the value changes in more than one ordered column, you can specify multiple columns and actions in separate BREAK commands. Study the following command:BREAK ON deptno SKIP 2 ON jobtitle SKIP 1 Is it equivalent to the following two command lines? BREAK ON deptno SKIP 2 BREAK ON jobtitle SKIP 1 False How to separate subsets by page? BREAK ON deptno SKIP PAGE You can organize the rows of a report into subsets with the BREAK command, and in the meantime perform various computations (e.g., SUM or AVG) on the rows in each subset as summery lines by using the COMPUTE command. Which of the following is a correct way to use the BREAK and COMPUTE commands? BREAK ON deptno SKIP 1 COMPUTE SUM OF monthly\_salary ON deptno The following COMPUTE command has no effect. WHY? BREAK ON deptno SKIP 1 COMPUTE SUM OF monthly\_salary ON empno Needs to be deptno Which of the following COMPUTE has no effect? BREAK ON deptno SKIP 1 COMPUTE SUM OF monthly\_salary ON jobtitle COMPUTE SUM OF monthly\_salary ON empno Complete the following COMPUTE command line that computes the sum for the two columns, monthly\_salary and commission, with a customized label 'Totl BREAK ON deptno SKIP 1 COMPUTE SUM LABEL Total OF monthly\_salary commission ON deptno The compute label can be suppressed by using the NOPRINT option of the COLUMN command on the break column. What does the following command set do? COLUMN DUMMY NOPRINT BREAK ON DUMMY SKIP 1 COMPUTE SUM OF monthly\_salary ON DUMMY SELECT deptno DUMMY, deptno, jobtitle, lastname, monthly\_salary FROM emp ORDER BY deptno, jobtitle, monthly\_salary; Does not print the dummy column Complete the following command set to calculate and print summary lines based on all values in a report (it will calculate and print the grand total of salaries for all employees in the company): BREAK ON REPORT COMPUTE SUM OF monthly\_salary commission ON REPORT **Lecture 7** Complete the following command lines that define a top and bottom titles: **TTITLE** CENTER "MONTHLY SALARY SUMMARY REPORT" SKIP 4 **BTITLE** LEFT "COMPANY CONFIDENTIAL" What does the following command do? TTITLE CENTER "MONTHLY SALARY SUMMARY REPORT" SKIP 2 - COL 50 FORMAT 9 'Page:' SQL.PNO Modifies the report to center the top title, set the bottom title to the left and add a page number What do the following command sets and query do? COLUMN TODAY NEW\_VALUE TODAY\_DATE SELECT TO\_CHAR(SYSDATE, 'fmMonth DD, YYYY') TODAY FROM DUAL; Displays the current date in the title TTITLE CENTER 'MONTHLY SALARY SUMMARY REPORT' SKIP – LEFT TODAY\_DATE RIGHT 'Page:' FORMAT 999 SQL.PNO SKIP 2 CENTER 'Prepared by Diane Smith' SKIP 4 Adds a top title that is centered and puts everything in a separate line. BREAK ON deptno SKIP 2 COMPUTE AVG LABEL Avg SUM LABEL Sum OF monthly\_salary ON deptno Compute avg and sum SELECT deptno, jobtitle, lastname, monthly\_salary FROM emp ORDER BY deptno, jobtitle, monthly\_salary; query Complete the following commands to create a master/detail report that displays 3 different managers' employee numbers (7566, 7839, 9698), each at the top of a separate page, and the people reporting to the manager on the same page as the manager's employee number: COLUMN mgr\_id NEW\_VALUE MGRVAR NOPRINT TTITLE LEFT 'Manager: ' FORMAT 9999 MGVAR SKIP 2 BREAK ON mgr\_id SKIP PAGE BTITLE OFF **Lecture 8** In competition ranking ranking, items that compare equal (i.e., a tie) receive the same ranking number, and then a gap is left in the ranking numbers. In dense ranking ranking, items that compare equal receive the same ranking number, and the next item(s) receive the immediately following ranking number. The DENSE\_RANK() function leaves no gaps in ranking sequence when there are ties. The range of values returned by **PERCENT\_RANK()** is 0 to 1, inclusive Complete the following query that produces the output shown as below: SELECT o.order#, TO\_CHAR(SUM(paideach\*quantity), '$999.99') "Sales$", RANK() OVER (ORDER BY SUM(paideach\*quantity) DESC NULLS LAST) custom\_rank, **PERCENT\_RANK()** OVER (ORDER BY SUM(paideach\*quantity)) percent\_rank FROM customers c, books b, orders o, orderitems oi WHERE c.customer# = o.customer# AND o.order# = oi.order# AND oi.ISBN = b.ISBN HAVING SUM(paideach\*quantity) > 200 GROUP BY o.order# ORDER BY "Sales$" DESC; Complete the following query that produces the output shown as below:SELECT RANK(19350) WITHIN GROUP (ORDER BY monthly\_salary DESC) Rank\_$19350,

RANK(9450) WITHIN GROUP (ORDER BY monthly\_salary DESC) Rank\_$9450 FROM emp; The LAG and **LEAD** functions are useful for comparing values when the relative positions of rows can be known reliably. They work by specifying the count of rows which separate the target row from the current row. The LAG function provides access to a row at a given offset prior to the current position The LEAD function provides access to a row at a given offset after the current position. Complete the following query that lists the sales values of two rows prior to the current row: SELECT order#, SUM(paideach\*quantity) sales, LAG(SUM(paideach\*quantity),2) OVER (ORDER BY order#) lag\_sales, FROM orders JOIN orderitems USING(order#) GROUP BY order#; ROW\_NUMBER()

is an analytic function. It assigns a unique number to each row to which it is applied (either each row in the partition or each row returned by the query), in the ordered sequence of rows specified in the order\_by\_clause, beginning with 1. Complete the following query that produces the output shown as below: SELECT ROW\_NUMBER() OVER (ORDER BY monthly\_salary DESC) row#, RANK() OVER (ORDER BY monthly\_salary DESC) salary\_rank1, DENSE\_RANK() OVER (ORDER BY monthly\_salary DESC) salary\_rank2, monthly\_salary, empno, firstname || ' ' || lastname employee FROM emp; The ROWNUM function is a single-row functionthat returns a numeric value representing the order that a row is selected by Oracle from a table or joined tables. The first row has a ROWNUM of 1, the second has a ROWNUM of 2, and so on (based on how the data were entered into the table).Which of the following statements in NOT true? ROWNUM and ROW\_NUMBER are both an analytic function. Which of the following queries is invalid? SELECT ROW\_NUMBER(), title, cost, retail FROM books; How the ROWNUM values are determined when the output is produced by joining two tables? They will be given based on the sequence of the data entered in the CHILD table Conditions testing for ROWNUM values greater than a positive integer (e.g., WHERE ROWNUM > 10) are always false Which of the following queries will return a result instead of a message of "no rows selected"? SELECT ROWNUM, lastname, jobtitle FROM emp WHERE ROWNUM > 5; SELECT ROWNUM, lastname, jobtitle FROM emp WHERE ROWNUM >5 AND ROWNUM <10; SELECT ROWNUM, lastname, jobtitle FROM emp WHERE ROWNUM BETWEEN 5 AND 10; What does the following query do? SELECT \* FROM (SELECT \* FROM customers ORDER BY customer# ASC) cust\_3 WHERE ROWNUM <= 3; Retrieves the first 3 records from the customers table, sorted by customer# in ascending order Which of the following queries will list the top three most expensive books? SELECT ROWNUM top\_3, title, cost FROM (SELECT title, cost FROM books ORDER BY cost DESC) WHERE ROWNUM <=3; Which of the two queries are equivalent to each other? In other words, they produce the same result. SELECT rank\_cost top\_3, title, cost FROM (SELECT title, cost, ROW\_NUMBER() OVER (ORDER BY cost DESC) rank\_cost FROM books) WHERE rank\_cost <= 3; SELECT ROWNUM top\_3, title, cost FROM (SELECT title, cost FROM books ORDER BY cost DESC)

WHERE ROWNUM <=3; The following two queries are invalid? How do you fix them? SELECT rank\_cost top\_3, title, cost FROM (SELECT title, cost, ROWNUM OVER (ORDER BY cost DESC) rank\_cost FROM books) WHERE rank\_cost <=3; ROWNUM needs to be ROW\_NUMBER() SELECT ROW\_NUMBER() top\_3, title, cost FROM (SELECT title, cost FROM books ORDER BY cost DESC) WHERE ROW\_NUMBER() <=3;ROW\_NUMBER needs to be ROWNUM Complete the following two queries that retrieve the FIVE most recently placed orders:SELECT ROWNUM, order#, shipdate FROM (SELECT order#, shipdate FROM orders ORDER BY orderdate DESC) order\_last5 WHERE ROWNUM <= 5; SELECT rank top\_5, order#, shipdate FROM (SELECT order#, shipdate, ROW\_NUMBER()OVER (ORDER BY orderdate DESC) rank FROM orders) WHERE rank <=5; Which of the following queries finds out the five least expensive books (in cost)? SELECT ROWNUM bottom\_5, title, cost FROM (SELECT title, cost FROM books ORDER BY cost) WHERE ROWNUM <=5; Other questions: How do you retrieve the FIVE most recently shipped orders? SELECT ROWNUM bottom\_5, title, cost FROM (SELECT title, cost FROM books ORDER BY cost DESC) WHERE ROWNUM <=5; How about the top three most profitable orders in terms of sales? SELECT rank\_cost top\_3, title, cost FROM (SELECT title, cost, ROW\_NUMBER() OVER (ORDER BY cost DESC) rank\_cost FROM books)WHERE rank\_cost <=3; OR SELECT ROWNUM top\_3, title, cost FROM (SELECT title, cost FROM books ORDER BY cost DESC) WHERE ROWNUM <=3; Which of the following queries identifies the top five salary earners? SELECT R, lastname, monthly\_salary

FROM (SELECT lastname, monthly\_salary, ROW\_NUMBER() OVER (ORDER BY monthly\_salary DESC) R FROM emp) WHERE R BETWEEN 1 and 5; Complete the following inner-N query that selects all rows from the employees table but returns only the fifth through tenth rows (ranking from highest to lowest pays:SELECT R rank, lastname, monthly\_salary FROM (SELECT lastname, monthly\_salary, row\_NUMBER() OVER(ORDER BY monthly\_salary DESC) R FROM emp) WHERE R BETWEEN 5 and 10; **Lecture 9** Univariate analysis involves the examination across cases of one variable at a time Bivariate analysis: When a sample consists of two or more variables, descriptive statistics may be used to describe the relationship between pairs of variables In statistics, dispersion (also called variability, scatter, or spread) is the extent to which a distribution is stretched or squeezed. Common examples of measures of statistical dispersion are the variance, standard deviation, percentile, range, etc. In statistics and probability, quantiles are considered the "cut-points" that divide the range of a probability distribution into contiguous intervals with equal probabilities. (In other words, quantiles are the cut-points for equally dividing the observations in a sample.) There should be one less quantile than the number of groups created. For instance, quartiles are the three cut points that will divide a dataset into four equal-size groups. In descriptive statistics, the quantiles  of a ranked set of data values are the three points that divide the data set into four equal groups, each group comprising a quarter of the data. In other words, the value of the boundary at the 25th, 50th, or 75th percentiles of a frequency distribution divided into four parts, each containing a quarter of the population. With quartile, Q2 is the median of all data. In descriptive statistics, which of the following provides a measure of central tendency for the data? Mean The median of a sample will always equal the 50th percentile The 75th percentile is referred to as the third quartile The difference between the largest (i.e. max) and the smallest (i.e., min) data values is called the range Which of the following is measure of dispersion? Variance The most frequently occurring value of a data set is called the mode In statistics and probability, quantiles are considered the "cutpoints" that divide the range of a probability distribution into contiguous intervals with equal probabilities. TERTILE is also called 3-quantile where any of the two points that divide an ordered distribution into three parts, each containing a third of the population.Variance is a measure of variability based on the deviation about the mean In descriptive statistics, a decile is any of the nine values that divide the sorted data into ten equal parts, so that each part represents 1/10 of the sample or population. In Oracle, the NTILE function allows easy calculation of tertiles, quartiles, deciles and other common summary statistics. It is an analytic function. It divides an ordered data set into a number of buckets indicated by expr and assigns the appropriate bucket number to each row. Which of the queries is VALID? SELECT NTILE(3) OVER (ORDER BY monthly\_salary DESC) AS tertile,lastname, monthly\_salary FROM emp;

The MEDIAN function is a specific case of PERCENTILE\_CONT where the percentile value defaults to 0.5. Which of the following statements is false? PERCENTILE\_CONT and PERCENTILE\_DISC always return different results. SELECT deptno, PERCENTILE\_CONT(0.5) WITHIN GROUP (ORDER BY monthly\_salary DESC) FROM emp GROUP BY deptno;In the query, PERCENTILE\_CONT is used as a aggregate function.Which of the following queries is VALID?

SELECT jobtitle, PERCENTILE\_CONT(0.5) WITHIN GROUP (ORDER BY monthly\_salary DESC)FROM emp GROUP BY jobtitle; Complete the query producing the following output:BREAK ON deptno SKIP 1 SELECT deptno, lastname, monthly\_salary, PERCENTILE\_CONT(0.5) WITHIN GROUP (ORDER BY monthly\_salary DESC) OVER (PARTITION BY deptno) "Median\_Cont" FROM emp WHERE deptno IN (30, 40); VARIANCE returns the variance of expr. You can use it as an **aggregate** or **analytic** function.True In the following query, the VARIANCE function is used as an aggregate function.SELECT category, VARIANCE(cost)

FROM books GROUP BY category ORDER BY category; SELECT title, cost, pubdate, VARIANCE(cost) OVER (ORDER BY pubdate) var

FROM books WHERE category = 'COMPUTER'; In the following query, the VARIANCE function is used as an analytic function.The Standard deviation is a measure of how spread out numbers are. The standard deviation is the square root of the variance We can use STDDEV as an aggregate or analytic function **LECTURE 10** Bivariate analysis is considered multiple-dimensional analytics. With bivariate analysis, it involves a sample consisting of two or more variables. Descriptive statistics can be used to describe the relationship between different pairs of variables We use the term **dimension** to mean any **category** (e.g., book categories, gender, academic majors, years, departments) used in specifying questions.The values associated with a particular set of dimension values are usually referred to as facts. The facts might be sales in units or local currency, profits, customer counts, or anything else worth tracking. Crosstab sort and group data and produce a neat row by column table with a calculation for each row and column intersect. You then choose at least one column for the Row heading, another for the COLUMN HEADING and a third for AGGREGATE VALUE (SUM, AVG, COUNT, etc.) In data processing, **a crosstab table** is a useful type of table for analyzing two data variables Study the following query output: Which is the column heading? Product Category Bikes Which is the row heading? Country How many row headings to be defined? What are they? 4, State, customer, totalorderqty, totalsales How many column headings to be defined? What are they? 1 product category: clothing Complete the query producing the following output: SELECT category, COUNT(isbn) count\_books, SUM(CASE WHEN pubid = 1 THEN 1 END) AS pub1,SUM(CASE WHEN pubid = 2 THEN 1 END) AS pub2,SUM(CASE WHEN pubid = 3 THEN 1 END) AS pub3,SUM(CASE WHEN pubid = 4 THEN 1 END) AS pub4, SUM(CASE WHEN pubid = 5 THEN 1 END) AS pub5 FROM publisherS JOIN books USING(pubid) GROUP BY category ORDER BY category; Which of the following statements is NOT true with regard to a view? It is a table that actually contains data. What are the purposes of using a view? Reuse, simplify and secure Which of the following SQL statements will produce the following output? SELECT SUM(DECODE(pubid, 1, 1)) AS pub1\_total, SUM(DECODE(pubid, 2, 1)) AS pub2\_total,SUM(DECODE(pubid, 3, 1)) AS pub3\_total, SUM(DECODE(pubid, 4, 1)) AS pub4\_total, SUM(DECODE(pubid, 5, 1)) AS pub5\_total, SUM(DECODE(pubid, 1,1,2,1,3,1,4,1,5,1)) AS grandtotal

FROM publishers JOIN books USING(pubid); Complete the following SQL statement to product the result as shown below: BREAK ON name SKIP 1

SELECT name, category, COUNT(isbn) "# of BOOKS", TO\_CHAR(AVG(retail), '$999.99') "AVG RETAIL" FROM publishers JOIN books USING(pubid)

WHERE pubid IN (2,3,5) GROUP BY (name, category) ORDER BY name, category; Complete the following SQL statement to product the result as shown below:

SELECT category, count(isbn) "# of BOOKS", SUM(CASE WHEN pubid = 2 THEN 1 ELSE 0 END) AS pub2, SUM(CASE WHEN pubid = 3 THEN 1 ELSE 0 END) AS pub3, SUM(CASE WHEN pubid = 5 THEN 1 ELSE 0 END) AS pub5 FROM publishers JOIN books USING(pubid) WHERE pubid IN (2,3,5) GROUP BY category

ORDER BY category; What does the following query do? SELECT name, category, COUNT(isbn) "# of BOOKS", AVG(retail) avg\_retail FROM publishers JOIN books USING(pubid) GROUP BY GROUPING SETS (name, category, ()) ORDER BY name, category; Groups the data first by name and then by category Is the foregoing query (in Question 13) equivalent to the following query? SELECT name, category, COUNT(isbn) "# of BOOKS", TO\_CHAR(AVG(retail), '$999.99') "AVG RETAIL" FROM publisher JOIN books USING(pubid) GROUP BY CUBE(name, category) ORDER BY name, category; Yes