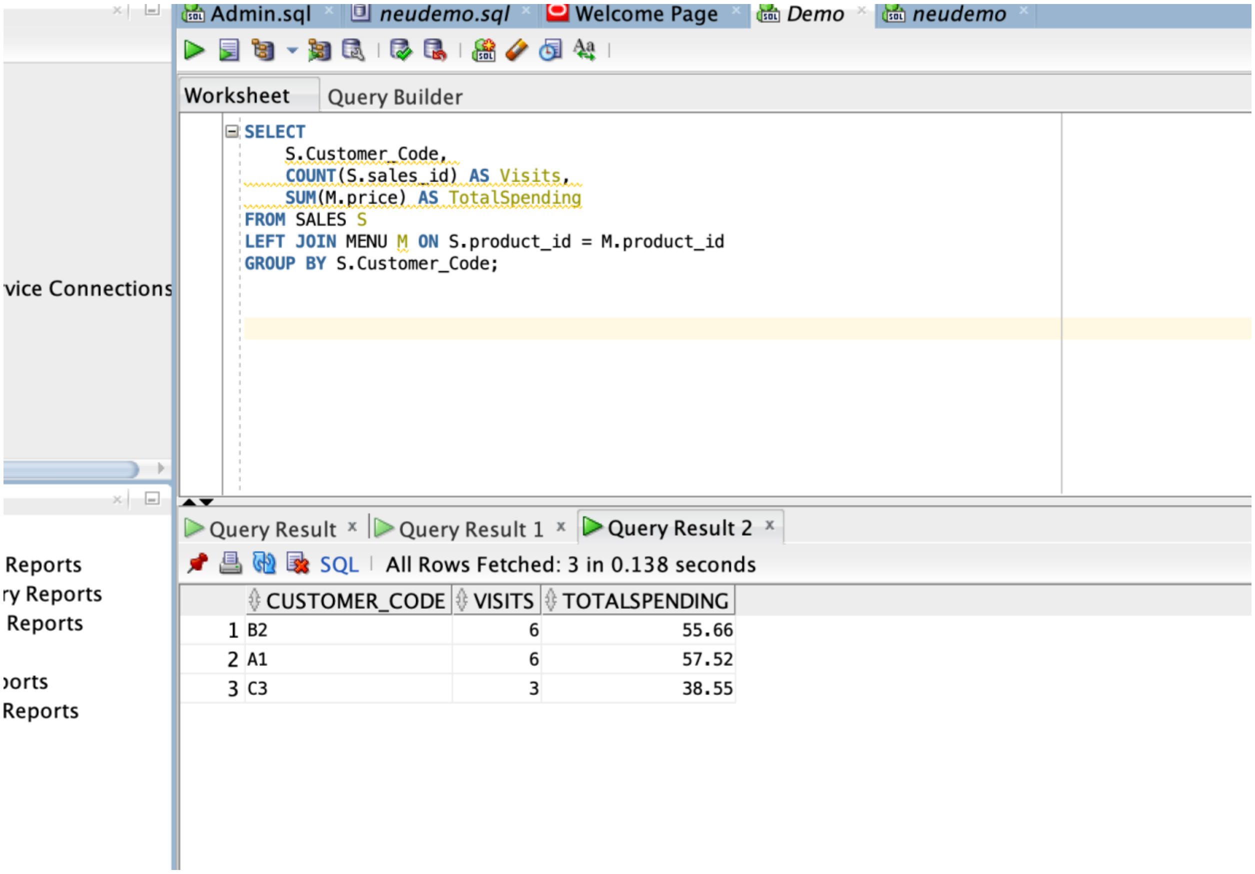
ASSIGNMENT - 3

Name - Uttkarsh Bharadia

NUID - 002872928

Question (1) use MENU, SALES, MEMBERS tables (wherever applicable) and you can include as many columns as you want to generate the below reports

1. customer wise total spending and number of visits

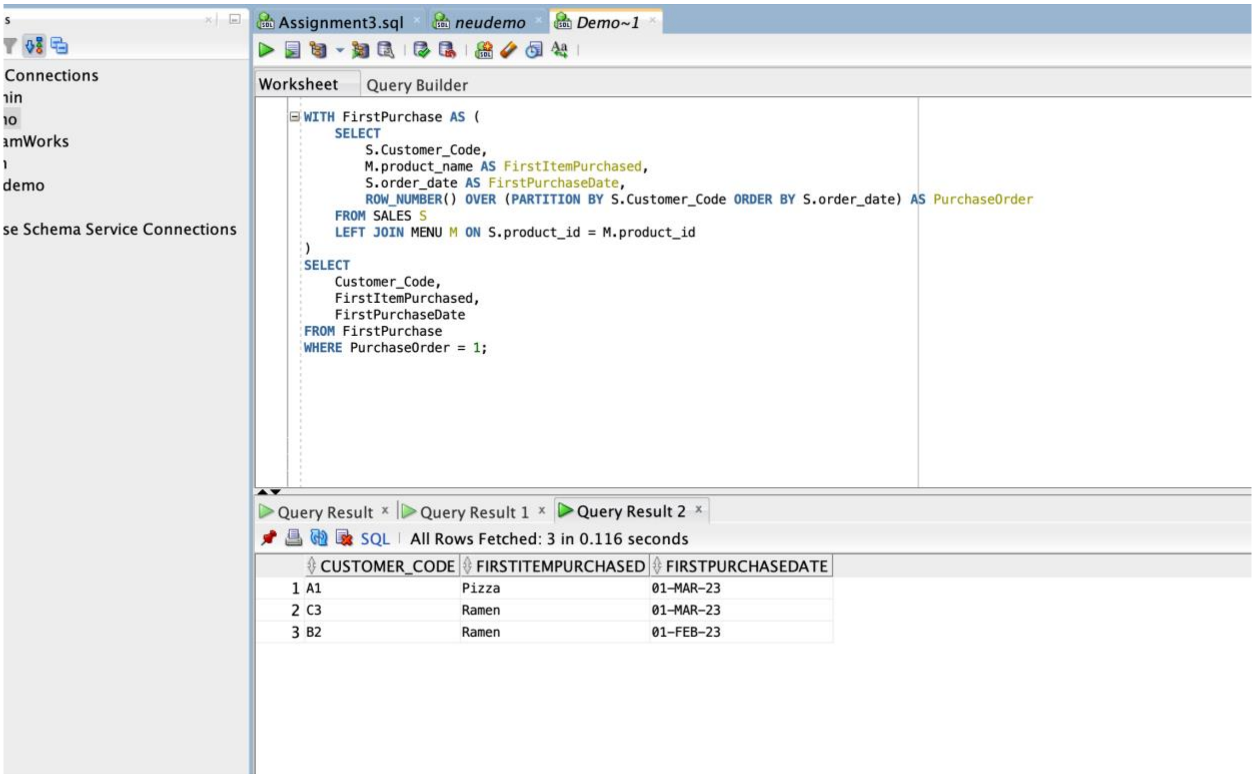


```
SELECT
     S.Customer_Code,
     COUNT(S.sales_id) AS Visits,
     SUM(M.price) AS TotalSpending
FROM SALES S
LEFT JOIN MENU M ON S.product_id = M.product_id
GROUP BY S.Customer_Code;
```

All Rows Fetched: 3 in 0.138 seconds

	CUSTOMER_CODE	VISITS	TOTALSPENDING
1	B2	6	55.66
2	A1	6	57.52
3	C3	3	38.55

2. Display customer wise first item purchased and show the purchase date as well.

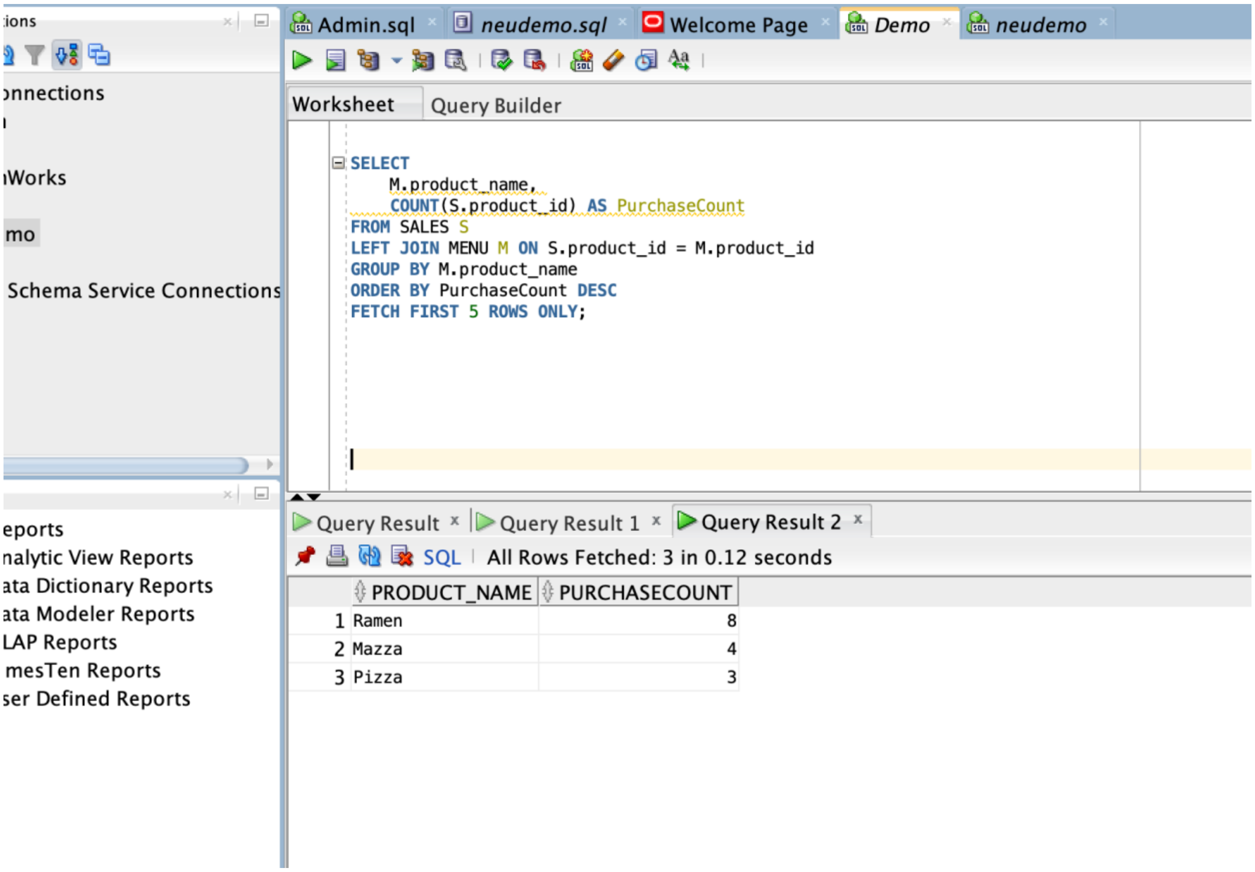


```
WITH FirstPurchase AS (
    SELECT
        S.Customer_Code,
        M.product_name AS FirstItemPurchased,
        S.order_date AS FirstPurchaseDate,
        ROW_NUMBER() OVER (PARTITION BY S.Customer_Code ORDER BY S.order_date) AS PurchaseOrder
    FROM SALES S
    LEFT JOIN MENU M ON S.product_id = M.product_id
)
SELECT
    Customer_Code,
    FirstItemPurchased,
    FirstPurchaseDate
FROM FirstPurchase
WHERE PurchaseOrder = 1;
```

All Rows Fetched: 3 in 0.116 seconds

```
   CUSTOMER_CODE  FIRSTITEMPURCHASED  FIRSTPURCHASEDATE
 1 A1             Pizza               01-MAR-23
 2 C3             Ramen               01-MAR-23
 3 B2             Ramen               01-FEB-23
```

3. Top 5 Most frequently purchased item.

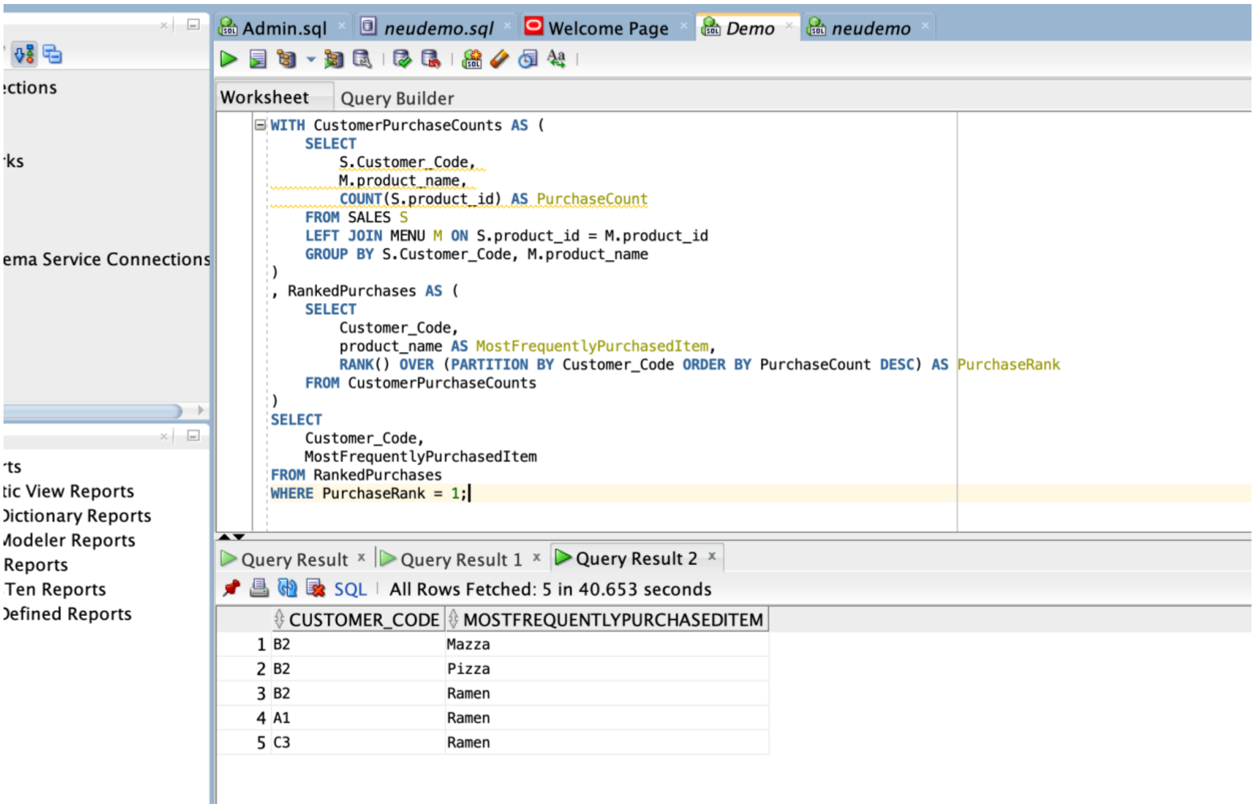


```sql
SELECT
    M.product_name,
    COUNT(S.product_id) AS PurchaseCount
FROM SALES S
LEFT JOIN MENU M ON S.product_id = M.product_id
GROUP BY M.product_name
ORDER BY PurchaseCount DESC
FETCH FIRST 5 ROWS ONLY;
```

All Rows Fetched: 3 in 0.12 seconds

	PRODUCT_NAME	PURCHASECOUNT
1	Ramen	8
2	Mazza	4
3	Pizza	3

4. Customer wise most frequently purchased item.

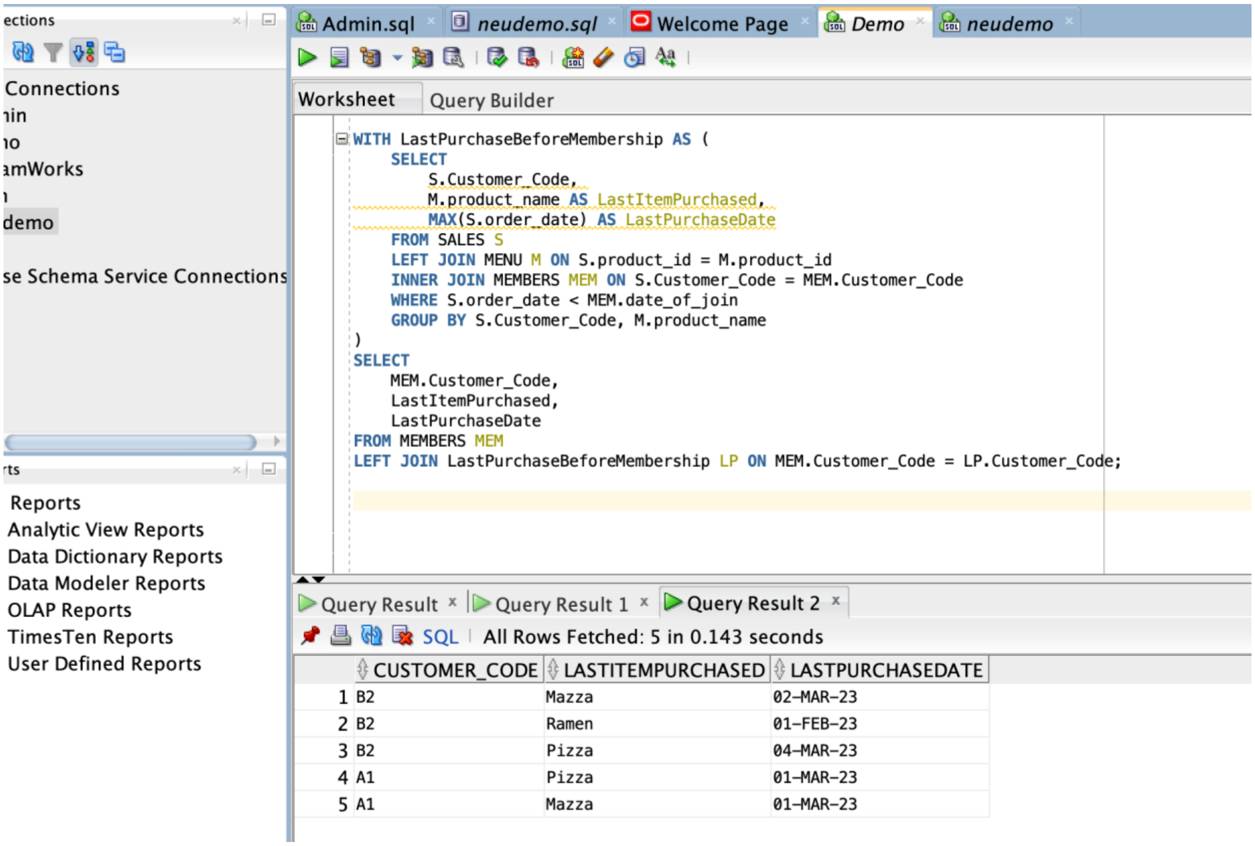


```sql
WITH CustomerPurchaseCounts AS (
    SELECT
        S.Customer_Code,
        M.product_name,
        COUNT(S.product_id) AS PurchaseCount
    FROM SALES S
    LEFT JOIN MENU M ON S.product_id = M.product_id
    GROUP BY S.Customer_Code, M.product_name
)
, RankedPurchases AS (
    SELECT
        Customer_Code,
        product_name AS MostFrequentlyPurchasedItem,
        RANK() OVER (PARTITION BY Customer_Code ORDER BY PurchaseCount DESC) AS PurchaseRank
    FROM CustomerPurchaseCounts
)
SELECT
    Customer_Code,
    MostFrequentlyPurchasedItem
FROM RankedPurchases
WHERE PurchaseRank = 1;
```

All Rows Fetched: 5 in 40.653 seconds

	CUSTOMER_CODE	MOSTFREQUENTLYPURCHASEDITEM
1	B2	Mazza
2	B2	Pizza
3	B2	Ramen
4	A1	Ramen
5	C3	Ramen

5. Customer wise show the last item that was purchased before becoming a member.



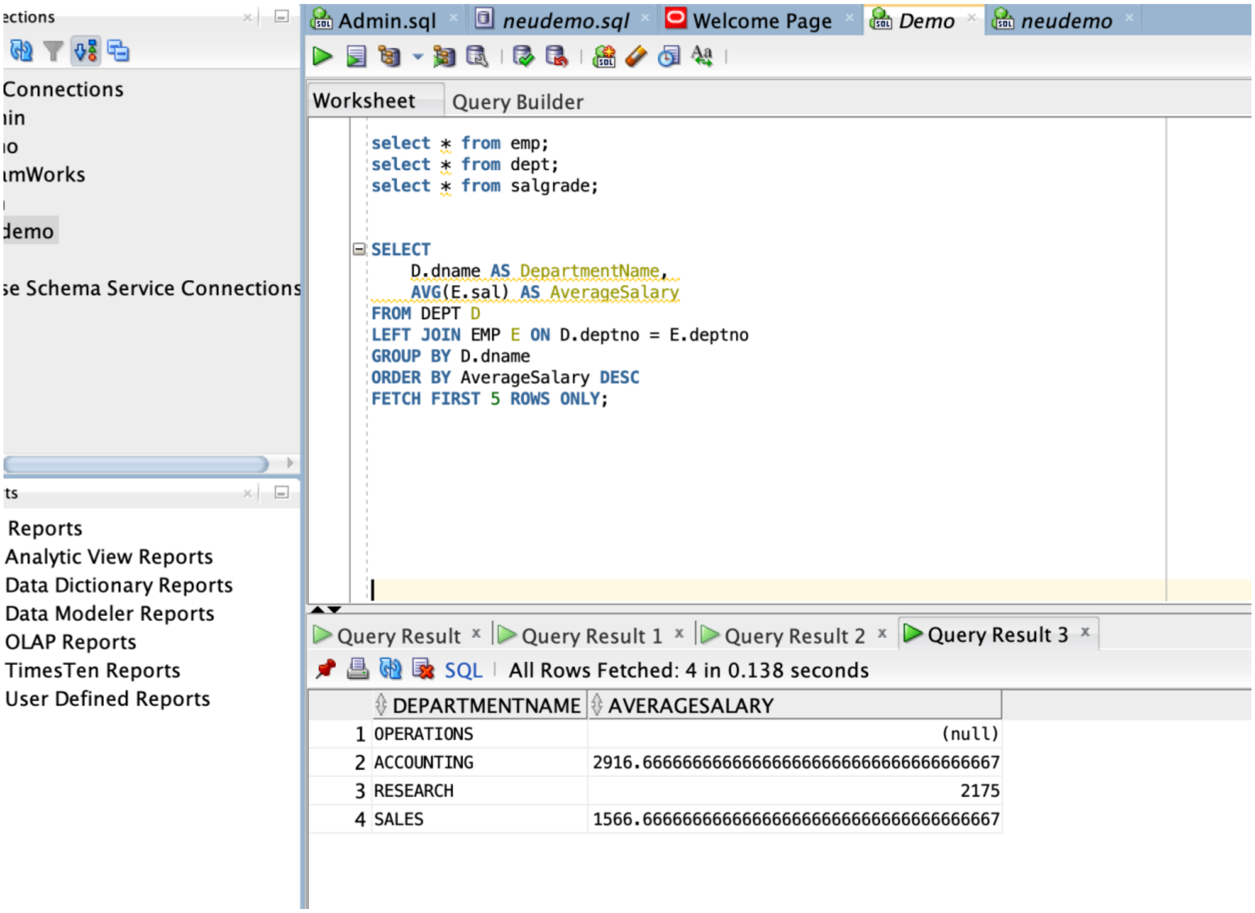
```sql
WITH LastPurchaseBeforeMembership AS (
    SELECT
        S.Customer_Code,
        M.product_name AS LastItemPurchased,
        MAX(S.order_date) AS LastPurchaseDate
    FROM SALES S
    LEFT JOIN MENU M ON S.product_id = M.product_id
    INNER JOIN MEMBERS MEM ON S.Customer_Code = MEM.Customer_Code
    WHERE S.order_date < MEM.date_of_join
    GROUP BY S.Customer_Code, M.product_name
)
SELECT
    MEM.Customer_Code,
    LastItemPurchased,
    LastPurchaseDate
FROM MEMBERS MEM
LEFT JOIN LastPurchaseBeforeMembership LP ON MEM.Customer_Code = LP.Customer_Code;
```

All Rows Fetched: 5 in 0.143 seconds

	CUSTOMER_CODE	LASTITEMPURCHASED	LASTPURCHASEDATE
1	B2	Mazza	02-MAR-23
2	B2	Ramen	01-FEB-23
3	B2	Pizza	04-MAR-23
4	A1	Pizza	01-MAR-23
5	A1	Mazza	01-MAR-23

Question (2) use EMP, DEPT and SALGRADE (wherever applicable) and answer below.

1. top 5 departments with the highest average salary. (Hint use FETCH FIRST 5 ROWS ONLY



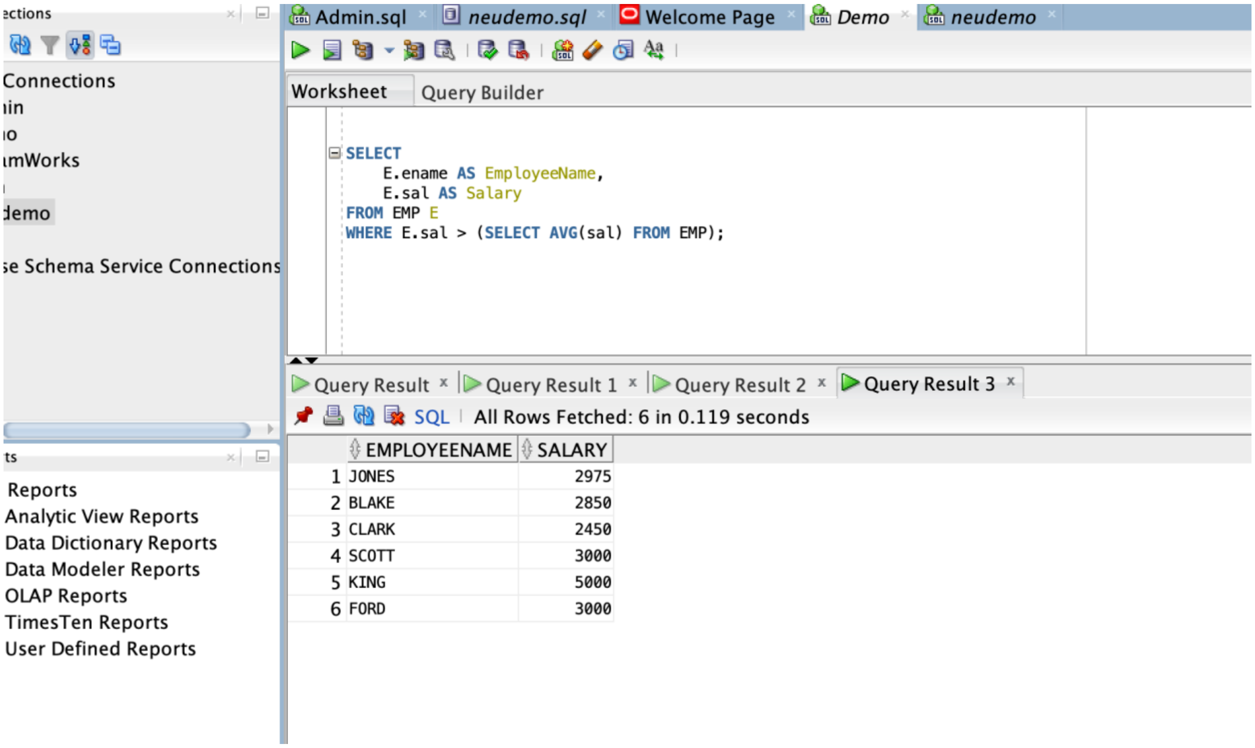
```
select * from emp;
select * from dept;
select * from salgrade;


SELECT
    D.dname AS DepartmentName,
    AVG(E.sal) AS AverageSalary
FROM DEPT D
LEFT JOIN EMP E ON D.deptno = E.deptno
GROUP BY D.dname
ORDER BY AverageSalary DESC
FETCH FIRST 5 ROWS ONLY;
```

All Rows Fetched: 4 in 0.138 seconds

	DEPARTMENTNAME	AVERAGESALARY
1	OPERATIONS	(null)
2	ACCOUNTING	2916.6666666666666666666666666666666667
3	RESEARCH	2175
4	SALES	1566.6666666666666666666666666666666667

2. List employees who have a salary greater than the average salary of all employees.

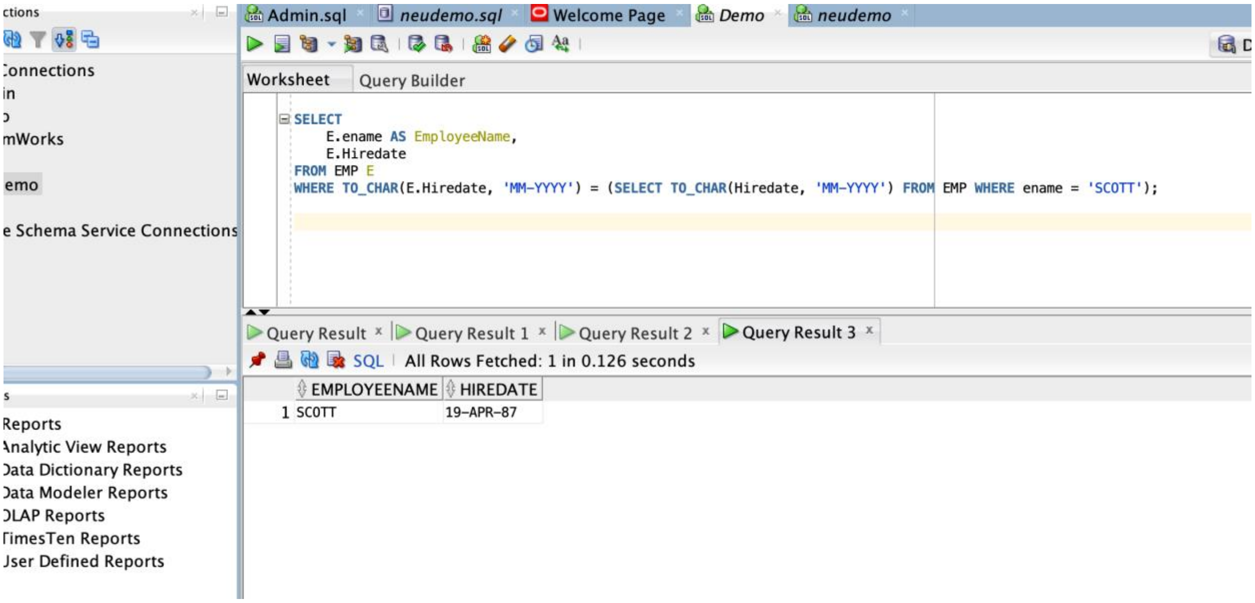


```sql
SELECT
     E.ename AS EmployeeName,
     E.sal AS Salary
FROM EMP E
WHERE E.sal > (SELECT AVG(sal) FROM EMP);
```

All Rows Fetched: 6 in 0.119 seconds

	EMPLOYEENAME	SALARY
1	JONES	2975
2	BLAKE	2850
3	CLARK	2450
4	SCOTT	3000
5	KING	5000
6	FORD	3000

3. List employees who were hired in the same month and year as the employee SCOTT

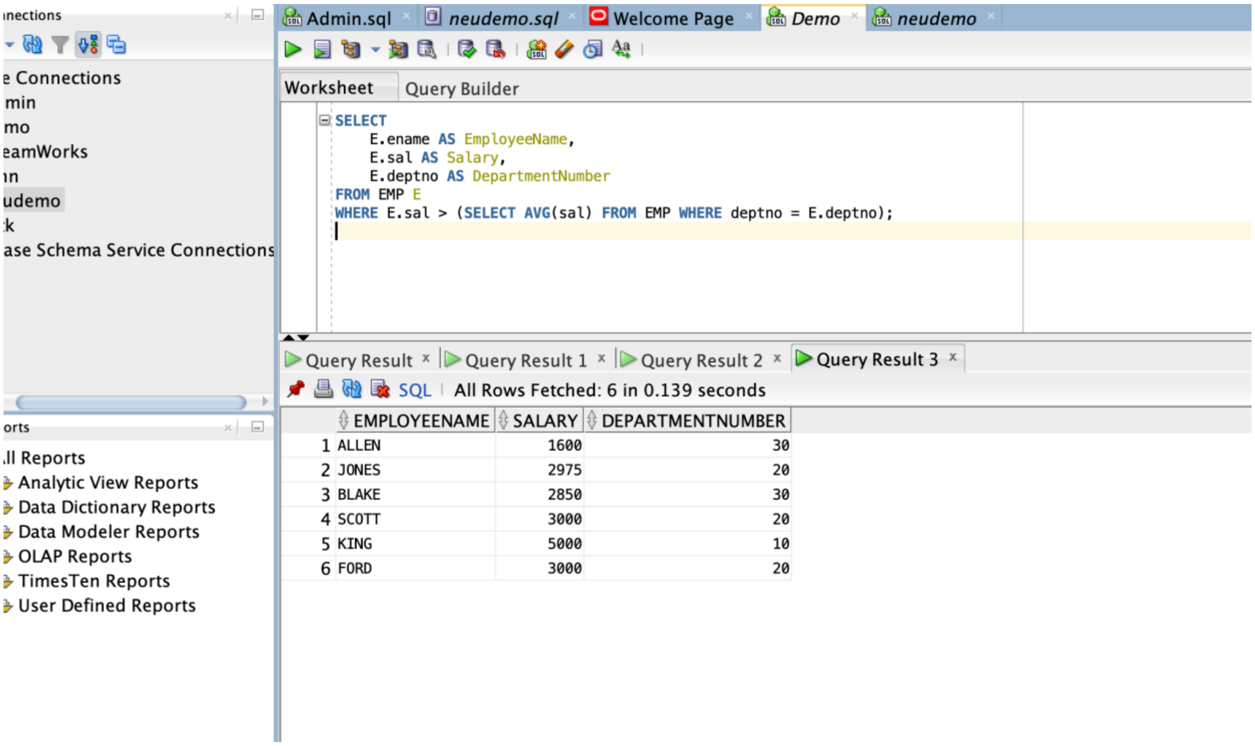


```sql
SELECT
    E.ename AS EmployeeName,
    E.Hiredate
FROM EMP E
WHERE TO_CHAR(E.Hiredate, 'MM-YYYY') = (SELECT TO_CHAR(Hiredate, 'MM-YYYY') FROM EMP WHERE ename = 'SCOTT');
```

All Rows Fetched: 1 in 0.126 seconds

	EMPLOYEENAME	HIREDATE
1	SCOTT	19-APR-87

4. List employee details who have a salary greater than average salary in their department

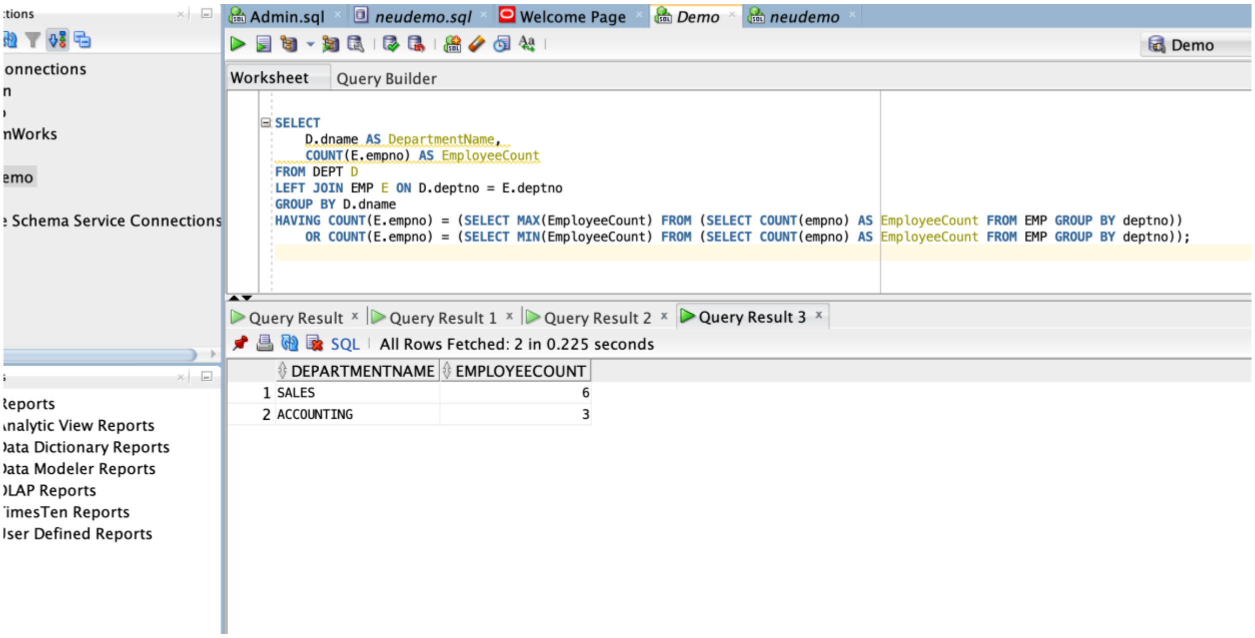


```
SELECT
    E.ename AS EmployeeName,
    E.sal AS Salary,
    E.deptno AS DepartmentNumber
FROM EMP E
WHERE E.sal > (SELECT AVG(sal) FROM EMP WHERE deptno = E.deptno);
```

All Rows Fetched: 6 in 0.139 seconds

	EMPLOYEENAME	SALARY	DEPARTMENTNUMBER
1	ALLEN	1600	30
2	JONES	2975	20
3	BLAKE	2850	30
4	SCOTT	3000	20
5	KING	5000	10
6	FORD	3000	20

5. Which departments has maximum and least number of employees?



```
SELECT
    D.dname AS DepartmentName,
    COUNT(E.empno) AS EmployeeCount
FROM DEPT D
LEFT JOIN EMP E ON D.deptno = E.deptno
GROUP BY D.dname
HAVING COUNT(E.empno) = (SELECT MAX(EmployeeCount) FROM (SELECT COUNT(empno) AS EmployeeCount FROM EMP GROUP BY deptno))
    OR COUNT(E.empno) = (SELECT MIN(EmployeeCount) FROM (SELECT COUNT(empno) AS EmployeeCount FROM EMP GROUP BY deptno));
```

All Rows Fetched: 2 in 0.225 seconds

	DEPARTMENTNAME	EMPLOYEECOUNT
1	SALES	6
2	ACCOUNTING	3

DDL SCRIPT:

```
select * from user_tables;
select * from menu;
select * from sales;
select * from members;

SELECT
   S.Customer_Code,
   COUNT(S.sales_id) AS Visits,
   SUM(M.price) AS TotalSpending
FROM SALES S
LEFT JOIN MENU M ON S.product_id = M.product_id
GROUP BY S.Customer_Code;

WITH FirstPurchase AS (
   SELECT
      S.Customer_Code,
      M.product_name AS FirstItemPurchased,
      S.order_date AS FirstPurchaseDate,
      ROW_NUMBER() OVER (PARTITION BY S.Customer_Code ORDER BY S.order_date) AS PurchaseOrder
   FROM SALES S
   LEFT JOIN MENU M ON S.product_id = M.product_id
)
SELECT
   Customer_Code,
   FirstItemPurchased,
   FirstPurchaseDate
FROM FirstPurchase
```

```
WHERE PurchaseOrder = 1;




SELECT
    M.product_name,
    COUNT(S.product_id) AS PurchaseCount
FROM SALES S
LEFT JOIN MENU M ON S.product_id = M.product_id
GROUP BY M.product_name
ORDER BY PurchaseCount DESC
FETCH FIRST 5 ROWS ONLY;

WITH CustomerPurchaseCounts AS (
    SELECT
        S.Customer_Code,
        M.product_name,
        COUNT(S.product_id) AS PurchaseCount
    FROM SALES S
    LEFT JOIN MENU M ON S.product_id = M.product_id
    GROUP BY S.Customer_Code, M.product_name
)
, RankedPurchases AS (
    SELECT
        Customer_Code,
        product_name AS MostFrequentlyPurchasedItem,
        RANK() OVER (PARTITION BY Customer_Code ORDER BY PurchaseCount DESC) AS PurchaseRank
    FROM CustomerPurchaseCounts
)
```

```
SELECT
    Customer_Code,
    MostFrequentlyPurchasedItem
FROM RankedPurchases
WHERE PurchaseRank = 1;
```

```
WITH LastPurchaseBeforeMembership AS (
    SELECT
        S.Customer_Code,
        M.product_name AS LastItemPurchased,
        MAX(S.order_date) AS LastPurchaseDate
    FROM SALES S
    LEFT JOIN MENU M ON S.product_id = M.product_id
    INNER JOIN MEMBERS MEM ON S.Customer_Code = MEM.Customer_Code
    WHERE S.order_date < MEM.date_of_join
    GROUP BY S.Customer_Code, M.product_name
)
SELECT
    MEM.Customer_Code,
    LastItemPurchased,
    LastPurchaseDate
FROM MEMBERS MEM
LEFT JOIN LastPurchaseBeforeMembership LP ON MEM.Customer_Code = LP.Customer_Code;
```

```
select * from emp;
select * from dept;
select * from salgrade;
```

```sql
SELECT
    D.dname AS DepartmentName,
    AVG(E.sal) AS AverageSalary
FROM DEPT D
LEFT JOIN EMP E ON D.deptno = E.deptno
GROUP BY D.dname
ORDER BY AverageSalary DESC
FETCH FIRST 5 ROWS ONLY;
```

```sql
SELECT
    E.ename AS EmployeeName,
    E.sal AS Salary
FROM EMP E
WHERE E.sal > (SELECT AVG(sal) FROM EMP);
```

```sql
SELECT
    E.ename AS EmployeeName,
    E.Hiredate
FROM EMP E
WHERE TO_CHAR(E.Hiredate, 'MM-YYYY') = (SELECT TO_CHAR(Hiredate, 'MM-YYYY') FROM EMP WHERE ename = 'SCOTT');
```

```sql
SELECT
    E.ename AS EmployeeName,
    E.sal AS Salary,
    E.deptno AS DepartmentNumber
FROM EMP E
WHERE E.sal > (SELECT AVG(sal) FROM EMP WHERE deptno = E.deptno);
```

```
SELECT
    D.dname AS DepartmentName,
    COUNT(E.empno) AS EmployeeCount
FROM DEPT D
LEFT JOIN EMP E ON D.deptno = E.deptno
GROUP BY D.dname
HAVING COUNT(E.empno) = (SELECT MAX(EmployeeCount) FROM (SELECT COUNT(empno) AS EmployeeCount FROM EMP GROUP BY deptno))
    OR COUNT(E.empno) = (SELECT MIN(EmployeeCount) FROM (SELECT COUNT(empno) AS EmployeeCount FROM EMP GROUP BY deptno));
```