DATA BASES I

ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ
UNIVERSITY OF WEST ATTICA

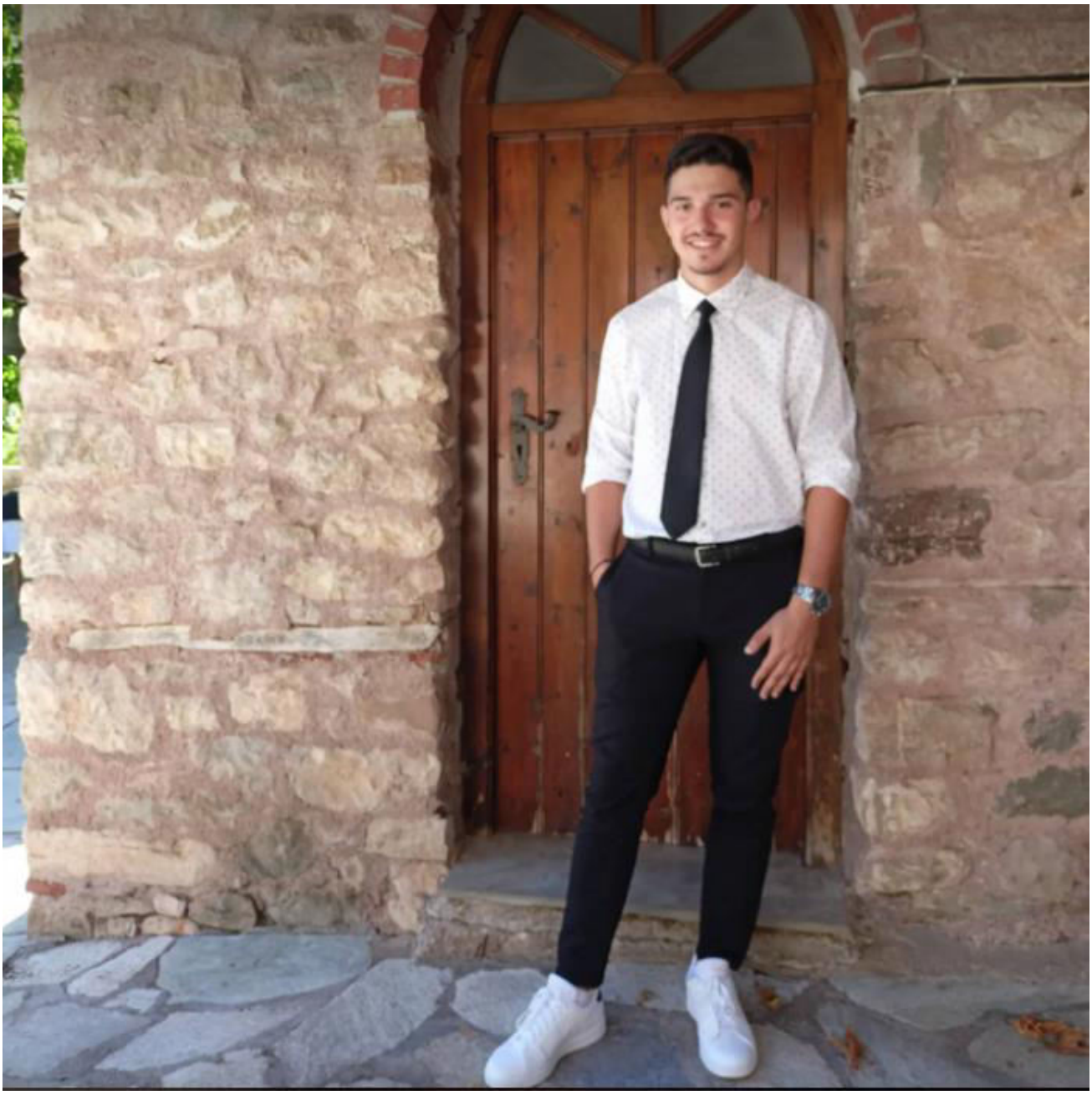
DEPARTMENT OF INFORMATION AND COMPUTER ENGINEERING

TASK 3
CLASSIFICATION AND SUGGESTIONS GROUP BY AND HAVING . JOIN

STUDENT DETAILS

NAME: ATHANASIOU VASILEIOS EVANGELOS
STUDENT ID: 19390005
STUDENT **SEMESTER:** 8th
STUDENT STATUS : UNDERGRADUATE
STUDY PROGRAM : UNIWA
LABORATORY DEPARTMENT : [2.1] WEDNESDAY 15:00 – 16:00
LABORATORY MANAGER : TSOLAKIDIS ANASTASIOS
DELIVERY DATE : 7/6/2023

STUDENT PHOTO:



CREATION BASE new_personnel

COMMANDS

```
DROP DATABASE IF EXISTS new_personnel;

CREATE DATABASE IF NOT EXISTS new_personnel;

USE new_personnel;


CREATE TABLE IF NOT EXISTS DEPT(DEPTNO INT(2) NOT NULL, DNAME
VARCHAR(14), LOC VARCHAR(14), PRIMARY KEY(DEPTNO));

INSERT INTO DEPT (DEPTNO, DNAME, LOC) VALUES (10, 'ACCOUNTING',
'ATHENS');

INSERT INTO DEPT (DEPTNO, DNAME, LOC) VALUES (20, 'SALES', 'LONDON');

INSERT INTO DEPT (DEPTNO, DNAME, LOC) VALUES (30, 'RESEARCH', 'ATHENS');

INSERT INTO DEPT (DEPTNO, DNAME, LOC) VALUES (40, 'PAYROLL', 'LONDON');

SELECT * FROM DEPT;


CREATE TABLE IF NOT EXISTS EMP (EMPNO INT(2) NOT NULL, ENAME VARCHAR(14),
JOB VARCHAR(14), HIREDATE DATE, MGR INT(2), SAL INT(4), COMM INT(3),
DEPTNO INT (2) NOT NULL, PRIMARY KEY(EMPNO), FOREIGN KEY(DEPTNO)
REFERENCES DEPT(DEPTNO));

INSERT INTO EMP (EMPNO, ENAME, JOB, HIREDATE, MGR, SAL, COMM, DEPTNO)
VALUES (10, 'CODD', 'ANALYST', '89/1/1', 15, 3000, NULL, 10);

INSERT INTO EMP (EMPNO, ENAME, JOB, HIREDATE, MGR, SAL, COMM, DEPTNO)
VALUES (15, 'ELMASRI', 'ANALYST', '95/5/2', 15, 1200, 150, 10);

INSERT INTO EMP (EMPNO, ENAME, JOB, HIREDATE, MGR, SAL, COMM, DEPTNO)
VALUES (20, 'NAVATHE', 'SALESMAN', '77/7/7', 20, 2000, NULL, 20);

INSERT INTO EMP (EMPNO, ENAME, JOB, HIREDATE, MGR, SAL, COMM, DEPTNO)
VALUES (30, 'DATE', 'PROGRAMMER', '04/5/4', 15, 1800, 200, 10);

SELECT * FROM EMP;


CREATE TABLE IF NOT EXISTS PROJ (PROJ_CODE INT(3) NOT NULL, DESCRIPTION
VARCHAR(14), PRIMARY KEY(PROJ_CODE));

INSERT INTO PROJ (PROJ_CODE, DESCRIPTION) VALUES (100, 'PAYROLL');

INSERT INTO PROJ (PROJ_CODE, DESCRIPTION) VALUES (200, 'PERSONNEL');

INSERT INTO PROJ (PROJ_CODE, DESCRIPTION) VALUES (300, 'SALES');
```

```
SELECT * FROM PROJ;


CREATE TABLE IF NOT EXISTS ASSIGN (EMPNO INT(2) NOT NULL, PROJ_CODE
INT(3) NOT NULL, A_TIME INT(3), PRIMARY KEY(EMPNO, PROJ_CODE), FOREIGN
KEY(EMPNO) REFERENCES EMP(EMPNO), FOREIGN KEY (PROJ_CODE) REFERENCES
PROJ(PROJ_CODE));
INSERT INTO ASSIGN (EMPNO, PROJ_CODE, A_TIME) VALUES (10, 100, 40);
INSERT INTO ASSIGN (EMPNO, PROJ_CODE, A_TIME) VALUES (10, 200, 60);
INSERT INTO ASSIGN (EMPNO, PROJ_CODE, A_TIME) VALUES (15, 100, 100);
INSERT INTO ASSIGN (EMPNO, PROJ_CODE, A_TIME) VALUES (20, 200, 100);
INSERT INTO ASSIGN (EMPNO, PROJ_CODE, A_TIME) VALUES (30, 100, 100);
SELECT * FROM ASSIGN;
```

RESULTS

```
SELECT * FROM DEPT;
+--------+------------+--------+
| DEPTNO | DNAME | LOC |
+--------+------------+--------+
| 10 | ACCOUNTING | ATHENS |
| 20 | SALES | LONDON |
| 30 | RESEARCH | ATHENS |
| 40 | PAYROLL | LONDON |
+--------+------------+--------+


SELECT * FROM EMP;
+-------+---------+-----------+------------+----- -+------+------+------
+
| EMPNO | ENAME | JOB | HIREDATE | MGR | SAL | COMM | DEPTNO |
+-------+---------+-----------+------------+----- -+------+------+------
+
| 10 | CODD | ANALYST | 1989-01-01 | 15 | 3000 | NULL | 10 |
| 15 | ELMASRI | ANALYST | 1995-05-02 | 15 | 1200 | 150 | 10 |
| 20 | NAVATHE | SALESMAN | 1977-07-07 | 20 | 2000 | NULL | 20 |
```

```
| 30 | DATE | PROGRAMMER | 2004-05-04 | 15 | 1800 | 200 | 10 |
+-------+---------+------------+------------+----- -+------+------+------
+
```

```
SELECT * FROM PROJ;
+-----------+-------------+
| PROJ_CODE | DESCRIPTION |
+-----------+-------------+
| 100 | PAYROLL |
| 200 | PERSONNEL |
| 300 | SALES |
+-----------+-------------+
```

```
SELECT * FROM ASSIGN;
+-------+-----------+---------+
| EMPNO | PROJ_CODE | A_TIME |
+-------+-----------+---------+
| 10 | 100 | 40 |
| 10 | 200 | 60 |
| 15 | 100 | 100 |
| 20 | 200 | 100 |
| 30 | 100 | 100 |
+-------+-----------+---------+
```

QUESTIONS

1. Locate the statement that will display the employees (ENAME,DEPTNO) who work in department 10, sorted by their commission

STATEMENT

```
SELECT ENAME " LAST NAME ", IFNULL(COMM, " ") " SUPPLY ",
DEPTNO " DEPARTMENT " FROM EMP
WHERE DEPTNO=10 ORDER BY COMM;
```

RESULTS

```
+---------------+------------------+---------- -+
| LAST NAME | SUPPLY | SECTION |
+---------------+------------------+---------- -+
| CODD | | 10 |
| ELMASRI | 150 | 10 |
| DATE | 200 | 10 |
+---------------+------------------+---------- -+
```

2. Locate the statement that will display the employees (ENAME,JOB,SAL) sorted by position (ascending order) and by their salary (descending order)

STATEMENT

```
SELECT ENAME " LAST NAME ", JOB " POSITION ", SAL " SALARY "
FROM EMP ORDER BY JOB, SAL DESC;
```

RESULTS

```
+---------------+-----------+-------------+
| LAST NAME | POSITION | SALARY |
+---------------+-----------+-------------+
| CODD | ANALYST | 3000 |
| ELMASRI | ANALYST | 1200 |
```

```
| DATE | PROGRAMMER | 1800 |
| NAVATHE | SALESMAN | 2000 |
+---------------+-----------+-------------+
```

3. Find the statement that will display the average salary per department when there are at least 1 employees.

STATEMENT

```
SELECT FORMAT(AVG(SAL), 0) "SALARY NO", DEPTNO "DEPARTMENT"
FROM EMP
GROUP BY DEPTNO HAVING COUNT(*) >= 1;
```

RESULTS

```
+--------------------+-----------+
| M.O. SALARY | SECTION |
+--------------------+-----------+
| 2,000 | 10 |
| 2,000 | 20 |
+--------------------+-----------+
```

4. Locate the statement that will show the average employment time (in years) by department (Table 1)

PART	PROVINCE
10	24.0
20	42.8

Table 1.

STATEMENT

```
SELECT DEPTNO "DEPARTMENT",
FORMAT(AVG(DATEDIFF('2020-4-15', HIREDATE)/365), 1) "SERVICE
(years)"
FROM EMP GROUP BY DEPTNO;
```

RESULTS

```
+-----------+--------------------------------+
| SECTION | TERM (years) |
+-----------+--------------------------------+
| 10 | 24.1 |
| 20 | 42.8 |
+-----------+--------------------------------+
```

5. Locate the statement that will display projects (PNAME), employed employees (ENAME), and their positions (JOB), sorted by project and by position.

STATEMENT

```
SELECT PROJ.DESCRIPTION "PROJECT", EMP.ENAME "LAST NAME", EMP.JOB
"POSITION"
FROM PROJ, ASSIGN, EMP WHERE
PROJ.PROJ_CODE = ASSIGN.PROJ_CODE AND
ASSIGN.EMPNO = EMP.EMPNO
ORDER BY PROJ.DESCRIPTION, EMP.JOB;
```

RESULTS

```
+-----------+---------------+------------+
| PROJECT | LAST NAME | POSITION |
+-----------+---------------+------------+
| PAYROLL | CODD | ANALYST |
| PAYROLL | ELMASRI | ANALYST |
| PAYROLL | DATE | PROGRAMMER |
| PERSONNEL | CODD | ANALYST |
| PERSONNEL | NAVATHE | SALESMAN |
+-----------+---------------+------------+
```

6. Locate the statement that will display all employees against their supervisors (Table 2). Results should be sorted by department name and employee name

Department	Manager	Employee
ACCOUNTING	ELMASRI	CODD
ACCOUNTING	ELMASRI	ELMASRI
ACCOUNTING	ELMASRI	DATE
RESEARCH	NAVATHE	NAVATHE

Table 2 .

STATEMENT

```
SELECT DEPT.DNAME "Department", MGR.ENAME "Manager", EMP.ENAME
"Employee"

FROM EMP EMP

JOIN DEPT ON EMP.DEPTNO = DEPT.DEPTNO

JOIN EMP MGR ON EMP.MGR = MGR.EMPNO

ORDER BY DEPT.DNAME, EMP.ENAME;
```

RESULTS

```
+------------+---------+----------+

| Department | Manager | Employee |

+------------+---------+----------+

| ACCOUNTING | ELMASRI | CODD |

| ACCOUNTING | ELMASRI | DATE |

| ACCOUNTING | ELMASRI | ELMASRI |

| SALES | NAVATHE | NAVATHE |

+------------+---------+----------+
```

7. Locate the statement that will display the names of the employees, their position and the location (loc) of the employees of the "RESEARCH" department (Table 3).

Ename	Job	Loc
NAVATHE	SALESMAN	DALLAS

Table 3 .

STATEMENT

```
SELECT EMP.ENAME "Ename", EMP.JOB "Job", DEPT.LOC "Loc"
FROM EMP, DEPT WHERE
DEPT.DEPTNO = EMP.EMPNO AND DEPT.DNAME='RESEARCH';
```

RESULTS

```
+-------+------------+--------+
| Ename | Job | Loc |
+-------+------------+--------+
| DATE | PROGRAMMER | ATHENS |
+-------+------------+--------+
```

8. Locate the statement that will display the names of employees participating in the "PAYROLL" project and working more than 50 hours (PTIME) for this project.

STATEMENT

```
SELECT EMP . ENAME "LAST NAME", PROJ . DESCRIPTION "PROJECT",
ASSIGN . A _ TIME "WORKING HOURS"
FROM EMP
JOIN ASSIGN ON ASSIGN.EMPNO = EMP.EMPNO
JOIN PROJ ON ASSIGN.PROJ_CODE = PROJ.PROJ_CODE
WHERE ASSIGN.A_TIME > 50 AND PROJ.DESCRIPTION='PAYROLL';
```

RESULT

```
+----------------+----------+--------------------- ------+
| LAST NAME | PROJECT | WORKING HOURS |
+----------------+----------+--------------------- ------+
| ELMASRI | PAYROLL | 100 |
| DATE | PAYROLL | 100 |
+----------------+----------+--------------------- ------+
```

Thank you for your attention.