COISSARD Kévin – ASPE

TP 4 – PL / SQL

```
set serveroutput on;
drop table jobs;
create table jobs as select * from hr.jobs;
drop table departments;
create table departments as select * from hr.departments;
drop table employees;
create table employees as select * from hr.employees;
```

-- Question 1

```
create or replace procedure ajout_job (p_id Varchar2, p_title Varchar2) is
BEGIN
  insert into jobs (job_id, job_title)
  values (p_id, p_title);
  commit;
END;
exec ajout_job('PR', 'Professeur');
```

-- Question 2

```
create or replace procedure modif_job (p_id varchar2, p_newtitle varchar2) is
    e_nojob exception;
BEGIN
    update jobs
    set job_title = p_newtitle
    where job_id = p_id;
    if (sql%rowcount = 0) then
        raise e_nojob;
    end if;
    commit;
    exception
```

```
      when e_nojob then
         raise_application_error(-20002, 'Le job ' || p_id || ' est inexistant.');
END;
exec modif_job('PR', 'Professionnel');
```

-- Question 3

```
create or replace procedure liste_emp_mgr is
  cursor cr_emp is
    select e1.first_name as emp_fn, e1.last_name as emp_ln, e2.first_name as mgr_fn, e2.last_name as mgr_ln
    from employees e1, employees e2
    where e2.employee_id = e1.manager_id;
BEGIN
  dbms_output.put_line('Employé  -  Manager');
  dbms_output.put_line('--------------------');
  for v_emp in cr_emp loop
    dbms_output.put_line(v_emp.emp_fn || ' ' || v_emp.emp_ln || ' - ' || v_emp.mgr_fn || ' ' || v_emp.mgr_ln);
  end loop;
  dbms_output.put_line(' ');
END;
exec liste_emp_mgr;
```

-- Question 4

```
create or replace procedure liste_emp_make_more_than (p_name varchar2) is
  cursor cr_emp (p_name varchar2) is
    select last_name
    from employees
    where salary + decode(commission_pct, null, 0, commission_pct * salary) > (
      select salary + decode(commission_pct, null, 0, commission_pct * salary)
```

```
    from employees
    where last_name = p_name);
BEGIN
 dbms_output.put_line('Nom');
 dbms_output.put_line('----------');
 for v_emp in cr_emp(p_name) loop
  dbms_output.put_line(v_emp.last_name);
 end loop;
 dbms_output.put_line(' ');
END;
exec liste_emp_make_more_than('Russell');
```

-- Question 5

```
create or replace procedure liste_same_job_make_more_than (p_name1 varchar2, p_name2
varchar2) is
 cursor cr_emp (p_name2 varchar2) is
   select last_name
   from employees
   where job_id in (
    select job_id
    from employees
    where last_name = p_name1)
   and salary > (
    select salary
    from employees
    where last_name = p_name2);
BEGIN
 dbms_output.put_line('Nom');
 dbms_output.put_line('----------');
 for v_emp in cr_emp(p_name2) loop
```

```
   dbms_output.put_line(v_emp.last_name);
  end loop;
  dbms_output.put_line(' ');
END;
exec liste_same_job_make_more_than('Kochhar', 'Fox');
```

-- Question 6

```
create or replace procedure liste_emp_bigger_sal (p_n number) is
  cursor cr_emp(p_n number) is
    select * from (
      select last_name
      from employees
      order by salary desc)
    where rownum <= p_n;
BEGIN
  dbms_output.put_line('Nom');
  dbms_output.put_line('----------');

  for v_emp in cr_emp(p_n) loop
    dbms_output.put_line(v_emp.last_name);
  end loop;
  dbms_output.put_line(' ');
END;

exec liste_emp_bigger_sal(5);
```

-- Question 7

```
create or replace procedure depts_without_emp is
```

```
  cursor cr_dept is
    select department_name as dname
    from departments
    where department_id not in (
      select distinct decode(department_id, null, 0, department_id)
      from employees)
    order by department_name desc;
BEGIN
  dbms_output.put_line('Département');
  dbms_output.put_line('-----------');

  for v_dept in cr_dept loop
    dbms_output.put_line(v_dept.dname);
  end loop;
  dbms_output.put_line(' ');
END;
exec depts_without_emp;
```

-- Question 8

```
create or replace procedure emp_ranking (p_n number) is
  cursor cr_emp(p_n number) is
    select * from (
      select last_name
      from employees
      where employee_id >= p_n
      order by employee_id asc);
BEGIN
  dbms_output.put_line('Nom');
  dbms_output.put_line('----------');
```

```
 for v_emp in cr_emp(p_n) loop
  dbms_output.put_line(v_emp.last_name);
 end loop;
 dbms_output.put_line(' ');
END;
exec emp_ranking(200);
```

-- Question 9

```
drop view dept_salary;
create view dept_salary as (
 select departments.department_id, department_name, sum(salary) as sum_sal
 from departments, employees
 where departments.department_id = employees.department_id
 group by departments.department_id, department_name);

create or replace procedure sum_sal_dept_greater_than (p_n number) is
 cursor cr_dept_sal(p_n number) is
  select * from (
   select department_id, department_name
   from dept_salary
   where sum_sal >= p_n
   order by department_id asc);
BEGIN
 dbms_output.put_line('Identifiant  -  Nom');
 dbms_output.put_line('--------------------');
 for v_dept_sal in cr_dept_sal(p_n) loop
  dbms_output.put_line(v_dept_sal.department_id || ' - ' || v_dept_sal.department_name);
 end loop;
 dbms_output.put_line(' ');
```

```
END;
exec sum_sal_dept_greater_than (10);
```

-- Question 10

```
create or replace procedure list_emp_bigger_sal_than_avg is
  cursor cr_emp_avg_sal is
    select last_name, dept_id, salary, avg_sal from employees, (
      select nvl(department_id,0) as dept_id, round(avg(salary),2) as avg_sal
      from employees
      group by nvl(department_id,0))
    where dept_id = employees.department_id
      and salary >= avg_sal
    order by dept_id;
BEGIN
  dbms_output.put_line('Nom');
  dbms_output.put_line('----------');

  for v_emp in cr_emp_avg_sal loop
    dbms_output.put_line(v_emp.last_name);
  end loop;
  dbms_output.put_line(' ');
END;
exec list_emp_bigger_sal_than_avg;
```

-- Question 11

```
create or replace function check_sal(p_empno employees.employee_id%type)
```

```
return Boolean is
  v_dept_id employees.department_id%type;
  v_sal employees.salary%type;
  v_avg_sal employees.salary%type;
BEGIN
  select salary, department_id into v_sal, v_dept_id from employees
  where employee_id = p_empno;
  select avg(salary) into v_avg_sal from employees
  where department_id = v_dept_id;
… ?
```