Coalesce

- Coalesce takes two or more expressions and returns the first expression that is not-null from the list of expressions
- Ordering is important
- Only if all expressions are null then is the result null
- The datatypes cannot be dynamic

COALESCE(EXPRESSION_1, EXPRESSION_2...EXPRESSION_N)

NVL

 The NVL function replaces null values from an expression with a specified string. It takes 2 parameters, the expression and replacement string

- If the expression in the first parameter is a character then it converts the replacement string to a character
- If it is numeric then Oracle determines which argument has the highest numeric precedence, implicitly converts the other argument to that datatype, and returns that datatype.

NVL(EXPRESSION, REPLACEMENT_STRING)

GREATEST / LEAST

The greatest function returns the greatest value in a list of expressions

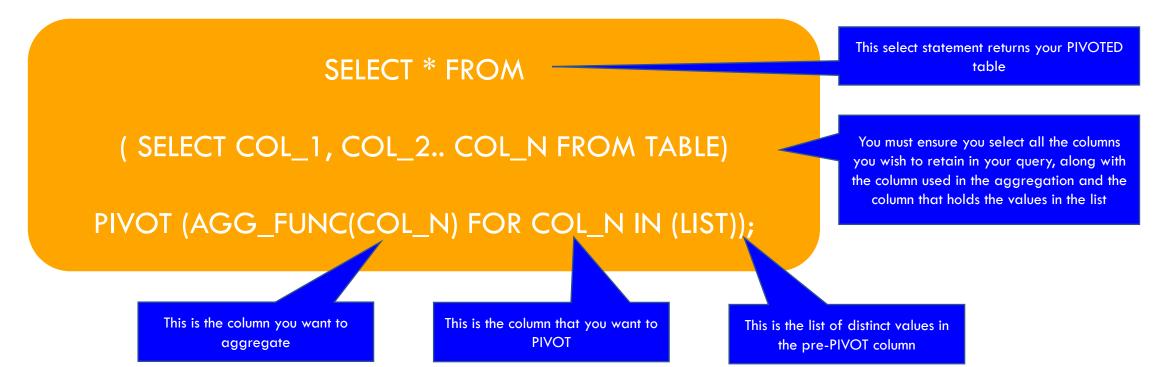
- The least function returns the lowest value in a list of expressions
- The first expression is used to determine the data type

GREATEST(EXPRESSION_1, EXPRESSION_2...EXPRESSION_N)

LEAST(EXPRESSION_1, EXPRESSION_2...EXPRESSION_N)

PIVOT

- PIVOT allows you to rotate rows into columns in a table
- You cannot input a subquery to obtain your list with PIVOT



TRIM

 TRIM enables you to trim leading or trailing characters (or both) from a character string

TRIM([LEADING | TRAILING | BOTH] trim_character FROM trim_source)

- If trim_character or trim_source is a character literal, then you must enclose it in single quotes
- The default trim_character is a whitespace. By default BOTH leading and trailing trim_characters are removed from the trim_source

LPAD, RPAD

- LPAD function pads the left side on an expression with a specific set of characters until it reaches its padded length
- RPAD function pads the right side on an expression with a specific set of characters until it reaches its padded length

LPAD(EXPRESSION, PADDED_LENGTH, PAD_CHARACTER)

RPAD(EXPRESSION, PADDED_LENGTH, PAD_CHARACTER)

UNPIVOT

UNPIVOT transforms columns to rows, which is the opposite of PIVOT

This select statement returns your UNPIVOTED table

SELECT * FROM TABLE

UNPIVOT (MEASURE_COL FOR NEW_COL_NAME IN (PIVOT_COL_LIST));

This will be a new column that holds the measure values

This will be a new column that holds the existing column names as records

This is a list of the PIVOTED columns that need to be UNPIVOTED