http://docs.oracle.com/cd/B28359\_01/appdev.111/b28370/dynamic.htm#LNPLS011

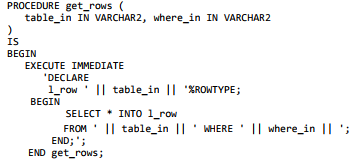
**Динамический SQL и PL\SQL**

Динамический SQL - команды SQL, которые конструируются и вызываются непосредственно во время вымы.

Статический SQL - жестко закодированные команды SQL, которые не изменяются с момента компиляции программы.

Динамическим PL/SQL - целые блоки кода PL/SQL, которые строятся динамически, а затем компилируются и выполняются.

**Пример динамического PL\SQL**



**Когда необходимо использовать:**

* Необходимость исполнение DDL команд в процедурах/функция
* На момент компиляции неизвестны текст SQL-выражения *или* количество *или* типы входных/выходных данных

**Способы реализации:**

* Native dynamic SQL (NDS)
* DBMS\_SQL

**NDS**

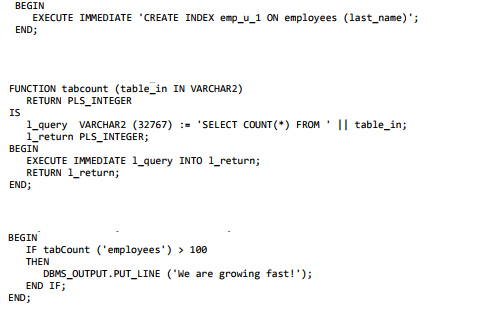
* **Execute immediate**
* **OPEN FOR**

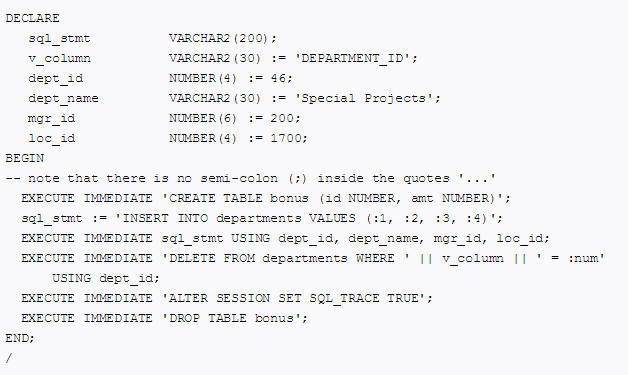
The SQL cursor attributes work the same way after native dynamic SQL INSERT, UPDATE, DELETE, and single-row SELECT statements as they do for their static SQL counterparts.

EXECUTE IMMEDIATE



* *строка\_SQL* — строковое выражение, содержащее команду SQL или блок PL/SQL;
* *переменная* — переменная, которой присваивается содержимое поля, возвращаемого за- просом;
* *запись* — запись, основанная на пользовательском типе или типе %ROWTYPE, при нимающая всю возвращаемую запросом строку;
* *аргумент* — либо выражение, значение которого передается команде SQL или блоку PL/SQL, либо идентификатор, являющийся входной и/или выходной переменной для функции или процедуры, вызываемой из блока PL/SQL.
* Секция *INTO* используется для однострочных запросов. Для каждого значения столбца, возвращаемого запросом, необходимо указать переменную или поле записи совместимого типа. Если INTO предшествует конструкция BULK COLLECT, появляется возможность выборки множественных строк в одну или несколько коллекций.
* Секция *USING* предназначена для передачи аргументов строке SQL. Она используется с динамическим SQL и PL/SQL, что и позволяет задать режим параметра. Этот режим актуален только для PL/SQL и секции RETURNING. По умолчанию для параметров используется режим IN (для команд SQL допустима только эта разновидность аргументов)



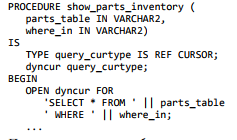


OPEN FOR

Реализации многострочных динамических запросов



* *курсорная\_переменная* — слаботипизированная курсорная переменная;
* *хост\_ переменная* — курсорная переменная, объявленная в хост-среде PL/SQL, например в программе OCI (Oracle Call Interface);
* *строка\_SQL* — команда SELECT, подлежащая динамическому выполнению.

****

**При выполнении OPEN FOR ядро PL/SQL:**

1) связывает курсорную переменную с командой SQL, заданной в строке запроса;

2) вычисляет значения параметров и заменяет ими формальные параметры в строке запроса;

3) выполняет запрос;

4) идентифицирует результирующий набор;

5) устанавливает курсор на первую строку результирующего набора;

6) обнуляет счетчик обработанных строк, возвращаемый атрибутом %ROWCOUNT.

**Для выполнения многострочного запроса** **необходимо:**

1) объявить тип REF CURSOR (или использовать встроенный тип SYS\_REFCURSOR);

2) объявить на его основе курсорную переменную;

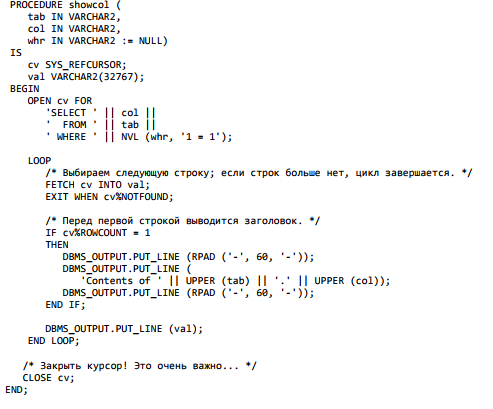
3) открыть курсорную переменную командой OPEN FOR;

4) с помощью команды FETCH по одной извлечь записи результирующего набора;

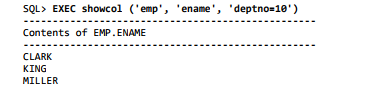
5) при необходимости проверить значения атрибутов (%FOUND, %NOTFOUND, %ROWCOUNT, %ISOPEN);

6) закрыть курсорную переменную обычной командой CLOSE. Как правило, после завершения работы с курсорной переменной следует явно закрыть ее.

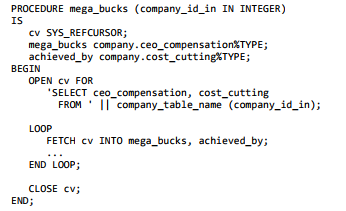
**Пример 1**

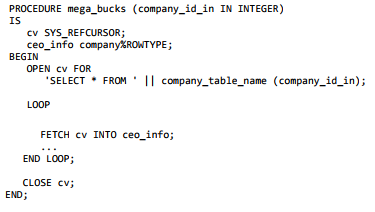
****

Пример результата:

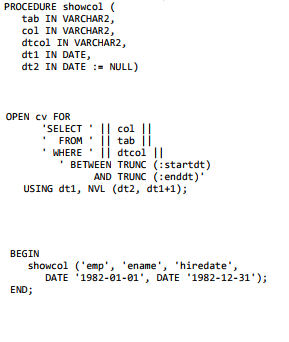
****

**Пример 2. (Выборка в запись)**



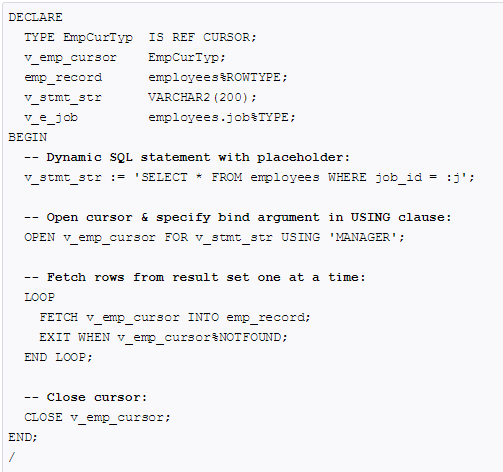


**Пример 3. ( Запрос с секцией USING)**

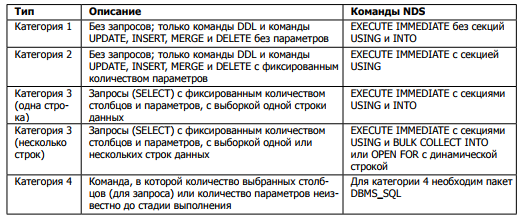




**Пример 4. (Обработка многострочного запроса)**

****

**4 Категории динамического SQL**

****

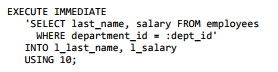
Категория 1



Категория 2



Категория 3



Категория 4



Команда FETCH?

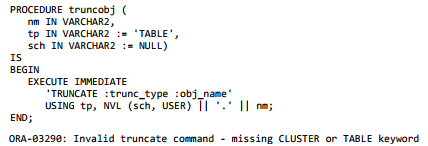
DBMS\_SQL или Динамический PL\SQL

**Передача параметров**

Ограничения

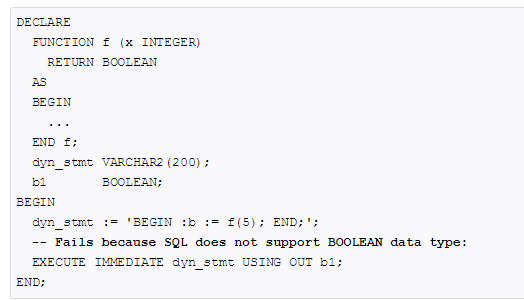
* Формальными параметрами не могут быть объекты схемы (например, имена таблиц или столбцов)
* В качестве параметров подаются поддерживаемые SQL типы данных ( например, BOOLEAN, ассоциативные массивы не могут быть использованы )
* Нельзя подать NULL в качестве параметра (необходимо использовать неинициализированные переменные, приведение типов).

**Пример использования имени таблицы как параметра (неправильно)**

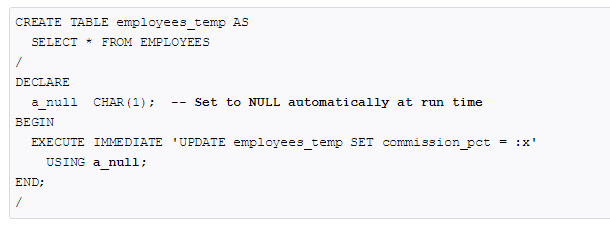
****

****

**Пример использования неподходящего типа (неправильно)**

****

**Пример подстановки NULL значения (NB)**

****

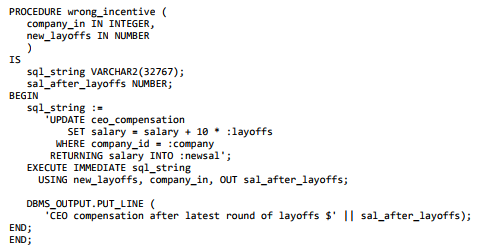
Режимы параметров

IN - режим для чтения, действует по умолчанию; динамический PL\SQL

OUT – только запись, реализуется через переменную, используется для RETURNING; динамический PL\SQL

IN OUT – динамический PL\SQL

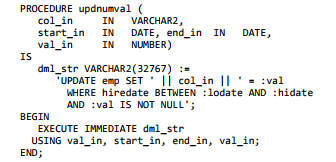
**Пример IN и IN OUT параметра**

****

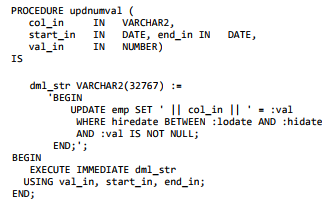
Дублирование параметров

* При выполнении динамической команды SQL (DML- или DDL-строки, не оканчивающейся точкой с запятой) параметр подстановки нужно задать для каждого формального параметра, даже если их имена повторяются. 
* При выполнении динамического блока PL/SQL (строки, оканчивающейся точкой с запятой) нужно указать параметр подстановки для каждого уникального формаль ного параметра.

**Пример дублирования параметра в динамическом SQL**

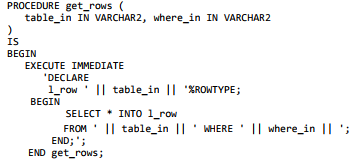
****

**Пример дублирования параметра в динамическом PL\SQL**

****

**Динамический PL\SQL (детальнее)**

**Пример динамического PL\SQL**

****

Позволяет:

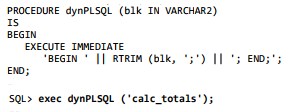
* создать программу, в том числе пакет, содержащий глобальные структуры данных; 
* получать и модифицировать значения глобальных переменных по именам; 
* вызывать функции и процедуры, имена которых не известны во время компиляции

Необходимо учитывать:

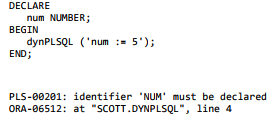
* Динамическая строка должна быть допустимым блоком PL/SQL, начинаться с ключевого слова DECLARE или BEGIN (которому может предшествовать метка или комментарий), а завершаться командой END и точкой с запятой (в противном случае она не считается кодом PL/SQL). 
* В динамическом блоке доступны только глобальные элементы PL/SQL (функции, процедуры и другие элементы, объявленные в спецификации пакета). Динамические блоки PL/SQL выполняются вне локального внешнего блока. 
* Ошибки, возникающие в динамическом блоке PL/SQL, могут быть перехвачены и обработаны локальным блоком, в котором с помощью команды EXECUTE IMMEDIATE выполнялись динамические команды SQL

Построение динамических блоков

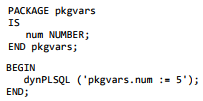
**Пример выполнения динамического блока**



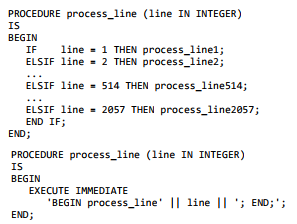
**Пример неправильного обращения к переменной**

****

**Пример неправильного обращения к переменной**

****

**Пример использования динамического PL\SQL**

****

Параметры или конкатенация

****

**???**