

*Advanced SQL*

Training Assignments

|  |  |
| --- | --- |
| Program Code | ASQL |
| Version | 3.1 |
| Effective Date | 01/11/2016 |

**Hanoi, 11/2016**

RECORD OF CHANGES

\*A - Added M - Modified D - Deleted

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Date | Changes | A\* M, D | Contents | Version |
| 14-Oct-2016 | Create | A | Add the new assignments. | v1.0 |
| 14-Oct-2018 | Update | M | Template. | v1.1 |
| 01-Jun-2019 | Update | M | Update Objective | v1.2 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Contents

[For the following assignments: 4](#_Toc17270521)

[Day 1. Unit 1: Advanced DML Statements 4](#_Toc17270522)

[Assignment 1\_Opt3: Employee Management System 4](#_Toc17270523)

|  |  |
| --- | --- |
|  | **CODE: ASQL\_Assignment1\_Opt3**  **TYPE: Medium**  **LOC: n/a**  **DURATION: 180 MINUTES** |

# For the following assignments:

* Print out respectively the screenshots to show the query results.
* Pack screenshots and SQL scripts or your answers into the zip file named ASQL\_Assignment1\_AccountName.zip (for instance: ASQL\_Assignment1\_NamNT.zip) then handle to the evaluator via email ([XYZ@fsoft.com.vn](mailto:XYZ@fsoft.com.vn) ) or follow the guidance of the class admin.

# Day 1. Unit 1: Advanced DML Statements

## Assignment 1\_Opt3: Employee Management System

**Barems**: a - 10%, b - 10%, c - 10%, d - 10%, e - 15%, f - 15%, g - 15%, h - 15%

**Objectives**: H5SD - SQL skills

**Problem Descriptions**:

In the database design for the Employee Management System of FSOFT Company, given tables of the application with below:

**EMPLOYEE**:

* EmpNo: employee code, primary key.
* EmpName: employee name.
* BirthDay: birth day of the employee.
* Email: email of the employee. Make sure that the database will not allow the value for Email to be inserted into a new row if that value has already been used in another row.
* DeptNo: department code of the employee.
* MgrNo: manager code, not null (default values to 0).
* StartDate: starting date of work.
* Salary: salary of the employee, data type is money (VND).
* Level: level of the employee (accepts value range from 1 to 7 only).
* Status: status of the employee (0: working, 1: unpaid leave, 2: out, default values to 0).
* Note: some note about employee, free text.

**EMP\_SKILL**:

* SkillNo: skill code, foreign key.
* EmpNo: employee code, foreign key.
* SkillLevel: skill level of the employee (accepts value range from 1 to 3 only).
* RegDate: registration date.
* Description: skill description, free text.

Primary key (SkillNo, EmpNo)

**SKILL**:

* SkillNo: skill code, primary key, auto increment.
* SkillName: name of skill.
* Note: some note about skill, free text.

**DEPARTMENT**:

* DeptNo: department code, primary key, auto increment.
* DeptName: department name.
* Note: some note about department, free text.

**Questions:**

1. Create the tables with the most appropriate/economic field/column constraints & types. Add at least 8 records into each created tables.
2. Specify name, email and department name of the employees that have been working at least six months.
3. Specify the names of the employees whore have either ‘C++’ or ‘.NET’ skills.
4. List allemployee names, manager names, manager emails of those employees.
5. Specify the departments which have >=2 employees, print out the list of departments’ employees right after each department.
6. List all name, email and skill number of the employees and sortascending order by employee’s name.
7. Use SUB-QUERY technique to list out the different employees (include name, email, birthday) who are working and have multiple skills.
8. Create a view to list all employees are working (include: name of employee and skill name, department name)

**-- THE END --**