

1

Объявление переменных

Использование переменных

Переменные могут использоваться для:

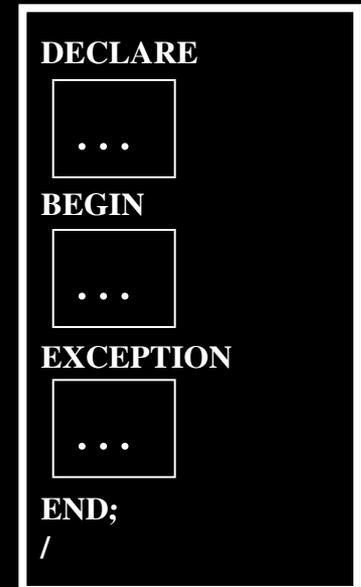
- Временного хранения данных
- Манипуляция хранимых значений
- Многократного использования данных

Существуют в рамках сеанса и не хранятся в бд (PGA)

Простота поддержки

Обработка переменных в PL/SQL

- Объявляют и инициализируют переменные в разделе объявлений
- Назначают новые значения переменным в разделе выполнения
- Передача значений в PL/SQL блоки через параметры
- Представление результатов происходит через переменные вывода



Правила именования переменных

- PL/SQL не учитывает регистр символов
- длина - до 30 символов, может содержать A-Z, 0-9, _, \$
- должны начинаться с буквы
- могут включать символы \$, _ и #
- не должны содержать пробелов
- не должны (а в большинстве случаев и не сможете) использовать зарезервированные слова : такие, как BEGIN, IF и т.д., есть специальный справочник :V\$RESERVED_WORDS ~ 2000 слов.

PLS-00103: Encountered the symbol "END" when expecting one of the following.

- некоторые из правил именования объектов можно нарушить, заключая идентификатор в кавычки
- переменные могут иметь одинаковые имена в случае, если находятся в разных блоках
- имя переменной не должно совпадать с именем поля таблицы, используемого в данном блоке

Правила именования переменных

Рабочая инструкция

“ПРАВИЛА Именования создания PL/SQL кода и объектов баз данных”

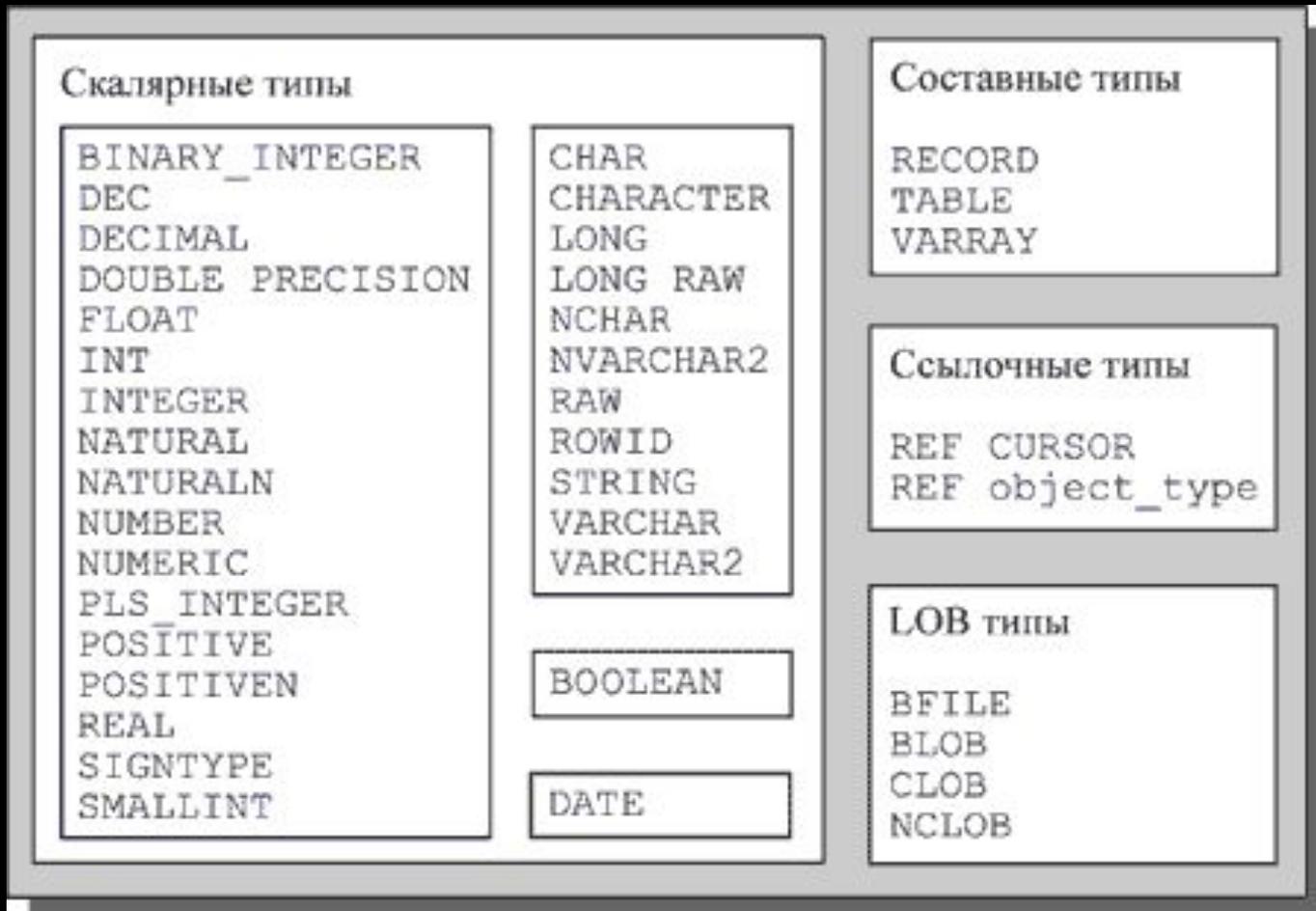
\\Fileserver.office.bercut.ru\CKC\RBT\DB monitoring_Обучение\PLSQL курс

Типы переменных в PL/SQL

Каждая константа, переменная и параметр имеют тип данных, который определяет формат хранения, ограничения и допустимый диапазон значений. PL/SQL предоставляет множество predefined типов данных.

- Скалярные (содержит одно значение и не имеет внутренних компонентов): числовые, символьные, даты, логические данные.
- Составные типы (Composite) – содержат внутренние элементы, которые могут быть скалярного или составного типов. Записи и таблицы PL/SQL.
- Ссылочные типы данных (Reference) – содержат значения, называемые указателями (Pointers), указывающие на другие места хранения.
- Типы данных LOB (Large object (LOB)) – содержат так называемые указатели места (locators), указывающие местоположение больших объектов (например, графических образов), которые хранятся отдельно от других столбцов строки (out of line).

Типы переменных в PL/SQL



Объявление переменных в PL/SQL

Name [CONSTANT] datatype [NOT NULL] [:= | DEFAULT expr];

DECLARE

```
v_hiredate DATE;  
v_deptno NUMBER(2) NOT NULL := 10;  
v_location VARCHAR2(13) := 'Atlanta';  
c_comm CONSTANT NUMBER := 1400;
```

Рекомендации объявления переменных

- Использовать правила именования объектов
- Одна строка – одно объявление переменной
- Для констант и не пустых значений указываем NOT NULL или CONSTANT
- Для объявления переменной использовать выражение (:=) или ключевое слово DEFAULT

```
identifier := expr ;
```

Переменные и ключевые слова

```
identifier := expr;
```

- Установка даты приема на работу:

```
v_hiredate := '31-DEC-08';
```

- Установка predetermined имени служащего :

```
v_ename := 'Ivanov';
```

Объявление скалярных переменных

```
v_hiredate    DATE := SYSDATE + 7;  
v_deptno     NUMBER(2) NOT NULL := 10;  
v_location   VARCHAR2(13);  
v_valid      BOOLEAN NOT NULL := TRUE;  
v_count      BINARY_INTEGER := 0;
```

Скалярные типы данных

Хранят одиночное значение

- The SQL data types , ANSI, DB2, and SQL/DS Data Types (ANSI SQL Data Type -> Oracle Data Type)

http://docs.oracle.com/database/121/SQLRF/sql_elements001.htm#SQLRF00213
INTEGER , INT, SMALLINT -> NUMBER(p,0)

- BOOLEAN

- PLS_INTEGER или BINARY_INTEGER (идентичны +/- $2^{32}/2$) ,
SIMPLE_INTEGER (NOT NULL с 11g)

Database SQL Language Reference:

https://docs.oracle.com/database/121/SQLRF/sql_elements001.htm#SQLRF0021

Database PL/SQL Language Reference:

<http://docs.oracle.com/database/121/LNPLS/datatypes.htm#LNPLS003>

Основные скалярные типы данных:

Символьные типы

- CHAR / NCHAR [(*maximum_length*)] – фиксированная длина, 2000 байт* задается в [CHAR | BYTE], если больше: CLOB.
- VARCHAR2 / NVARCHAR2 (*maximum_length*) – переменная длина, max 4000 байт* задается в [CHAR | BYTE]
- Строковые подтипы и эквивалентные типы данных PL/SQL**
, RAW, LONG RAW поддерживается для обеспечения обратной совместимости. до 32767 байт Для столбца базы данных типа RAW максимальный размер - 2000 байт.

Основные скалярные типы данных

- NUMBER — тип с фиксированной точностью
- PLS_INTEGER и BINARY_INTEGER — целочисленные, Арифметические операции выполняются на уровне машинных команд. Значения этих типов не могут храниться в базе данных.
- SIMPLE_INTEGER — с 11g. ~ BINARY_INTEGER, но не допускает хранение NULL
- BINARY_FLOAT и BINARY_DOUBLE — двоичные типы с плавающей запятой IEEE-754 одинарной и двойной точности.
- SIMPLE_FLOAT и SIMPLE_DOUBLE — типы появились в Oracle Database 11g. Они поддерживают тот же диапазон, что и BINARY_FLOAT с BINARY_DOUBLE, не могут NULL
- INTEGER и DECIMAL. Они представляют собой не что иное, как альтернативные имена для перечисленных основных числовых типов.

Основные скалярные типы данных

1 Общее количество значащих цифр в числе равно 9

NUMBER(9,2)

99999999.99

2 Количество цифр после запятой положительно, поэтому отсчитывать следует влево от этой позиции

3 Отсчитав две позиции влево, ставим десятичную запятую

Таблица 9.5. Предопределенные числовые подтипы данных

Подтип	Совместимость	Соответствующий тип данных Oracle
DEC (A, B)	ANSI	NUMBER (A,B)
DECIMAL (A, B)	IBM	NUMBER (A,B)
DOUBLE PRECISION	ANSI	NUMBER, точность 126 двоичных цифр
FLOAT	ANSI, IBM	NUMBER, точность 126 двоичных цифр
FLOAT (двоичная_точность)	ANSI, IBM	NUMBER, с точностью до 126 двоичных цифр (по умолчанию)
INT	ANSI	NUMBER(38)
INTEGER	ANSI, IBM	NUMBER(38)
NATURAL	N/A	PLS_INTEGER*, но только с неотрицательными значениями (0 и выше)
NATURALN	N/A	То же, что NATURAL, но с запретом NULL
NUMERIC (A, B)	ANSI	NUMBER (A,B)
POSITIVE	N/A	PLS_INTEGER*, но только с положительными значениями (1 и выше)
POSITIVEN	N/A	То же, что POSITIVE, но с запретом NULL
REAL	ANSI	NUMBER, точность 63 цифры
SIGNTYPE	N/A	PLS_INTEGER* с возможными значениями -1, 0 и 1
SMALLINT	ANSI, IBM	NUMBER (38)

Скалярные типы данных

- DATE - дата и время с точностью до секунды.
- TIMESTAMP –DATE, с точностью до миллиардной доли секунды.
- TIMESTAMP WITH TIME ZONE - дата и время с часовым поясе с точностью до девяти десятичных позиций.
- TIMESTAMP WITH LOCAL TIME ZONE - дата и время с точностью до девяти десятичных позиций. Автоматически преобразуются между часовым поясом базы данных и местным (сеансовым) часовым поясом. При хранении в базе данных значения преобразуются к часовому поясу базы данных, а при выборке они преобразуются к местному (сеансовому) часовому поясу.
- Oracle поддерживает два типа данных INTERVAL. Оба типа были введены в Oracle9i, и оба соответствуют стандарту ISO SQL:
- INTERVAL YEAR TO MONTH —интервал в годах и месяцах;
- INTERVAL DAY TO SECOND —интервал времени в днях, часах, минутах и секундах (с долями секунд).

Основные скалярные типы данных: булевы переменные

- Булева переменная может принимать три значения: TRUE, FALSE, NULL.
- Сравнить логические переменные с помощью операторов (OR, AND, NOT)
- Переменные всегда возвращают TRUE, FALSE или NULL.

Основные скалярные типы данных: Объявление булевых переменных

```
v_sal1 := 50000;  
v_sal2 := 60000;
```

Следующее выражение возвращает ИСТИНУ:

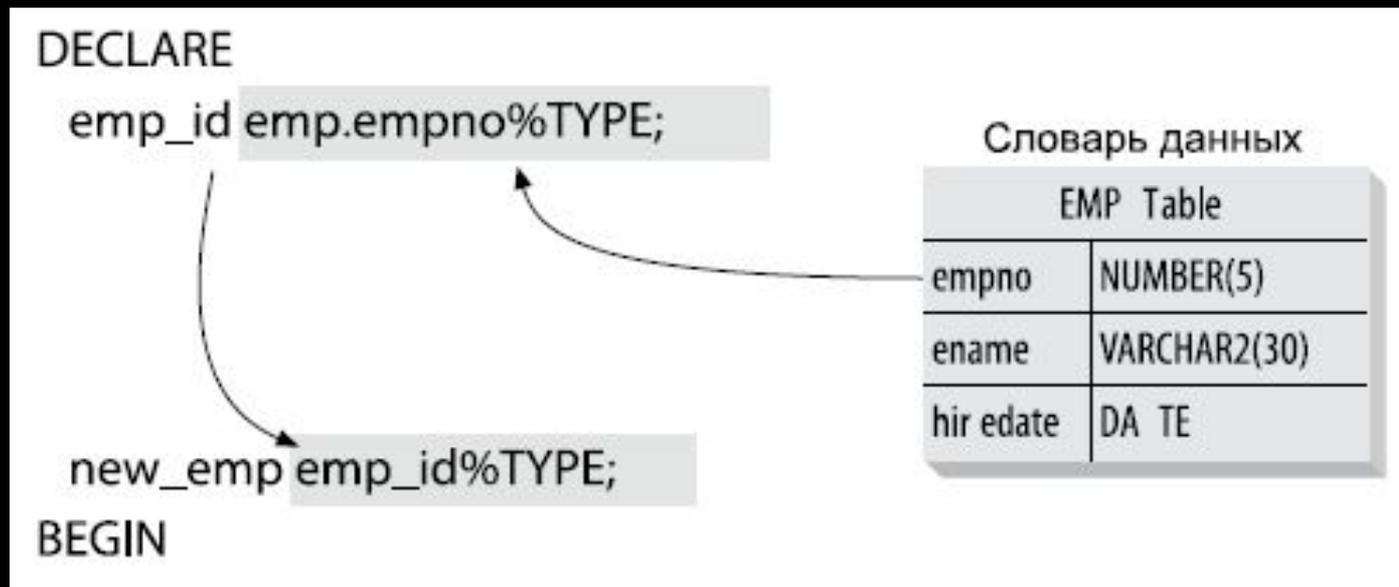
```
v_sal1 < v_sal2
```

Объявление и инициализация Булевой переменной:

```
DECLARE  
    v_flag BOOLEAN := FALSE;  
BEGIN  
    v_flag := TRUE;  
END;
```

Привязанная переменная атрибут %TYPE

Объявляя «привязанную» переменную, мы устанавливаем ее тип данных на основании типа уже определенной структуры данных. Таковой может являться другая переменная PL/SQL, заранее определенный тип, конкретный столбец таблицы.



Объявление переменных с привязкой

```
Identifier_name Table_name.column_name%TYPE;
```

```
v_name employees.first_name%TYPE;
```

```
v_balance NUMBER(7,2);
```

```
v_min_balance v_balance%TYPE := 10;
```

```
...
```

Составные типы данных

- Составной содержит внутренние компоненты, управление которыми осуществляется индивидуально.
- Составные типы данных (также известные как коллекции) используют табличные типы для хранения.

```
TYPE is_SmplRec IS RECORD (  
  m_Fld1 VARCHAR2(10),  
  m_Fld2 VARCHAR2(30) := 'Buber',  
  m_DtFld DATE,  
  m_Fld3 INTEGER := 1000,  
  m_Fld4 VARCHAR2(100) NOT NULL := 'System');  
MY_SMPL is_SmplRec;
```

```
TYPE is_Customers IS RECORD (  
  m_COMPANY CUSTOMERS.COMPANY%TYPE,  
  m_CUST_REP CUSTOMERS.CUST_REP%TYPE,  
  );  
MY_CUST is_Customers;
```

Двоичные типы данных и LOB-типы

Неструктурированные данные, не интерпретируемых и не обрабатываемых Oracle. Для загрузки объектов LOB предусмотрен пакет *PL/SQL DBMS_LOB*.

- CLOB для внутреннего символьного объекта, книги
- BLOB для внутреннего двоичного объекта, фото
- BFILE для внешнего двоичного файла, фильм
- NCLOB для внутреннего символьного объекта, учитывающего национальный набор символов.

Типы данных ROWID и UROWID

Oracle поддерживает два типа данных ROWID и UROWID, предназначенных для предоставления адреса строки в таблице. Тип ROWID представляет уникальный физический адрес строки в таблице, а тип UROWID — логическую позицию строки в индексной таблице (Index-Organized Table, IOT). Тип ROWID также является псевдостолбцом SQL, который может включаться в инструкции SQL.

ИТОГИ

- Рассмотрено устройство блока PL/SQL
- Блоки PL/SQL – анонимные, функции, процедуры
- Переменные PL/SQL – скалярные, ссылки, составные, LOB

Практика №1!

25 минут