

## ◆ cursors (cursor with parameter & sys\_refcursor & for update)

در این بخش قصد داریم به معرفی cursor ، انواع و کاربرد آن در برنامه های PL/SQL بپردازیم

✚ سوال cursor : چیست ؟

در واقع cursor قسمتی از فضایی حافظه تخصیص داده شده به اوراکل است که یا توسط اوراکل بصورت اتوماتیک باز و بسته میشود و یا توسط برنامه نویس در بخش declaration تعریف و استفاده میشود

✚ چند نوع cursor وجود دارد ؟

میتوان cursor ها را به صورت زیر دسته بندی کرد

1. cursor implicit (ضمنی)
2. cursor explicit (صریح)
  - Cursor
  - sys\_refcursor

۱- اوراکل به ازای هر دستور DML یا دستور select ی که در بخش) executable کدهای نوشته در begin و end) نوشته میشود بصورت اتوماتیک از cursor ضمنی استفاده میکند ، در واقع اوراکل برای پیمایش رکوردهای مورد نظر دستورات نوشته شده از cursor ضمنی استفاده میکند ، تنها استفاده ای که میتوانید از cursor های ضمنی داشته باشید این است که تا قبل از صادر کردن دستور commit یا rollback از خصوصیت های آن میتوانید استفاده کنیم ، خصوصیت های cursor های ضمنی شامل موارد زیر است 🏠🏠🏠

- sql % rowcount
- sql % found
- sql % notfound

۲- هرگاه نیاز داشتهیم که دستور پرس و جویی بنویسیم که خروجی بیش از یک سطر داشته باشد باید از cursor صریح استفاده میکنیم ، cursor صریح باید توسط برنامه نویس و در قسمت declaration تعریف شود ، مدیریت cursor های صریح توسط برنامه نویس انجام میشود، به مثال ذیل دقت نمایید 🏠🏠🏠

```

declare
cursor c is
select * from employees emp;
v_emp_row employees % rowtype;
begin

open c;
loop
fetch c
into v_emp_row;
exit when c % notfound;
dbms_output.put_line(c % rowcount);
insert into employees_1 values v_emp_row;

end loop;

close c;
commit;
end;

```

در مثال فوق در بخش declare یک cursor تعریف شده است که در بخش executable مدیریت آن توسط دستورات آن ذیل انجام شده است

- open <cursor\_namer> ==> باز کردن
- fetch <cursor\_namer> into ==> واکنشی یک سطر
- exit when <cursor\_name> % notfound; ==> شرط خروج از حلقه
- close <cursor\_namer> ==> بستن

👉 میتوانید از پارامتر در CURSOR استفاده کنید، به مثال ذیل دقت نمایید 📌📌

```

declare
cursor c(p_dept_id employees.department_id % type) is
select * from employees emp where emp.department_id = p_dept_id;
v_emp_row employees % rowtype;
v_counter pls_integer := 0;

```



```

begin
open c(20);
loop
fetch c
into v_emp_row;
exit when c%notfound;
end loop;
dbms_output.put_line(c%rowcount);

close c;

for r in c(30) loop
v_counter := v_counter + 1;

end loop;
dbms_output.put_line(v_counter);

end;

```

👉 میتوانید از عبارت for update در cursor استفاده کنید ، این کار باعث میشود که رکوردهای خروجی پرس و جو lock شود 🙅🙅

```

declare
cursor c is
select *
from employees emp
where emp.department_id = 80
for update nowait;
begin
for r in c loop
update employees emp
set emp.salary = emp.salary + 0.5
where current of c;

end loop;
end;

```



اگر نیاز داشتید از cursor استفاده کنید ولی پرس و جوهای آن متفاوت باشد میتوانید از sys\_refcursor استفاده کنید ،  
به مثال ذیل دقت نمایید

```
declare  
v_field_result number;  
v_ref sys_refcursor;  
begin  
open v_ref for 'select ' || p_field_name || ' from ' || p_table_name;  
loop  
fetch v_ref  
into v_field_result;  
exit when v_ref % notfound;  
dbms_output.put_line(v_field_result);  
end loop;  
close v_ref;  
end;
```