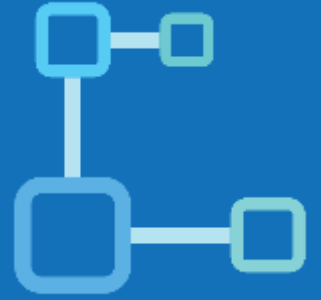




[www.netlink.com](http://www.netlink.com)

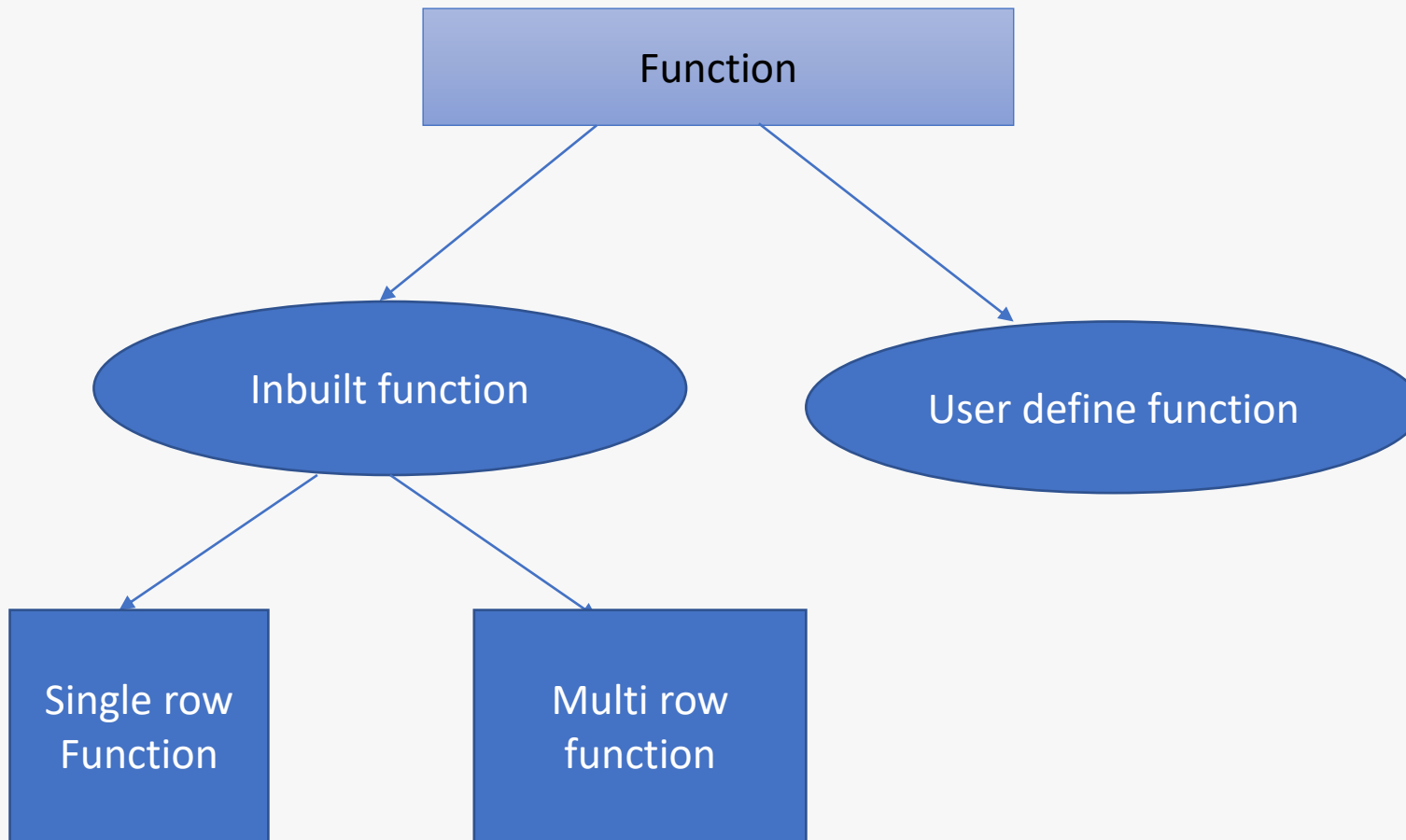


# DATABASE



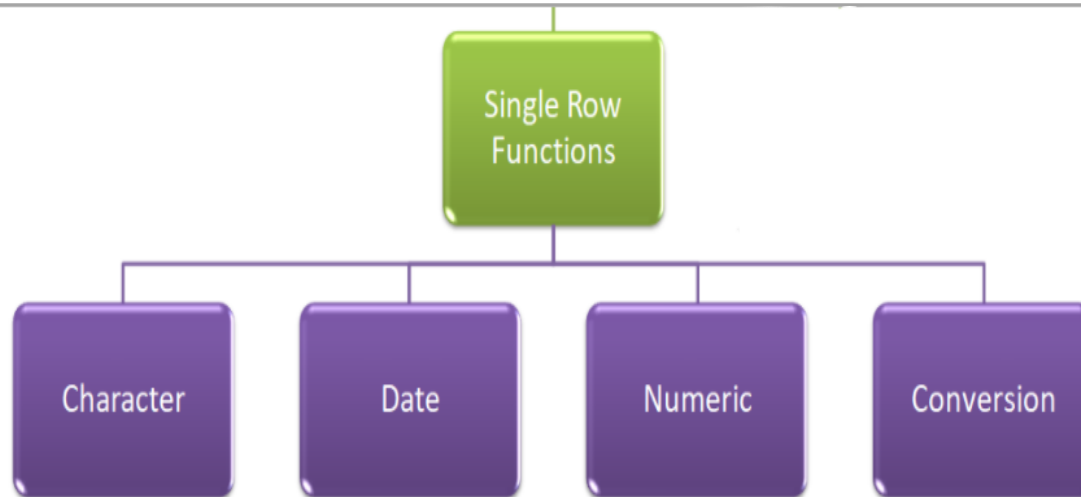
# Function

Function is a set of instruction on block of code which is used to perform some specific task



# Single Row Function

- ✓ **Single row function in SQL are the ones who work on a single row and return one output per row.**
- ✓ **Single row function execute row by row.**
- ✓ **Single row function in SQL can be character, numeric, date, and conversion functions. these functions are used to modify data items. These functions need one or more input and operate on each row, thereby returning one output value for each row.**



## Types of Single row function In SQL

- **Character Function**
- **General Function**
- **Case conversion Function**
- **Date Function**
- **Number Function**

# Single Line function

✓ **General functions** - Usually contain NULL handling functions. The functions under the categories COALESCE.

✓ **Case Conversion functions** - Accepts character input and returns a character value. Functions under the category are UPPER, LOWER and INITCAP.

- ❖ UPPER function converts a string to upper case.
- ❖ LOWER function converts a string to lower case.
- ❖ INITCAP function converts only the initial alphabets of a string to upper case.

✓ **Character functions** - Accepts character input and returns number or character value. Functions under the category are CONCAT, LENGTH, SUBSTR, INSTR, LPAD, RPAD, TRIM and REPLACE.

✓ **Date functions** - Date arithmetic operations return date or numeric values. Functions under the category are

- ❖ CURRENT\_DATE
- ❖ CURRENT\_TIME ,
- ❖ LOCALTIME,
- ❖ LOCALTIMESTAMP
- ❖ age('09-APR-1996', now())
- ❖ clock\_timestamp();
- ❖ select now();

✓ **Number functions** - Accepts numeric input and returns numeric values. Functions under the category are ROUND, TRUNC, and MOD.

# Multi Row Function

1. **Max()**----→ **Max (Column\_name /Expression)**
2. **Min()**----→ **Min(Column\_name /Expression )**
3. **Sum()**----→ **Sum (Column\_name/Expression)**
4. **Aug()**----→ **Avg (Column\_name/Expression)**
5. **Count**---→  
**Count(\*)/Column\_name/Expression)**

- ✓ **Multi-Row Function execute group by group**
- ✓ **We can use multi row function in select Clause**
- ✓ **We can not use multi row function in where clause**
- ✓ **Multi row Function can accept single argument column\_name /Expression.**
- ✓ **Multi row function ignore full values**
- ✓ **We can not use next multi row function  
select max(min(sal)) this is not possible**

# User-define Function

## Introduction to Create Function statement

PostgreSQL **functions**, also known as Stored Procedures,

Syntax :

```
create [or replace] function
function_name(param_list)
  returns return_type
  language plpgsql
as
$$
declare
-- variable declaration
begin
-- logic
end;
$$
```

- First, specify the name of the function after the create function keywords. If you want to replace the existing function, you can use the or replace keywords.
- Then, specify the function parameter list surrounded by parentheses after the function name. A function can have zero or many parameters.
- Next, specify the datatype of the returned value after the returns keyword.
- After that, use the language plpgsql to specify the procedural language of the function. Note that PostgreSQL supports many procedural languages, not just plpgsql.

# Example

Question : Check max salary of all employee

```
create or replace function check_min_sal(min_sal int)  
returns int  
language plpgsql  
as  
$$  
declare  
copies int;  
begin  
select count(*) into copies from emp where sal >  
min_sal;  
return copies;  
end;  
$$
```

Question: insert data in table

```
create or replace function insertData(empno int,  
ename varchar(10),  
job varchar(9),  
hiredate date,  
dept int)  
  
returns varchar  
language plpgsql  
as  
$$  
declare  
-- copies int;  
begin  
insert into  
public.emp(empno,ename,job,hiredate,dept)  
values(empno,ename,job,hiredate,dept);  
return 'inserted';  
end;  
$$
```



[www.netlink.com](http://www.netlink.com)

*Thank you*