## Co je to kurzor? K čemu se využívá? Výhody a nevýhody. Uveďte příklad

* Kurzory jsou pomocné proměnné, které nám umožní procházet výsledek nějakého SQL příkazu
* Existují dva typy
  + implicitní - vytváří se automaticky po provedení příkazu INSERT, UPDATE, DELETE
  + explicitní - definuje se v definiční části procedury podobně jako proměnná, takový kurzor je často spojen s příkazem SELECT, který vrací více než jeden řádek
* Pomocí kurzoru můžeme postupně procházet jednotlivé záznamy výsledku SELECT příkazu a pracovat s jednotlivými hodnotami

**PL/SQL**

* DECLARE

CURSOR c\_surname IS SELECT \* FROM Student;

v\_rec VARCHAR2(30);

BEGIN

OPEN c\_surname;

LOOP

FETCH c\_surname INTO v\_rec;

EXIT WHEN c\_surname%NOTFOUND;

---Do something---

END LOOP;

CLOSE c\_surname;

END;

Případně pomocí FOR – jednodušší zápis, nemusíme uzavírat CLOSE

BEGIN

FOR one\_surname IN c\_surname

LOOP

---------

END LOOP;

END;

**T-SQL**

* DECLARE @id INT

DECLARE productList CURSOR FOR select top 5 id from Product

OPEN productList

FETCH NEXT FROM productList INTO @id

WHILE @@FETCHSTATUS = 0

BEGIN

--do something—

FETCH NEXT FROM productList INTO @id

END

CLOSE productList

DEALLOCATE productList

**Jaké příkazy musíme v PL/SQL spouštět pomocí dynamického SQL? Popište rozdíl mezi statickým a dynamickým PL/SQL**

* Statické dotazy jsou na rozdíl od dynamických předkompilovány a mají hotový plán vykonávání dotazu, spouštění statického SQL je rychlejší.
* Statické využívají vázané proměnné
* Dynamické SQL umožňuje sestavit a volat jakýkoliv SQL příkaz za běhu aplikace
* Dynamické SQL je náchylné na SQL injection, proto je používáme jen když nemůžeme použít statické SQL
* Příklad dynamického SQL u PL/SQL: proměnná typu VARCHAR, do které je postupně vkládán dotaz, spouští se pomocí EXECUTE IMMEDIATE
* T-SQL: sp\_executesql

**Napište minispecifikaci zvolené netriviální funkce z vašeho projektu obsahující alespoň tři DB operace (ne převod částky, kaskádové mazání..). Popište i vstupní parametry a výstup funkce**

Vložení nového řidiče

Procedura, pomocí které vložíme do DB řidiče a zároveň jeho skupinu (řidič nemůže být bez skupiny). Procedura je transakce.

Vstupy: $jmeno, $id\_skupiny

1. Do proměnné @uID uložíme ID nového řidiče:

SELECT MAX(uID) FROM ridic + 1

1. Do proměnné @cislo\_rp vygenerujeme nové číslo ŘP:

SELECT MAX(číslo\_rp) FROM Ridic + 1

1. Transakci začneme vložením nového řidiče:

INSERT INTO Ridic VALUES(@jmeno, @cislo\_rp)

1. Zároveň vložíme do tabulky Ridicovy\_skupiny záznam a dokončíme transakci

INSERT INTO Ridicovy\_skupiny values(@uID, $id\_skupiny)

**Rozdíl mezi shlukováním a indexy**

**K čemu používáme vícenásobné operace**

**Naimplementovat 2 různé kurzory**

**Rozdíl mezi T-SQL a PL/SQL**

* **T-SQL**
* Nejsou operátory %TYPE, %ROWTYPE, nepodporuje CREATE OR REPLACE PROCEDURE
* Musíme při práci s kurzory používat OPEN, FETCH, CLOSE a DEALLOCATE. Nutí nás k dvojitému FETCH u kurzorů
* Musíme definovat u parametrů procedur a funkcí délku datového typu
* U triggerů tabulky INSERTED, DELETED
* Proměnné @name

DECLARE @cnt INT

* **PL/SQL**
  + U triggerů :OLD, :NEW