Date: October 26, 2023

Group 4 Topic # 26 Payroll Management DBMS		
Names	Student #	Signature
Hamza Malik	501112545	Malik
Omer Zulfiqar	501101201	omerz
Amanat Sodhi	501108395	AS

Assignment 5 Goal:

Prepare at least 5 interesting advanced queries similar to the queries 9 to 23 of the lecture notes including join, set operations, statistical and aggregation functions and grouping queries. For UI in that stage Unix shell programming and command line is expected.

Changes Made in this LAB:

ADDED 5 advanced queries. UI implementation complete.

Script Code:

```
-- Drop tables in an order where tables with no foreign keys are dropped first
DROP TABLE DEDUCTION;
DROP TABLE TAX;
DROP TABLE PAYMENT;
DROP TABLE SALARY;

-- Now drop the tables with foreign key constraints
DROP TABLE EMPLOYEE;
DROP TABLE DESIGNATION;

-- Now, recreate the tables

-- DESIGNATION table
CREATE TABLE DESIGNATION (
    DESIGNATION_ID VARCHAR2(100) NOT NULL PRIMARY KEY,
    TITLE VARCHAR2(100) NOT NULL UNIQUE
);
```

```
-- EMPLOYEE table
CREATE TABLE EMPLOYEE (
   EMPLOYEE_ID VARCHAR2(100) NOT NULL PRIMARY KEY,
   NAME VARCHAR2(100) NOT NULL,
   DESIGNATION_ID VARCHAR2(100) UNIQUE REFERENCES
DESIGNATION(DESIGNATION_ID)
);

-- SALARY table
CREATE TABLE SALARY (
   SALARY_ID VARCHAR2(100) NOT NULL PRIMARY KEY,
   EMPLOYEE_ID VARCHAR2(100) UNIQUE REFERENCES EMPLOYEE(EMPLOYEE_ID),
   AMOUNT NUMBER(10,2) NOT NULL CHECK (AMOUNT >= 0)
);

-- PAYMENT table
CREATE TABLE PAYMENT (
   PAYMENT_ID VARCHAR2(100) NOT NULL PRIMARY KEY,
   EMPLOYEE_ID VARCHAR2(100) REFERENCES EMPLOYEE(EMPLOYEE_ID),
   AMOUNT NUMBER(10,2) NOT NULL CHECK (AMOUNT >= 0),
   DATE_RECEIVED DATE DEFAULT SYSDATE
);

-- TAX table
CREATE TABLE TAX (
   TAX_ID VARCHAR2(100) NOT NULL PRIMARY KEY,
   EMPLOYEE_ID VARCHAR2(100) REFERENCES EMPLOYEE(EMPLOYEE_ID),
   TAX_AMOUNT NUMBER(10,2) NOT NULL CHECK (TAX_AMOUNT >= 0)
);

-- DEDUCTION table
CREATE TABLE DEDUCTION (
   DEDUCTION_ID VARCHAR2(100) NOT NULL PRIMARY KEY,
   EMPLOYEE_ID VARCHAR2(100) REFERENCES EMPLOYEE(EMPLOYEE_ID),
   DEDUCTION_AMOUNT NUMBER(10,2) NOT NULL CHECK (DEDUCTION_AMOUNT >= 0),
   REASON VARCHAR2(200)
);

INSERT INTO DESIGNATION (DESIGNATION_ID, TITLE) VALUES ('D1', 'Manager');
INSERT INTO DESIGNATION (DESIGNATION_ID, TITLE) VALUES ('D2', 'Engineer');
INSERT INTO DESIGNATION (DESIGNATION_ID, TITLE) VALUES ('D3', 'Lawyer');

INSERT INTO EMPLOYEE (EMPLOYEE_ID, NAME, DESIGNATION_ID) VALUES ('E1', 'Alice',
'D1');
```

```
INSERT INTO EMPLOYEE (EMPLOYEE_ID, NAME, DESIGNATION_ID) VALUES ('E2', 'Bob',
'D2');
INSERT INTO EMPLOYEE (EMPLOYEE_ID, NAME, DESIGNATION_ID) VALUES ('E3',
'Charlie', 'D3');

INSERT INTO SALARY (SALARY_ID, EMPLOYEE_ID, AMOUNT) VALUES ('S1', 'E1', 60000);
INSERT INTO SALARY (SALARY_ID, EMPLOYEE_ID, AMOUNT) VALUES ('S2', 'E2', 50000);
INSERT INTO SALARY (SALARY_ID, EMPLOYEE_ID, AMOUNT) VALUES ('S3', 'E3', 40000);

INSERT INTO PAYMENT (PAYMENT_ID, EMPLOYEE_ID, AMOUNT, DATE_RECEIVED)
VALUES ('P1', 'E1', 1000, TO_DATE('15-OCT-2023', 'DD-MON-YYYY'));
INSERT INTO PAYMENT (PAYMENT_ID, EMPLOYEE_ID, AMOUNT, DATE_RECEIVED)
VALUES ('P2', 'E2', 800, TO_DATE('14-OCT-2023', 'DD-MON-YYYY'));

INSERT INTO TAX (TAX_ID, EMPLOYEE_ID, TAX_AMOUNT) VALUES ('T1', 'E1', 500);
INSERT INTO TAX (TAX_ID, EMPLOYEE_ID, TAX_AMOUNT) VALUES ('T2', 'E2', 400);


INSERT INTO DEDUCTION (DEDUCTION_ID, EMPLOYEE_ID, DEDUCTION_AMOUNT,
REASON) VALUES ('Dd1', 'E1', 100, 'Health Insurance');
INSERT INTO DEDUCTION (DEDUCTION_ID, EMPLOYEE_ID, DEDUCTION_AMOUNT,
REASON) VALUES ('Dd2', 'E2', 50, 'Transport Fee');
```

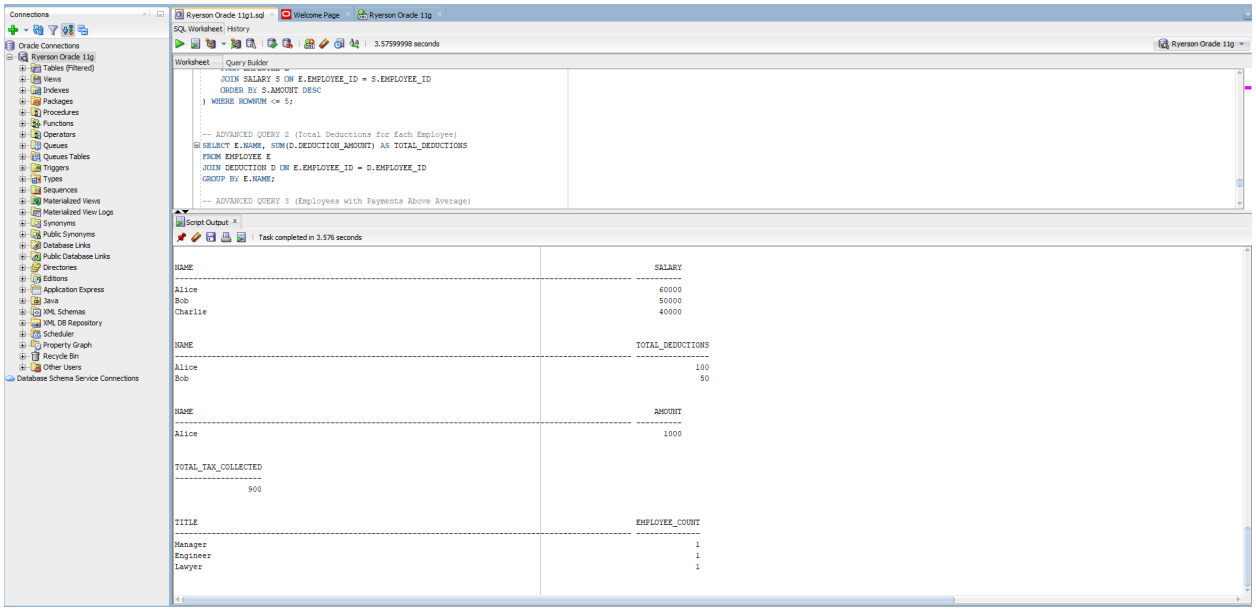
```
-- QUERY 1: Retrieve all managers from DESIGNATION table
SELECT * FROM DESIGNATION WHERE TITLE = 'Manager';
-- QUERY 2
SELECT NAME, DESIGNATION_ID FROM EMPLOYEE WHERE NAME LIKE 'A%';
--QUERY 3
SELECT EMPLOYEE_ID, AMOUNT FROM SALARY WHERE AMOUNT > 50000;
-- QUERY 4
SELECT EMPLOYEE_ID, AMOUNT, DATE_RECEIVED FROM PAYMENT WHERE
DATE_RECEIVED > '01-JAN-2023';
--QUERY 5
SELECT EMPLOYEE_ID, TAX_AMOUNT FROM TAX WHERE TAX_AMOUNT > 1000;
-- QUERY 6
SELECT EMPLOYEE_ID, DEDUCTION_AMOUNT, REASON FROM DEDUCTION WHERE
DEDUCTION_AMOUNT < 500;

-- COMPLEX
```

```
SELECT E.NAME, D.TITLE, S.AMOUNT
FROM EMPLOYEE E, DESIGNATION D, SALARY S
WHERE E.DESIGNATION_ID = D.DESIGNATION_ID AND E.EMPLOYEE_ID =
S.EMPLOYEE_ID;


DROP VIEW EmployeeTotalAmount;
DROP VIEW EmployeeDeductionsTaxes;


-- VIEW for Total Amount for Each Employee
CREATE VIEW EmployeeTotalAmount AS
SELECT E.EMPLOYEE_ID, E.NAME, NVL(S.AMOUNT, 0) + NVL(P.AMOUNT, 0) AS
TOTAL_AMOUNT
FROM EMPLOYEE E
LEFT JOIN SALARY S ON E.EMPLOYEE_ID = S.EMPLOYEE_ID
LEFT JOIN PAYMENT P ON E.EMPLOYEE_ID = P.EMPLOYEE_ID;

-- VIEW for Total Deductions and Taxes for Each Employee
CREATE VIEW EmployeeDeductionsTaxes AS
SELECT E.EMPLOYEE_ID, E.NAME, NVL(D.DEDUCTION_AMOUNT, 0) AS
TOTAL_DEDUCTIONS, NVL(T.TAX_AMOUNT, 0) AS TOTAL_TAXES
FROM EMPLOYEE E
LEFT JOIN DEDUCTION D ON E.EMPLOYEE_ID = D.EMPLOYEE_ID
LEFT JOIN TAX T ON E.EMPLOYEE_ID = T.EMPLOYEE_ID;

-- Advanced Queries

-- Query 7
SELECT E.NAME, DE.TITLE, P.AMOUNT AS TOTAL_PAYMENT, TX.TAX_AMOUNT AS
TOTAL_TAX
FROM EMPLOYEE E
JOIN DESIGNATION DE ON E.DESIGNATION_ID = DE.DESIGNATION_ID
LEFT JOIN PAYMENT P ON E.EMPLOYEE_ID = P.EMPLOYEE_ID
LEFT JOIN TAX TX ON E.EMPLOYEE_ID = TX.EMPLOYEE_ID;

-- Query 8
SELECT E.NAME, SUM(D.DEDUCTION_AMOUNT) AS TOTAL_DEDUCTION
FROM EMPLOYEE E
JOIN DEDUCTION D ON E.EMPLOYEE_ID = D.EMPLOYEE_ID
GROUP BY E.NAME
HAVING SUM(D.DEDUCTION_AMOUNT) > 1000;

-- Query 9
```

```
SELECT A.EMPLOYEE_ID, A.NAME, (A.TOTAL_AMOUNT - B.TOTAL_DEDUCTIONS -
B.TOTAL_TAXES) AS NET_AMOUNT
FROM EmployeeTotalAmount A
JOIN EmployeeDeductionsTaxes B ON A.EMPLOYEE_ID = B.EMPLOYEE_ID
WHERE (A.TOTAL_AMOUNT - B.TOTAL_DEDUCTIONS - B.TOTAL_TAXES) > 50000;
```

```
-- ADVANCED QUERY 1 (Employees with Highest Salaries)
SELECT * FROM (
   SELECT E.NAME, S.AMOUNT AS SALARY
   FROM EMPLOYEE E
   JOIN SALARY S ON E.EMPLOYEE_ID = S.EMPLOYEE_ID
   ORDER BY S.AMOUNT DESC
) WHERE ROWNUM <= 5;


-- ADVANCED QUERY 2 (Total Deductions for Each Employee)
SELECT E.NAME, SUM(D.DEDUCTION_AMOUNT) AS TOTAL_DEDUCTIONS
FROM EMPLOYEE E
JOIN DEDUCTION D ON E.EMPLOYEE_ID = D.EMPLOYEE_ID
GROUP BY E.NAME;

-- ADVANCED QUERY 3 (Employees with Payments Above Average)
SELECT E.NAME, P.AMOUNT
FROM EMPLOYEE E
JOIN PAYMENT P ON E.EMPLOYEE_ID = P.EMPLOYEE_ID
WHERE P.AMOUNT > (SELECT AVG(AMOUNT) FROM PAYMENT);

-- ADVANCED QUERY 4 (Total Tax Amount Collected)
SELECT SUM(T.TAX_AMOUNT) AS TOTAL_TAX_COLLECTED
FROM TAX T;

-- ADVANCED QUERY 5 (Count of Employees in Each Designation)
SELECT DE.TITLE, COUNT(E.EMPLOYEE_ID) AS EMPLOYEE_COUNT
FROM EMPLOYEE E
JOIN DESIGNATION DE ON E.DESIGNATION_ID = DE.DESIGNATION_ID
GROUP BY DE.TITLE;
```

RUN SCRIPT OUTPUT:



IMPLEMENTATION FOR UI CODE:

```
#!/bin/bash

while true; do
    echo "--------------------------------------"
    echo "            Menu Options              "
    echo "--------------------------------------"
    echo "1. Drop Tables"
    echo "2. Create Tables"
    echo "3. Populate Tables"
    echo "4. Query Tables"
    echo "5. Exit"
    echo "--------------------------------------"
    read -p "Enter your choice [1-5]: " choice

    case $choice in
```

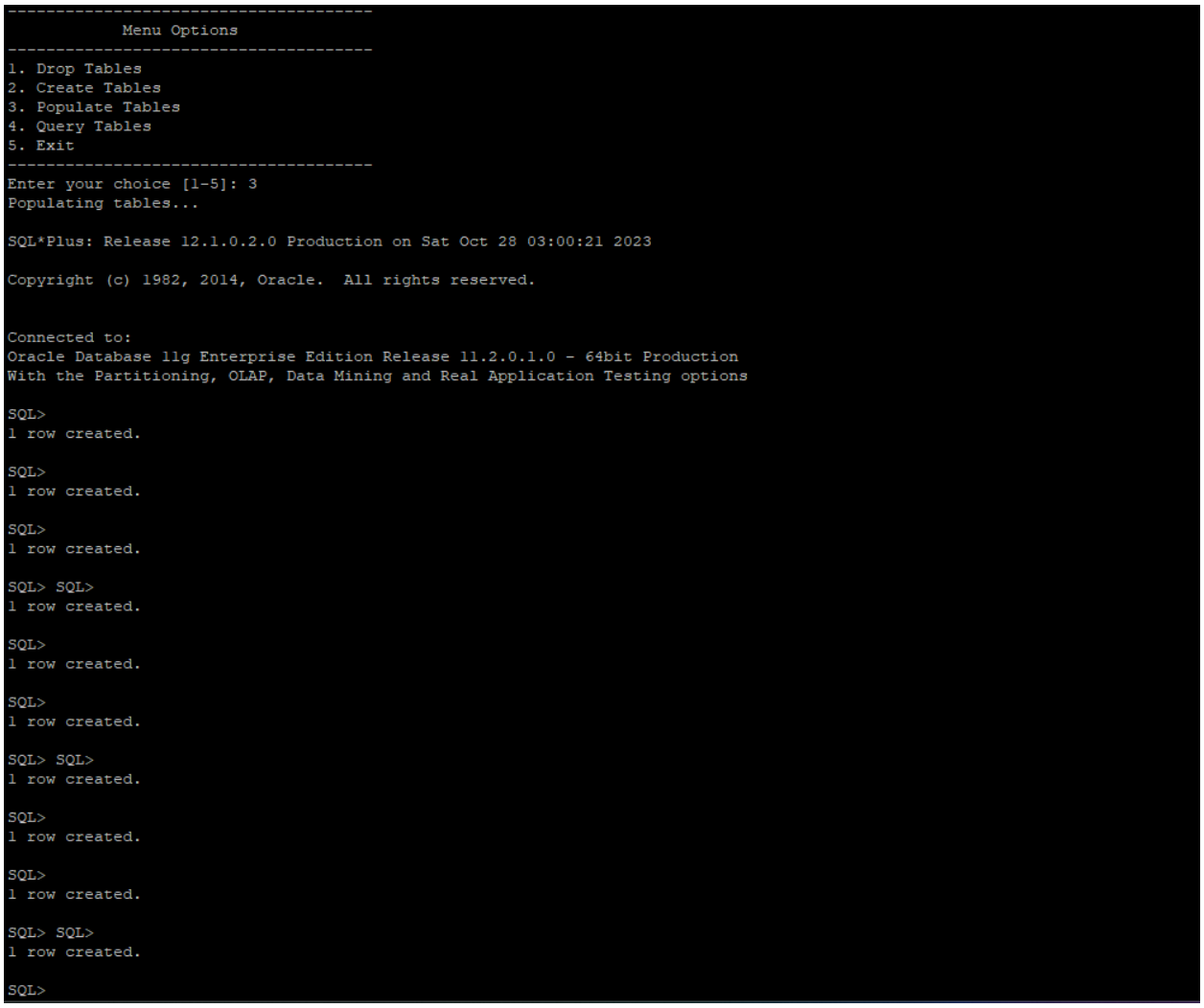
SCREENSHOT OF EACH OPTIONS:

```
---------------------------------------
           Menu Options
---------------------------------------
1. Drop Tables
2. Create Tables
3. Populate Tables
4. Query Tables
5. Exit
---------------------------------------
Enter your choice [1-5]: 1
Dropping tables...

SQL*Plus: Release 12.1.0.2.0 Production on Sat Oct 28 02:38:39 2023

Copyright (c) 1982, 2014, Oracle.  All rights reserved.


Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options

SQL>
Table dropped.

SQL>
Table dropped.

SQL>
Table dropped.

SQL>
Table dropped.

SQL>
Table dropped.

SQL>
Table dropped.

SQL> Disconnected from Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options
---------------------------------------
```

OPTION 1 ABOVE

```
           Menu Options
---------------------------------------
1. Drop Tables
2. Create Tables
3. Populate Tables
4. Query Tables
5. Exit
---------------------------------------
Enter your choice [1-5]: 2
Creating tables...

SQL*Plus: Release 12.1.0.2.0 Production on Sat Oct 28 02:50:42 2023

Copyright (c) 1982, 2014, Oracle.  All rights reserved.


Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options

SQL> SQL> SQL>   2    3    4
Table created.

SQL> SQL> SQL>   2    3    4    5
Table created.

SQL> SQL> SQL>   2    3    4    5
Table created.

SQL> SQL> SQL>   2    3    4    5    6
Table created.

SQL> SQL> SQL>   2    3    4    5
Table created.

SQL> SQL> SQL>   2    3    4    5    6
Table created.

SQL> SQL> Disconnected from Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options
---------------------------------------
```

OPTION 2 created tables.



OPTION 3 populate table

```
--------------------------------------
           Menu Options
--------------------------------------
1. Drop Tables
2. Create Tables
3. Populate Tables
4. Query Tables
5. Exit
--------------------------------------
Enter your choice [1-5]: 4
Querying tables...

SQL*Plus: Release 12.1.0.2.0 Production on Sat Oct 28 03:04:10 2023

Copyright (c) 1982, 2014, Oracle.  All rights reserved.


Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options

SQL> SQL> SQL>
DESIGNATION_ID
--------------------------------------------------------------------------------
TITLE
--------------------------------------------------------------------------------
D1
Manager


SQL> SQL>
NAME
--------------------------------------------------------------------------------
DESIGNATION_ID
--------------------------------------------------------------------------------
Alice
D1


SQL> SQL>
EMPLOYEE_ID
--------------------------------------------------------------------------------
    AMOUNT
----------
E1
     60000


SQL> SQL>
EMPLOYEE_ID
--------------------------------------------------------------------------------
    AMOUNT DATE_RECE
---------- ---------
E1
      1000 15-OCT-23

E2
       800 14-OCT-23


SQL> SQL>
no rows selected
```

Option 4

```
NAME
--------------------------------------------------------------------------------
DESIGNATION_ID
--------------------------------------------------------------------------------
Alice
D1


SQL> SQL>
EMPLOYEE_ID
--------------------------------------------------------------------------------
    AMOUNT
----------
E1
     60000


SQL> SQL>
EMPLOYEE_ID
--------------------------------------------------------------------------------
    AMOUNT DATE_RECE
---------- ---------
E1
      1000 15-OCT-23

E2
       800 14-OCT-23


SQL> SQL>
no rows selected

SQL> SQL>
EMPLOYEE_ID
--------------------------------------------------------------------------------
DEDUCTION_AMOUNT
----------------
REASON
--------------------------------------------------------------------------------
E1
            100
Health Insurance

E2
             50
Transport Fee

EMPLOYEE_ID
--------------------------------------------------------------------------------
DEDUCTION_AMOUNT
----------------
REASON
--------------------------------------------------------------------------------


SQL> SQL> SQL>   2    3
NAME
--------------------------------------------------------------------------------
TITLE
--------------------------------------------------------------------------------
```

OPTION 4

```
Bob
     50000

Charlie
     40000


SQL> SQL> SQL> SQL>   2    3    4
NAME
--------------------------------------------------------------------------------
TOTAL_DEDUCTIONS
----------------
Alice
             100

Bob
              50


SQL> SQL> SQL>   2    3    4
NAME
--------------------------------------------------------------------------------
    AMOUNT
----------
Alice
      1000


SQL> SQL> SQL>   2
TOTAL_TAX_COLLECTED
-------------------
                900

SQL> SQL> SQL>   2    3    4
TITLE
--------------------------------------------------------------------------------
EMPLOYEE_COUNT
--------------
Manager
             1

Engineer
             1

Lawyer
             1


SQL> SQL> SQL> SQL> Disconnected from Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options
---------------------------------------
            Menu Options
---------------------------------------
1. Drop Tables
2. Create Tables
3. Populate Tables
4. Query Tables
5. Exit
```

MORE OPTION 4