

Tema 3 – PL/SQL

1.

```
DECLARE
v_id wf_countries.country_id%TYPE:=1;
v_country_name wf_countries.country_name%TYPE;
BEGIN
LOOP
SELECT country_name INTO v_country_name
FROM wf_countries
WHERE country_id=v_id;
DBMS_OUTPUT.PUT_LINE('Country id: '||v_id||' Country name: '||v_country_name);
v_id:=v_id+1;
if v_id>3 THEN EXIT;
END IF;
END LOOP;
END;
```

2.

```
BEGIN
FOR v_out IN 60..65 LOOP
  FOR v_inn IN 100..110 LOOP
    DBMS_OUTPUT.PUT_LINE(v_out||'-'||v_inn);
  END LOOP;
END LOOP;
END;
```

3.

```
DECLARE
CURSOR wf_holidays IS
SELECT country_name, national_holiday_date, national_holiday_name
FROM wf_countries
WHERE region_id=5 AND national_holiday_date IS NOT NULL;
v_country_name wf_countries.country_name%TYPE;
v_national_holiday_date wf_countries.national_holiday_date%TYPE;
v_national_holiday_name wf_countries.national_holiday_name%TYPE;
BEGIN
DBMS_OUTPUT.PUT_LINE('Tarile din regiunea 5 si sarbatorile lor nationale sunt:');
OPEN wf_holidays;
LOOP
  FETCH wf_holidays INTO v_country_name, v_national_holiday_date,
  v_national_holiday_name;
  EXIT WHEN wf_holidays%NOTFOUND;
```

```
        DBMS_OUTPUT.PUT_LINE(v_country_name||' '||v_national_holiday_date||
        '||v_national_holiday_name);
END LOOP;
CLOSE wf_holidays;
END;
```

4.

```
DECLARE
CURSOR c_nr_emp IS
SELECT d.department_name, count(e.employee_id)
  FROM departments d, employees e
  WHERE d.department_id=e.department_id
GROUP BY d.department_name
ORDER BY 2 DESC;
v_dept_name departments.department_name%TYPE;
v_nr NUMBER;
BEGIN
DBMS_OUTPUT.PUT_LINE('Primele 5 departamente:');
OPEN c_nr_emp;
LOOP
  FETCH c_nr_emp INTO v_dept_name, v_nr;
  EXIT WHEN c_nr_emp%ROWCOUNT>5;
  DBMS_OUTPUT.PUT_LINE(v_dept_name||', '||v_nr);
END LOOP;
CLOSE c_nr_emp;
END;
```

5.

```
DECLARE
CURSOR wf_currencies_cur IS
SELECT currency_code, currency_name
  FROM wf_currencies
  ORDER BY currency_name;
v_curr_code wf_currencies.currency_code%TYPE;
v_curr_name wf_currencies.currency_name%TYPE;
BEGIN
OPEN wf_currencies_cur;
LOOP
  FETCH wf_currencies_cur
  INTO v_curr_code, v_curr_name;
  EXIT WHEN wf_currencies_cur%NOTFOUND;
  DBMS_OUTPUT.PUT_LINE(v_curr_code || ' ' || v_curr_name);
END LOOP;
CLOSE wf_currencies_cur;
END;
```

The solution:

```
DECLARE
CURSOR wf_currencies_cur IS
SELECT currency_code, currency_name
FROM wf_currencies
ORDER BY currency_name;
BEGIN
FOR v_wf_currencies_cur IN wf_currencies_cur
LOOP
    DBMS_OUTPUT.PUT_LINE(v_wf_currencies_cur.currency_code || ' ' ||
    v_wf_currencies_cur.currency_name);
END LOOP;
END;
```

6.

```
declare
    cursor c_evaluate is
        select * from evaluate;
```

```
maxim evaluate.nota%type;
minim evaluate.nota%type;
```

```
begin
    select min(nota) into minim from evaluate;
    select max(nota) into maxim from evaluate;
```

```
FOR rand IN c_evaluate LOOP
    exit when c_evaluate%notfound;
```

```
    if rand.nota = maxim then DBMS_OUTPUT.PUT_LINE(rand.ume_student||' are nota
    || rand.nota||', nota maxima la SGBD');
    elsif rand.nota = minim then DBMS_OUTPUT.PUT_LINE(rand.ume_student||' are
    nota || rand.nota||', nota minima la SGBD');
    else DBMS_OUTPUT.PUT_LINE(rand.ume_student||' are nota || rand.nota||' la data
    ||rand.data);
    end if;
```

```
end loop;
end;
```

SAU cu diverse variante

```
set serveroutput on;
--exercitiul 1 tema 3
```

```
/*DECLARE
v_id countries.country_id%TYPE:=1;
v_country_name countries.country_name%TYPE;
BEGIN
LOOP
-- sau cu declarare exceptie
--Begin
SELECT country_name INTO v_country_name
FROM countries WHERE country_id=v_id;

DBMS_OUTPUT.PUT_LINE('Country id: '||v_id||' Country name: '||v_country_name);
--Exception
--when no_data_found then dbms_output.put_line(' nu exista tara cu id-il '||v_id);
--End;

v_id:=v_id+1;
--if v_id>3 THEN EXIT;
exit when v_id >3; -- sau ies direct din bucla fara if
-- END IF;
END LOOP;
END;
*/
--sau cu cursor, pentru tarile cu id egal 1, 2, 3
/*declare
cursor tari is select * from countries order by country_id; -- where country_id in (1,2,3);
begin
for rand in tari loop
--exit when tari%rowcount >3;
DBMS_OUTPUT.PUT_LINE('Country id: '||rand.country_id||' Country name:
'||rand.country_name);

end loop;
end;
*/
--exercitiul 2 tema 3
/*
BEGIN
FOR v_out IN 60..65 LOOP
FOR v_inn IN 100..110 LOOP
DBMS_OUTPUT.PUT_LINE(v_out||'-'||v_inn);
END LOOP;
END LOOP;
END;
*/
--exercitiul 3 tema 3
```

```
/*DECLARE
  CURSOR holidays_cursor IS
    SELECT country_name, national_holiday_date, national_holiday_name
    FROM countries
    WHERE region_id=5 AND national_holiday_date IS NOT NULL;
  v_country_name countries.country_name%TYPE;
  v_national_holiday_date countries.national_holiday_date%TYPE;
  v_national_holiday_name countries.national_holiday_name%TYPE;
BEGIN
  DBMS_OUTPUT.PUT_LINE('Tarile din regiunea 5 si sarbatorile lor nationale sunt:');
  OPEN holidays_cursor;
  LOOP
    FETCH holidays_cursor INTO v_country_name, v_national_holiday_date,
v_national_holiday_name;
    EXIT WHEN holidays_cursor%NOTFOUND;
    DBMS_OUTPUT.PUT_LINE(v_country_name||          '||v_national_holiday_date||'
'v_national_holiday_name);
  END LOOP;
  CLOSE holidays_cursor;
END;*/
---- sau ---
/*DECLARE
  CURSOR holidays_cursor IS
    SELECT country_name, national_holiday_date, national_holiday_name
    FROM countries
    WHERE region_id=5 AND national_holiday_date IS NOT NULL;

BEGIN
  DBMS_OUTPUT.PUT_LINE('Tarile din regiunea 5 si sarbatorile lor nationale sunt:');

  FOR inregistrare_din_cursor IN holidays_cursor
  LOOP
    DBMS_OUTPUT.PUT_LINE(inregistrare_din_cursor.country_name||
'v_national_holiday_date||'
'v_national_holiday_name);
  END LOOP;

END;
*/
--exercitiul 4 tema 3
/*
DECLARE
  CURSOR c_nr_emp IS
    SELECT d.department_name, count(e.employee_id) v_nr
    FROM departments d, employees e
```

Informatica II - Sursa Oracle Academy

```
WHERE d.department_id=e.department_id
GROUP BY d.department_name
ORDER BY 2 DESC;
-- v_dept_name departments.department_name%TYPE;
-- v_nr NUMBER;
BEGIN
  DBMS_OUTPUT.PUT_LINE('Primele 5 departamente:');
-- OPEN c_nr_emp;
  FOR inregistrare IN c_nr_emp
  LOOP
--   FETCH c_nr_emp INTO v_dept_name, v_nr;
    EXIT WHEN c_nr_emp%ROWCOUNT>5;
    DBMS_OUTPUT.PUT_LINE(inregistrare.department_name||', '||inregistrare.v_nr);
  END LOOP;
-- CLOSE c_nr_emp;
END;
*/
--exercitiul 5 tema 3
/*
DECLARE
  CURSOR currencies_cursor IS
    SELECT currency_code, currency_name
    FROM currencies
    ORDER BY currency_name;
BEGIN
  FOR v_currencies_cursor IN currencies_cursor
  LOOP
    DBMS_OUTPUT.PUT_LINE(v_currencies_cursor.currency_code || ' ' ||
v_currencies_cursor.currency_name);
  END LOOP;
END;
*/
```