

PRAKTIKUM basis data lanjutan

Praktikum 6



RIZKY CAHYA ZULIYANTO (3122522018)

PRODI D3 TEKNIK INFORMATIKA

DEPARTEMEN TEKNIK INFORMATIKA DAN KOMPUTER

PENS PSDKU SUMENEP

## KEGIATAN PRAKTIKUM

1. Pada tiap assignment berikut, yang manakah statement yang valid dan mana yang tidak valid, kemudian nyatakan tipe datanya !

- a. `v_days_to_go := v_due_date - SYSDATE;`  
valid dikarenakan Tipe data yang sesuai tergantung pada tipe data `v_due_date` dan `SYSDATE`.
- b. `v_sender := USER || ' ' || TO_CHAR(v_dept_no);`  
valid dikarenakan Tipe data yang sesuai adalah `VARCHAR2` karena hasilnya adalah string.
- c. `v_sum := $100,000 + $250,000`  
tidak valid di karena penggunaan simbol "\$" bukanlah cara yang valid untuk mendeklarasikan variabel atau nilai dalam PL/SQL.
- d. `v_flag := TRUE;`  
valid dikarenakan Tipe data yang sesuai adalah `BOOLEAN` karena Anda menginisialisasi variabel `v_flag` dengan nilai `TRUE`.
- e. `v_n1 := v_n2 > (2 * v_n3);`  
valid dikarenakan hasil perbandingan `(v_n2 > (2 * v_n3))` adalah nilai boolean (`TRUE` atau `FALSE`).
- f. `v_value := NULL;`  
valid dikarena Tipe data yang sesuai tergantung pada tipe data `v_value` yang mendukung nilai `NULL`

## TUGAS PRAKTIKUM

1. Tuliskan output yang dihasilkan dari blok anonym PL/SQL tersebut, dan jika terdapat error tunjukkan baris ke berapa dan mengapa? Lakukan pembenahan terhadap error yang terjadi!.

```

DECLARE
    varKdKelas NUMBER := 7;
    varNmKelas VARCHAR2(20);
    var2 NUMBER;
    hasil VARCHAR2(20);
BEGIN
    SELECT namakelas into varNmKelas FROM kelas WHERE
    ROWNUM=1;
    var2 := 10;
    hasil := varNmKelas + var2;
    dbms_output.put_line (hasil);
    dbms_output.put_line (VARNMKELAS);
    dbms_output.put_line (var2);
END;
/

```

Script Output x

Task completed in 0.159 seconds

\*Cause: An arithmetic, numeric, string, conversion, or constraint error occurred. For example, this error occurs if an attempt is made to assign the value NULL to a variable declared NOT NULL, or if an attempt is made to assign an integer larger than 99 to a variable declared NUMBER(2).

\*Action: Change the data, how it is manipulated, or how it is declared so that values do not violate constraints.

Kode salah dikarenakan adanya kesalahan konversi karakter ke angka yang disebabkan oleh penggabungan variabel varNmKelas (tipe data VARCHAR2) dan var2 (tipe data NUMBER) tanpa konversi yang benar.

```

SET SERVEROUTPUT ON;
DECLARE
    varKdKelas NUMBER := 7;
    varNmKelas VARCHAR2(20);
    var2 NUMBER;
    hasil VARCHAR2(20);
BEGIN
    SELECT namakelas INTO varNmKelas FROM kelas WHERE ROWNUM=1;
    var2 := 10;
    hasil := varNmKelas || TO_CHAR(var2);
    dbms_output.put_line (hasil);
    dbms_output.put_line (varNmKelas);
    dbms_output.put_line (TO_CHAR(var2));
END;
/

```

Script Output x

Task completed in 0.118 seconds

PIS-10-0110  
PIS-10-01  
10

PL/SQL procedure successfully completed.

Kode di perbaiki dengan menggunakan TO\_CHAR untuk mengkonversi nilai dari var2 ke VARCHAR2 sebelum menggabungkannya dengan varNmKelas.

2.

```

DECLARE
    a CONSTANT INT := 10;
    b INT := 40;
    c NUMBER;
BEGIN
    a :=10;
    b :=40;
    c := (a + b - a) / b * a;
    DBMS_OUTPUT.PUT_LINE('Result=|'|c|' dollar');
END;
/

```

- a. Sebutkan bagian yang menjadi blok header, blok body dan blok exception blok header :

```

DECLARE
    a CONSTANT INT := 10;
    b INT := 40;
    c NUMBER;

```

blok body :

```

BEGIN
    a :=10;
    b :=40;
    c := (a + b - a) / b * a;
    DBMS_OUTPUT.PUT_LINE('Result=|c|' dollar');
END;
/

```

blok exception :

tidak ada di program

b. Sebutkan nama variable serta tipe datanya (sebutkan baris ke berapa)

- Baris 2: variable a dengan tipe data INT dengan nilai awal 10.
- Baris 3: variable b dengan tipe data INT dengan nilai awal 40.
- Baris 4: variable c dengan tipe data NUMBER.
- Baris 7: memberikan nilai 10 ke variabel c.
- Baris 8: mengganti nilai variabel c menjadi 40.
- Baris 9: menghitung ekspresi  $(a + b - a) / (b * a)$  dan menetapkan hasilnya ke variabel c.
- Baris 10: menggunakan DBMS\_OUTPUT.PUT\_LINE untuk mencetak hasil dengan pesan..

c. Tuliskan output dari blok anonym PL/SQL tersebut!

Kode setelah di perbaiki hasil nya akan seperti ini

```
SET SERVEROUTPUT ON;
DECLARE
  a CONSTANT INT := 10;
  b INT := 40;
  c NUMBER;
BEGIN
  c := (a + b - a) / b * a;
  DBMS_OUTPUT.PUT_LINE('Result=' || TO_CHAR(c) || ' dollar');
END;
/
```

Script Output x

Task completed in 0.108 seconds

PL/SQL procedure successfully completed.

Result=10 dollar

PL/SQL procedure successfully completed.

3.

```
DECLARE
  kode_brg NUMBER;
  harga_brg NUMBER;
  jumlah_brg NUMBER;
  diskon NUMBER;
  jumlah_bayar NUMBER;
BEGIN
  kode_brg := &kode_barang;
  harga_brg := &harga_barang;
  jumlah_brg := &jumlah_barang;
  diskon := (harga_brg*0.1)*jumlah;
  jumlah_bayar := (harga*jumlah)-diskon;
  DBMS_OUTPUT.PUT_LINE('jumlah bayar: ');
EXCEPTION
  WHEN OTHERS THEN
    DBMS_OUTPUT.PUT_LINE(SQLERRM)
END;
/
```

a. Pada baris program ditemukan tanda &(dan) apakah itu?

tanda & digunakan untuk menunjukkan bahwa nilai-nilai untuk variabel-variabel seperti kode\_barang, harga\_barang, dan jumlah\_barang harus dimasukkan secara interaktif atau manual oleh pengguna saat program dijalankan.

- b. Kapan exception akan dilakukan?

EXCEPTION akan dieksekusi atau berjalan jika di program terjadi kesalahan selama eksekusi kode di dalam blok "BEGIN"

- c. Jika pada baris 6 jumlah\_bayar diganti menjadi jbayer, baris berapakah yang harus dilakukan perbaikan agar program tetap berjalan dengan baik?

Jika baris ke 6 di ganti jumlah\_bayar menjadi jbayer maka pada baris ke 8 juga harus di ubah menjadi jbayer supaya program bisa berjalan lagi

Seperti:

```
DECLARE
    kode_brg NUMBER;
    harga_brg NUMBER;
    jumlah_brg NUMBER;
    diskon NUMBER;
    jbayer NUMBER;
BEGIN
    kode_brg := &kode_barang;
    harga_brg := &harga_barang;
    jumlah_brg := &jumlah_barang;
    diskon := (harga_brg*0.1)*jumlah;
    jbayer := (harga*jumlah)-diskon;
    DBMS_OUTPUT.PUT_LINE('jumlah bayar: ');
EXCEPTION
    WHEN OTHERS THEN
        DBMS_OUTPUT.PUT_LINE(SQLERRM)
END;
/
```

Kode yang sudah di perbaiki :

```

DECLARE
    kode_brg NUMBER;
    harga_brg NUMBER;
    jumlah_brg NUMBER;
    diskon NUMBER;
    jumlah_bayar NUMBER;
BEGIN
    kode_brg := &kode_barang;
    harga_brg := &harga_barang;
    jumlah_brg := &jumlah_barang;

    diskon := (harga_brg * 0.1) * jumlah_brg;
    jumlah_bayar := (harga_brg * jumlah_brg) - diskon;

    DBMS_OUTPUT.PUT_LINE('Jumlah bayar: ' || TO_CHAR(jumlah_bayar));
EXCEPTION
    WHEN OTHERS THEN
        DBMS_OUTPUT.PUT_LINE('Terjadi kesalahan: ' || SQLERRM);
END;
/

```

```

BEGIN
    kode_brg := 12;
    harga_brg := 13;
    jumlah_brg := 15;

    diskon := (harga_brg * 0.1) * jumlah_brg;
    jumlah_bayar := (harga_brg * jumlah_brg) - diskon;

    DBMS_OUTPUT.PUT_LINE('Jumlah bayar: ' || TO_CHAR(jumlah_bayar));
EXCEPTION
    WHEN OTHERS THEN
        DBMS_OUTPUT.PUT_LINE('Terjadi kesalahan: ' || SQLERRM);
END;
Jumlah bayar: 175,5

```

PL/SQL procedure successfully completed.