

### Exemplul 6.2

```
BEGIN
  UPDATE  produse
  SET     pret_unitar =
           pret_unitar + pret_unitar*v_procent
  WHERE   id_produs = v_id;

EXCEPTION

  RAISE_APPLICATION_ERROR (-20000, 'Nu exista produsul');
END;
```

### Exemplul 6.5

```
DECLARE
  v_den   categorii.denumire%TYPE :=          ;
  v_nivel categorii.nivel%TYPE :=          ;
  v_parinte categorii.id_parinte%TYPE;

  (id categorii.id_categorie%TYPE          ,
   den   categorii.denumire%TYPE,
   nivel categorii.nivel%TYPE,
   parinte categorii.id_parinte%TYPE)
IS
BEGIN
  INSERT INTO categorii
  VALUES (                                     );

  DBMS_OUTPUT.PUT_LINE('Au fost inserate '||
    SQL%ROWCOUNT ||' linii');
END;
BEGIN

END;
```

**Exemplul 6.6**

```

CREATE OR REPLACE PROCEDURE proc_ex6
(p_id          categorii.id_categorie%TYPE,
 p_den        categorii.denumire%TYPE,
 p_parinte    categorii.id_parinte%TYPE) IS
BEGIN
    SELECT denumire, id_parinte INTO
    FROM    categorii
    WHERE   id_categorie =          ;
END;
/

DECLARE
        categorii.denumire%TYPE;
        categorii.id_parinte%TYPE;

BEGIN

    DBMS_OUTPUT.PUT_LINE(          || ' ' ||          );
END;

```

**Exemplul 6.7**

```

CREATE OR REPLACE PROCEDURE proc_ex6
(p_id          categorii.id_categorie%TYPE,
 p_den        categorii.denumire%TYPE) IS
BEGIN
    SELECT denumire, id_parinte INTO
    FROM    categorii
    WHERE   id_categorie =          ;
END;
/

DECLARE
        categorii.denumire%TYPE;
        categorii.id_parinte%TYPE          ;

BEGIN

    DBMS_OUTPUT.PUT_LINE(          || ' ' ||          );
END;

```

#### Exemplul 6.8

```
BEGIN
  SELECT COUNT(DISTINCT id_produc)
  FROM    facturi f, facturi_produce fp

  AND     id_client = p_id;

END;
```

#### Exemplul 6.14

```
CREATE OR REPLACE FUNCTION func_ex14(v_id NUMBER)
RETURN VARCHAR2
IS

BEGIN
  FOR i IN (SELECT denumire
            FROM   categorii

            ) LOOP

    END LOOP;

END;
/

SELECT id_produc, denumire,

FROM   produse
WHERE  ROWNUM <=10;
```

**Exemplul 6.15**

```
DECLARE
  v_nr_trimestru NUMBER(10);
  v_nr_luna      NUMBER(10);

  (p_id produse.id_produs%TYPE,
   )

SELECT SUM(cantitate)
  INTO rezultat
  FROM facturi f, facturi_produce fp
  WHERE f.id_factura = fp.id_factura
  AND    fp.id_produs = p_id

  (p_id produse.id_produs%TYPE,
   )

SELECT SUM(cantitate)
  INTO rezultat
  FROM facturi f, facturi_produce fp
  WHERE f.id_factura = fp.id_factura
  AND    fp.id_produs = p_id

BEGIN

  DBMS_OUTPUT.PUT_LINE('Nr produse vandute in luna
    aprilie: '||v_nr_luna);

  DBMS_OUTPUT.PUT_LINE('Nr produse vandute in
    trimestrul 2: '||v_nr_trimestru);
END;
```

### Exemplul 6.16

```
CREATE OR REPLACE FUNCTION
```

```
    RETURN NUMBER
```

```
IS
```

```
BEGIN
```

```
END;
```