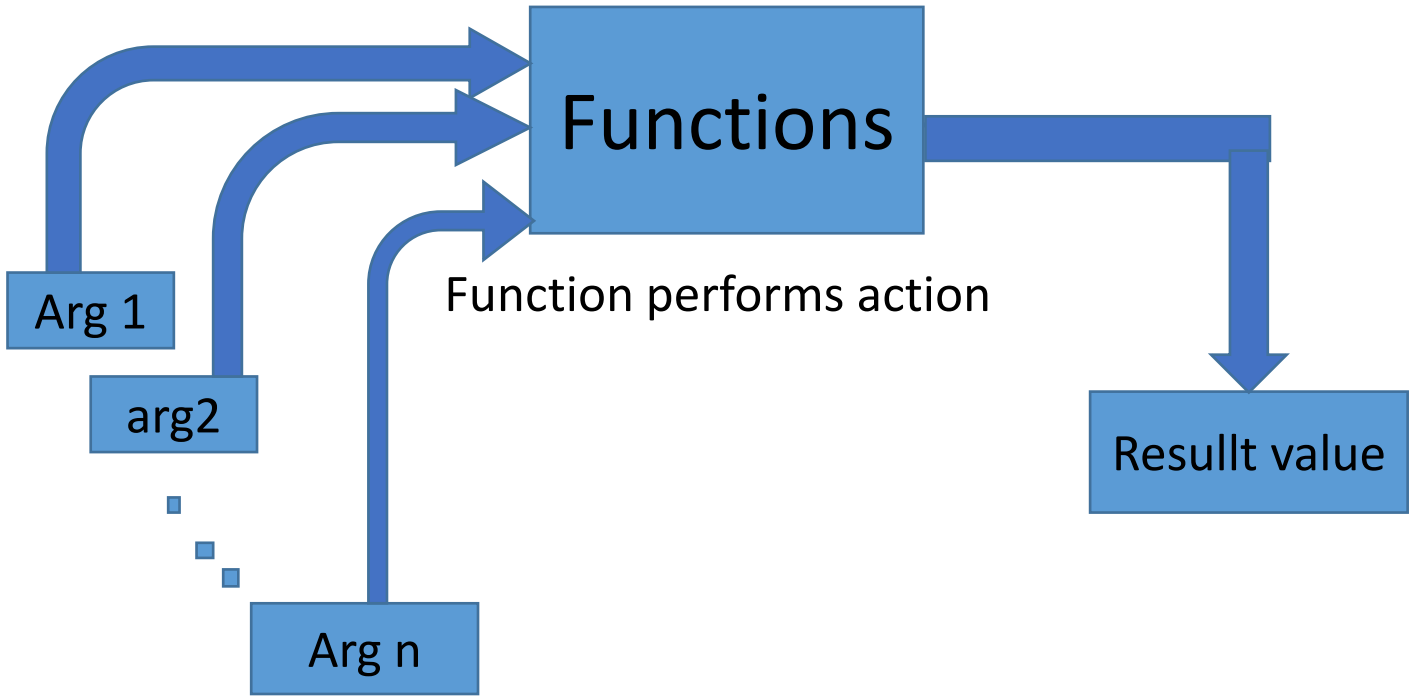
SQL FUNCTIONS

Functions

- Functions are very powerful feature of SQL used to manipulate data items .

- SQL functions are built into oracle database and are operated for use in various appropriate SQL statements.

SQL FUNCTION Diagrammatic Representation

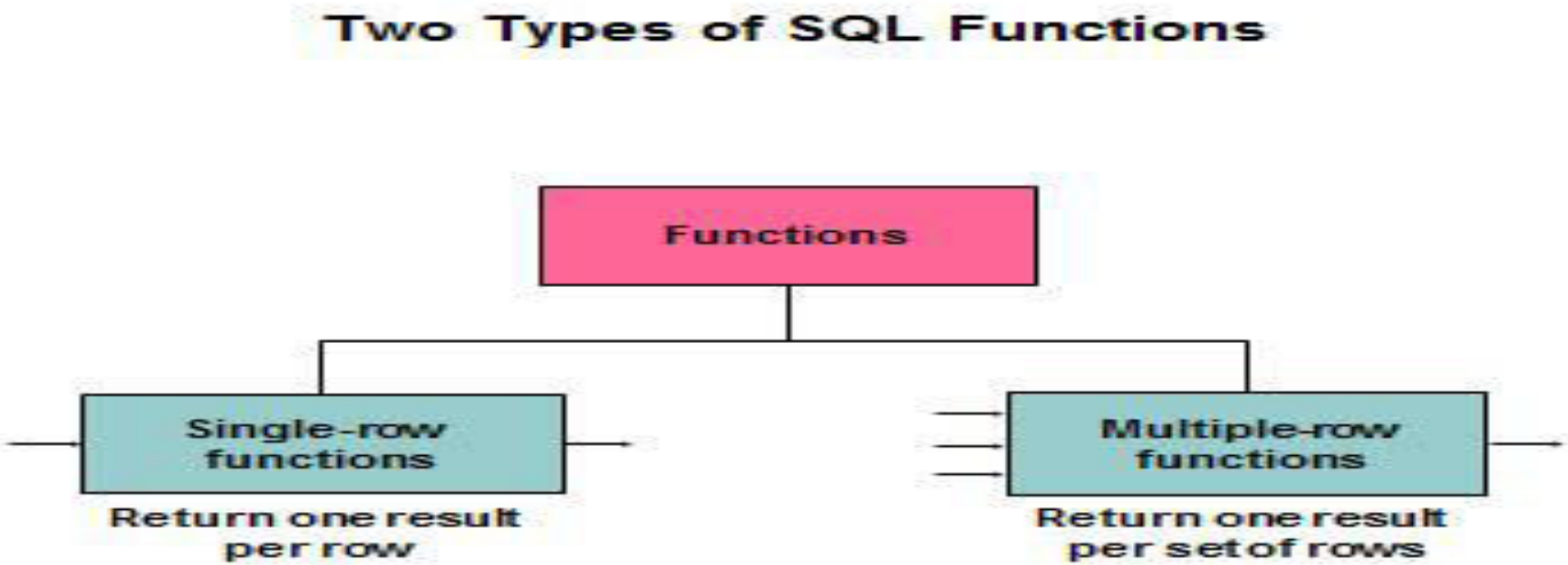


Advantages of function

1. Function can be used to perform complex calculations on data.
2. Functions can modify individual data items
3. Function can very easily manipulate output for groups of rows.
4. Function can manipulate character as well as numeric type of data.
5. function can alter date formats for display

TYPES OF FUNCTION

- There are two types of function:

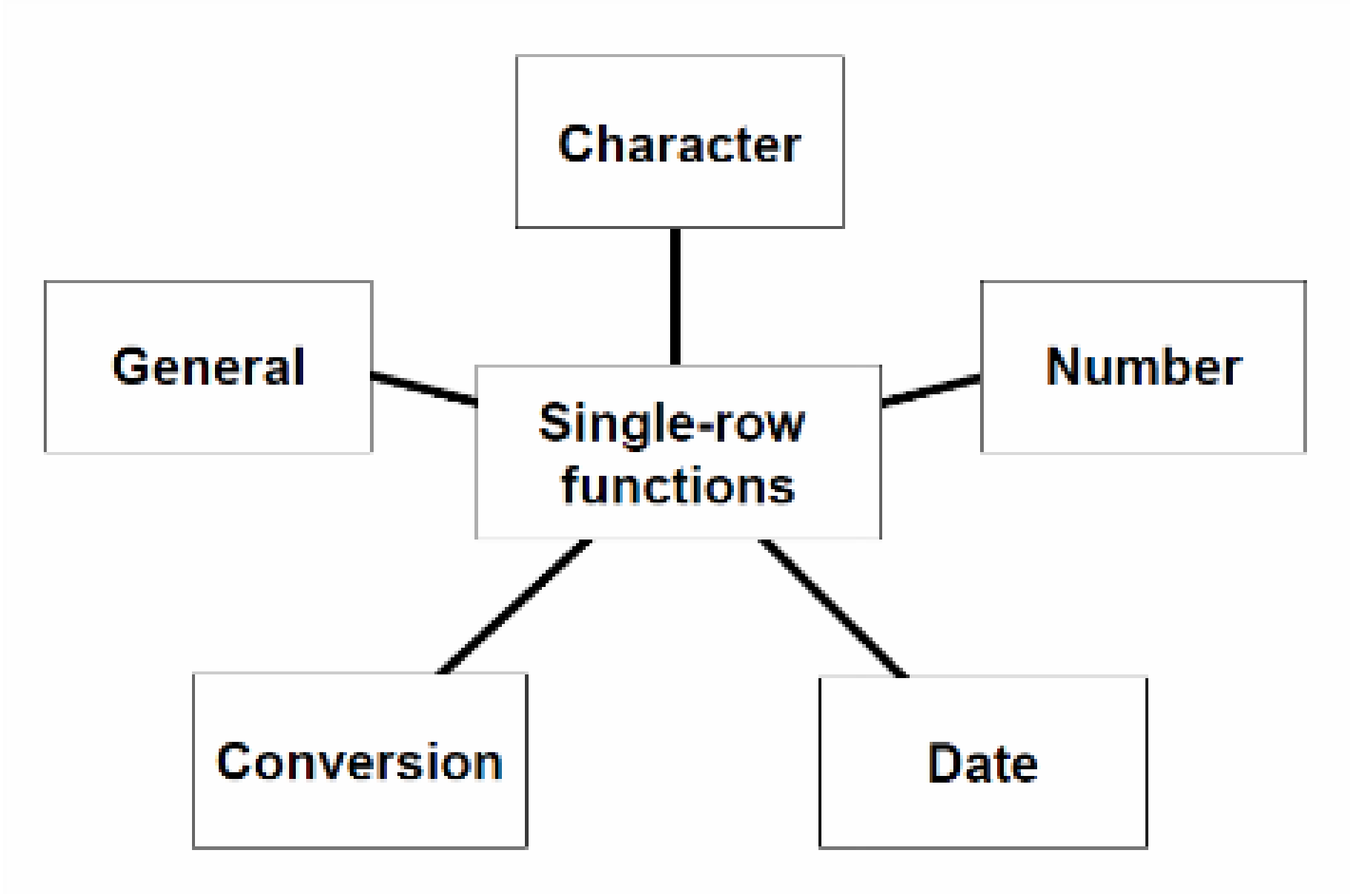


Single row function

- These function operate on single rows only and return one value for ach row, column name or an expression. Single-row functions can be used in SELECT. WHERE and ORDER by clauses.
- **Syntax of using a single-row function is function_name [(arg1, arg2,…..)]**
- Where, **function_name** is the name of the function. **arg1**,**arg2** is any argument to be used by the function. This can be represented by a user-supplied constant value, variable value, column name or an expression.

Types of single row functions

- Character functions
- arithmetic functions
- Date functions
- conversion functions
- General functions
- Aggregate functions


Character
General
Single-row functions
Number
Conversion
Date

String/character function

- 1.**LOWER**:- returns char, with all letters in lowercase.

  Syntax:-lower(char)

  e.g. select lower('IVAN BAYROSS') from dual;

  Output=ivan bayross

2.**INITCAP**:- returns a string with the first letter of each word in upper case.

Syntax:- initcap(char)

e.g. select initcap('IVAN BAYROSS') from dual;

Output=Ivan Bayross

- **3.UPPER**:- returns char, with all letters in uppercase.

  syntax:- upper(char)

  e.g. select upper('ivan bayross') from dual;

  Output= IVAN BAYROSS

4.SUBSTR:-returns a portion of characters beginning at character m, and going up to character n. if n is omitted the result returned is up to the last character in the string. The first position of char is 1.

Syntax:- substr(<string>,<start_position>,[<length>])

- Where string is source string
- start_position is the position for extraction. The first position in the string is always 1.
- Length is the number of character is extract.

e.g. select substr(“secure”,3,4) from dual;

Output= cure

5.**ASCII**:-returns the number code that represents the specified character. If more than one character is entered, the function will return the value for the first character and ignore all the characters after the first.

syntax:-ascii(character)

e.g. select ascii(‘a’) from dual;

ouput= 97

6.LENGTH:- returns a length of a word.

Syntax:- length(word)

e.g. select length(‘sharanam’) from dual;

Output= 8

Continue;

NUMERIC FUNCTIONS.....

- 1. **ABS**:- returns the absolute value of ‘n’.

  syntax:- ABS(-15)

  e.g. Select ABS(-15) from
  dual;

- 2.**POWER**:- returns m raised to the nth power.
  n must be an integer else an error is returned.

  syntax:-power(m,n)

  e.g. Select power(3,2) from
  dual;

- 3.**Round**:-returns n, rounded to m places to the right of the decimal point.
- m must be an integer

syntax:-round(n,[m])

e.g. select round(15.91,1) from dual;

output=15.2

4.**SQRT**:- returns square root of n.

syntax:-sqrt(n)

e.g. select sqrt(25) from dual;

output=5

- **5.EXP**:-returns e raised t the nth power where e=2.71828183

  syntax:- exp(n)

  E.g. select exp(5) from dual;

  Output=148.413159

- **6.EXTRACT**:-returns a value extracted from a date or an integer value. A date can be used only to extract year, month and day, while a timestamp with a time zone data type can be used only to extract timezone_hour and timezone_minute.

  E.g. select extract(year from date '2004-07-02')"year", extract(month from sysdate)"month" from dual;

  Output=2004 7

- **9.MOD** :-returns the remainder of a first number divided by second number passed a parameter. If the second number is zero the result of the same as the first number

  Syntax:-mod(m,n)

  e.g. select mod(15,7) from dual;

  Output= 1

11. FLOOR:- return a largest integer value that is equal to less than a number.

Syntax:-floor(n)

e.g. select floor(24.8) from dual;

Output=24

12.CEIL:-return the smallest integer value that is greater than or equal to a number.

Syntax:-ceil(n)

e.g. select ceil(24.8) from dual;

Output= 25