CURSORES

SELECT last\_name

INTO l\_last\_name

FROM employees

WHERE employee\_id = 138;

-- retorna uma única linha

SELECT last\_name

INTO l\_last\_name

FROM employees

-- retorna várias linhas, TOO\_MANY\_ROWS exception raised

Ciclo de vida de um cursor:

1. **DECLARE**

CURSOR <cursor\_name>[(parameters\_list)] IS <select statement>;

1. **OPEN**

OPEN <cursor\_name>[(parameters\_list)];

1. **FETCH, num ciclo LOOP, FOREACH**

FETCH <cursor\_name> INTO <variable>;

1. **CLOSE**

CLOSE <cursor\_name>;

EXCEÇÕES

Ciclo de vida de uma exceção:

1. **DECLARE**

<exception\_name> EXCEPTION;

1. **RAISE**

RAISE <exception\_name>;

1. **EXCEPTION HANDLING**

EXCEPTION

WHEN < exception\_name > THEN <exception\_handling\_routine>;

-- tabelas

drop table ArmazemProduto;

drop table Produto;

drop table Armazem;

create table Produto(cod\_produto integer primary key);

create table Armazem(cod\_armazem integer primary key);

create table ArmazemProduto(

cod\_armazem integer,

cod\_produto integer,

stock integer default 0 not null,

stock\_minimo integer default 0 not null,

corredor varchar(10),

prateleira varchar(10),

constraint pk\_ArmazemProduto primary key(cod\_armazem, cod\_produto),

constraint fk\_ArmazemProduto\_Armazem foreign key (cod\_armazem) references Armazem(cod\_armazem),

constraint fk\_ArmazemProduto\_Produto foreign key (cod\_produto) references Produto(cod\_produto)

);

-- dados para teste

insert into Armazem values(1);

insert into Armazem values(2);

insert into Armazem values(3);

insert into Armazem values(4);

insert into Armazem values(5);

insert into Produto values(1);

insert into Produto values(2);

insert into Produto values(3);

insert into Produto values(4);

insert into Produto values(5);

insert into Produto values(6);

insert into Produto values(7);

insert into Produto values(8);

insert into ArmazemProduto values(1, 1, 10, 5, 'A23', '2A');

insert into ArmazemProduto values(1, 2, 18, 10, 'A81', '1S');

insert into ArmazemProduto values(1, 3, 8, 1, 'A23', '1W');

insert into ArmazemProduto values(1, 4, 121, 50, 'A81', '2A');

insert into ArmazemProduto values(1, 5, 60, 5, 'A23', '2S');

insert into ArmazemProduto values(1, 6, 154, 10, 'A23', '2B');

insert into ArmazemProduto values(1, 7, 25, 20, 'A23', '2C');

insert into ArmazemProduto values(1, 8, 113, 10, 'A23', '2S');

insert into ArmazemProduto values(2, 2, 26, 5, '2B98', '22EW');

insert into ArmazemProduto values(2, 4, 87, 10, '2B12', '22B2');

insert into ArmazemProduto values(3, 3, 44, 12, '3B15', '32B3');

insert into ArmazemProduto values(3, 1, 110, 5, '3C13', '42B2');

insert into ArmazemProduto values(4, 7, 120, 10, '4C65', '42NW');

insert into ArmazemProduto values(4, 6, 90, 5, '4B35', '42SS');

-- procedimento

CREATE OR REPLACE PROCEDURE repor\_stocks(armazem\_a\_reabastecer ArmazemProduto.cod\_armazem%type) AS

qtd\_a\_encomendar ArmazemProduto.stock%type;

c\_produto ArmazemProduto.cod\_produto%type;

c\_stock ArmazemProduto.stock%type;

c\_stock\_minimo ArmazemProduto.stock\_minimo%type;

**CURSOR c\_produtos(armazem ArmazemProduto.cod\_armazem%type)** IS

SELECT cod\_produto, stock, stock\_minimo

FROM ArmazemProduto

WHERE cod\_armazem = armazem;

armazem\_inexistente EXCEPTION;

low\_sales EXCEPTION;

vendas\_reduzidas BOOLEAN := TRUE;

dummy Armazem.cod\_armazem%type;

BEGIN

SELECT COUNT(\*) INTO dummy FROM Armazem WHERE cod\_armazem = armazem\_a\_reabastecer;

IF dummy = 0 THEN

RAISE armazem\_inexistente;

ELSE

OPEN c\_produtos(armazem\_a\_reabastecer);

LOOP

FETCH c\_produtos INTO c\_produto, c\_stock, c\_stock\_minimo;

EXIT WHEN c\_produtos%notfound;

qtd\_a\_encomendar := 12 \* c\_stock\_minimo - c\_stock;

IF qtd\_a\_encomendar > 0 THEN

Vendas\_reduzidas := FALSE;

dbms\_output.put\_line(c\_produto || ' - ' || qtd\_a\_encomendar);

END IF;

END LOOP;

CLOSE c\_produtos;

IF vendas\_reduzidas THEN

RAISE low\_sales;

END IF;

END IF;

**EXCEPTION**

WHEN armazem\_inexistente THEN

dbms\_output.put\_line('Não existe nenhum armazém com o código ' || armazem\_a\_reabastecer);

WHEN low\_sales THEN

dbms\_output.put\_line('O armazém ' || armazem\_a\_reabastecer || ' tem vendas reduzidas.');

WHEN others THEN

);

END;

/

-- teste

SET serveroutput ON;

EXEC repor\_stocks(1);

-- resultado esperado com os dados de teste

-- 1 - 50

-- 2 - 102

-- 3 - 4

-- 4 - 479

-- 7 - 215

-- 8 - 7

EXEC repor\_stocks(2);

-- resultado esperado com os dados de teste

-- 2 - 34

-- 4 - 33

EXEC repor\_stocks(3);

-- resultado esperado com os dados de teste

-- 3 – 100

EXEC repor\_stocks(4);

-- resultado esperado com os dados de teste

-- O armazém 4 não tem stock de nenhum produto.

EXEC repor\_stocks(5);

-- resultado esperado com os dados de teste

-- O armazém 5 não tem stock de nenhum produto.

EXEC repor\_stocks(100);

-- resultado esperado com os dados de teste

-- Não existe nenhum armazém com o código 100