Atrybuty kursora

		%FOUND	%ISOPEN	%NOTFOUND	%ROWCOUNT
OPEN	**before**	exception	FALSE	exception	exception
	after	NULL	TRUE	NULL	0
First FETCH	**before**	NULL	TRUE	NULL	0
	after	TRUE	TRUE	FALSE	1
Next FETCH(es)	**before**	TRUE	TRUE	FALSE	1
	after	TRUE	TRUE	FALSE	data dependent
Last FETCH	**before**	TRUE	TRUE	FALSE	data dependent
	after	FALSE	TRUE	TRUE	data dependent
CLOSE	**before**	FALSE	TRUE	TRUE	data dependent
	after	exception	FALSE	exception	exception

Notes:

1. Referencing %FOUND, %NOTFOUND, or %ROWCOUNT before a cursor is opened or after it is closed raises INVALID_CURSOR.
2. After the first FETCH, if the result set was empty, %FOUND yields FALSE, %NOTFOUND yields TRUE, and %ROWCOUNT yields 0.

Kursor niejawny

```
set serveroutput on

begin
   begin
      insert into dept(deptno,dname) values (21,'IMSI');
      insert into dept(deptno,dname) values (22,'IMSI');
      insert into dept(deptno,dname) values (23,'IMSI');
      insert into dept(deptno,dname) values (24,'IMSI');
      commit;
   end;
   begin
      delete dept where dname='IMSI';
      IF SQL%FOUND THEN
      dbms_output.put_line(SQL%ROWCOUNT ||' rekordy
     zostały skasowane');
      END IF;
      commit;
   end;
end;
```

Kursor jawny

```
- zadeklarowanie kursora np.:
CURSOR C IS SELECT …FROM;
- otworzenie kursora, które powoduje wykonanie
instrukcji SELECT:
OPEN c;
- pobranie danych z kursora instrukcją FETCH:
FETCH c INTO zmienne;
- zamknięcie kursora
CLOSE c;
```

UWAGA:
Specjalna formuła pętli FOR- LOOP – umożliwia łatwiejsze przetwarzanie całych wyników zapytać zastępując wywołania OPEN, FETCH, CLOSE i wykonując zawartość pętli do momentu braku wyników.

Kursor jawny

```
set serveroutput on;
declare
cursor kur1 is select
sal
from emp order by 1;
zm number(10);
begin
open kur1;
loop
fetch kur1 into zm;
exit when kur1%notfound;
dbms_output.put_line(zm)
;
end loop;
close kur1;
end;
```

```
set serveroutput on;
declare
cursor kur1 is select sal
from emp order by 1;
zm emp.sal%type;
begin
open kur1;
loop
fetch kur1 into zm;
exit when kur1%notfound;
dbms_output.put_line(zm);
end loop;
close kur1;
end;
```

```
set serveroutput on;

declare
cursor kur1 is select sal
from emp order by 1;
zm kur1%rowtype;
begin
open kur1;
loop
fetch kur1 into zm;
--exit when kur1%notfound;
dbms_output.put_line(zm.sal
);
end loop;
close kur1;
end;
```

Kursor jawny - cd.

```
set serveroutput on;
declare
  cursor   kur1(n   number)   is
select sal from emp
where sal > n order by 1;
  zm kur1%ROWTYPE;
begin
open kur1(2000);
loop
fetch kur1 into zm;
exit when kur1%notfound;
dbms_output.put_line(zm.sal);
end loop;
close kur1;
end;
```

```
set serveroutput on;
  declare
  cursor kur1 is select sal from
emp order by 1;
  zm kur1%rowtype;
begin
for zm in kur1
loop
dbms_output.put_line(zm.sal);
--exit when kur1%notfound;
end loop;
end;
```