

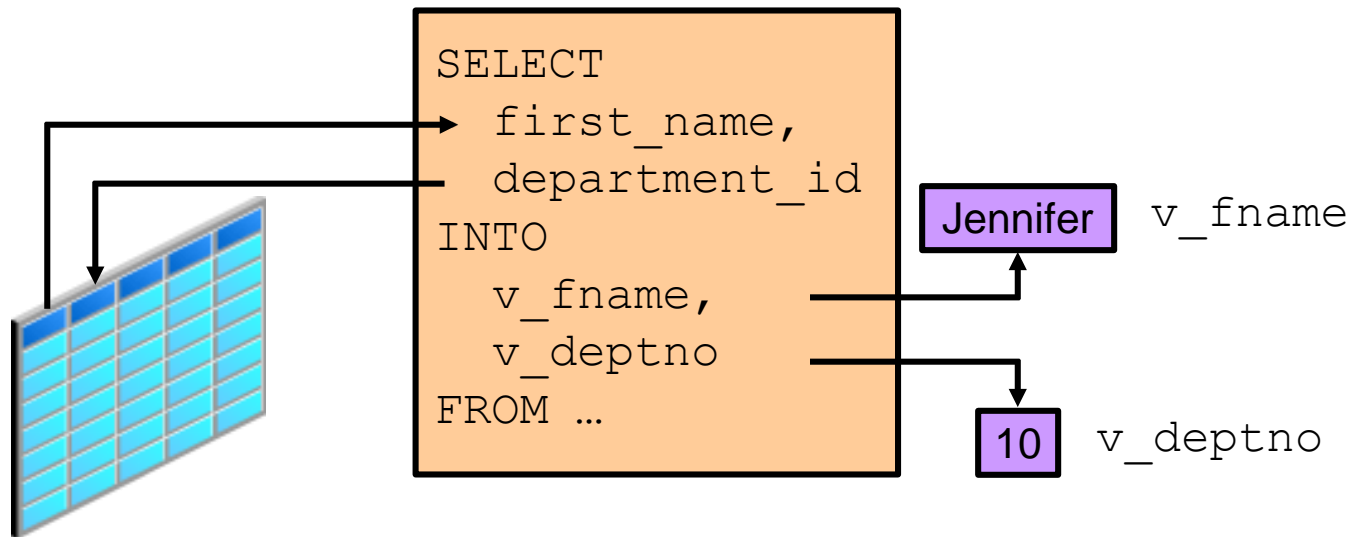
Объявление переменных в PL/SQL

Использование переменных

Переменная – область памяти для хранения данных. У нее есть имя (название) и тип (какого типа данные хранятся)

Переменные используются для:

- Временного хранения данных
- Работы с сохраненными значениями
- Повторного использования данных



Требования к названию переменных

Имя переменной:

- Должно начинаться с буквы
- Может содержать латинские буквы и цифры
- Может содержать спец символы (\$, _, и #)
- Состоит не более, чем из 30 символов
- Не может быть зарезервированным словом (SELECT, DECLARE, BEGIN и т.д.)



Работа с переменными в PL/SQL

Переменные:

- Объявляются (и опционально инициализируются) в секции DECLARE
- Используются в исполняемой секции (BEGIN...END)
- Передаются как параметры в подпрограммы PL/SQL
- Принимают вывод подпрограмм PL/SQL

Объявление и инициализация переменных в PL/SQL

Синтаксис:

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

Примеры:

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

Объявление и инициализация переменных в PL/SQL

```
DECLARE
```

```
    v_myName  VARCHAR(20);
```

1

```
BEGIN
```

```
    DBMS_OUTPUT.PUT_LINE('My name is: ' || v_myName );
```

```
    v_myName  := 'John';
```

```
    DBMS_OUTPUT.PUT_LINE('My name is: ' || v_myName );
```

```
END;
```

```
/
```

```
DECLARE
```

```
    v_myName  VARCHAR2(20) := 'John';
```

2

```
BEGIN
```

```
    v_myName  := 'Steven';
```

```
    DBMS_OUTPUT.PUT_LINE('My name is: ' || v_myName);
```

```
END;
```

```
/
```

Объединение строк

```
DECLARE
    v_event VARCHAR2(15);
BEGIN
    v_event := q'!Father's day!';
    DBMS_OUTPUT.PUT_LINE('3rd Sunday in June is :
    '|| v_event );
    v_event := q'[Mother's day]';
    DBMS_OUTPUT.PUT_LINE('2nd Sunday in May is :
    '|| v_event );
END;
/
```

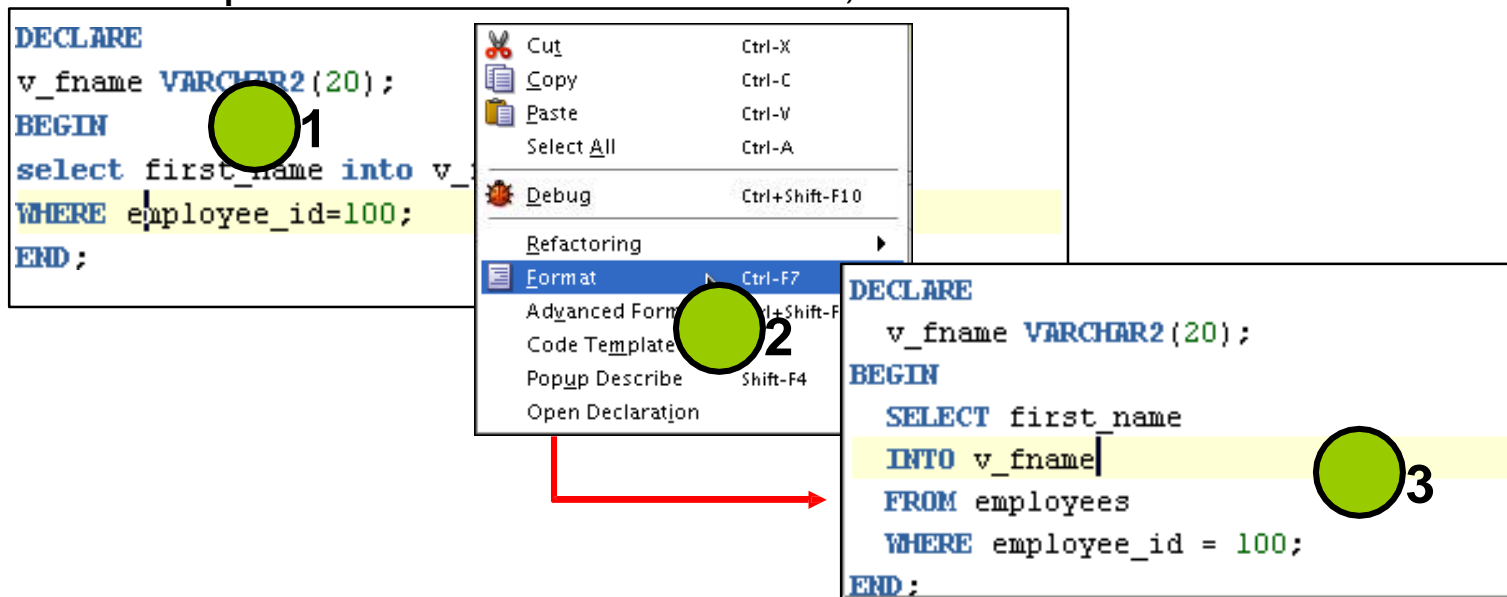
```
anonymous block completed
3rd Sunday in June is : Father's day
2nd Sunday in May is : Mother's day
```

Написание кода в PL/SQL

Синтаксис блоков PL/SQL

- Использование литералов
 - Текст заключается в одинарные кавычки.
 - Числа в виде значений

```
v_name := 'Henderson';
```
- Выражение может быть записано на несколько строк.
Завершается точкой с запятой ;



Комментарии в коде

- Однострочные комментарии (--).
- Многострочный блок с комментариями между /* и */.

Пример:

```
DECLARE
...
v_annual_sal NUMBER (9,2);
BEGIN
/* Compute the annual salary based on the
   monthly salary input from the user */
v_annual_sal := monthly_sal * 12;
--The following line displays the annual salary
DBMS_OUTPUT.PUT_LINE(v_annual_sal);
END;
/
```

Вызов функций SQL из PL/SQL: пример

- Получить длину строки:

```
v_desc_size INTEGER(5);  
v_prod_description VARCHAR2(70):='You can use this  
product with your radios for higher frequency';  
  
-- get the length of the string in prod_description  
v_desc_size:= LENGTH(v_prod_description);
```

- Узнать количество месяцев, которые проработал сотрудник:

```
v_tenure:= MONTHS_BETWEEN (CURRENT_DATE, v_hiredate);
```

Преобразование типов данных

1

-- Неявное преобразование типов

```
v_date_of_joining DATE:= '02-Feb-2000';
```

2

-- Ошибка приведения типа

```
v_date_of_joining DATE:= 'February 02,2000';
```

3

-- Явное преобразование типа

```
v_date_of_joining DATE:= TO_DATE('February  
02,2000','Month DD, YYYY');
```