oracle中对排序的总结 - CSDN博客

```
-- 按字符方式排序

    select * from perexl order by perexl.xx

    -- 按字符对应的数字方式排序

    select * from perexl order by to_number(perexl.xx)

    -- 按拼音排序 (系统默认)

    select * from perexl order by nlssort(danwei,'NLS_SORT=SCHINESE_PINYIN_M');

    -- 按部首排序

    select * from perexl order by nlssort(danwei,'NLS_SORT=SCHINESE_STROKE_M');

    -- 按笔画排序

    select * from perexl order by nlssort(danwei,'NLS_SORT=SCHINESE_RADICAL_M');

    --排序后获取第一行数据

    select * from (select * from perexl order by nlssort(danwei,'NLS_SORT=SCHINESE_PINYIN_M')
)C where rownum=1

    --降序排序

    select * from perexl order by zongrshu desc

    --升序排序

    select * from perexl order by zongrshu asc

    --将nulls始终放在最前

    select * from perexl order by danwei nulls first

    --将nulls始终放在最后

    select * from perexl order by danwei desc nulls last

    --decode函数比nvl函数更强大，同样它也可以将输入参数为空时转换为一特定值

    select * from perexl order by decode(danwei,null,'单位是空', danwei)

    -- 标准的rownum分页查询使用方法
```

select *from (select c.*, rownum rn from personnel c)where rn >= 1and rn <= 5

--在oracle语句rownum对排序分页的解决方案

--但是如果，加上order by 姓名 排序则数据显示不正确

select *from (select c.*, rownum rn from personnel c order by 出生年月)where rn >= 1and rn <= 5

--解决方法，再加一层查询，则可以解决

select *from (select rownum rn, t.*from (select 姓名, 出生年月 from personnel order by 出生年月 desc) t)where rn >= 1and rn <= 5

--如果要考虑到效率的问题，上面的还可以优化成（主要两者区别）

select *from (select rownum rn, t.*from (select 姓名,出生年月 from personnel order by 出生年月 desc) t where rownum <= 10) where rn >= 3

--nvl函数可以将输入参数为空时转换为一特定值,下面就是当单位为空的时候转换成“单位是空”

select * from perexl order by nvl(danwei,'单位是空')