### **✅ محاضرة PL/SQL عملي**

#### **: أنواع البيانات في PL/SQL**

* **أنواع بسيطة (Scalar):** تخزن قيمة واحدة فقط (NUMBER) ،( VARCHAR2).
* **أنواع مركبة (Composite):** تخزن أكثر من قيمة:
  + RECORD
  + %ROWTYPE

“البيانات المركبة في PL/SQL”

* **البيانات المركبة** = نوع بيانات بيخزن أكثر من قيمة في متغير واحد.
* أمثلة في الحياة العملية: بيانات موظف، بيانات طالب، بيانات طلب شراء.

**: RECORD**

* **التعريف:** نوع بيانات مركب بننشئه بأنفسنا عشان يخزن أكثر من حقل.
* **حالات الاستخدام:**
  + صف واحد من جدول (مش كل الأعمدة).
  + نخزن بيانات مرتبطة في متغير واحد.
* **مثال بسيط:**

**DECLARE  
 TYPE emp\_rec IS RECORD (  
 id NUMBER,  
 name VARCHAR2(50),  
 salary NUMBER  
 );  
 v\_emp emp\_rec;  
BEGIN  
 v\_emp.id := 101;  
 v\_emp.name := 'Ali';  
 v\_emp.salary := 5000;  
END;**

#### **%ROWTYPE**

* **التعريف:** متغير بياخد شكل صف كامل من جدول أو نتيجة Cursor.
* **حالات الاستخدام:**
  + لما نحتاج كل الأعمدة من جدول.
* **مثال:**

DECLARE  
 v\_emp employees%ROWTYPE;  
BEGIN  
 SELECT \* INTO v\_emp FROM employees WHERE emp\_id = 101;  
END;

**: جدول مقارنة**

|  |  |
| --- | --- |
| النوع | متى نستخدمه |
| RECORD | صف واحد + أعمدة محددة |
| %ROWTYPE | صف واحد + كل الأعمدة |

**“إنتوا مبرمجين في شركة SmartTech، الشركة عندها قاعدة بيانات للموظفين والأقسام والمشاريع. المدير طلب تقارير مختلفة.”**

#### **الجداول المستخدمة**

**EMPLOYEES:** (EMP\_ID, EMP\_NAME, SALARY, DEPT\_ID)

**DEPARTMENTS:** (DEPT\_ID, DEPT\_NAME)

**PROJECTS:** (PROJ\_ID, PROJ\_NAME, EMP\_ID)

**-- 1️⃣ إنشاء جدول DEPARTMENTS**

**CREATE TABLE DEPARTMENTS (**

**DEPT\_ID NUMBER PRIMARY KEY,**

**DEPT\_NAME VARCHAR2(50)**

**);**

**-- 2️⃣ إنشاء جدول EMPLOYEES مع مفتاح أجنبي يشير لـ DEPARTMENTS**

**CREATE TABLE EMPLOYEES (**

**EMP\_ID NUMBER PRIMARY KEY,**

**EMP\_NAME VARCHAR2(50),**

**SALARY NUMBER,**

**DEPT\_ID NUMBER,**

**CONSTRAINT FK\_EMP\_DEPT FOREIGN KEY (DEPT\_ID)**

**REFERENCES DEPARTMENTS(DEPT\_ID)**

**);**

**-- 3️⃣ إنشاء جدول PROJECTS مع مفتاح أجنبي يشير لـ EMPLOYEES**

**CREATE TABLE PROJECTS (**

**PROJ\_ID NUMBER PRIMARY KEY,**

**PROJ\_NAME VARCHAR2(50),**

**EMP\_ID NUMBER,**

**CONSTRAINT FK\_PROJ\_EMP FOREIGN KEY (EMP\_ID)**

**REFERENCES EMPLOYEES(EMP\_ID)**

**);**

**ادخال السجلات**

-- جدول DEPARTMENTS

INSERT INTO DEPARTMENTS (DEPT\_ID, DEPT\_NAME) VALUES (1, 'HR');

INSERT INTO DEPARTMENTS (DEPT\_ID, DEPT\_NAME) VALUES (2, 'IT');

INSERT INTO DEPARTMENTS (DEPT\_ID, DEPT\_NAME) VALUES (3, 'Finance');

INSERT INTO DEPARTMENTS (DEPT\_ID, DEPT\_NAME) VALUES (4, 'Marketing');

INSERT INTO DEPARTMENTS (DEPT\_ID, DEPT\_NAME) VALUES (5, 'Sales');

-- جدول EMPLOYEES

INSERT INTO EMPLOYEES (EMP\_ID, EMP\_NAME, SALARY, DEPT\_ID) VALUES (101, 'Ahmed', 3000, 1);

INSERT INTO EMPLOYEES (EMP\_ID, EMP\_NAME, SALARY, DEPT\_ID) VALUES (102, 'Sara', 3500, 2);

INSERT INTO EMPLOYEES (EMP\_ID, EMP\_NAME, SALARY, DEPT\_ID) VALUES (103, 'Omar', 3200, 3);

INSERT INTO EMPLOYEES (EMP\_ID, EMP\_NAME, SALARY, DEPT\_ID) VALUES (104, 'Laila', 2800, 4);

INSERT INTO EMPLOYEES (EMP\_ID, EMP\_NAME, SALARY, DEPT\_ID) VALUES (105, 'Mona', 4000, 5);

-- جدول PROJECTS

INSERT INTO PROJECTS (PROJ\_ID, PROJ\_NAME, EMP\_ID) VALUES (201, 'Project A', 101);

INSERT INTO PROJECTS (PROJ\_ID, PROJ\_NAME, EMP\_ID) VALUES (202, 'Project B', 102);

INSERT INTO PROJECTS (PROJ\_ID, PROJ\_NAME, EMP\_ID) VALUES (203, 'Project C', 103);

INSERT INTO PROJECTS (PROJ\_ID, PROJ\_NAME, EMP\_ID) VALUES (204, 'Project D', 104);

INSERT INTO PROJECTS (PROJ\_ID, PROJ\_NAME, EMP\_ID) VALUES (205, 'Project E', 105);

-- احفظ التغييرات

COMMIT;

#### **(RECORD)**

* المطلوب: جلب بيانات موظف محدد.
* الكود:

DECLARE  
 TYPE emp\_rec IS RECORD (  
 id employees.emp\_id%TYPE,  
 name employees.emp\_name%TYPE,  
 salary employees.salary%TYPE  
 );  
 v\_emp emp\_rec;  
BEGIN  
 SELECT emp\_id, emp\_name, salary INTO v\_emp.id, v\_emp.name, v\_emp.salary  
 FROM employees WHERE emp\_id = 101;  
 DBMS\_OUTPUT.PUT\_LINE('Name: ' || v\_emp.name);  
END;

#### **(%ROWTYPE)**

* المطلوب: جلب صف كامل لموظف.
* الكود:

DECLARE  
 v\_emp employees%ROWTYPE;  
BEGIN  
 SELECT \* INTO v\_emp FROM employees WHERE emp\_id = 101;  
 DBMS\_OUTPUT.PUT\_LINE('Name: ' || v\_emp.emp\_name);  
END;