СУБД

Курсоры

Лекция 12

Доцент Кафедри Інформатики ХНУРЕ, начальник Відділу Контроля Якості ПЗ INFOSTROY LTD Матат О.О.
E-mail: helen.matat@infostroy.com.ua

Основные понятия

Для обработки SQL-оператора Oracle выделяет область памяти, называемую контекстной областью (context area).

Включая

Информация, необходимая для завершения обработки

- число строк, обрабатываемых оператором
- указатель на представление этого оператора после проведения синтаксического анализа
- активный набор (active set), т.е. набор строк, возвращаемых запросом

Курсор (cursor) - это указатель на контекстную область. С его помощью программа PL/SQL может управлять контекстной областью и ее состоянием во время обработки оператора.

Доцент Кафедри Інформатики ХНУРЕ, начальник Відділу Контроля Якості ПЗ INFOSTROY LTD Матат О.О.
E-mail: helen.matat@infostroy.com.ua

Основные понятия

Курсор - это указатель на зарезервированный участок памяти в SGA, где обрабатывается SQL (SELECT) или DML (UPDATE/INSERT/DELETE) предложение или получаемый при выполнении запроса результирующий набор и связанный с ним указатель текущей записи.

В PL/SQL поддерживаются три основных типов курсоров:

- Неявный,
- Явный
- Курсорные циклы FOR.

Курсор может возвращать одну строку, несколько строк или ни одной строки.

Для запросов, возвращающих более одной строки, можно использовать только явный курсор.

Курсор может быть объявлен в секциях объявлений любого блока PL/SQL, подпрограммы или пакета.

Доцент Кафедри Інформатики ХНУРЕ, начальник Відділу Контроля Якості ПЗ INFOSTROY LTD Матат О.О.
E-mail: helen.matat@infostroy.com.ua

Неявные курсоры

Неявный курсор управляется автоматически. Неявный курсор не требует объявления. В процессе выполнения курсор открывается, из него выбираются данные и закрывается, все за один шаг.Неявные курсоры используются только в том случае, если надо возвратить единственную строку из таблицы.

```
SELECT addr1, addr2, city, state, zip5
INTO v_addr1, v_addr2, v_city, v_state, v_zip5
FROM addresses
WHERE …  -- извлечь по уникальному ключу
      -- (exact match on unique key)
```

Если неявный курсор должен возвратить более, чем одну строку, срабатывает исключение (TOO_MANY_ROWS). Дело в том, что Вы можете выбирать (SELECT INTO) только в скалярные переменные (имеющие только единственное значение).

Если неявный курсор вообще не возвращает строк, срабатывает исключение (NO_DATA_FOUND).

Доцент Кафедри Інформатики ХНУРЕ, начальник Відділу Контроля Якості ПЗ INFOSTROY LTD Матат О.О.
E-mail: helen.matat@infostroy.com.ua

Типичные исключения при работе с курсором

Неявный курсор необходимо использовать вместе с обработчиком исключений.

Типичные исключения, которые могут возникать при работе с курсором:

Exception Name	ORACLE Error	SQLCODE Value
CURSOR_ALREADY_OPEN	ORA-06511	-6511
DUP_VAL_ON_INDEX	ORA-00001	-1
INVALID_CURSOR	ORA-01001	-1001
INVALID_NUMBER	ORA-01722	-1722
LOGIN_DENIED	ORA-01017	-1017
NO_DATA_FOUND	ORA-01403	+100
NOT_LOGGED_ON	ORA-01012	-1012
PROGRAM_ERROR	ORA-06501	-6501
STORAGE_ERROR	ORA-06500	-6500
TIMEOUT_ON_RESOURCE	ORA-00051	-51
TOO_MANY_ROWS	ORA-01422	-1422
TRANSACTION_BACKED_OUT	ORA-00061	-61
VALUE_ERROR	ORA-06502	-6502
ZERO_DIVIDE	OAR-01476	-1476

Доцент Кафедри Інформатики ХНУРЕ, начальник Відділу Контроля Якості ПЗ INFOSTROY LTD Матат О.О.
E-mail: helen.matat@infostroy.com.ua

Неявные курсоры

Если объявлена переменная типа record, поля которой точно совпадают с данными, запрашиваемыми в операторе SELECT, то можно использовать конструкцию SELECT INTO.

```
DECLARE
address_rec addresses%ROWTYPE;
BEGIN
SELECT *
INTO address_rec
FROM addresses
WHERE …
```

Неявные курсоры были оптимизированы для выбора единственной строки, и, следовательно, они срабатывают быстрее, чем заданные явно: открытие, выборка и закрытие курсора.

Доцент Кафедри Інформатики ХНУРЕ, начальник Відділу Контроля Якості ПЗ INFOSTROY LTD Матат О.О.
E-mail: helen.matat@infostroy.com.ua

Явные курсоры

Явные курсоры используются, когда ожидается, что запрос может вернуть как ни одной, так и много строк. Он же позволяет выполнить сложные действия (логику) при выборке каждой строки. Управление явными курсорами осуществляется в программном коде PL/SQL и включает открытие, извлечение данных и закрытие (OPEN, FETCH, CLOSE) курсора.

При использовании явных курсоров можно выделить 4 основных этапа:

1. Объявление курсора.
2. Открытие (OPEN) курсора.
3. Извлечение (FETCH) данных из курсора один или более раз.
4. Закрытие (CLOSE) курсор.

При применении неявного курсора ***нельзя*** использовать операторы управления курсором OPEN, FETCH и CLOSE.

Доцент Кафедри Інформатики ХНУРЕ, начальник Відділу Контроля Якості ПЗ INFOSTROY LTD Матат О.О.
E-mail: helen.matat@infostroy.com.ua

Формальное описание явного курсора

```
CURSOR cursor_name
   [(parameter[,parameter]...)]
   [RETURN return_type]
   IS select_statement;
```

Каждый параметр parameter определяется как:

```
cursor_parameter_name [IN]
      datatype [{:= | DEFAULT} expr]
```

Параметр return_type определяет запись или строку базы данных, используемую для возвращаемых значений. Тип возвращаемого значения должен соответствовать столбцам, перечисленным в операторе SELECT. Список параметров определяет параметры курсора, передаваемые на сервер каждый раз при выполнении оператора OPEN.

Одновременно с созданием результирующего набора можно выполнить блокировку выбираемых строк. Для этого в операторе SELECT следует указать фразу FOR UPDATE..

Доцент Кафедри Інформатики ХНУРЕ, начальник Відділу Контроля Якості ПЗ INFOSTROY LTD Матат О.О.
E-mail: helen.matat@infostroy.com.ua

Этапы работы с явными курсорами

Курсор определятся в декларативной части блока PL/SQL, подпрограммы или пакета путем задания его имени и специфицирования запроса. Потом возможно манипулирование курсором при помощи трех команд: OPEN, FETCH и CLOSE:

– предложение OPEN выполняет запрос, ассоциированный с курсором, идентифицирует активное множество и позиционирует курсор перед его первой строкой;

– предложение FETCH извлекает текущую строку и продвигает курсор к следующей строке;. FETCH можно повторять неоднократно, пока не будут извлечены все строки;

– предложение CLOSE закрывает курсор.

–

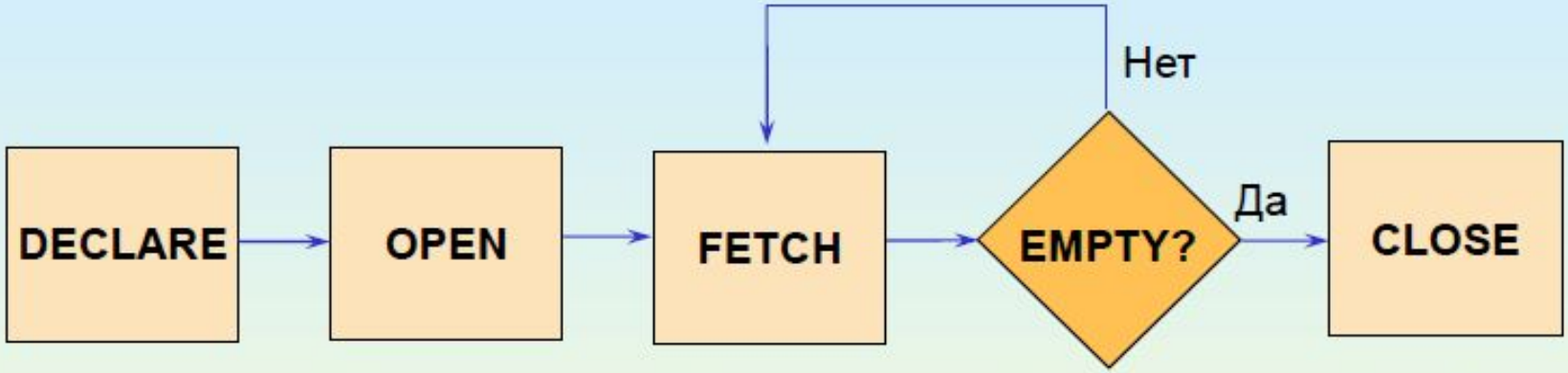
Пример объявления курсора с именем c1:

```
DECLARE
CURSOR c1 IS SELECT ename, deptno FROM emp
WHERE sal > 2000;
...
BEGIN
...
```

Доцент Кафедри Інформатики ХНУРЕ, начальник Відділу Контроля Якості ПЗ INFOSTROY LTD Матат О.О.
E-mail: helen.matat@infostroy.com.ua

Явные курсоры

Управление явными курсорами



- Создание именованной рабочей области SQL
- Выявление активного набора строк
- Загрузка текущей строки в перемен-ные
- Проверка на наличие строки
- Возврат к FETCH если строка обнаружена
- Освобож-дение активного набора строк

Доцент Кафедри Інформатики ХНУРЕ, начальник Відділу Контроля Якості ПЗ INFOSTROY LTD Матат О.О.
E-mail: helen.matat@infostroy.com.ua

Этапы работы с явными курсорами

Курсоры могут принимать параметры, как показывает следующий пример. Параметр курсора может появляться в запросе всюду, где допускается появление константы.

Например:

CURSOR c1 (median IN NUMBER) IS

SELECT job, ename FROM emp WHER sal > median;

Для объявления формальных параметров курсора используется синтаксис

CURSOR имя [(параметр [, параметр, ...])] IS

SELECT ...

где "параметр", в свою очередь, имеет следующий синтаксис:

имя_переменной [IN] тип_данных [{:= | DEFAULT} значение]

Доцент Кафедри Інформатики ХНУРЕ, начальник Відділу Контроля Якості ПЗ INFOSTROY LTD Матат О.О.
E-mail: helen.matat@infostroy.com.ua

Этапы работы с явными курсорами

Открытие курсора предложением **OPEN** исполняет предложение SELECT и идентифицирует АКТИВНОЕ МНОЖЕСТВО, т.е. все строки, удовлетворяющие поисковым условиям запроса. Для курсоров, объявленных с фразой FOR UPDATE, предложение OPEN также осуществляет блокировку этих строк.

Например:

OPEN c1;

Предложение OPEN не извлекает строк активного множества. Для этого используется предложение FETCH.

Пример передачи параметров при объявлении курсора:

CURSOR c1 (my_ename CHAR, my_comm NUMBER) IS SELECT ...

OPEN c1('ATTLEY', 300);

OPEN c1(employee_name, 150);

OPEN c1('THURSTON', my_comm);

Доцент Кафедри Інформатики ХНУРЕ, начальник Відділу Контроля Якості ПЗ INFOSTROY LTD Матат О.О.
E-mail: helen.matat@infostroy.com.ua

Этапы работы с явными курсорами

Предложение **FETCH** извлекает очередную строку из активного множества. При каждом выполнении FETCH курсор продвигается к следующей строке в активном множестве.

Например:

```
FETCH c1 INTO my_empno, my_ename, my_deptno;
```

Для каждого значения столбца, извлекаемого запросом, ассоциированного с курсором, в списке INTO должна быть соответствующая переменная, имеющая совместимый с этим столбцом тип данных.

Например:

```
...
OPEN c1;
LOOP
FETCH c1 INTO my_record;
EXIT WHEN c1%NOTFOUND;
-- обработать извлеченные данные
END LOOP;
```

Доцент Кафедри Інформатики ХНУРЕ, начальник Відділу Контроля Якості ПЗ INFOSTROY LTD Матат О.О.
E-mail: helen.matat@infostroy.com.ua

Этапы работы с явными курсорами

Предложение **CLOSE** деактивирует курсор освобождает занимаемые им ресурсы, и активное множество становится неопределенным.

Например:

CLOSE c1;

После того, как курсор закрыт, можно снова открыть его.

Любая иная операция на закрытом курсоре возбуждает предопределенное исключение INVALID_CURSOR, которое соответствует ошибке ORACLE с кодом ORA-01001.

Доцент Кафедри Інформатики ХНУРЕ, начальник Відділу Контроля Якості ПЗ INFOSTROY LTD Матат О.О.
E-mail: helen.matat@infostroy.com.ua

Атрибуты явного курсора

Для работы с курсором можно использовать следующие атрибуты, указываемые после имени курсора:

Атрибут	Тип	Описание
%ISOPEN	Boolean	Истинно (TRUE), если курсор открыт.
%NOTFOUND	Boolean	Истинно (TRUE), если последняя команда FETCH не вернула строку.
%FOUND	Boolean	Истинно (TRUE), пока последняя команда FETCH возвращает строку.
%ROWCOUNT	Number	Общее количество строк, выбранных на данный момент.

Для обращения к атрибуту его имя присоединяется к имени курсора. Атрибуты явного курсора можно использовать в процедурных предложениях, но не в предложениях SQL.

Доцент Кафедри Інформатики ХНУРЕ, начальник Відділу Контроля Якості ПЗ INFOSTROY LTD Матат О.О.
E-mail: helen.matat@infostroy.com.ua

Атрибут %NOTFOUND

Когда курсор открыт, строки, удовлетворяющие ассоциированному запросу, идентифицированы и образуют активное множество. Эти строки извлекаются операцией FETCH по одной за раз. Если последняя операция FETCH вернула строку, %NOTFOUND дает FALSE. Если последняя операция FETCH не смогла вернуть строку (так как активное множество исчерпано), %NOTFOUND дает TRUE. Операция FETCH должна в конце концов исчерпать активное множество, так что, когда это происходит, никакого исключения не возбуждается.

Пример использования %NOTFOUND:

```
LOOP
FETCH c1 INTO my_ename, my_deptno;
EXIT WHEN c1%NOTFOUND;
...
END LOOP;
```

Доцент Кафедри Інформатики ХНУРЕ, начальник Відділу Контроля Якості ПЗ INFOSTROY LTD Матат О.О.
E-mail: helen.matat@infostroy.com.ua

Атрибут %FOUND

%FOUND логически противоположен атрибуту %NOTFOUND. После открытия явного курсора, но до первой операции FETCH, %FOUND дает NULL. Впоследствии он дает TRUE, если последняя операция FETCH вернула строку, или FALSE, если последняя операция FETCH не смогла извлечь строку, так как больше нет доступных строк.

Пример использования %FOUND:

```
LOOP
FETCH c1 INTO my_ename, my_deptno;
IF c1%FOUND THEN -- извлечение успешно
INSERT INTO temp VALUES (...);
ELSE
EXIT;
...
END LOOP;
```

Доцент Кафедри Інформатики ХНУРЕ, начальник Відділу Контроля Якості ПЗ INFOSTROY LTD Матат О.О.
E-mail: helen.matat@infostroy.com.ua

Атрибут %FOUND

Когда вы открываете курсор, его атрибут %ROWCOUNT обнуляется. Перед первой операцией FETCH %ROWCOUNT возвращает 0. Впоследствии, %ROWCOUNT возвращает число строк, извлеченных операциями FETCH из активного множества на данный момент. Это число увеличивается, если последняя FETCH вернула строку.

Пример использования %ROWCOUNT:

```
LOOP
FETCH c1 INTO my_ename, my_deptno;
IF c1%ROWCOUNT > 10 THEN
-- выбрано больше 10 строк
...
END IF;
END LOOP;
```

Доцент Кафедри Інформатики ХНУРЕ, начальник Відділу Контроля Якості ПЗ INFOSTROY LTD Матат О.О.
E-mail: helen.matat@infostroy.com.ua

Атрибут %ROWCOUNT

Когда вы открываете курсор, его атрибут %ROWCOUNT обнуляется. Перед первой операцией FETCH %ROWCOUNT возвращает 0. Впоследствии, %ROWCOUNT возвращает число строк, извлеченных операциями FETCH из активного множества на данный момент. Это число увеличивается, если последняя FETCH вернула строку.

Пример использования %ROWCOUNT:

```
LOOP
FETCH c1 INTO my_ename, my_deptno;
IF c1%ROWCOUNT > 10 THEN
-- выбрано больше 10 строк
...
END IF;
END LOOP;
```

Доцент Кафедри Інформатики ХНУРЕ, начальник Відділу Контроля Якості ПЗ INFOSTROY LTD Матат О.О.
E-mail: helen.matat@infostroy.com.ua

Атрибут %ISOPEN

%ISOPEN дает TRUE, если явный курсор открыт, и FALSE в противном случае.

Пример использования %ISOPEN:

```
IF c1%ISOPEN THEN -- курсор открыт
...
ELSE -- курсор закрыт, открыть его
OPEN c1;
END IF;
```

Доцент Кафедри Інформатики ХНУРЕ, начальник Відділу Контроля Якості ПЗ INFOSTROY LTD Матат О.О.
E-mail: helen.matat@infostroy.com.ua

Использование курсора в цикле FOR

Вместо управления курсором операторами OPEN, FETCH и CLOSE язык PL/SQL предоставляет возможность последовательной обработки курсора в цикле FOR.

Цикл FOR с курсором выполняет следующие действия:

1. Неявно объявляет переменную цикла как запись %ROWTYPE.
2. Открывает курсор.
3. При каждой итерации извлекает следующую строку из результирующего набора в поля неявно объявленной записи.
4. По достижении конца результирующего набора закрывает курсор.

Доцент Кафедри Інформатики ХНУРЕ, начальник Відділу Контроля Якості ПЗ INFOSTROY LTD Матат О.О.
E-mail: helen.matat@infostroy.com.ua

Использование курсора в цикле FOR

Циклы FOR с курсорами: синтаксис

```
FOR record_name IN cursor_name LOOP
  statement1;
  statement2;
  . . .
END LOOP;
```

Циклы FOR с курсорами: пример

```
FOR item_record IN item_cursor LOOP
        -- неявное открытие и неявная выборка
  v_order_total := v_order_total +
     (item_record.price * item_record.quantity);
  i := i + 1;
  product_id_table (i) := item_record.product_id;
  order_total_table (i) := v_order_total;
END LOOP;                    -- неявное закрытие
```

Доцент кафедри інформатики ХНУРЕ, начальник відділу контроля якості ПЗ INFOSTROY LTD Матат О.О.
E-mail: helen.matat@infostroy.com.ua

Использование курсора в цикле FOR

Например:

```
DECLARE
  CURSOR c1 IS
    SELECT f1, f2 FROM tbl2
    WHERE f1 = 10;
BEGIN
  - Неявное объявление rec1
  FOR rec1 IN c1 LOOP
      /* Выбрана следующая
        строка таблицы */
      INSERT INTO temp_tbl
      VALUES (rec1.f2);
  END LOOP;
  COMMIT;
END;
```

Доцент Кафедри Інформатики ХНУРЕ, начальник Відділу Контроля Якості ПЗ INFOSTROY LTD Матат О.О.
E-mail: helen.matat@infostroy.com.ua

Тип REF CURSOR

При объявлении переменной курсора создается указатель типа REF CURSOR. Переменная типа REF CURSOR может передаваться через механизм RPC (вызовы удаленных процедур) между клиентским приложением и сервером Oracle. При использовании переменных курсора для передачи результирующих наборов между хранимыми подпрограммами PL/SQL и различными клиентскими приложениями каждое из приложений разделяет указатель на рабочую область, в которой расположен результирующий набор. Это позволяет одновременно ссылаться на одну и ту же рабочую область как клиентским приложениям, разработанным в Oracle Forms, в Visual Studio или в Delphi, так и OCI-клиентам или серверу Oracle.

Создание переменной курсора выполняется в два этапа: сначала определяется тип REF CURSOR, а затем объявляется переменная этого типа.

Определить тип REF CURSOR можно в любых блоках PL/SQL, подпрограммах или пакетах.

Матат Кафедри Інформатики ХНУРЕ, начальник Відділу Контроля Якості ПЗ INFOSTROY LTD Матат О.О.
E-mail: helen.matat@infostroy.com.ua

Тип REF CURSOR

Определение типа REF CURSOR может иметь следующее формальное описание:

TYPE ref_type_name IS REF

CURSOR [RETURN return_type];

Параметр ref_type_name задает имя создаваемого типа, а параметр return_type должен определять запись или строку таблицы базы данных.

Если опция RETURN не указана, то переменную курсора можно использовать более гибко, ссылаясь на различные запросы, которые имеют разные типы записей. Однако применение опции RETURN обеспечивает более высокий уровень надежности, позволяя компилятору PL/SQL выполнять проверку совместимости типа переменной курсора с типом результатов запроса. Переменная курсора не может быть сохранена в базе данных.

Переменные курсора в PL/SQL аналогичны указателям языка C++.

Доцент Кафедри Інформатики ХНУРЕ, начальник Відділу Контроля Якості ПЗ INFOSTROY LTD Матат О.О.
E-mail: helen.matat@infostroy.com.ua

Тип REF CURSOR

При выполнении SQL-запроса Oracle создает неименованную рабочую область, в которой содержится сформированный результирующий набор. Для доступа к этому результирующему набору может использоваться:

- явный курсор, который именует рабочую область;
- переменная курсора, которая указывает на эту рабочую область.

Однако явный курсор всегда указывает только на одну и ту же рабочую область, а переменная курсора может ссылаться на различные рабочие области.

Рабочая область будет оставаться доступной до тех пор, пока на нее ссылается хотя бы одна переменная курсора.

Переменная курсора может быть использована в качестве формального параметра процедуры или функции.

Доцент Кафедри Інформатики ХНУРЕ, начальник Відділу Контроля Якості ПЗ INFOSTROY LTD Матат О.О.
E-mail: helen.matat@infostroy.com.ua

Тип REF CURSOR

Для управления переменной курсора используются операторы OPEN-FOR, FETCH и CLOSE.

Оператор OPEN-FOR связывает переменную курсора с запросом, выполняет запрос и получает результирующий набор.

Оператор OPEN-FOR может иметь следующее формальное описание:

OPEN {cursor_variable_name |

:host_cursor_variable_name}

FOR select_statement;

Параметр указывает переменную курсора host_cursor_variable_name, которая была объявлена как переменная основного языка, если блок PL/SQL выполняется в режиме встроенного SQL. Для ссылки на хост-переменную перед ней необходимо указывать символ двоеточия.

Переменная курсора, в отличие от самого курсора, не может иметь параметров.

Запрос, указываемый для переменной курсора, может использовать:

- хост-переменные;
- переменные PL/SQL;
- параметры;
- функции.

Запрос, на который ссылается переменная курсора, не может содержать фразу FOR UPDATE.

Доцент Кафедри Інформатики ХНУРЕ, начальник Відділу Контроля Якості ПЗ INFOSTROY LTD Матат О.О.
E-mail: helen.matat@infostroy.com.ua

Тип REF CURSOR

Для переменной курсора можно использовать атрибуты %FOUND, %NOTFOUND, %ISOPEN и %ROWCOUNT.

Например:

```
DECLARE
  TYPE VarCur IS REF CURSOR
  RETURN tbl1%ROWTYPE;
  - Объявление переменной курсора
  t1 VarCur;
BEGIN
  IF NOT t1%ISOPEN THEN
    - Открываем
    - переменную курсора
    OPEN t1 FOR SELECT *
    FROM tbl1;
    OPEN t1 FOR SELECT *
    FROM tbl1
    - Повторно открываем
    - переменную курсора
          WHERE f2>100;
  END IF;
```

Доцент Кафедри Інформатики ХНУРЕ, начальник Відділу Контроля Якості ПЗ INFOSTROY LTD Матат О.О.
E-mail: helen.matat@infostroy.com.ua

Тип REF CURSOR

При попытке повторно открыть ранее открытую переменную курсора для другого результирующего набора предыдущий запрос будет потерян, а переменная будет ассоциирована с новым запросом.

Если попытаться повторно открыть уже открытый курсор, то Oracle инициирует исключение CURSOR_ALREADY_OPEN.

Если переменная курсора объявляется как формальный параметр подпрограммы, открывающей переменную курсора, то такой параметр должен быть указан с опцией IN OUT.

Доцент Кафедри Інформатики ХНУРЕ, начальник Відділу Контроля Якості ПЗ INFOSTROY LTD Матат О.О.
E-mail: helen.matat@infostroy.com.ua

Тип REF CURSOR

Следующий пример демонстрирует применение в качестве переменной хост-переменной основного языка программирования.

```
/* Объявление хост-переменных */
EXEC SQL BEGIN DECLARE
  /* Объявление переменной курсоров */
  SQL_CURSOR cur1;
EXEC SQL END DECLARE SECTION;
/* Инициализация хост-переменной */
EXEC SQL ALLOCATE : cur1;
EXEC SQL EXECUTE
/* Передача хост-переменной
  курсора в блок PL/SQL */
BEGIN
    OPEN :cur1 FOR SELECT *
    FROM emp;
    - :
    OPEN :cur1 FOR SELECT *
    FROM temp_emp;
END;
END-EXEC;
```

Доцент Кафедри Інформатики ХНУРЕ, начальник Відділу Контроля Якості ПЗ INFOSTROY LTD Матат О.О.
E-mail: helen.matat@infostroy.com.ua

Тип REF CURSOR

Применение **переменных курсора** имеет ряд следующих **ограничений**:

- переменные курсора не могут быть объявлены в пакете (и сохранены в базе данных);
- нельзя использовать механизм RPC для передачи переменных курсора между различными серверами;
- удаленная подпрограмма не может получать значения от переменных курсора с другого сервера;
- запрос, указываемый для переменной курсора оператором OPEN-FOR, не должен содержать фразы FOR UPDATE;
- для переменных курсора не применимы операторы равенства, сравнения или эквивалентности значению NULL;
- переменной курсора не может быть присвоено значение NULL;
- типы REF CURSOR не могут использоваться для определения типа столбцов в SQL-операторах CREATE TABLE и CREATE VIEW, а также для определения типов элементов коллекций;
- переменные курсора не могут быть использованы в динамическом SQL;
- переменную курсора нельзя использовать вместо курсора в цикле FOR-IN-LOOP.

Доцент Кафедри Інформатики ХНУРЕ, начальник Відділу Контроля Якості ПЗ INFOSTROY LTD Матат О.О.
E-mail: helen.matat@infostroy.com.ua

Тип REF CURSOR

Выборка строк из результирующего набора выполняется оператором FETCH, который может иметь следующее формальное описание:

```
FETCH {cursor_variable_name |
    :host_cursor_variable_name}
      INTO {variable_name
      [, variable_name]... |
      record_name};
```

Доцент Кафедри Інформатики ХНУРЕ, начальник Відділу Контроля Якості ПЗ INFOSTROY LTD Матат О.О.
E-mail: helen.matat@infostroy.com.ua

Тип REF CURSOR

Например:

```
DECLARE
  TYPE VarCur IS REF CURSOR
  RETURN tbl1%ROWTYPE;
  - Объявление переменной курсора
  t1 VarCur;
  tbl1_rec tbl1%ROWTYPE;
BEGIN
  IF NOT t1%ISOPEN THEN
  - Открываем
  - переменную курсора
    OPEN t1 FOR SELECT * FROM tbl1;
    LOOP
      - Извлечение строки
      FETCH t1 INTO tbl1_rec;
      - Проверка атрибута
      EXIT WHEN t1%NOTFOUND;
    END LOOP;
  END IF;
  CLOSE t1;
```

Доцент Кафедри Інформатики ХНУРЕ, начальник Відділу Контроля Якості ПЗ INFOSTROY LTD Матат О.О.
E-mail: helen.matat@infostroy.com.ua