

TP PL/SQL

Ex1

a)

```
SET SERVEROUTPUT ON;
DECLARE
    a NUMBER;
    b NUMBER;
BEGIN
    a := &a;
    b := &b;
    dbms_output.put_line('La somme est : ' || (a + b));
END;
/
```

b)

```
SET SERVEROUTPUT ON;
DECLARE
    n NUMBER;
    i NUMBER := 1;
BEGIN
    n := &n;
    WHILE i <= 10 LOOP
        dbms_output.put_line(n || ' x ' || i || ' = ' || (n * i));
        i := i + 1;
    END LOOP;
END;
/
```

c)

```
CREATE OR REPLACE FUNCTION puissance(x IN NUMBER, n IN NUMBER) RETURN
NUMBER IS
BEGIN
    IF n = 0 THEN
        RETURN 1;
    ELSE
        RETURN x * puissance(x, n - 1);
    END IF;
END;
```

```
END;  
/
```

d)

```
-- Création de la table  
CREATE TABLE resultatFactoriel (  
    nombre NUMBER PRIMARY KEY,  
    factorielle NUMBER  
);  
  
-- Bloc PL/SQL  
SET SERVEROUTPUT ON;  
DECLARE  
    n NUMBER;  
    res NUMBER := 1;  
BEGIN  
    n := &n;  
    FOR i IN 1..n LOOP  
        res := res * i;  
    END LOOP;  
    INSERT INTO resultatFactoriel (nombre, factorielle) VALUES (n,  
res);  
    dbms_output.put_line('Factorielle de ' || n || ' est ' || res);  
END;  
/
```

e)

```
-- Création de la table  
CREATE TABLE resultatsFactoriels (  
    nombre NUMBER PRIMARY KEY,  
    factorielle NUMBER  
);  
  
-- Bloc PL/SQL  
SET SERVEROUTPUT ON;  
DECLARE  
    res NUMBER;  
BEGIN  
    FOR n IN 1..20 LOOP
```

```

        res := 1;
        FOR i IN 1..n LOOP
            res := res * i;
        END LOOP;
        INSERT INTO resultatsFactoriels (nombre, factorielle) VALUES
(n, res);
        dbms_output.put_line('Factorielle de ' || n || ' est ' || res);
    END LOOP;
END;
/

```

Ex2

a)

```

SET SERVEROUTPUT ON;
DECLARE
    v_employe emp%ROWTYPE;
BEGIN
    v_employe.matr := 4;
    v_employe.nom := 'Youcef';
    v_employe.sal := 2500;
    v_employe.adresse := 'avenue de la République';
    v_employe.dep := 92002;
    INSERT INTO emp VALUES v_employe;
END;
/

```

b)

```

SET SERVEROUTPUT ON;
DECLARE
    v_nb_lignes NUMBER;
BEGIN
    DELETE FROM emp WHERE dep = 10;
    v_nb_lignes := SQL%ROWCOUNT;
    dbms_output.put_line('Nombre de lignes supprimées : ' ||
v_nb_lignes);
END;
/

```

c)

```

SET SERVEROUTPUT ON;
DECLARE

```

```

    v_salaire emp.sal%TYPE;
    v_total emp.sal%TYPE := 0;
    CURSOR c_salaires IS SELECT sal FROM emp;
BEGIN
    OPEN c_salaires;
    LOOP
        FETCH c_salaires INTO v_salaire;
        EXIT WHEN c_salaires%NOTFOUND;
        IF v_salaire IS NOT NULL THEN
            v_total := v_total + v_salaire;
        END IF;
    END LOOP;
    CLOSE c_salaires;
    dbms_output.put_line('Somme des salaires : ' || v_total);
END;
/

```

d)

```

SET SERVEROUTPUT ON;
DECLARE
    v_salaire emp.sal%TYPE;
    v_total emp.sal%TYPE := 0;
    v_count NUMBER := 0;
    CURSOR c_salaires IS SELECT sal FROM emp;
BEGIN
    OPEN c_salaires;
    LOOP
        FETCH c_salaires INTO v_salaire;
        EXIT WHEN c_salaires%NOTFOUND;
        IF v_salaire IS NOT NULL THEN
            v_total := v_total + v_salaire;
            v_count := v_count + 1;
        END IF;
    END LOOP;
    CLOSE c_salaires;
    dbms_output.put_line('Salaire moyen : ' || (v_total / v_count));
END;
/

```

e)

```

SET SERVEROUTPUT ON;
DECLARE

```

```

    v_total emp.sal%TYPE := 0;
BEGIN
    FOR rec IN (SELECT sal FROM emp) LOOP
        IF rec.sal IS NOT NULL THEN
            v_total := v_total + rec.sal;
        END IF;
    END LOOP;
    dbms_output.put_line('Somme des salaires : ' || v_total);
END;
/

-- Pour le salaire moyen
SET SERVEROUTPUT ON;
DECLARE
    v_total emp.sal%TYPE := 0;
    v_count NUMBER := 0;
BEGIN
    FOR rec IN (SELECT sal FROM emp) LOOP
        IF rec.sal IS NOT NULL THEN
            v_total := v_total + rec.sal;
            v_count := v_count + 1;
        END IF;
    END LOOP;
    dbms_output.put_line('Salaire moyen : ' || (v_total / v_count));
END;
/

```

f)

```

SET SERVEROUTPUT ON;
DECLARE
    CURSOR c(p_dep emp.dep%TYPE) IS
        SELECT dep, nom FROM emp WHERE dep = p_dep;
BEGIN
    FOR v_employe IN c(92000) LOOP
        dbms_output.put_line('Dep 92000 : ' || v_employe.nom);
    END LOOP;

    FOR v_employe IN c(75000) LOOP
        dbms_output.put_line('Dep 75000 : ' || v_employe.nom);
    END LOOP;
END;
/

```


Ex3

a)

```
-- Spécification
CREATE OR REPLACE PACKAGE gestion_clients AS
    PROCEDURE ajouter_client(p_id IN NUMBER, p_nom IN VARCHAR2);
    PROCEDURE ajouter_client(p_id IN NUMBER, p_nom IN VARCHAR2,
p_adresse IN VARCHAR2);
END gestion_clients;
/

-- Corps du package
CREATE OR REPLACE PACKAGE BODY gestion_clients AS

    PROCEDURE ajouter_client(p_id IN NUMBER, p_nom IN VARCHAR2) IS
    BEGIN
        INSERT INTO client (id, nom) VALUES (p_id, p_nom);
        dbms_output.put_line('Client ajouté sans adresse.');
```

```
    EXCEPTION
        WHEN OTHERS THEN
            dbms_output.put_line('Erreur : ' || SQLERRM);
    END;

    PROCEDURE ajouter_client(p_id IN NUMBER, p_nom IN VARCHAR2,
p_adresse IN VARCHAR2) IS
    BEGIN
        INSERT INTO client (id, nom, adresse) VALUES (p_id, p_nom,
p_adresse);
        dbms_output.put_line('Client ajouté avec adresse.');
```

```
    EXCEPTION
        WHEN OTHERS THEN
            dbms_output.put_line('Erreur : ' || SQLERRM);
    END;

END gestion_clients;
/
```