PostgreSql迁移：

**DBUtil类：**

    isOracle()，判断是否是Oracle数据库；isPostgreSql()，判断是否是PostgreSql数据库

    pagination(String sql, int startL, int limitL)，公共分页方法，将sql包装成分页的sql

**创建类似oralce 的 dual视图**

     create or replace view dual as select null::unknown as  unknown;

**类型变化**

     VARCHAR2 → varchar

     Number → Numeric

     DATE → timestamp

**分页**

     limit n  offset s

     rownum     row\_number() over() 但不能用在where中

**系统时间**

     localtimestamp

     now()

**中文排序**

     nlssort(a.text ,'NLS\_SORT=SCHINESE\_STROKE\_M') → convert\_to(a.text ,'GBK')

**时间增加小时**

     localtimestamp - interval '1 hour'

     localtimestamp - interval '2 hours'

**时间相减计算分钟**

     round((EXTRACT(EPOCH from CURRENT\_TIMESTAMP) - EXTRACT(EPOCH from ag.last\_heartbeat\_time)) / 60)

**函数替换：**

nvl

     coalesce

instr函数替换

--  
-- instr functions that mimic Oracle's counterpart  
-- Syntax: instr(string1, string2, [n], [m]) where [] denotes optional parameters.  
--  
-- Searches string1 beginning at the nth character for the mth occurrence  
-- of string2.  If n is negative, search backwards.  If m is not passed,  
-- assume 1 (search starts at first character).  
--

CREATE or replace FUNCTION instr(varchar, varchar) RETURNS integer AS $$  
DECLARE  
    pos integer;  
BEGIN  
    pos:= instr($1, $2, 1);  
    RETURN pos;  
END;  
$$ LANGUAGE plpgsql STRICT IMMUTABLE;  
  
CREATE or replace FUNCTION instr(string varchar, string\_to\_search varchar, beg\_index integer)  
RETURNS integer AS $$  
DECLARE  
    pos integer NOT NULL DEFAULT 0;  
    temp\_str varchar;  
    beg integer;  
    length integer;  
    ss\_length integer;  
BEGIN  
    IF beg\_index > 0 THEN  
        temp\_str := substring(string FROM beg\_index);  
        pos := position(string\_to\_search IN temp\_str);  
        IF pos = 0 THEN  
            RETURN 0;  
        ELSE  
            RETURN pos + beg\_index - 1;  
        END IF;  
    ELSE  
        ss\_length := char\_length(string\_to\_search);  
        length := char\_length(string);  
        beg := length + beg\_index - ss\_length + 2;  
        WHILE beg > 0 LOOP  
            temp\_str := substring(string FROM beg FOR ss\_length);  
            pos := position(string\_to\_search IN temp\_str);  
            IF pos > 0 THEN  
                RETURN beg;  
            END IF;  
            beg := beg - 1;  
        END LOOP;  
        RETURN 0;  
    END IF;  
END;  
$$ LANGUAGE plpgsql STRICT IMMUTABLE;  
  
CREATE or replace FUNCTION instr(string varchar, string\_to\_search varchar,  
                      beg\_index integer, occur\_index integer)  
RETURNS integer AS $$  
DECLARE  
    pos integer NOT NULL DEFAULT 0;  
    occur\_number integer NOT NULL DEFAULT 0;  
    temp\_str varchar;  
    beg integer;  
    i integer;  
    length integer;  
    ss\_length integer;  
BEGIN  
    IF beg\_index > 0 THEN  
        beg := beg\_index;  
        temp\_str := substring(string FROM beg\_index);  
        FOR i IN 1..occur\_index LOOP  
            pos := position(string\_to\_search IN temp\_str);  
            IF i = 1 THEN  
                beg := beg + pos - 1;  
            ELSE  
                beg := beg + pos;  
            END IF;  
            temp\_str := substring(string FROM beg + 1);  
        END LOOP;  
        IF pos = 0 THEN  
            RETURN 0;  
        ELSE  
            RETURN beg;  
        END IF;  
    ELSE  
        ss\_length := char\_length(string\_to\_search);  
        length := char\_length(string);  
        beg := length + beg\_index - ss\_length + 2;  
        WHILE beg > 0 LOOP  
            temp\_str := substring(string FROM beg FOR ss\_length);  
            pos := position(string\_to\_search IN temp\_str);  
            IF pos > 0 THEN  
                occur\_number := occur\_number + 1;  
                IF occur\_number = occur\_index THEN  
                    RETURN beg;  
                END IF;  
            END IF;  
            beg := beg - 1;  
        END LOOP;  
        RETURN 0;  
    END IF;  
END;  
$$ LANGUAGE plpgsql STRICT IMMUTABLE;