جامعة حلب في المناطق المحرّرة
كليّة الهندسة المعلوماتيّة
مقرّر: قواعد معطيات -2- (عملي)

المحاضرة الخامسة
الحلقات التكرارية

د. ياسر اليوسف
م. عبد الله الحايك

تطبيقات عملية

التحقق من كلمة المرور باستخدام بعض الدوال

```
SQL> set serveroutput on;
SQL> declare
  2  v_length_check BOOLEAN;
  3  v_uppercase_check BOOLEAN;
  4  v_lowercase_check BOOLEAN;
  5  v_digit_check BOOLEAN;
  6  v_password VARCHAR2(50) := '&Password';
  7  begin
  8  --  فحص الطول (يجب أن يكون على الأقل 8أحرف)
  9  v_length_check := LENGTH(v_password) >= 8;
 10  -- فحص وجود حرف كبير
 11  v_uppercase_check := REGEXP_LIKE(v_password, '[A-Z]');
 12  -- فحص وجود حرف صغير
 13  v_lowercase_check := REGEXP_LIKE(v_password, '[a-z]');
 14  -- فحص وجود رقم
 15  v_digit_check := REGEXP_LIKE(v_password, '[0-9]');
 16  if v_length_check = True AND v_uppercase_check = True AND v_lowercase_check = True AND v_digit_check = True
 17  then DBMS_OUTPUT.PUT_LINE('The password is correct');
 18  else DBMS_OUTPUT.PUT_LINE('The password is incorrect');
 19  end if;
 20  end;
 21  /
Enter value for password: Ab05sdghahfda
old   6: v_password VARCHAR2(50) := '&Password';
new   6: v_password VARCHAR2(50) := 'Ab05sdghahfda';
The password is correct
```

بنى الحلقات

```
LOOP
   statement1;
   . . .
   EXIT [WHEN condition];
END LOOP;
```

```
WHILE condition LOOP
   statement1;
   statement2;
   . . .
END LOOP;
```

```
FOR counter IN [REVERSE]
      lower_bound..upper_bound LOOP
   statement1;
   statement2;
   . . .
END LOOP;
```

تطبيقات عملية

```
SQL> set serveroutput on;
SQL> DECLARE
  2  i NUMBER := 1;
  3  BEGIN
  4  LOOP
  5  DBMS_OUTPUT.PUT_LINE('THE NUMBER: ' || i);
  6  i := i + 1;
  7  EXIT WHEN i = 6;
  8  END LOOP;
  9  END;
 10  /
THE NUMBER: 1
THE NUMBER: 2
THE NUMBER: 3
THE NUMBER: 4
THE NUMBER: 5

PL/SQL procedure successfully completed.
```

```
SQL> set serveroutput on;
SQL> DECLARE
  2  i NUMBER := 1;
  3  BEGIN
  4  WHILE i <= 5 LOOP
  5  DBMS_OUTPUT.PUT_LINE('THE NUMBER: ' || i);
  6  i := i + 1;
  7  END LOOP;
  8  END;
  9  /
THE NUMBER: 1
THE NUMBER: 2
THE NUMBER: 3
THE NUMBER: 4
THE NUMBER: 5

PL/SQL procedure successfully completed.
```

```
SQL> set serveroutput on;
SQL> DECLARE
  2  i NUMBER;
  3  BEGIN
  4  FOR i IN 1..5 LOOP
  5  DBMS_OUTPUT.PUT_LINE('THE NUMBER: ' || i);
  6  END LOOP;
  7  END;
  8  /
THE NUMBER: 1
THE NUMBER: 2
THE NUMBER: 3
THE NUMBER: 4
THE NUMBER: 5

PL/SQL procedure successfully completed.
```

تطبيقات عملية

$$sum = x + x^2 + x^3 + \cdots + x^n$$

```
SQL> SET SERVEROUTPUT ON;
SQL> DECLARE
  2  X NUMBER := &X; -- الأساس
  3  N NUMBER := &N; -- الأس
  4  S NUMBER := 0; -- المجموع
  5  P NUMBER := 1; -- خزان المضروبات
  6  BEGIN
  7  FOR i IN 1..n LOOP
  8  P := P * X;
  9  S := S + P;
 10  END LOOP;
 11  DBMS_OUTPUT.PUT_LINE('The Result is: ' || S);
 12  END;
 13  /
Enter value for x: 3
old   2: X NUMBER := &X; -- ??????
new   2: X NUMBER := 3; -- ??????
Enter value for n: 3
old   3: N NUMBER := &N; -- ????
new   3: N NUMBER := 3; -- ????
The Result is: 39

PL/SQL procedure successfully completed.
```

Composite

```
SQL> set serveroutput on
SQL> declare
  2  x emp%rowtype;
  3  begin
  4  select *  into x from emp where emp.no=5;
  5  dbms_output.put_line(x.no ||' the number of '|| x.name||' that has the salary of '|| x.sal);
  6  end;
  7  /
```

```
SQL> set serveroutput on
SQL> declare
  2  x emp%rowtype;
  3  begin
  4  select no,name,birth into x.no,x.name,x.birth from emp where emp.no=1;
  5  dbms_output.put_line(x.no ||'  '|| x.name||'     '||x.birth);
  6  end;
  7  /
```

السجلات و المصفوفات

تعبئة المصفوفة من خلال الاستعلام

```
SQL> CREATE TABLE Emp (
  2      Employee_ID NUMBER PRIMARY KEY,
  3      Employee_Name VARCHAR2(50),
  4      Salary NUMBER
  5  );

Table created.

SQL> INSERT INTO Emp VALUES (1, 'John', 50000);

1 row created.

SQL> INSERT INTO Emp VALUES (2, 'Jane', 60000);

1 row created.

SQL> INSERT INTO Emp VALUES (3, 'Bob', 55000);

1 row created.
```

```
SQL> SET SERVEROUTPUT ON;
SQL> DECLARE
  2      TYPE employee_record IS RECORD (
  3          employee_id NUMBER,
  4          employee_name VARCHAR2(50),
  5          salary NUMBER
  6      );
  7
  8      TYPE emp_array IS TABLE OF employee_record INDEX BY PLS_INTEGER;
  9      emp emp_array;
 10
 11  BEGIN
 12      -- استعلام لاسترجاع بيانات الموظفين
 13      FOR emp_rec IN (SELECT employee_id, employee_name, salary FROM emp) LOOP
 14          emp(emp_rec.employee_id).employee_id := emp_rec.employee_id;
 15          emp(emp_rec.employee_id).employee_name := emp_rec.employee_name;
 16          emp(emp_rec.employee_id).salary := emp_rec.salary;
 17      END LOOP;
 18
 19      -- طباعة بيانات الموظفين
 20      FOR i IN emp.FIRST..emp.LAST LOOP
 21          DBMS_OUTPUT.PUT_LINE('employee_id: ' || emp(i).employee_id || ', employee_name: ' || emp(i).employee_name || ', salary: ' || emp(i).salary);
 22      END LOOP;
 23  END;
 24  /
employee_id: 1, employee_name: John, salary: 50000
employee_id: 2, employee_name: Jane, salary: 60000
employee_id: 3, employee_name: Bob, salary: 55000

PL/SQL procedure successfully completed.
```