Exam Advanced Databases and Big Data

II.2314

January 2021

Before starting, carefully read the following instructions:

- This is a ***closed-book*** exam, meaning that you cannot use additional resources or notes.
- You **MUST return this document** along with your answers and it is strictly forbidden to leave the classroom with any document (original or copy)!
- read questions carefully and answer all parts as completely as possible. Vague, off-subject, or illegible answers **will not earn credit**.
- This exam has a duration of **three hours**.
- You must fill answer in appropriate **empty blocks after each answer**. If there is not enough space, you can add additional sheets.
- **Write your name** in the footer of each page.

Lastname: **Firstname:**

Block PL/SQL

```
DECLARE
        -- constants/variables declaration
BEGIN
        --Commands/instructions
EXCEPTION
        --Processing runtime errors
END;
/
```

Declare Variable

```
Variable_Name VARIABLE_Type;
```

Affect value

```
Variable_Name := value;
SELECT attribute INTO Variable_ Name
FROM table;
```

Triggers

```
CREATE [OR REPLACE] TRIGGER
Trigger_name
{BEFORE|AFTER}
{INSERT|DELETE|UPDATE} ON
<table_name>
[FOR EACH ROW]
<trigger_body (PL/SQL)>
```

Declare Exception

```
Exception_Name EXCEPTION;
```

Raise Exception

```
RAISE Exception_Name;
```

Treat Exception

```
WHEN Exception_Name THEN --
Instructions
```

Function

```
CREATE [OR REPLACE ] FUNCTION
nom_fonction
        [ argument [ IN ] type, ...]]
RETURN return_type
[ IS | AS ] block PL/SQL
```

Cursor

```
CURSOR cursor_name
IS
  SELECT_statement;
```

Fetch statement

```
FETCH cursor_name INTO variable_name;
```

PL/SQL Cheat sheet

Lastname: **Firstname:**

Part I: Advanced SQL and PL/SQL

1. What is a database index? Name a data structure used to maintain indexes.

2. Is the response time (in seconds) a good metric to analyze queries? Why?

Lastname: **Firstname:**

3. What is a database *transaction* and what is a database *lock*?

4. What is a *role* in a database and what it is useful for?

5. What is the relationship between database *views* and database *security*?

Lastname: **Firstname:**

Consider the following table for questions 6, 7, 8, 9 and 10:

```
CREATE TABLE employees (
   employee_id NUMBER(6) CONSTRAINT emp_id_nn NOT NULL,
   first_name VARCHAR2(20),
   last_name VARCHAR2(25) CONSTRAINT emp_last_name_nn NOT NULL,
   email VARCHAR2(25) CONSTRAINT emp_email_nn NOT NULL,
   phone_number VARCHAR2(20),
   hire_date DATE CONSTRAINT emp_hire_date_nn NOT NULL,
   job_id VARCHAR2(10) CONSTRAINT emp_job_nn NOT NULL,
   salary NUMBER(8,2),
   commission_pct NUMBER(2,2),
   manager_id NUMBER(6),
   department_id NUMBER(4),
   CONSTRAINT emp_salary_min CHECK (salary > 0),
   CONSTRAINT emp_email_uk UNIQUE (email)
 ) ;
```

6. Write a SQL query that prints for each employee, its last name followed by the last name of her manager. Note that manager are also employees.

Lastname: **Firstname:**

7. Create a PL/SQL block that, given an employee's id, inserts an asterisk in the stars column for every $1000 if the employee's salary. For example, if the employee earns $8000, the string of asterisks should contain eight asterisks. If the employee earns $12500, the string of asterisks should contain 13 asterisks. Hints:

- NVL(*e1*, *e2*): The Oracle NVL() function allows you to replace null with a more meaningful alternative in the results of a query. The NVL() function accepts two arguments. If *e1* evaluates to null, then NVL() function returns *e2*. If *e1* evaluates to non-null, the NVL() function returns *e1*.
- ROUND(*n*): function to round a number *n*
- *s1* || *s2*: operator for concatenating strings *s1* and *s2*.

Lastname: **Firstname:**

8. Create a PL/SQL block that does the following:

(a) In the declarative section, declare a variable *v_deptno* of type NUMBER and assign a value that holds the department ID. You can assign the value 10 to it.

(b) Declare a cursor, *emp_cursor*, that retrieves the last_name, salary, and manager_id of the employees working in the department specified in *v_deptno*.

(c) In the executable section, use the cursor FOR loop to operate on the data retrieved. If the salary of the employee is less than 5000 and if the manager ID is either 101 or 124, display the message "<<last_name>> due for a raise." Otherwise, display the message "<<last_name>> not due for a raise."

Lastname: **Firstname:**

9. Create a PL/SQL block that determines the top *n* salaries of the employees. Follow the next instructions:

b) In the declarative section, declare a variable *v_num* of type NUMBER that holds a number *n* representing the number of top *n* earners from the employees table. For example, to view the top five salaries, assign 5 to *n*. Declare another variable *sal* of type employees.salary. Declare a cursor, *c_emp_cursor*, that retrieves the salaries of employees in descending order. Remember that the salaries should not duplicated.

c) In the executable section, open the loop and fetch top *n* salaries and insert them into the top_salaries table. You can use a simple loop to operate the data. Also, use %ROWCOUNT and %FOUND attributes for the exit condition. Indeed, make sure you add an exit condition to avoid having an infinite loop.

Lastname: **Firstname:**

10. Write a trigger that verifies that the inserted value for an employee's salary is equal or lesser than $100000. If it is greater, it raises the exception INVALID_SALARY.

Lastname: **Firstname:**

Part II: Big Data

11. According to you, what led to the emergence of Big Data?

12. What defines Big Data? Explain the Big Data's 3 Vs.

Lastname: **Firstname:**

13. Do you know other characteristics "V" (for example, the 5 Vs) associated to Big Data? If you do, which ones?

14. What are horizontal and vertical scaling? What are their advantages and disadvantages?

15. What does A.C.I.D. mean?

Lastname: **Firstname:**

16. Explain the CAP theorem

17. Cite 4 types of NoSQL databases, give an example of application and an example of a database management system for each type.

Lastname: **Firstname:**

Transform the following MongoDB queries into SQL (or explain what they display):

18. db.person.find{ $or: [{ sexe: "F" } ,{ age: 50 }] })

19. db.person.find({ user_id: { $exists: true } }).count()

20. db.person.update({ age: { $gt: 25 } }, { $set: { sexe: "F" } }, { multi: true })

Lastname: Firstname: