

## TP FUNCTIONS Solutions :

```
--function number to day

CREATE OR REPLACE FUNCTION toDAY(n INTEGER)RETURN VARCHAR2 IS
dayR VARCHAR2(50):= '';
BEGIN
    CASE n
        When 0 Then dayR := 'Sunday';
        When 1 Then dayR := 'Monday';
        WHEN 2 THEN dayR:= 'Tuesday';
        WHEN 3 THEN dayR:= 'Wednesday';
        WHEN 4 THEN dayR:= 'Thursday';
        WHEN 5 THEN dayR:= 'Friday';
        WHEN 6 THEN dayR:= 'Saturday';
        Else return 'Invalid Number!';
    END CASE;
    return dayR;
END;

--nombre de jours entre 2
CREATE OR REPLACE FUNCTION BetweenTwo(date1 date , date2 date) RETURN INTEGER
IS
    result NUMBER := ABS(TRUNC(date1) - TRUNC(date2));
    BEGIN
        return result;
    END;

--NAME CONDUCTEUR
CREATE OR REPLACE FUNCTION nomConducteur(numero NUMBER) RETURN VARCHAR2 IS
nomC VARCHAR2(100) := '';
BEGIN
    SELECT CNOM INTO nomC FROM CONDUCTEUR WHERE CNUM = numero;
    RETURN nomC;
END;

--procedure AUGMENTE
set SERVEROUTPUT ON
CREATE OR REPLACE PROCEDURE AUGMENTE (aug IN OUT number,numero NUMBER) IS
salaireActuel NUMBER;
BEGIN
SELECT SALAIRE INTO salaireActuel FROM CONDUCTEUR WHERE cNUM = numero;
salaireActuel := salaireActuel + aug;

UPDATE CONDUCTEUR SET Salaire = salaireActuel WHERE cNUM = numero;
aug := salaireActuel;
dbms_output.put_line('Done! Nouveau Salaire : ' || aug);
END;

--Factoriel

CREATE OR REPLACE FUNCTION FACTORIEL(N number) RETURN NUMBER IS
```

```

BEGIN

    IF(N = 0) THEN
        return 1;
    ELSE
        return N * FACTORIEL(N-1);
    END IF;
END;

---tests
SELECT SALAIRE FROM CONDUCTEUR ;
SELECT * FROM CONDUCTEUR;
SELECT FACTORIEL(5) FROM DUAL;
Declare
augm NUMBER := 2000;
BEGIN
AUGMENTE(augm,1);
END;
--test
SELECT nomConducteur(1) FROM DUAL;
SELECT BETWEEN TWO(TO_DATE('2023-11-15','YYYY-MM-DD'),TO_DATE('2023-11-21','YYYY-MM-DD')) from dual;
SELECT toDay(10) from dual;

```