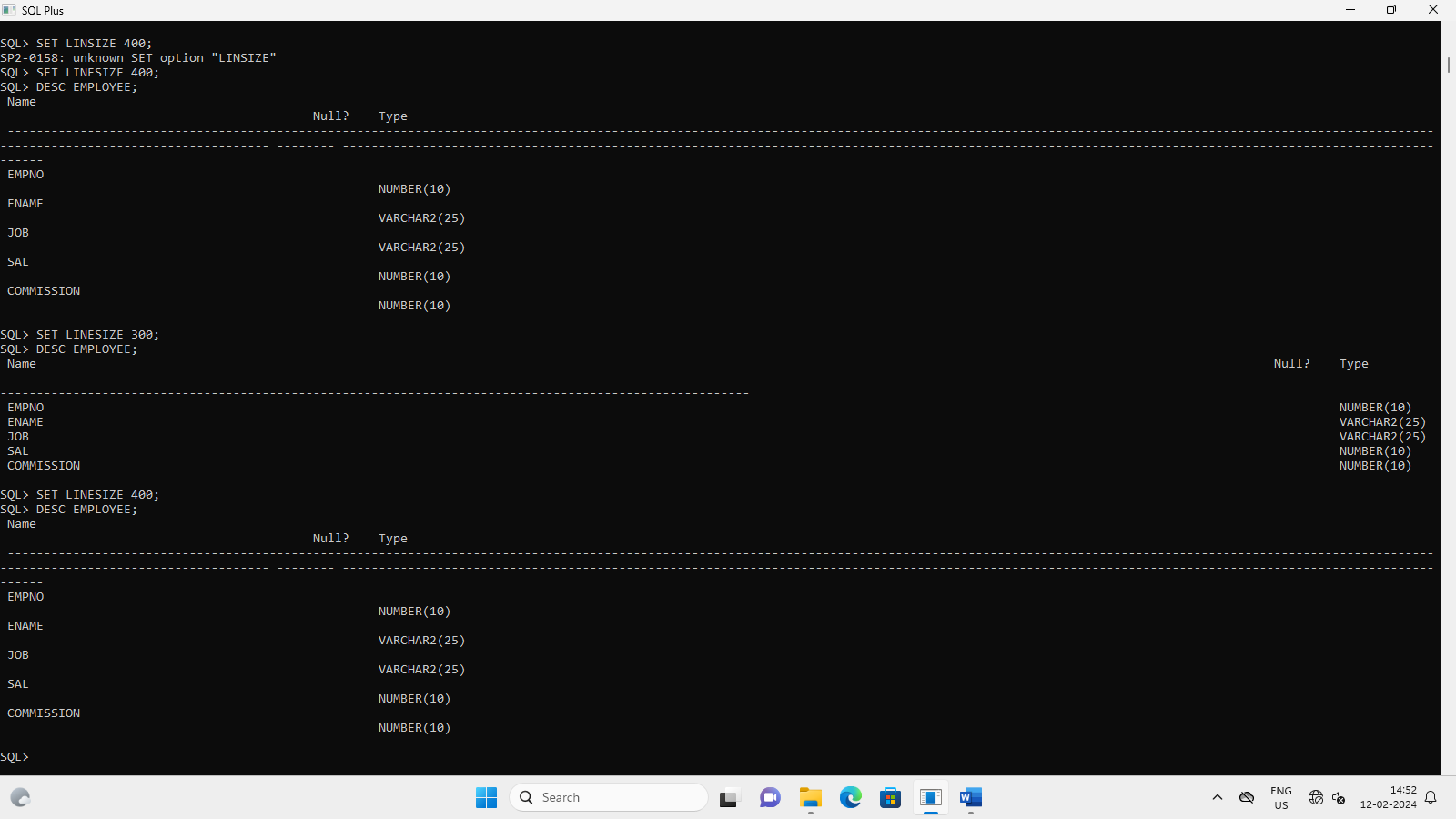
**Q) Create a table called Employee with the following structures:**

|  |  |
| --- | --- |
| **Name** | **Type** |
| **Empno** | **Number** |
| **Ename** | **Varchar2(20)** |
| **Job** | **Varchar2(20)** |
| **Sal** | **Number** |

CREATE TABLE EMPLOYEE(EMPNO NUMBER(10), ENAME VARCHAR2(25), JOB VARCHAR2(25), SAL NUMBER(10));

1. **Add a column commission with domain to the Employee table**

DESC EMPLOYEE;



1. **Insert records into the table**

INSERT INTO EMPLOYEE VALUES(&EMPNO,'&ENAME','&JOB',&SAL,&COMMISSION);

Enter value for empno: 51

Enter value for ename: MEGHA

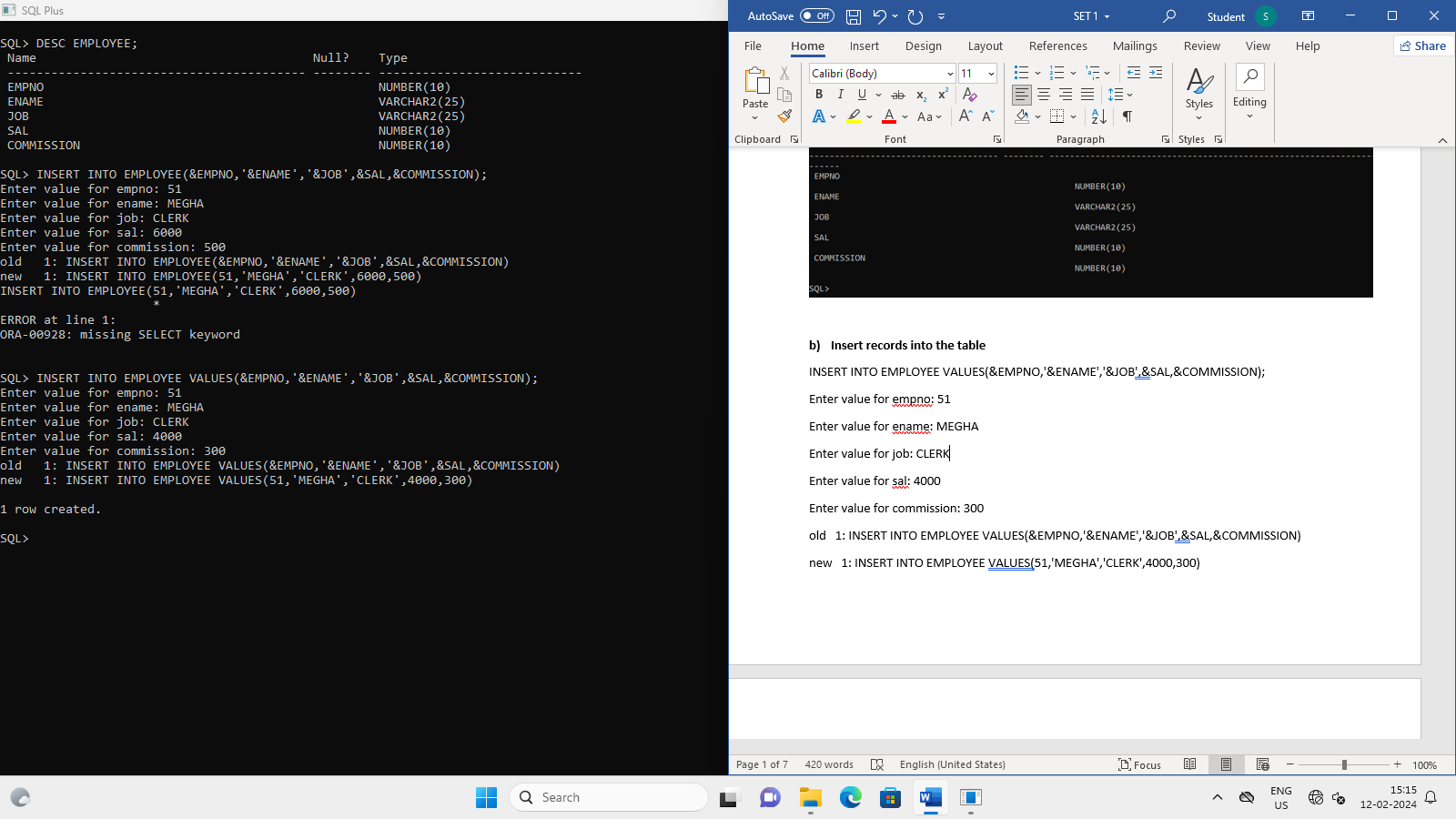
Enter value for job: CLERK

Enter value for sal: 4000

Enter value for commission: 300

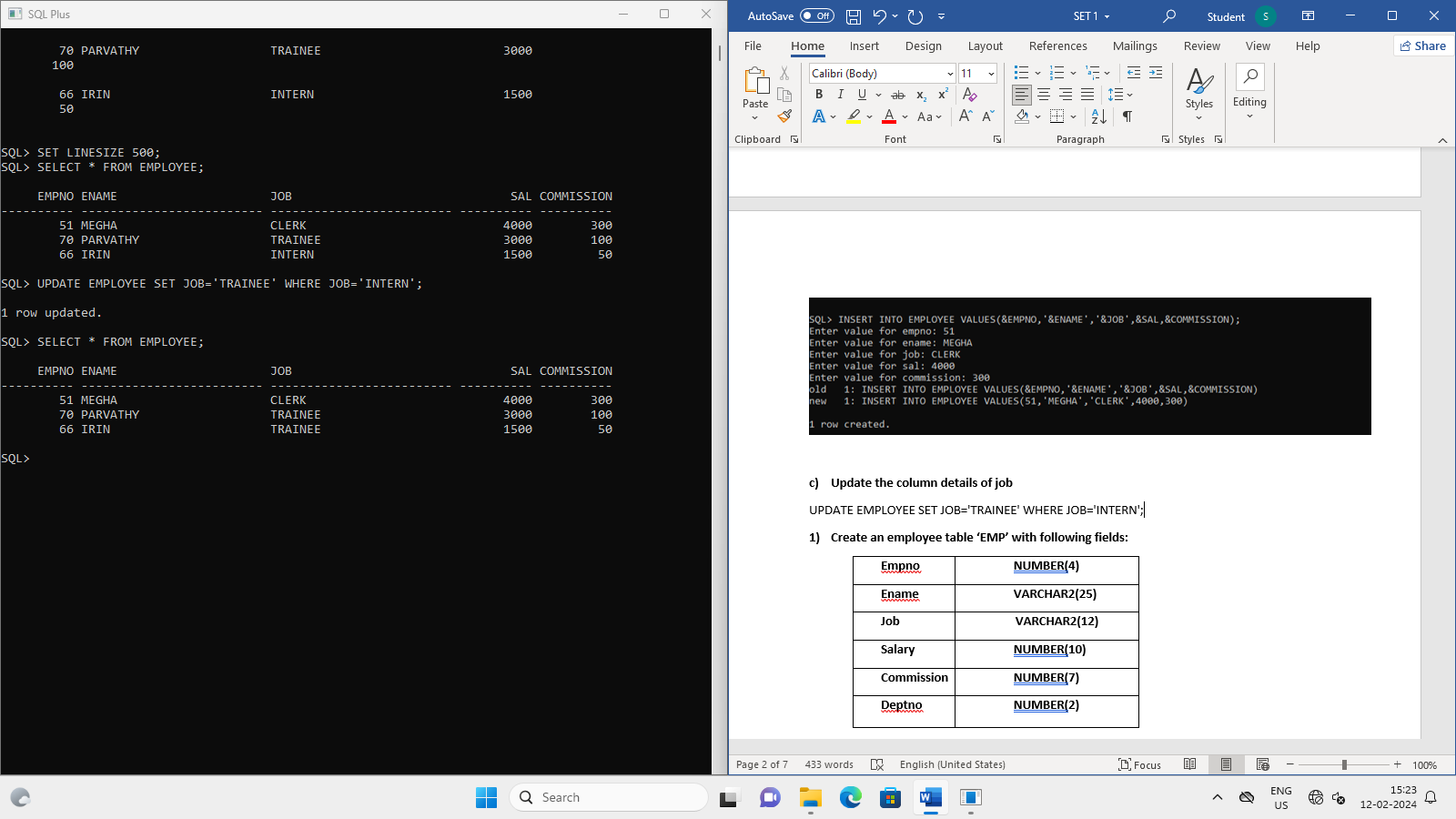
old 1: INSERT INTO EMPLOYEE VALUES(&EMPNO,'&ENAME','&JOB',&SAL,&COMMISSION)

new 1: INSERT INTO EMPLOYEE VALUES(51,'MEGHA','CLERK',4000,300)



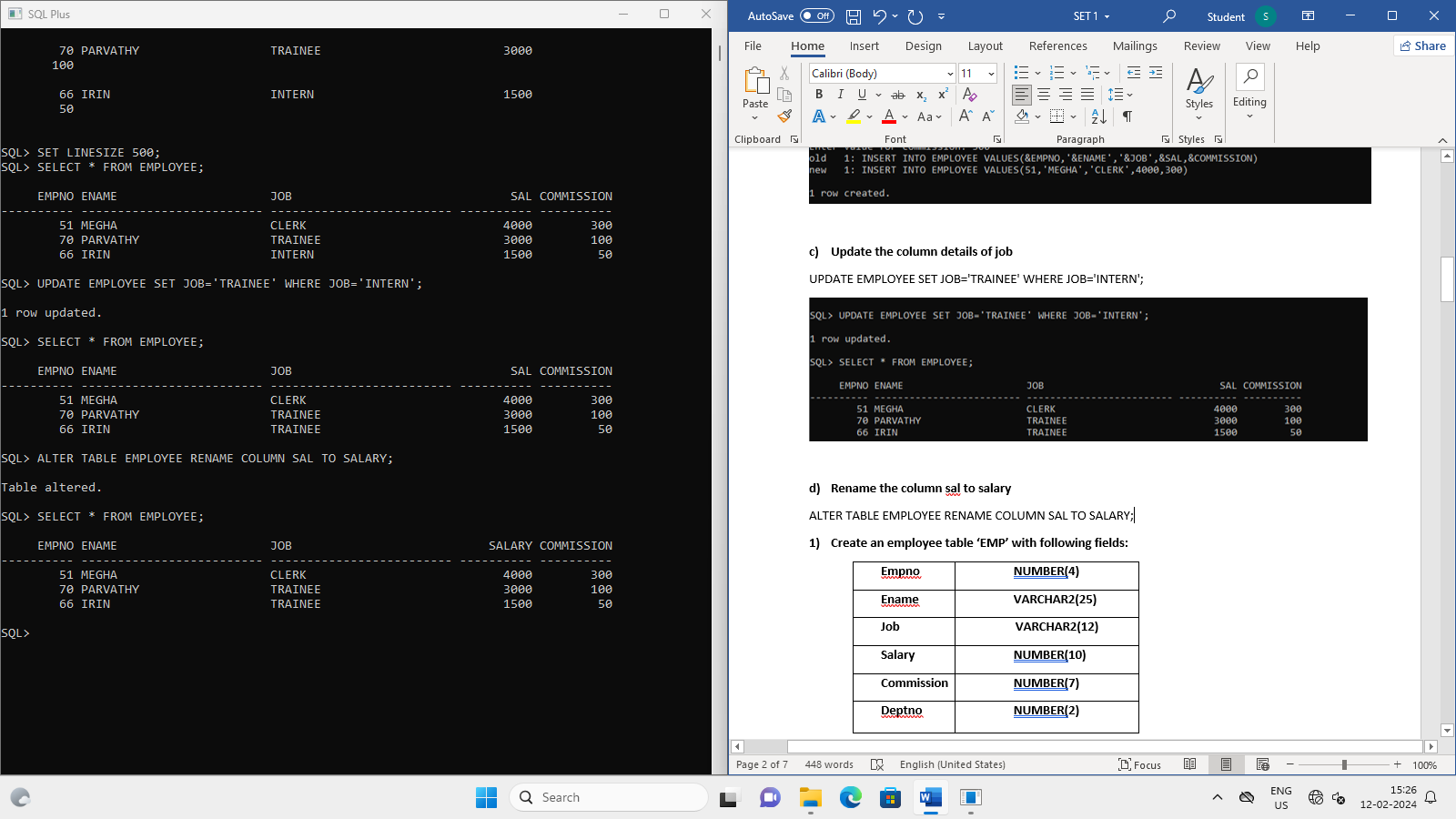
1. **Update the column details of job**

UPDATE EMPLOYEE SET JOB='TRAINEE' WHERE JOB='INTERN';



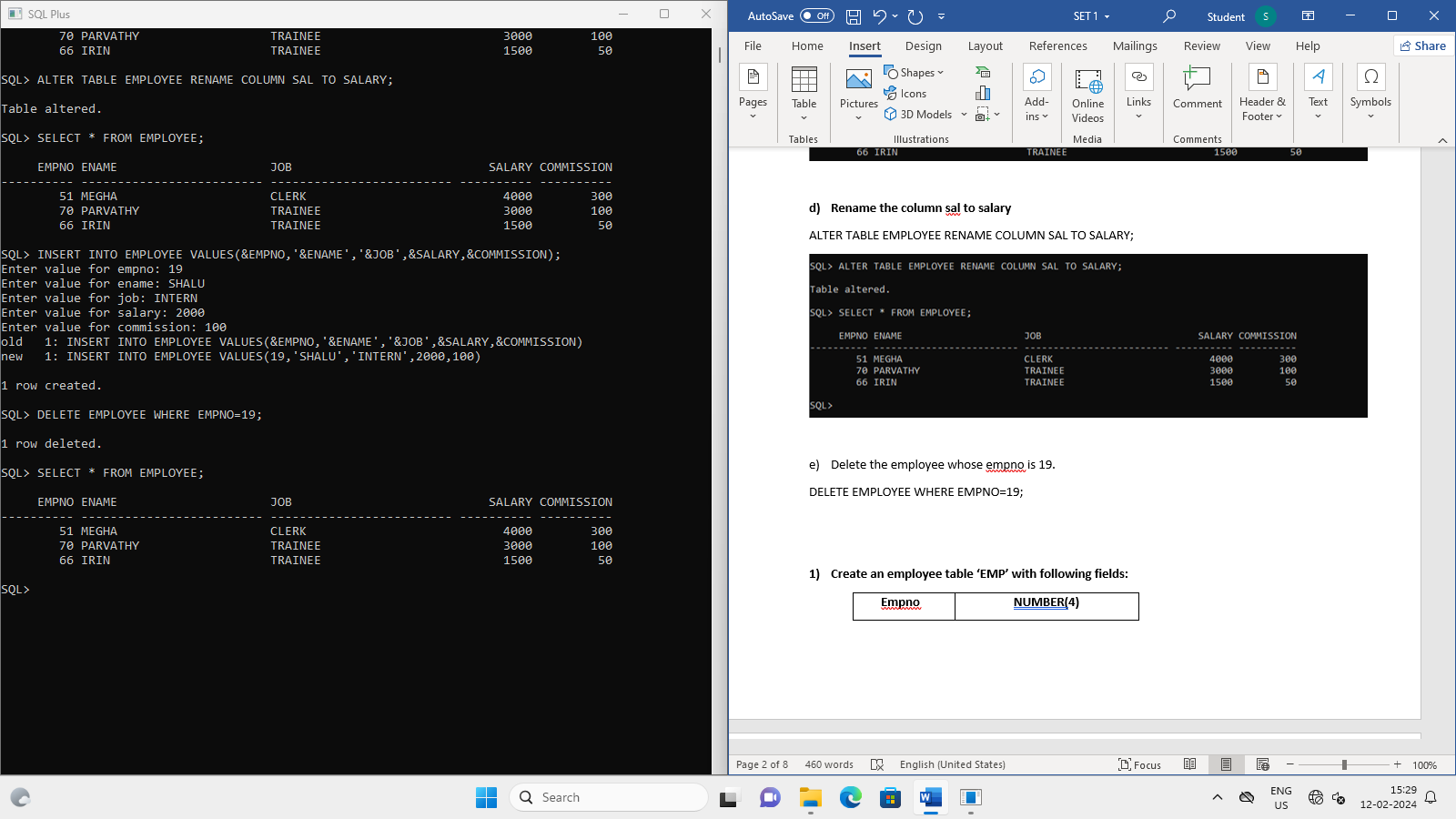
1. **Rename the column sal to salary**

ALTER TABLE EMPLOYEE RENAME COLUMN SAL TO SALARY;



1. **Delete the employee whose empno is 19.**

DELETE EMPLOYEE WHERE EMPNO=19;



**Q) Create department table with the following structures**

|  |  |
| --- | --- |
| **Name** | **Type** |
| Deptno | Number |
| Deptname | Varchar2(20) |
| Location | Varchar2(20) |

CREATE TABLE DEPARTMENT(DEPTNO NUMBER(10),DEPTNAME VARCHAR2(20),LOCATION VARCHAR2(20));

1. **Add a column designation to the department table**

ALTER TABLE DEPARTMENT ADD(DESIGNATION VARCHAR2(20));

1. **Insert values into the table**

INSERT INTO DEPARTMENT VALUES(&DEPTNO,'&DEPTNAME','&LOCATION','&DESIGNATION');

Enter value for deptno: 5

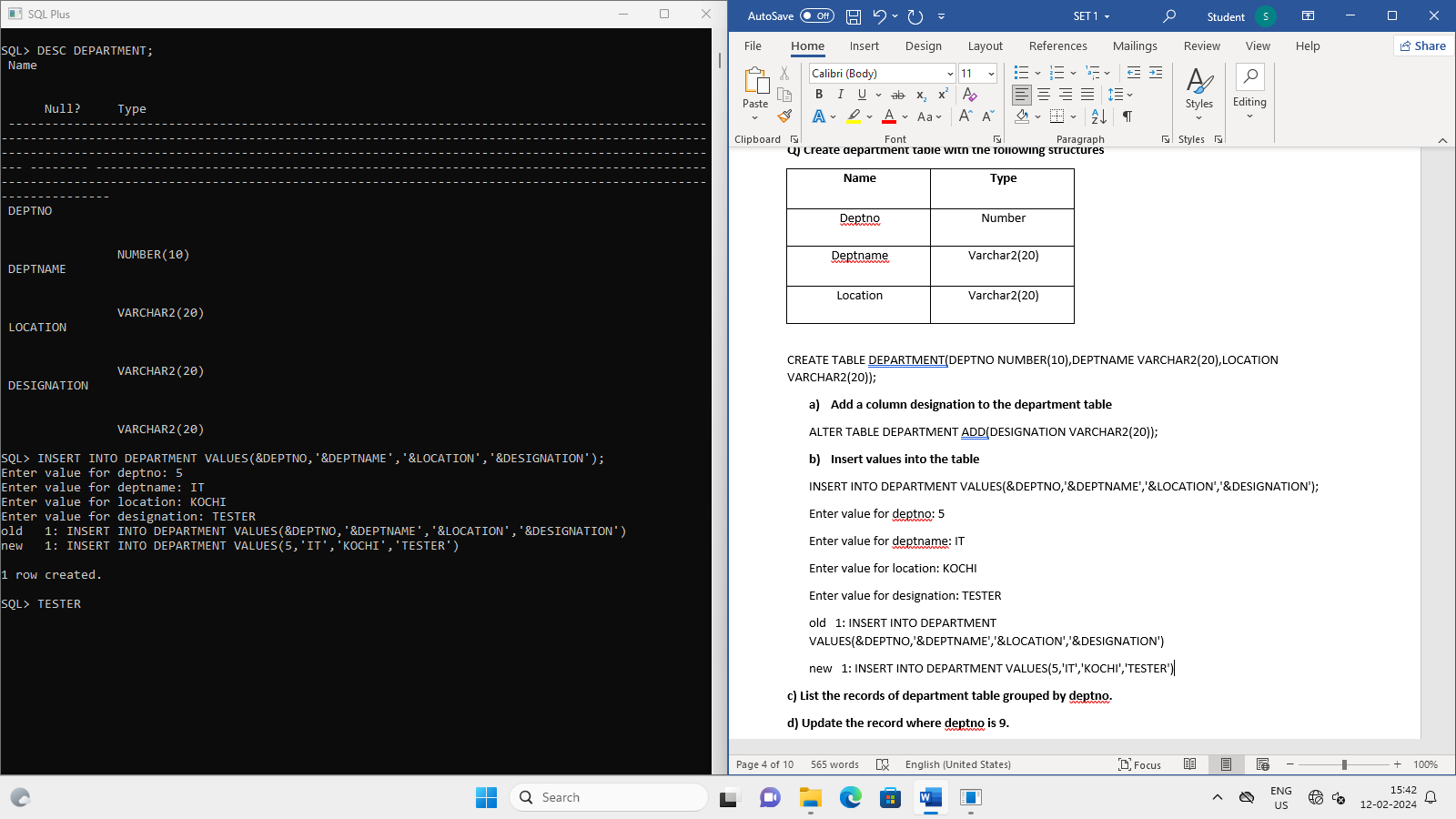
Enter value for deptname: IT

Enter value for location: KOCHI

Enter value for designation: TESTER

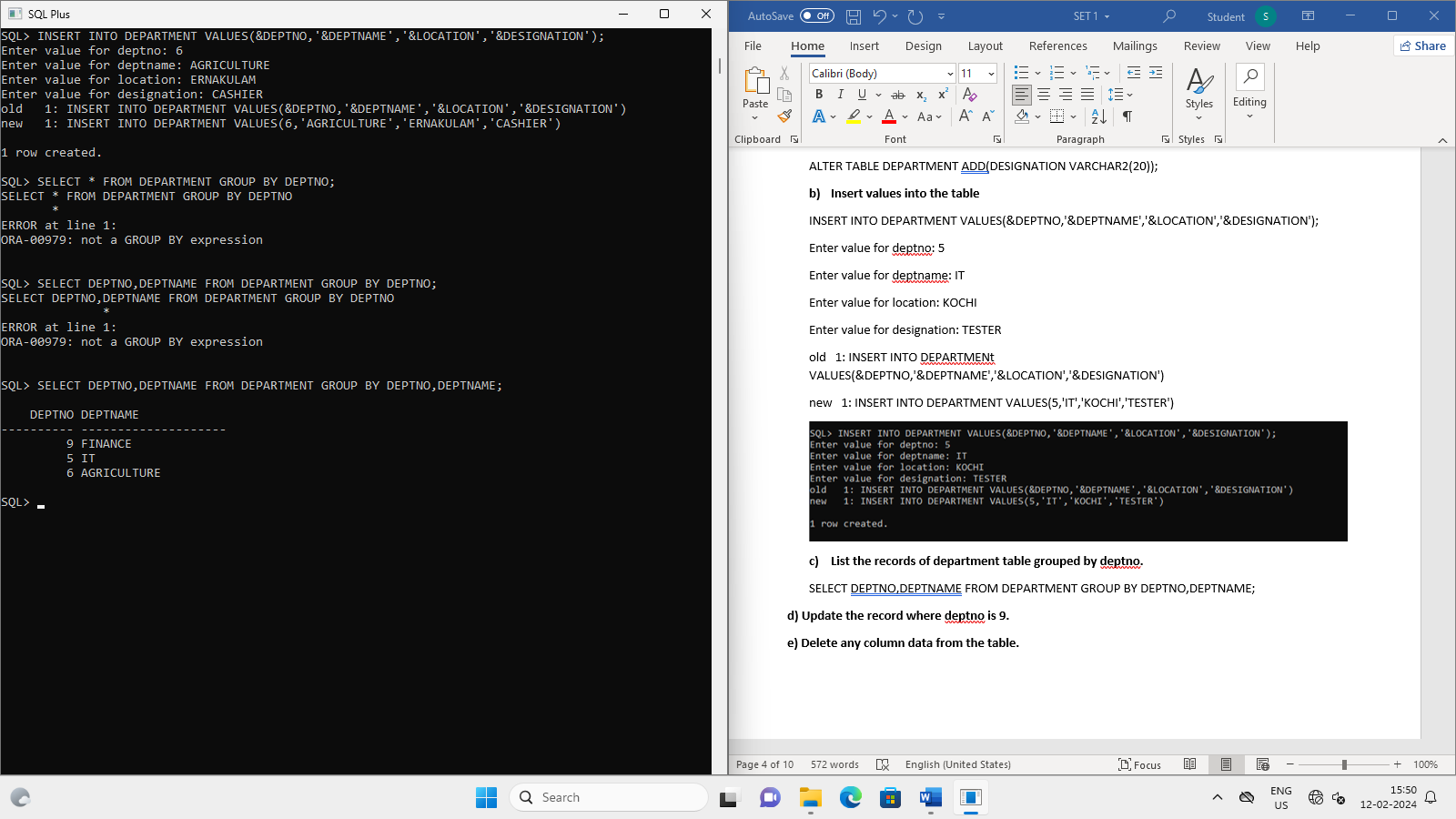
old 1: INSERT INTO DEPARTMENT VALUES(&DEPTNO,'&DEPTNAME','&LOCATION','&DESIGNATION')

new 1: INSERT INTO DEPARTMENT VALUES(5,'IT','KOCHI','TESTER')



1. **List the records of department table grouped by deptno.**

SELECT DEPTNO,DEPTNAME FROM DEPARTMENT GROUP BY DEPTNO,DEPTNAME;



1. **Update the record where deptno is 9.**

UPDATE DEPARTMENT SET DEPTNAME='ACCOUNTING' WHERE DEPTNO=9;

1. **Delete any column data from the table.**

ALTER TABLE DEPARTMENT DROP(LOCATION);

**AGGREGATE FUNCTIONS**

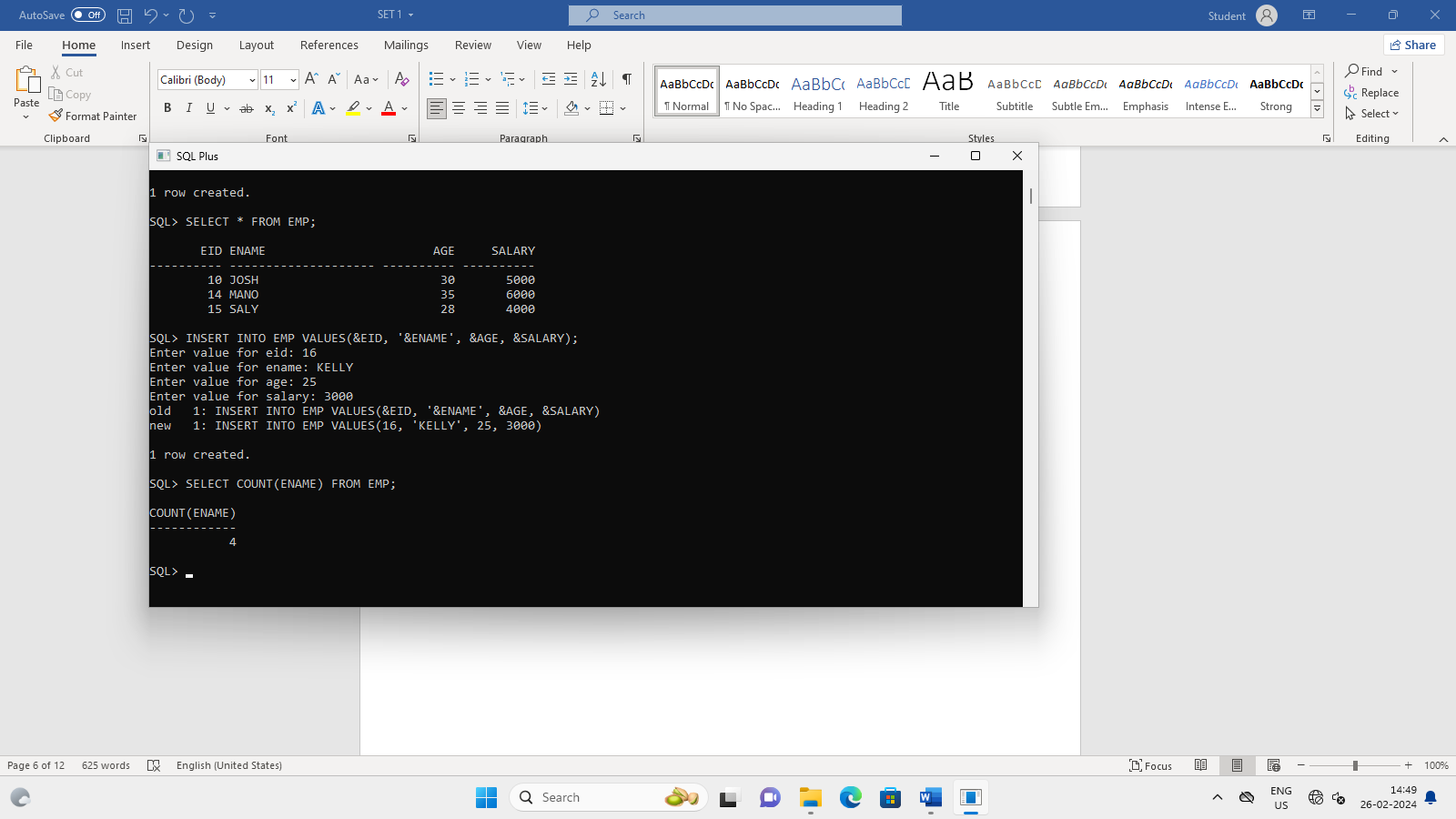
**Q) Create a table EMP with attributes EID, ENAME, AGE, SALARY**

CREATE TABLE EMP(EID NUMBER(10), ENAME VARCHAR2(20), AGE NUMBER(10), SALARY NUMBER(10));

INSERT INTO EMP VALUES(&EID, '&ENAME', &AGE, &SALARY);

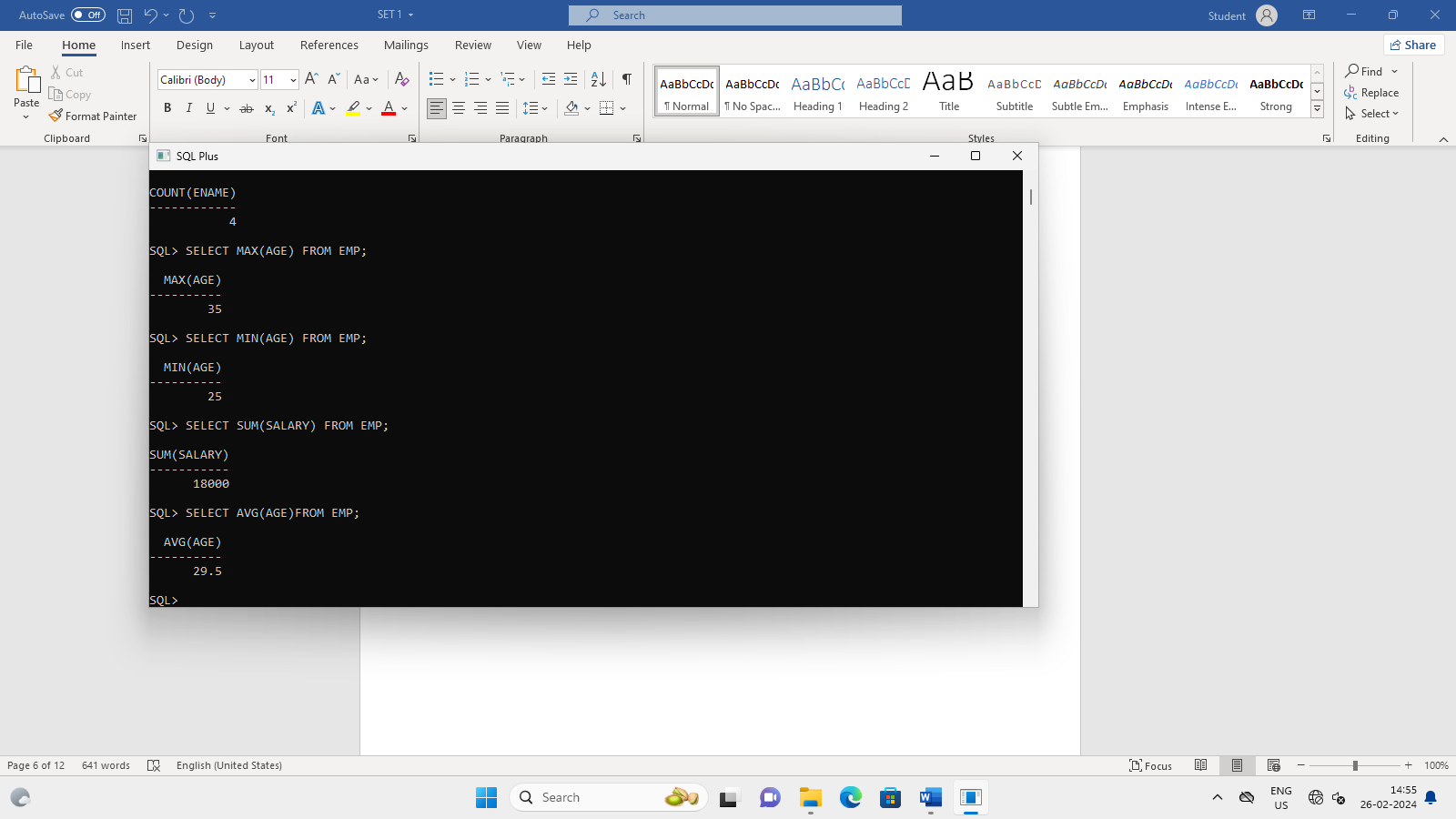
1. Find the count of the employees

SELECT COUNT(ENAME) FROM EMP;



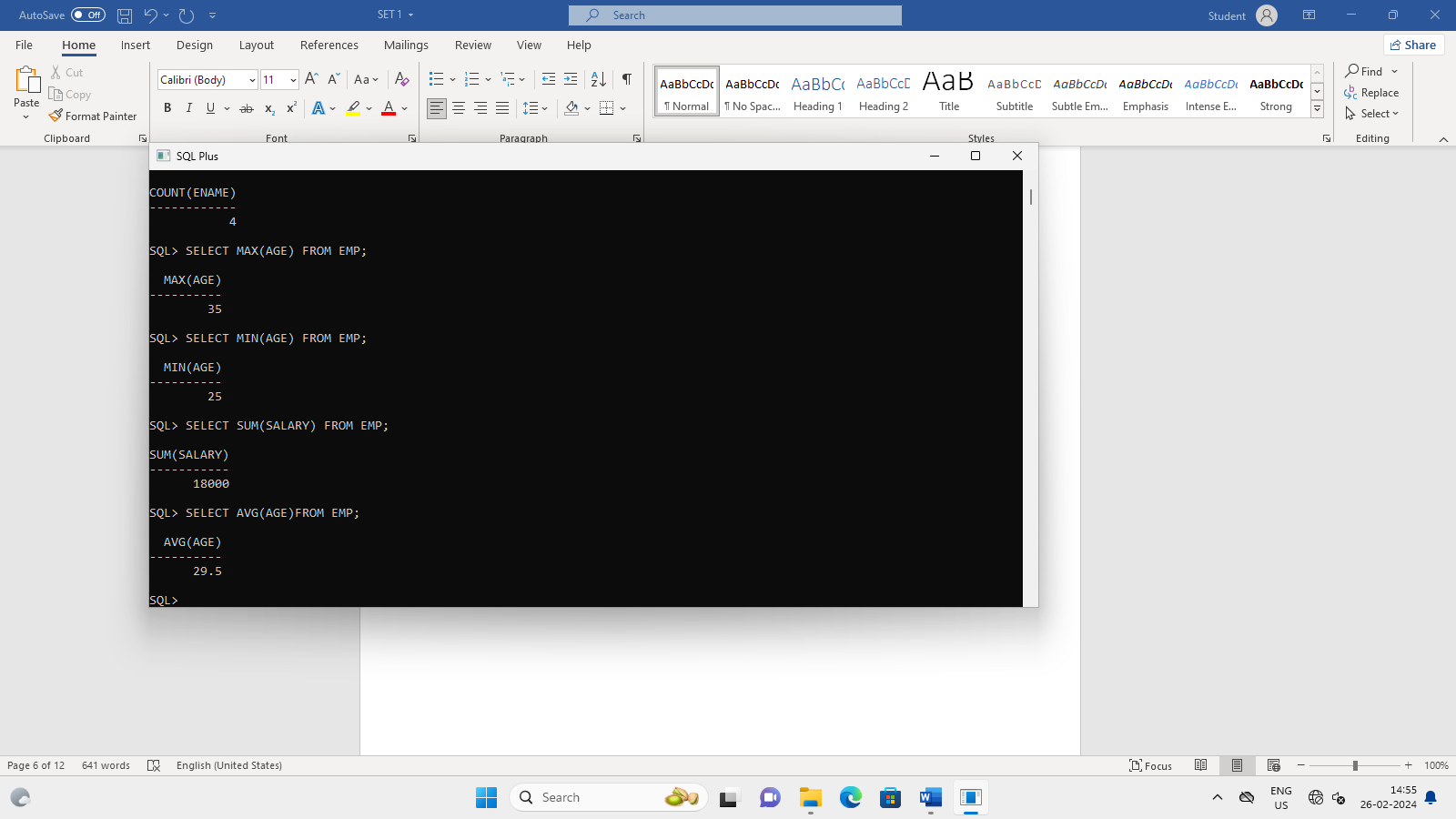
1. Find the maximum age from emp table

SELECT MAX(AGE) FROM EMP;



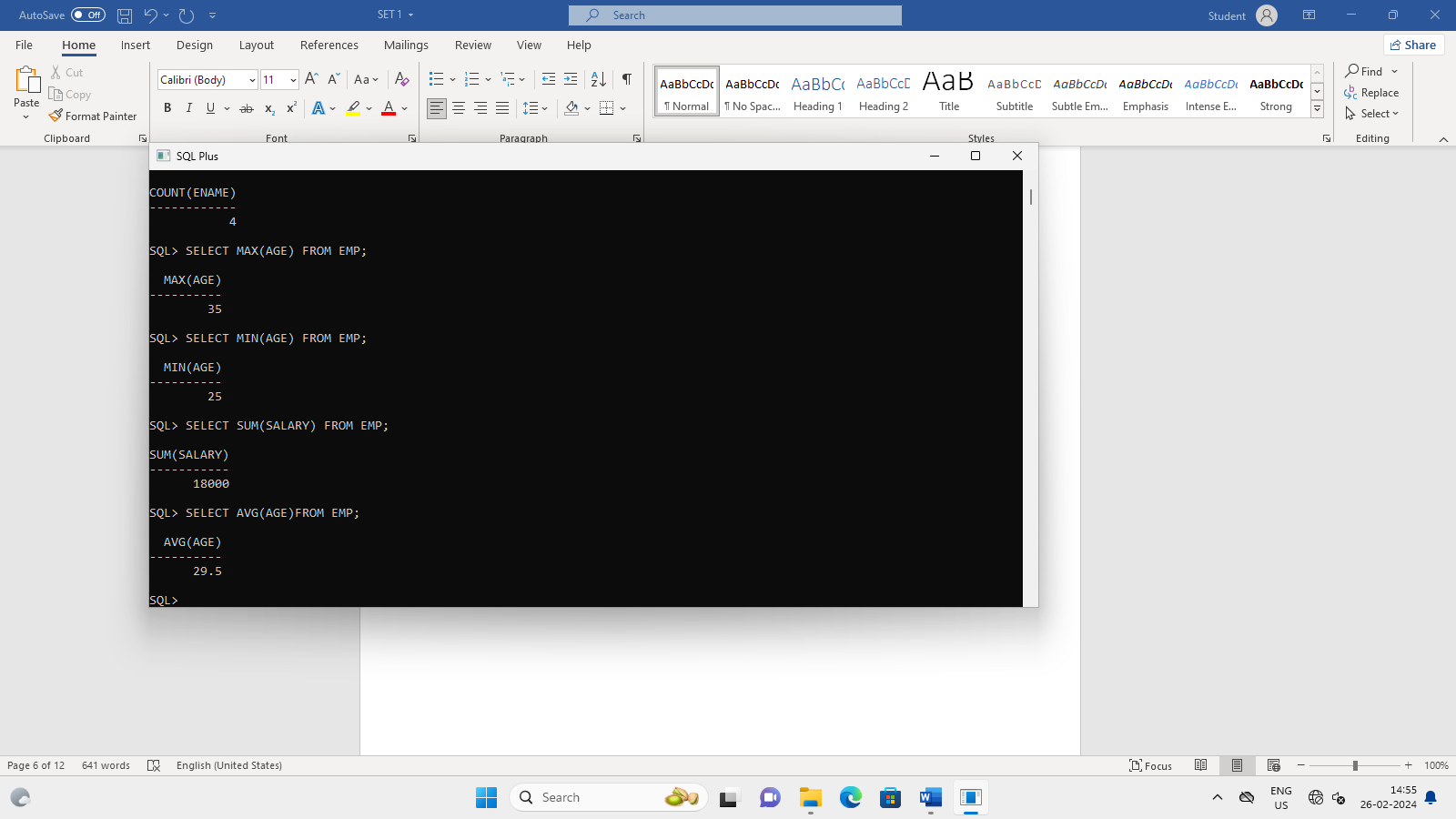
1. Find the minimum age

SELECT MIN(AGE) FROM EMP;



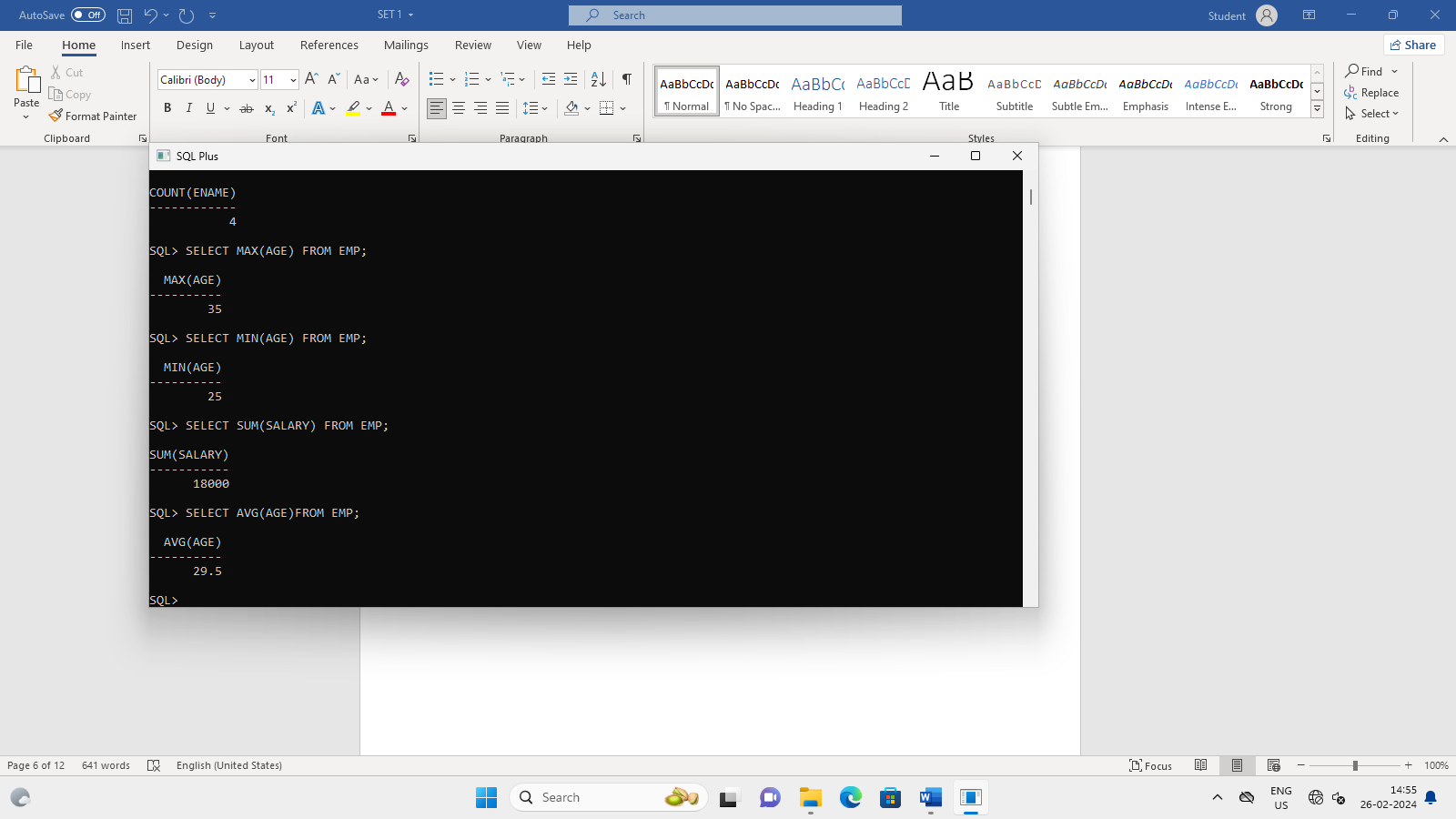
1. Find the sum of salary

SELECT SUM(SALARY) FROM EMP;



1. Find the average age

SELECT AVG(AGE)FROM EMP;



**VIEWS**

1. Create a view for name and age

CREATE OR REPLACE VIEW AgeView AS SELECT AGE, ENAME FROM EMP WHERE AGE<30;

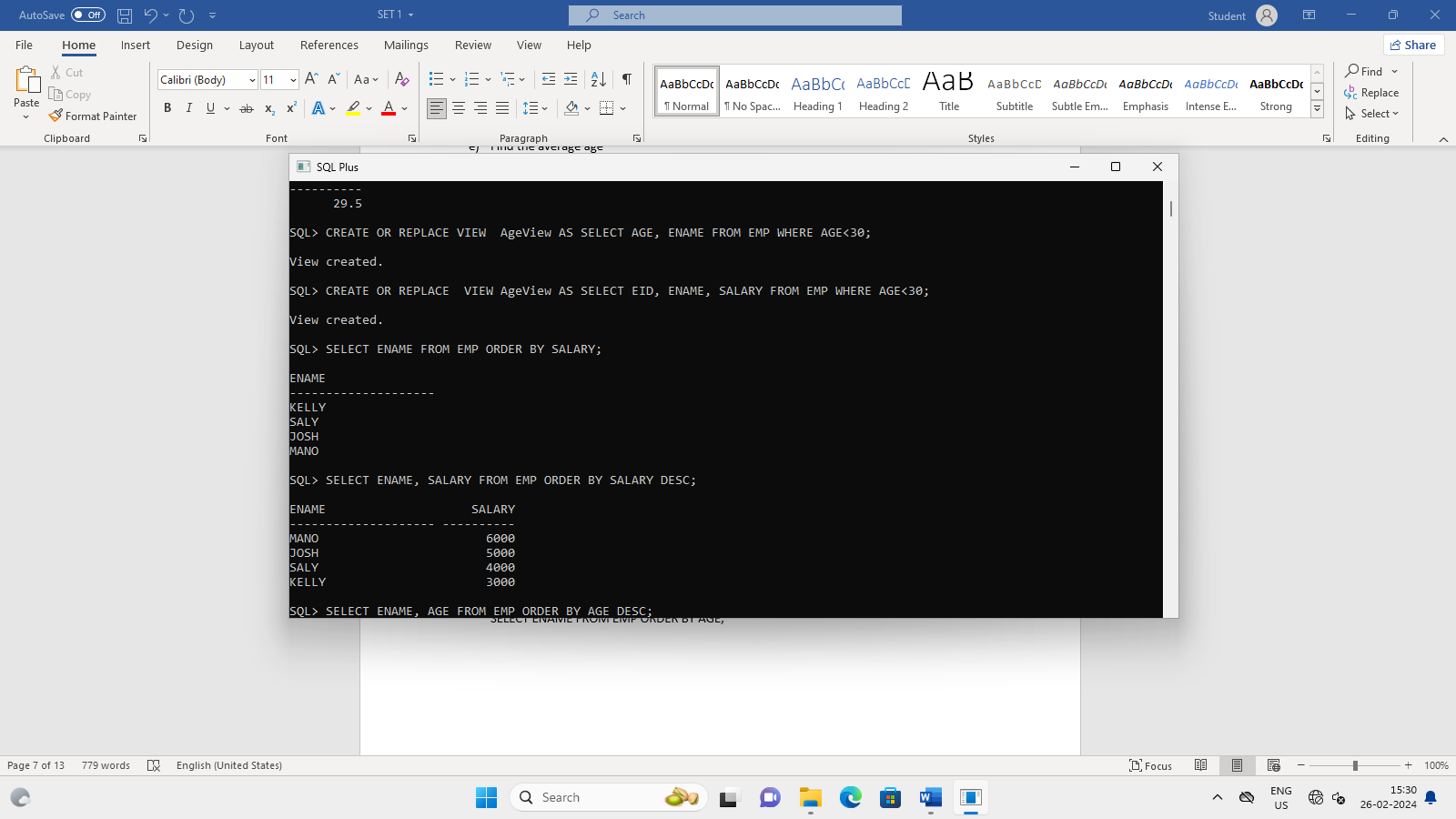
1. Create a view to store id, name, salary

CREATE OR REPLACE VIEW AgeView AS SELECT EID, ENAME, SALARY FROM EMP WHERE AGE<30;

1. Display the name of the employee in the descending order of salary

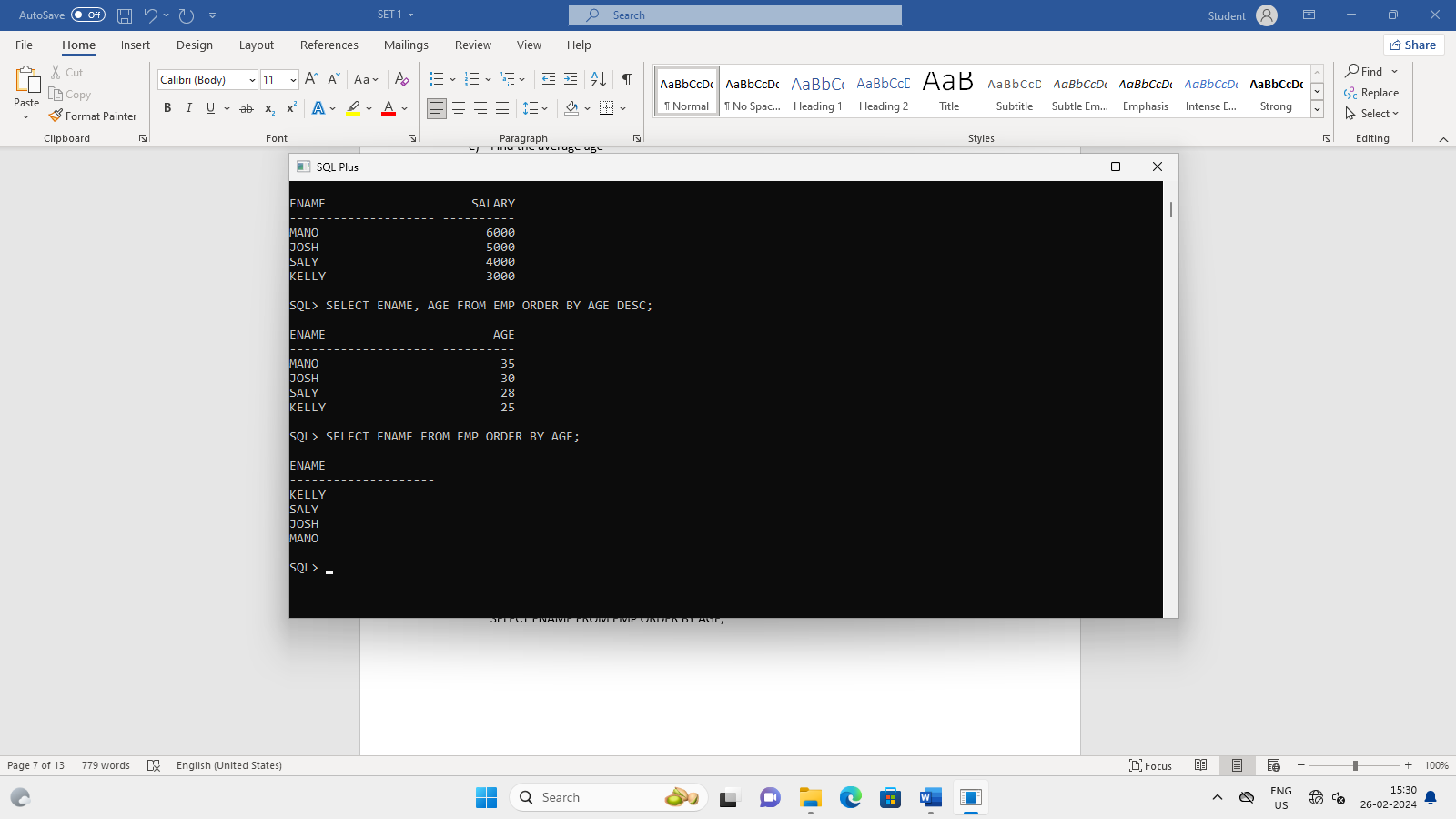
SELECT ENAME FROM EMP ORDER BY SALARY;

SELECT ENAME, SALARY FROM EMP ORDER BY SALARY DESC;



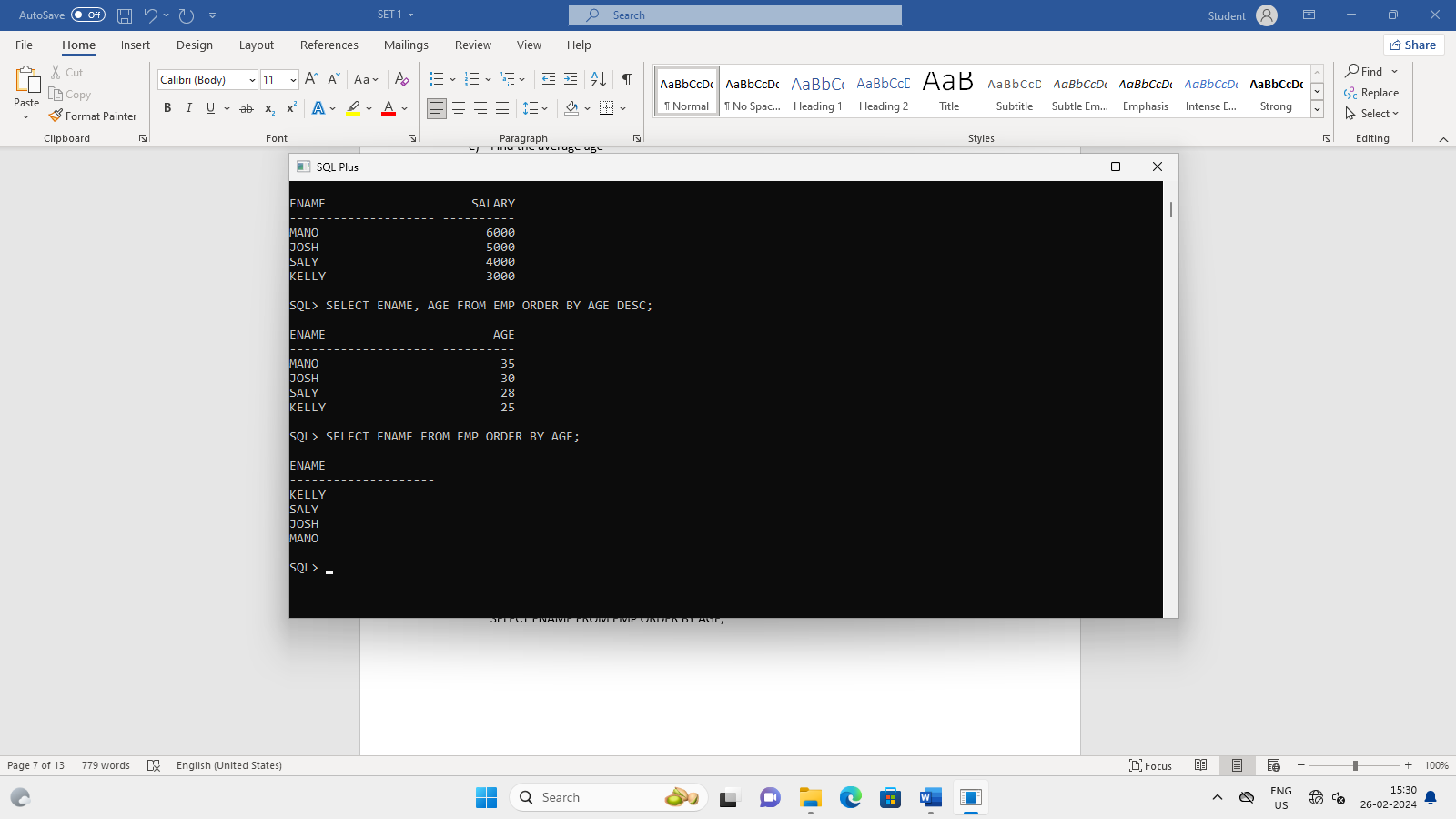
1. Print the name of employee in descending order by age

SELECT ENAME, AGE FROM EMP ORDER BY AGE DESC;



1. Print name of employee in ascending order by age

SELECT ENAME FROM EMP ORDER BY AGE;



**Q) Create a table called Customer table**

|  |  |
| --- | --- |
| **Name** | **Type** |
| **Cust Name** | **Varchar2(20)** |
| **Cust Street** | **Varchar2(20)** |
| **Cust City** | **Varchar2(20)** |

CREATE TABLE CUSTOMER(CUST\_NAME VARCHAR2(20), CUST\_STREET VARCHAR2(20), CUST\_CITY VARCHAR(20));

1. Insert records into the table

INSERT INTO CUSTOMER VALUES('&CUST\_NAME', '&CUST\_STREET', '&CUST\_CITY');

1. Add salary column to the table

ALTER TABLE CUSTOMER ADD(SALARY NUMBER(10));

1. Alter the table column domain

ALTER TABLE CUSTOMER RENAME COLUMN SALARY TO CUST\_SALARY;

1. Drop salary column of the customer table

ALTER TABLE CUSTOMER DROP(CUST\_SALARY);

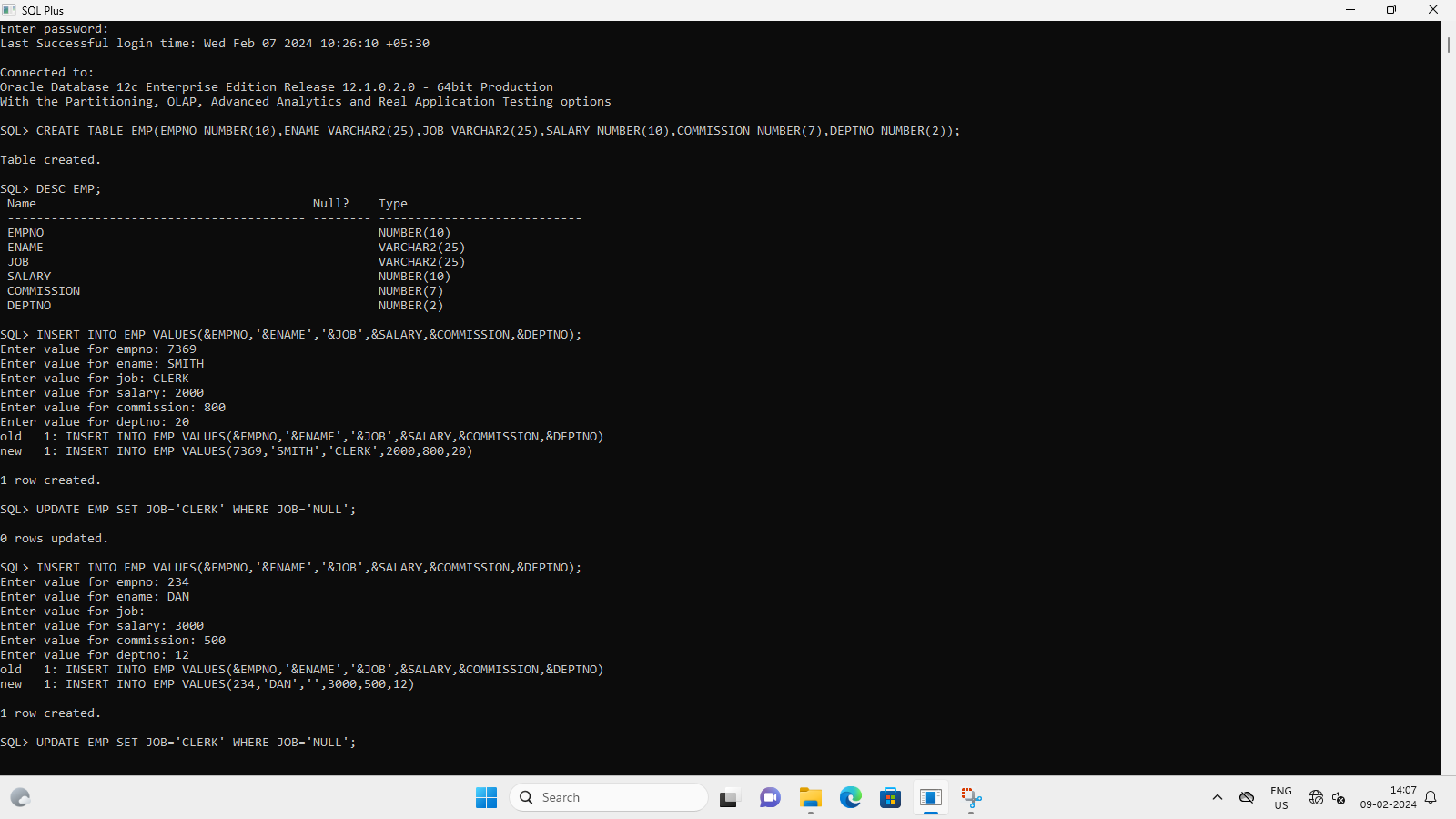
1. Delete the rows of customers table whose cust\_city is “hyderabad”.

DELETE CUSTOMER WHERE CUST\_CITY='HYDERABAD';

**SET-1**

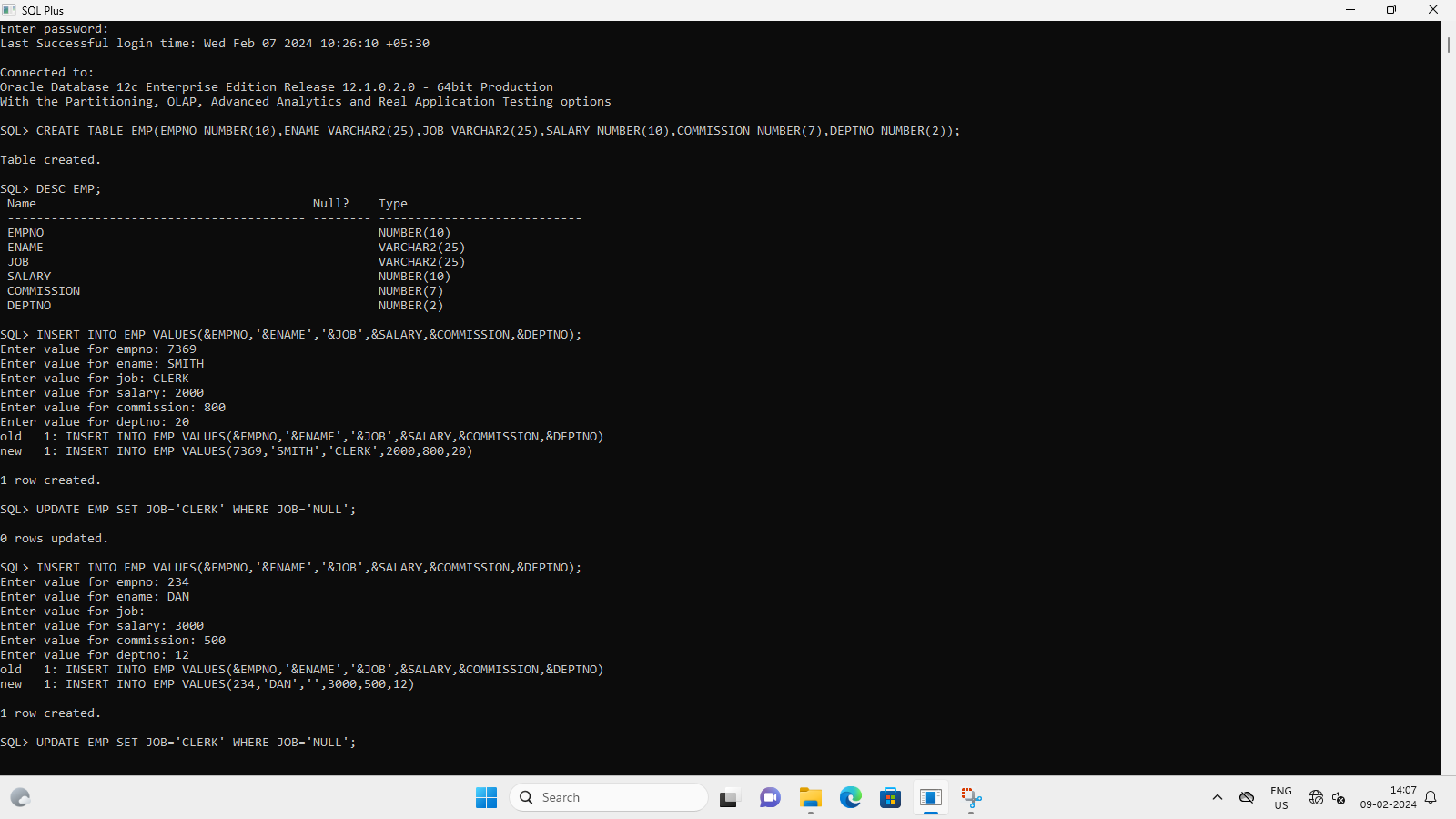
1. **Create an employee table ‘EMP’ with following fields:**

|  |  |
| --- | --- |
| **Empno** | **NUMBER(4)** |
| **Ename** | **VARCHAR2(25)** |
| **Job** | **VARCHAR2(12)** |
| **Salary** | **NUMBER(10)** |
| **Commission** | **NUMBER(7)** |
| **Deptno** | **NUMBER(2)** |



CREATE TABLE EMP(EMPNO NUMBER(10),ENAME VARCHAR2(25),JOB VARCHAR2(25),SALARY NUMBER(10),COMMISSION NUMBER(7),DEPTNO NUMBER(2));

1. **Display the structure of the table**



DESC EMP;

Name Null? Type

----------------------------------------- -------- ----------------------------

EMPID NUMBER(10)

EMPNAME VARCHAR2(25)

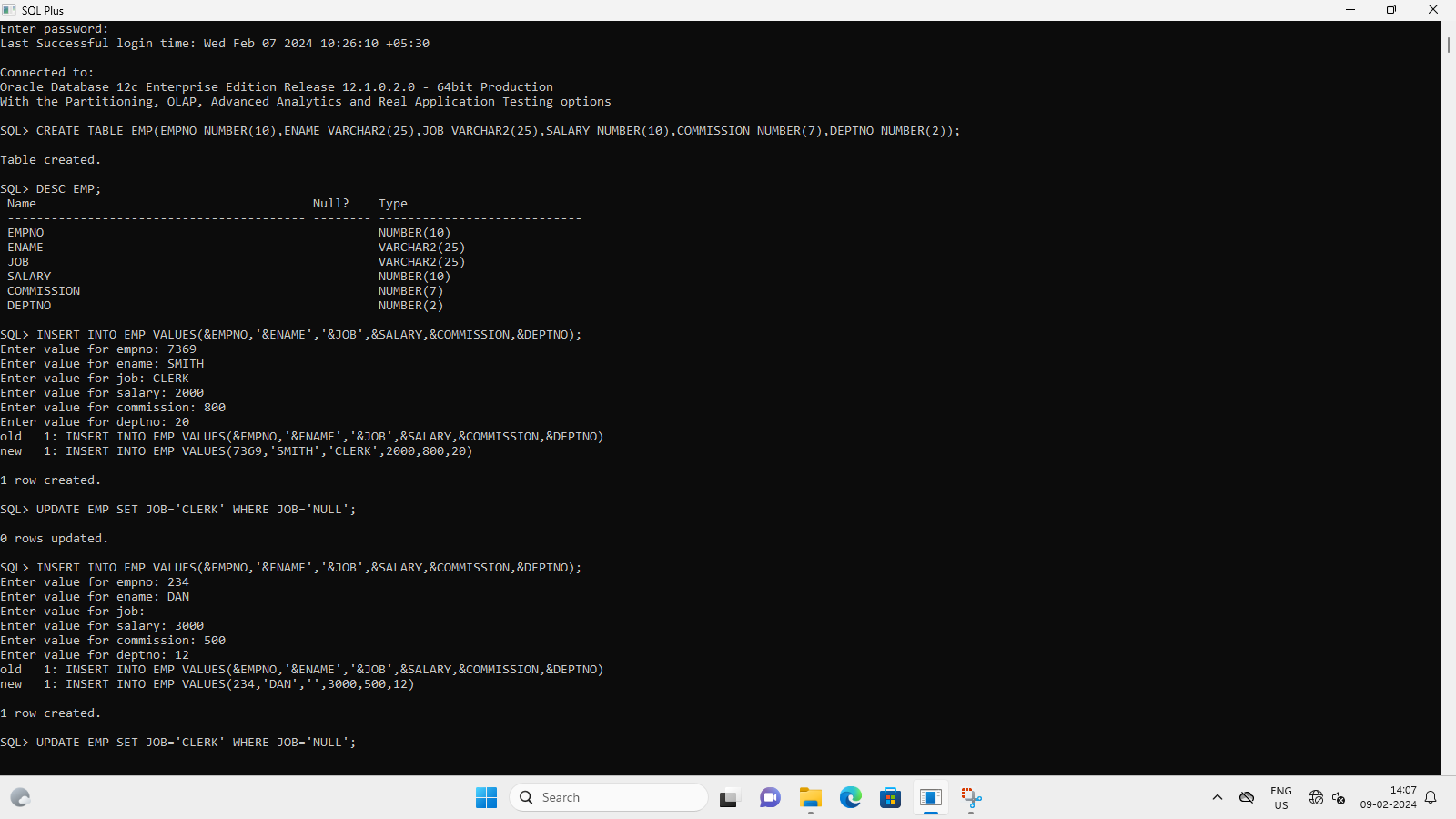
JOB VARCHAR2(25)

SALARY NUMBER(10)

DEPTNO NUMBER(2)

DOJ DATE

1. **Insert records into the table**



INSERT INTO EMP VALUES(&EMPNO,'&ENAME','&JOB',&SALARY,&COMMISSION,&DEPTNO);

Enter value for empno: 7369

Enter value for ename: SMITH

Enter value for job: CLERK

Enter value for salary: 2000

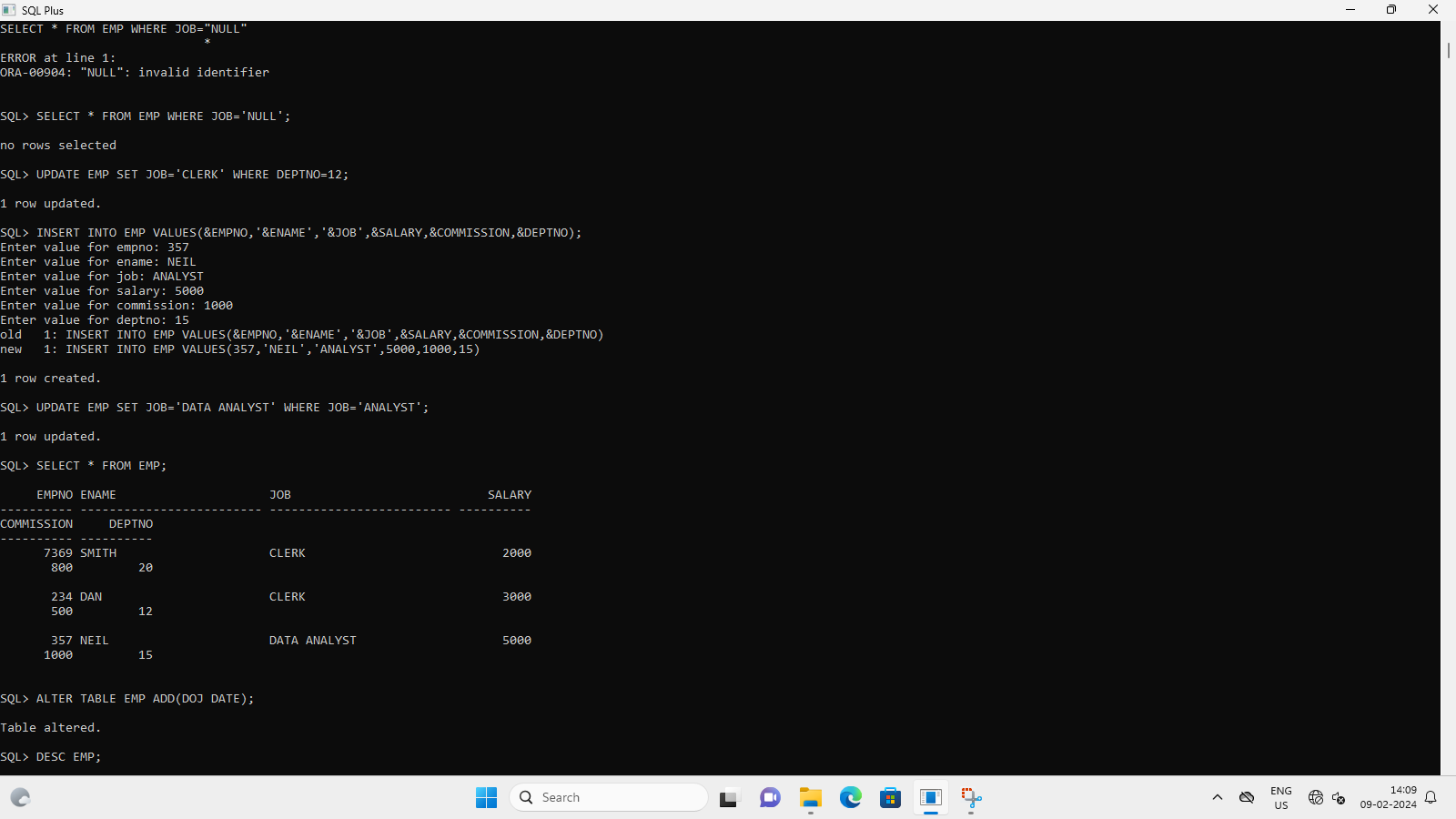
Enter value for commission: 800

Enter value for deptno: 20

1. **Insert job as ‘CLERK’ for all NULL job types**

UPDATE EMP SET JOB='CLERK' WHERE JOB='NULL';

1. **Change the job type ‘ANALYST’ to ‘DATA ANALYST’**

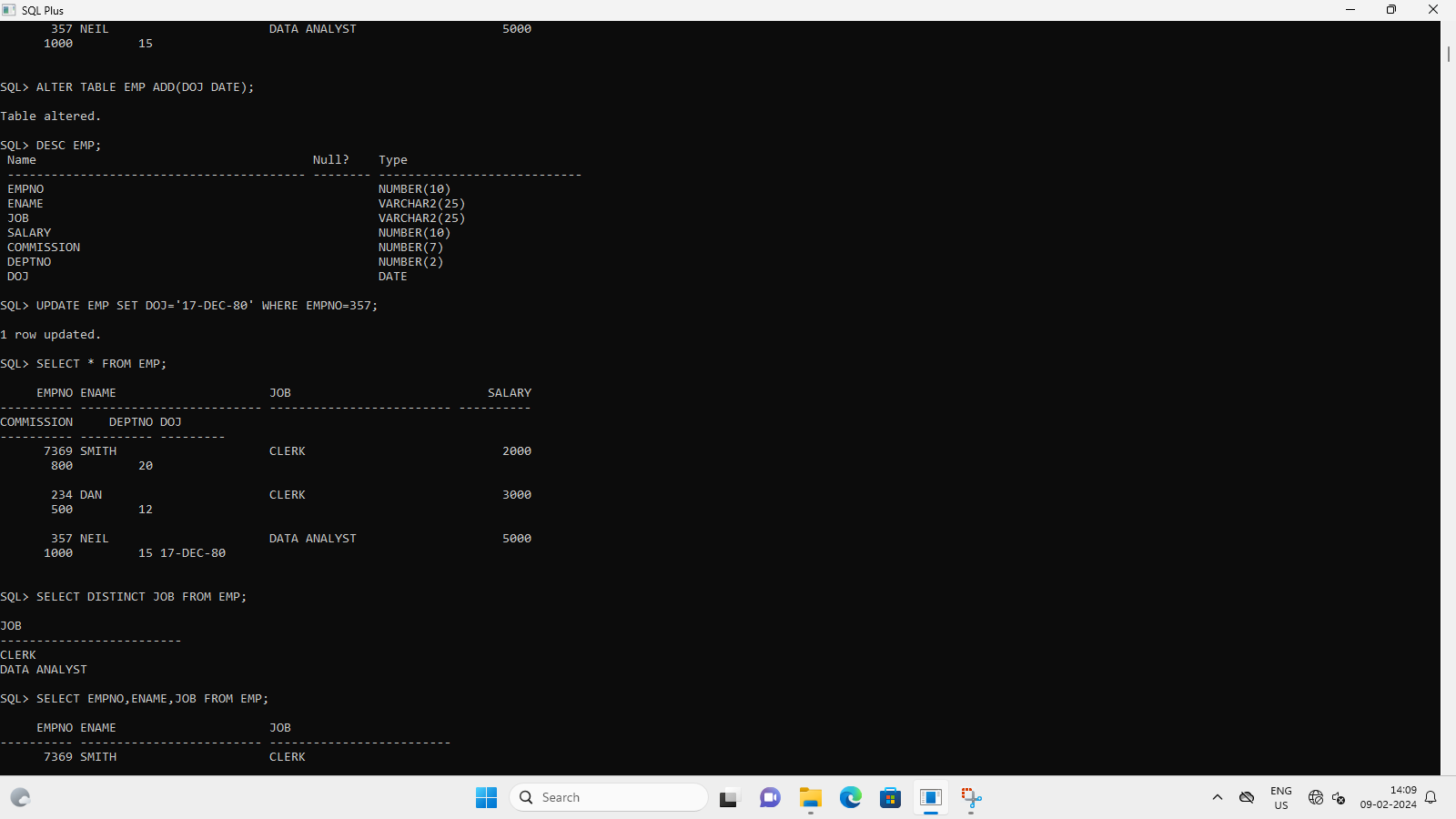


UPDATE EMP SET JOB='DATA ANALYST' WHERE JOB='ANALYST';

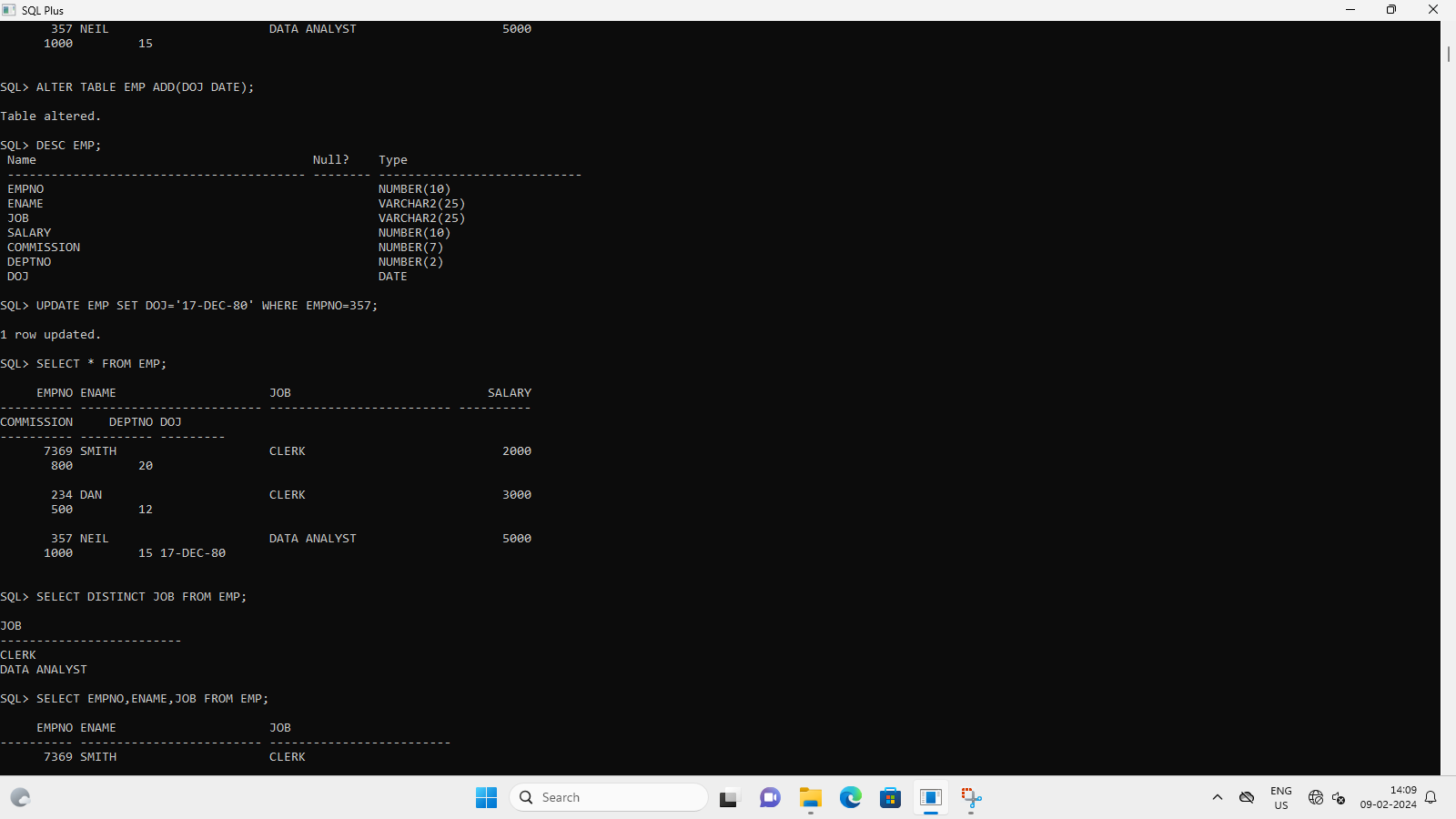
1. **Add a new field ‘doj’ and insert the date 17-DEC-80**

ALTER TABLE EMP ADD(DOJ DATE);

UPDATE EMP SET DOJ='17-DEC-80' WHERE EMPNO=357;

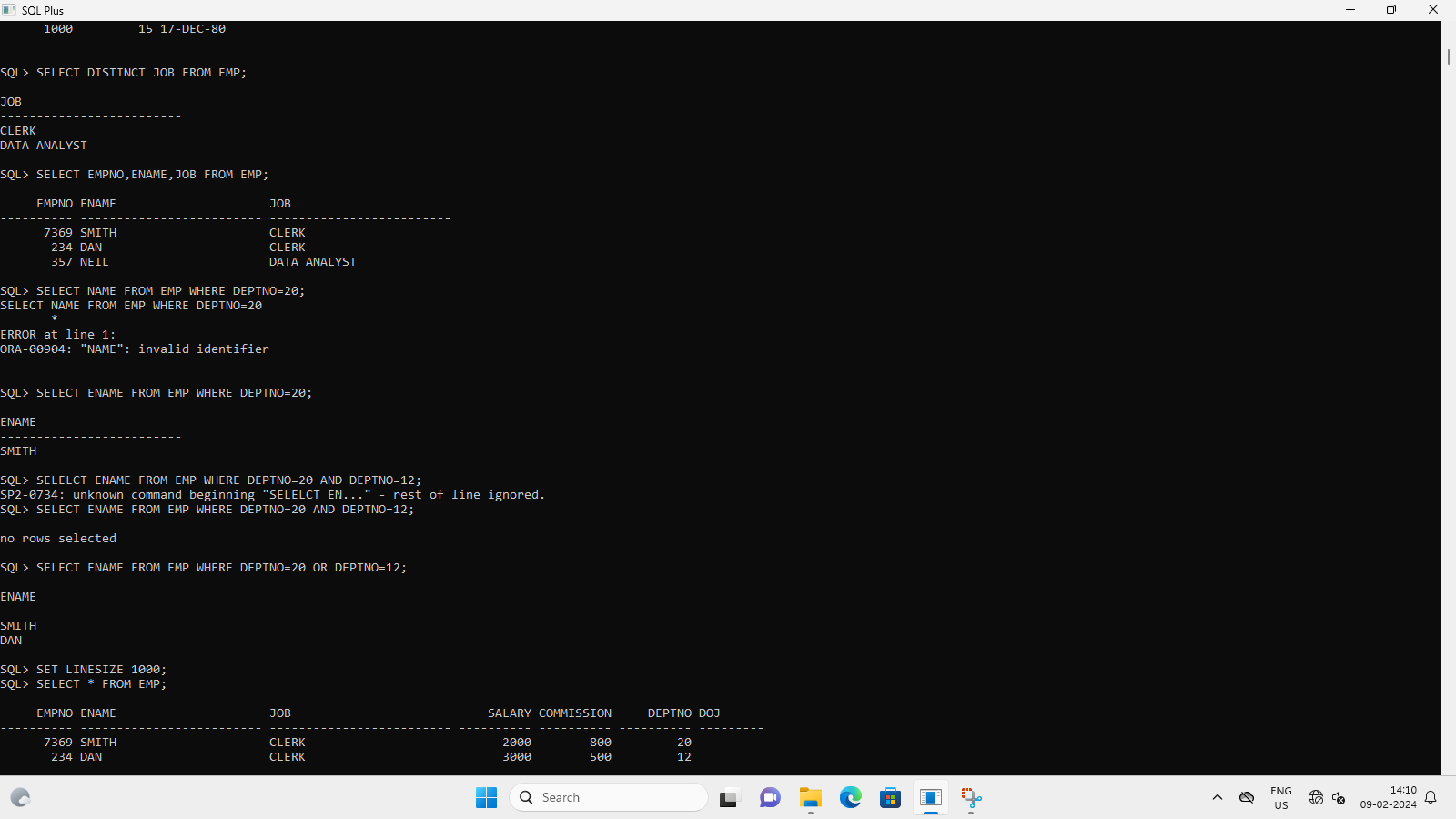


1. **Display all distinct job type**



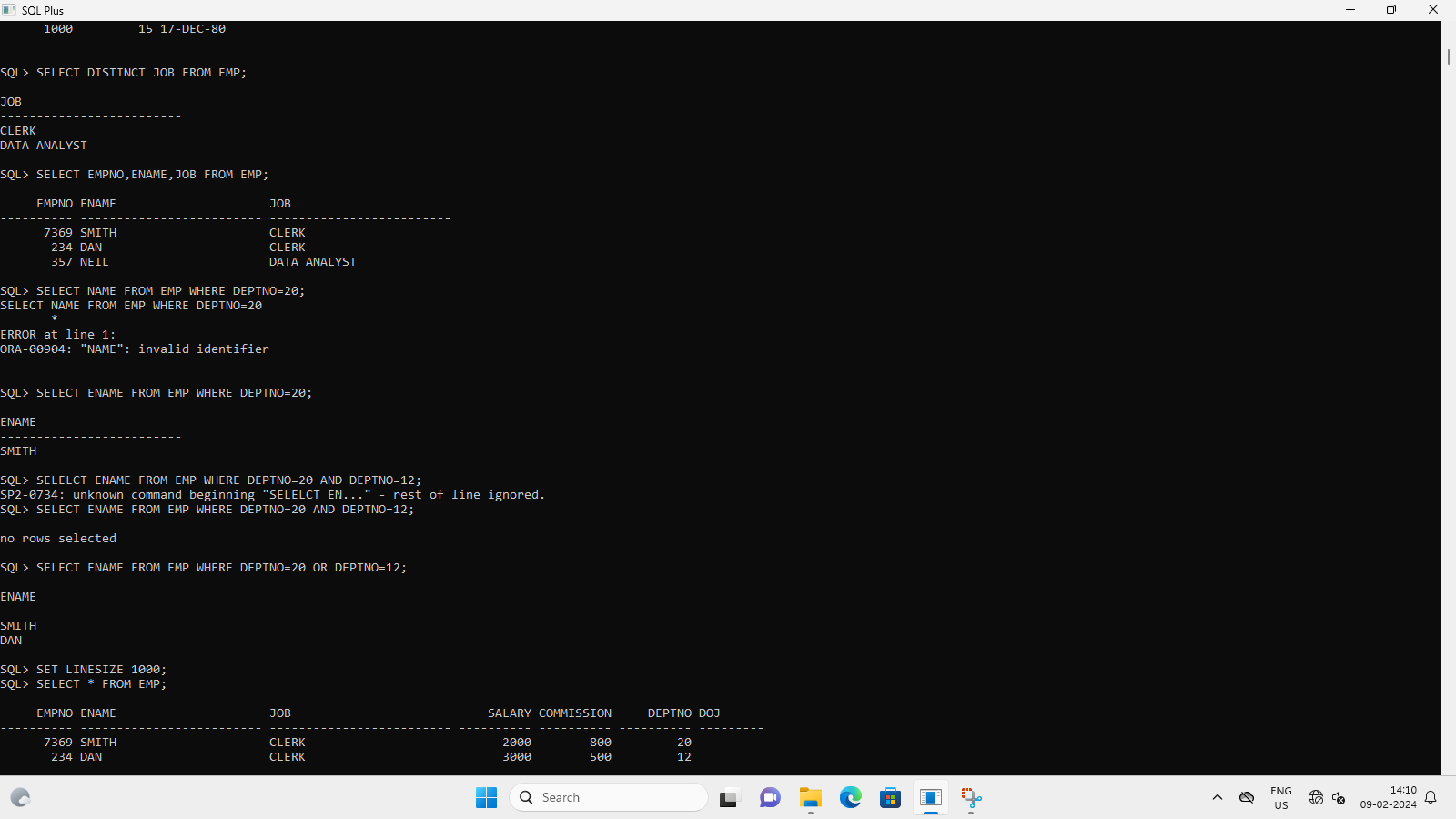
SELECT DISTINCT JOB FROM EMP;

1. **Display empno, name, job from table emp**



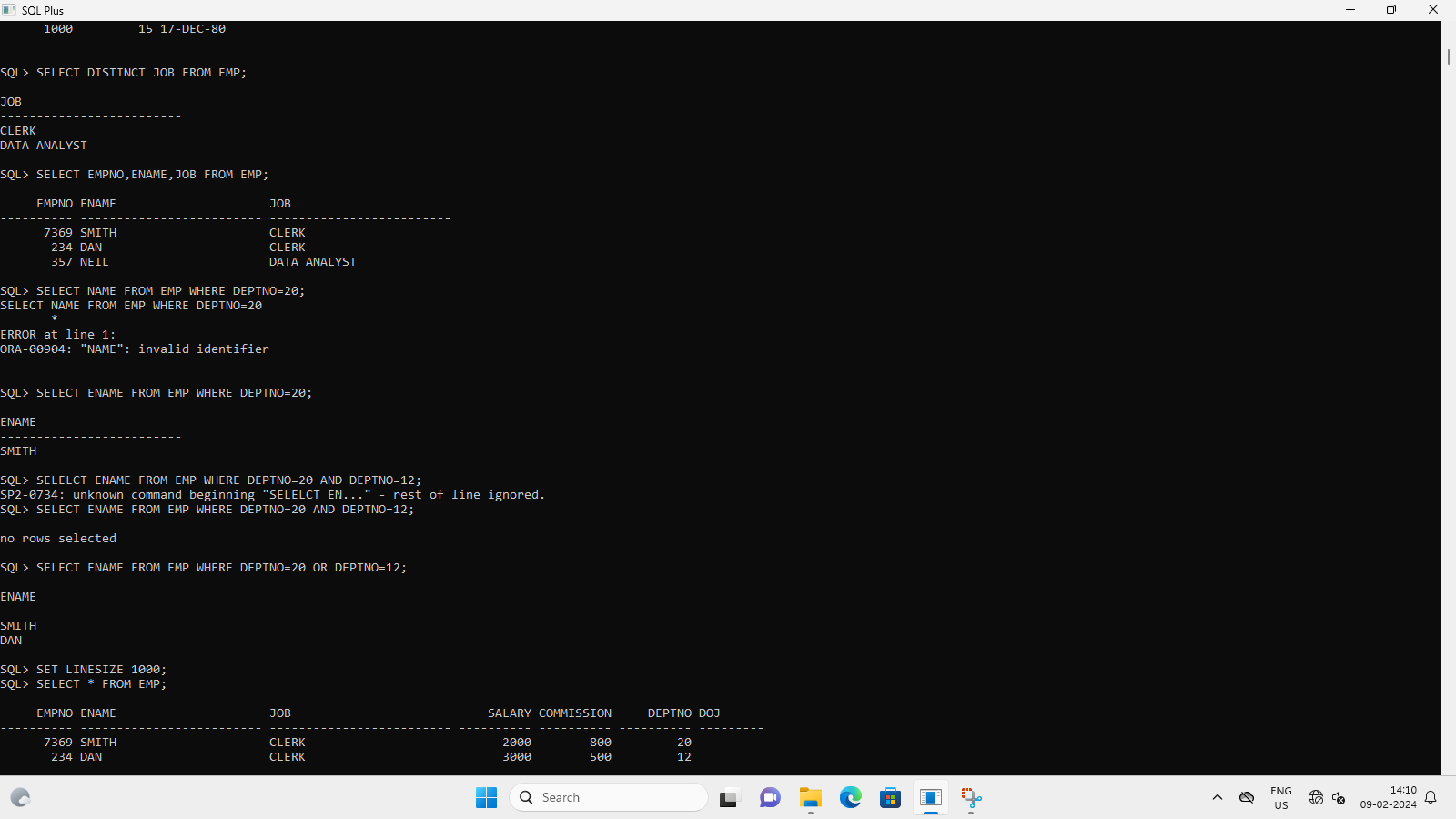
SELECT EMPNO,ENAME,JOB FROM EMP;

1. **Display name of all employees in dept 20**



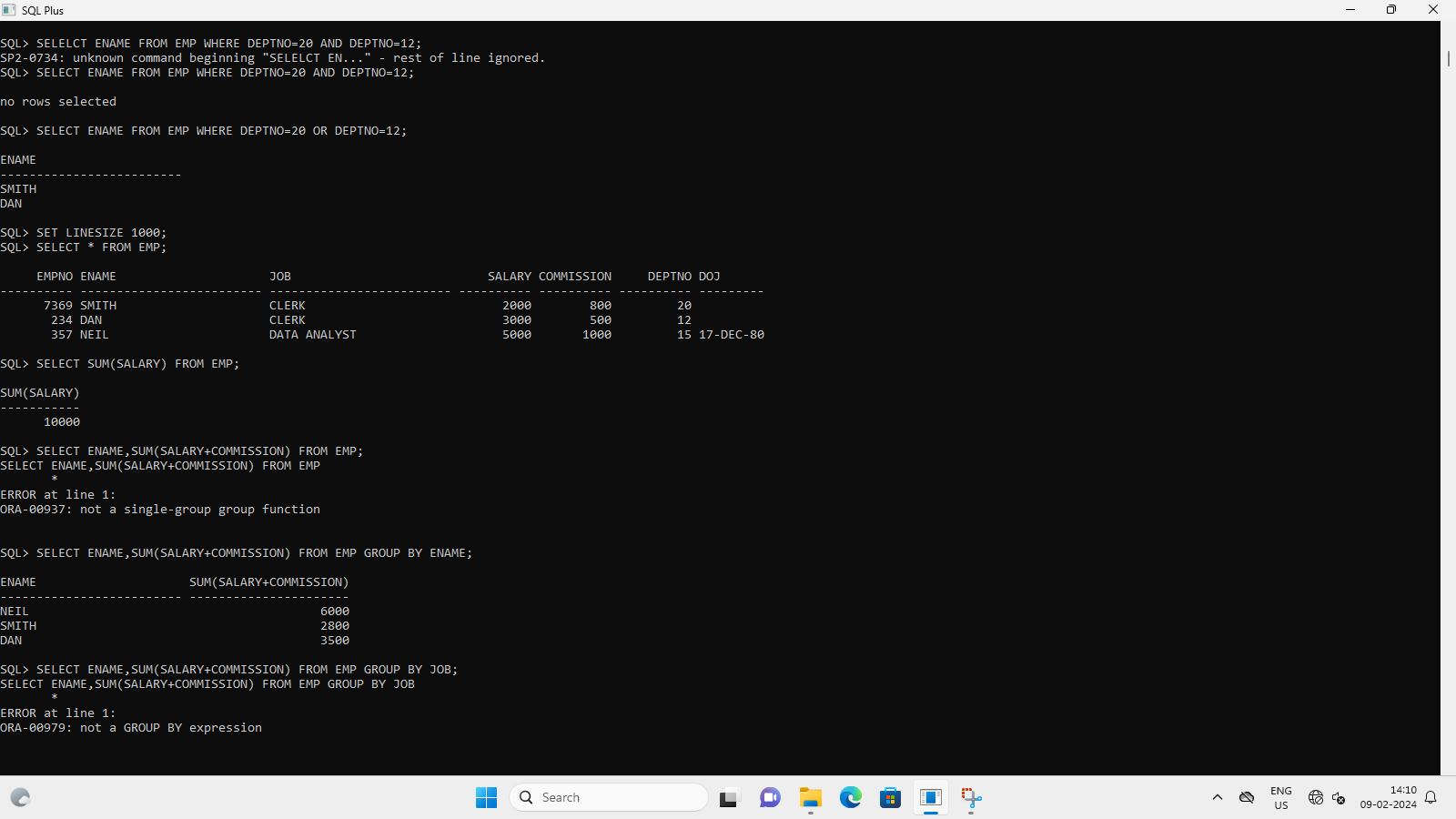
SELECT ENAME FROM EMP WHERE DEPTNO=20;

1. **Display name of all employees in dept 20 and 15**



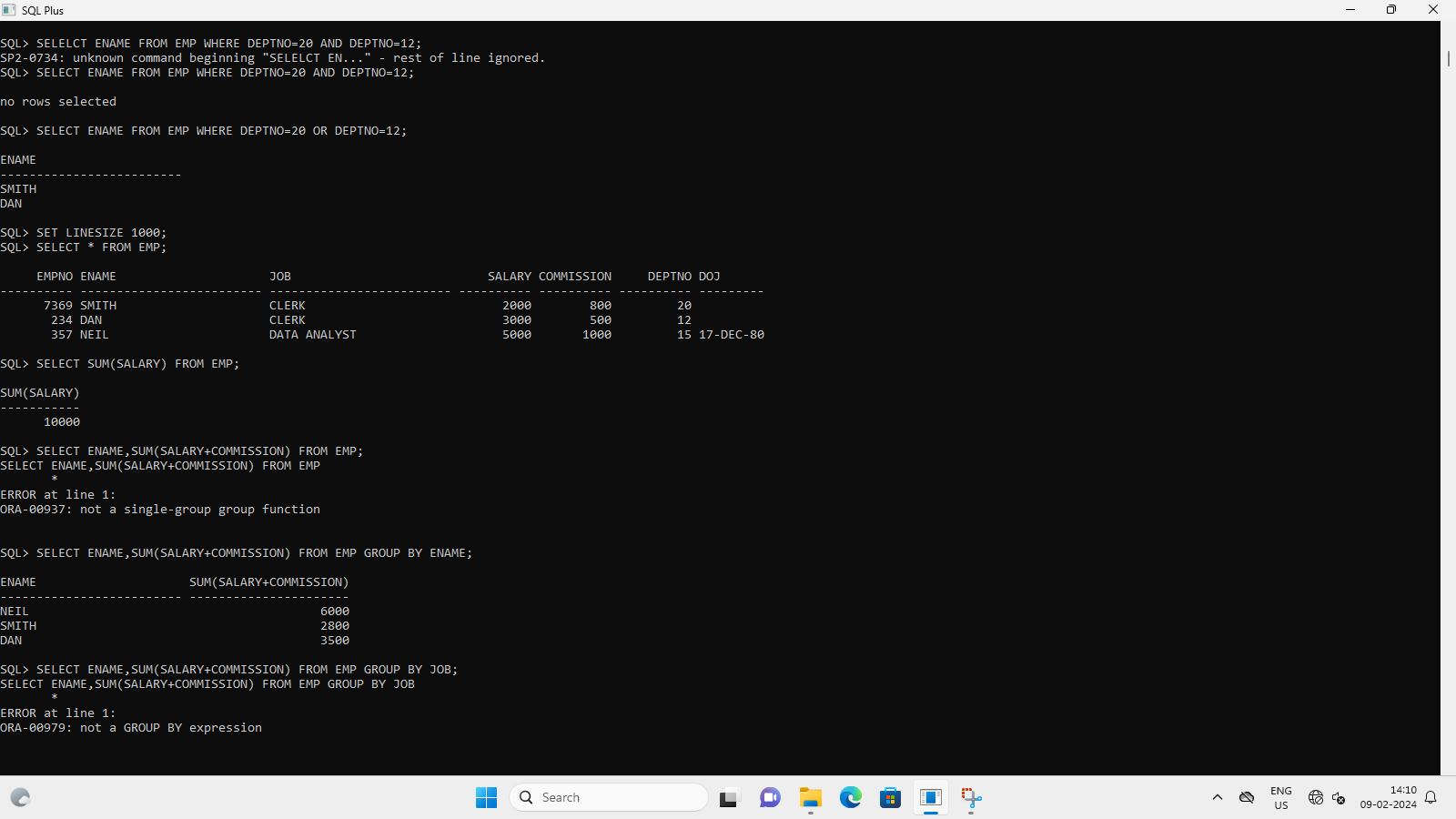
SELECT ENAME FROM EMP WHERE DEPTNO=20 OR DEPTNO=12;

1. **Find the total salary of all employees**



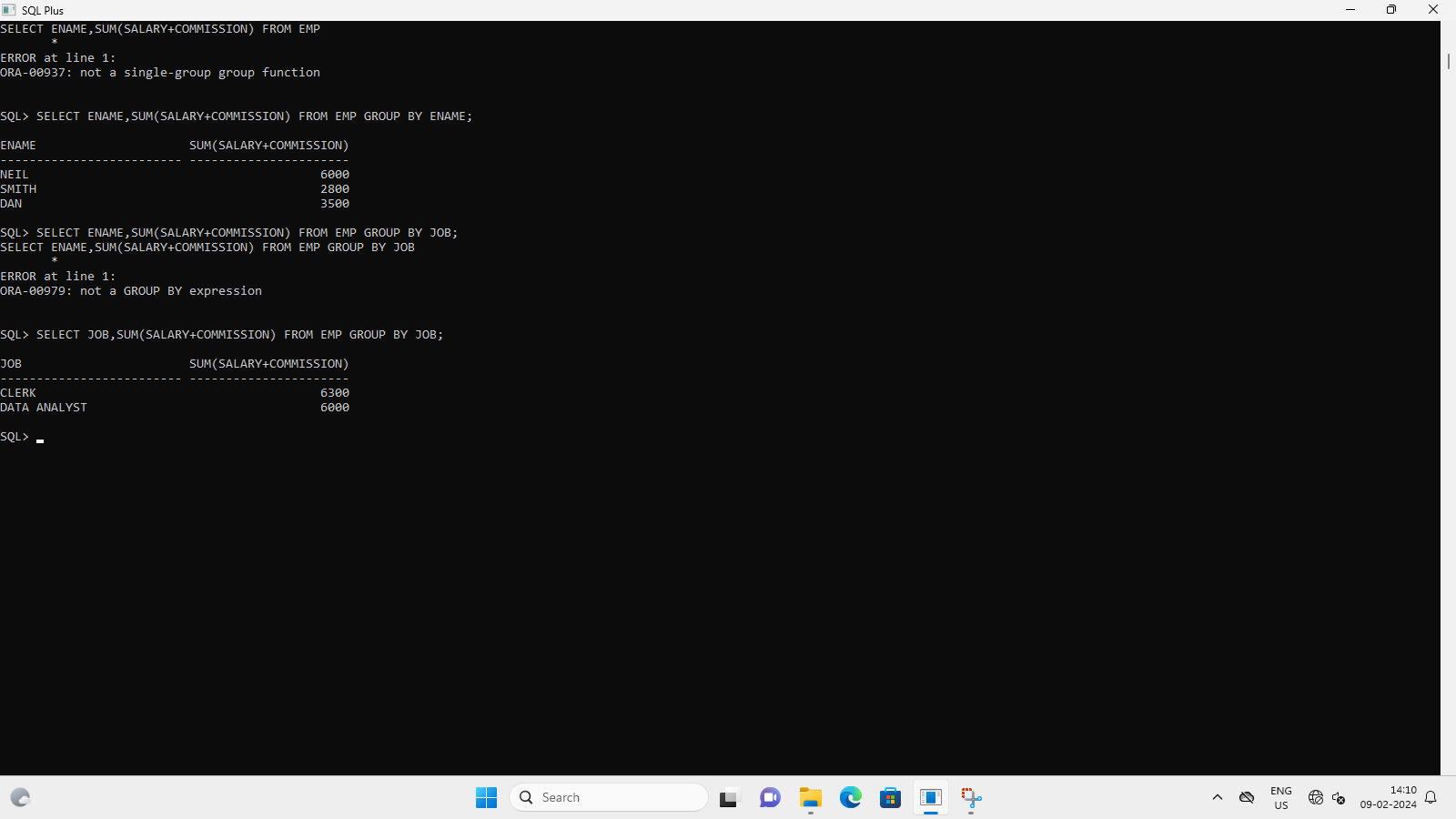
SELECT SUM(SALARY) FROM EMP;

1. **List name and total salary. ie, salary+commission**



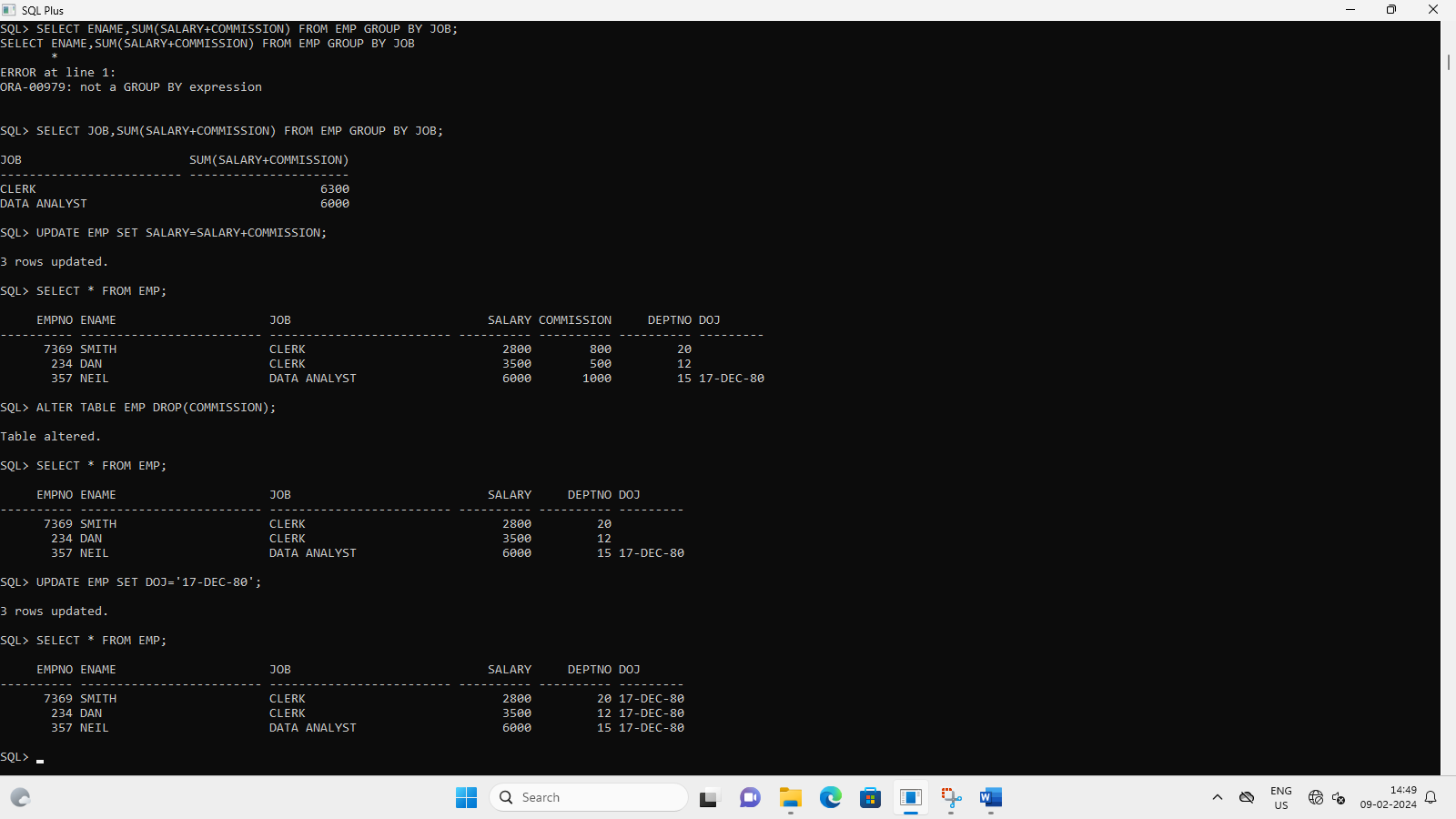
SELECT ENAME,SUM(SALARY+COMMISSION) FROM EMP GROUP BY ENAME;

1. **Group by job**



SELECT JOB,SUM(SALARY+COMMISSION) FROM EMP GROUP BY JOB;

1. **Delete column commission and update salary as salary**



UPDATE EMP SET SALARY=SALARY+COMMISSION;

ALTER TABLE EMP DROP(COMMISSION);

1. **Display the employees with the same salary**

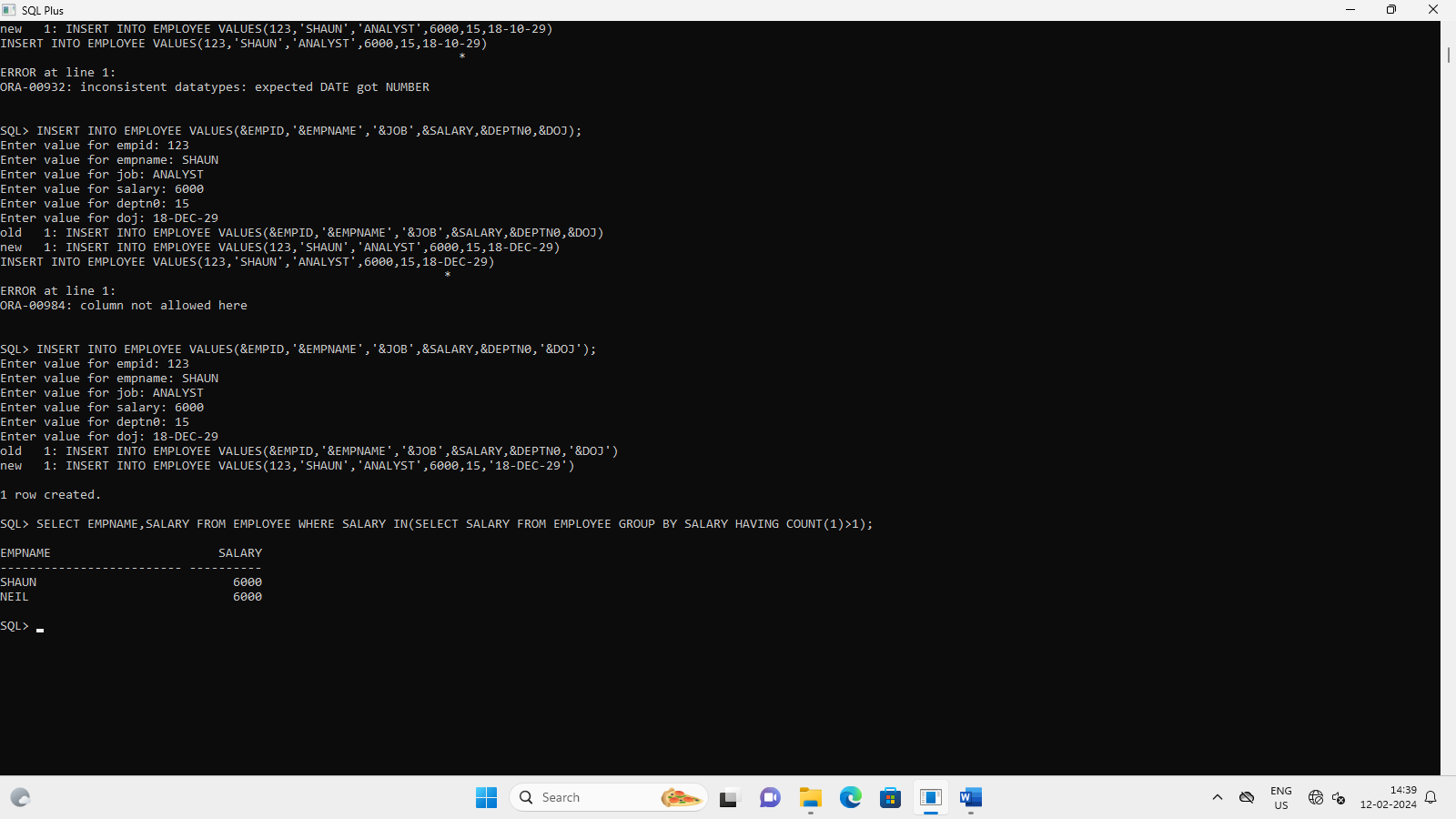
SQL> SELECT EMPNAME,SALARY FROM EMPLOYEE WHERE SALARY IN(SELECT SALARY FROM EMPLOYEE GROUP BY SALARY HAVING COUNT(1)>1);

EMPNAME SALARY

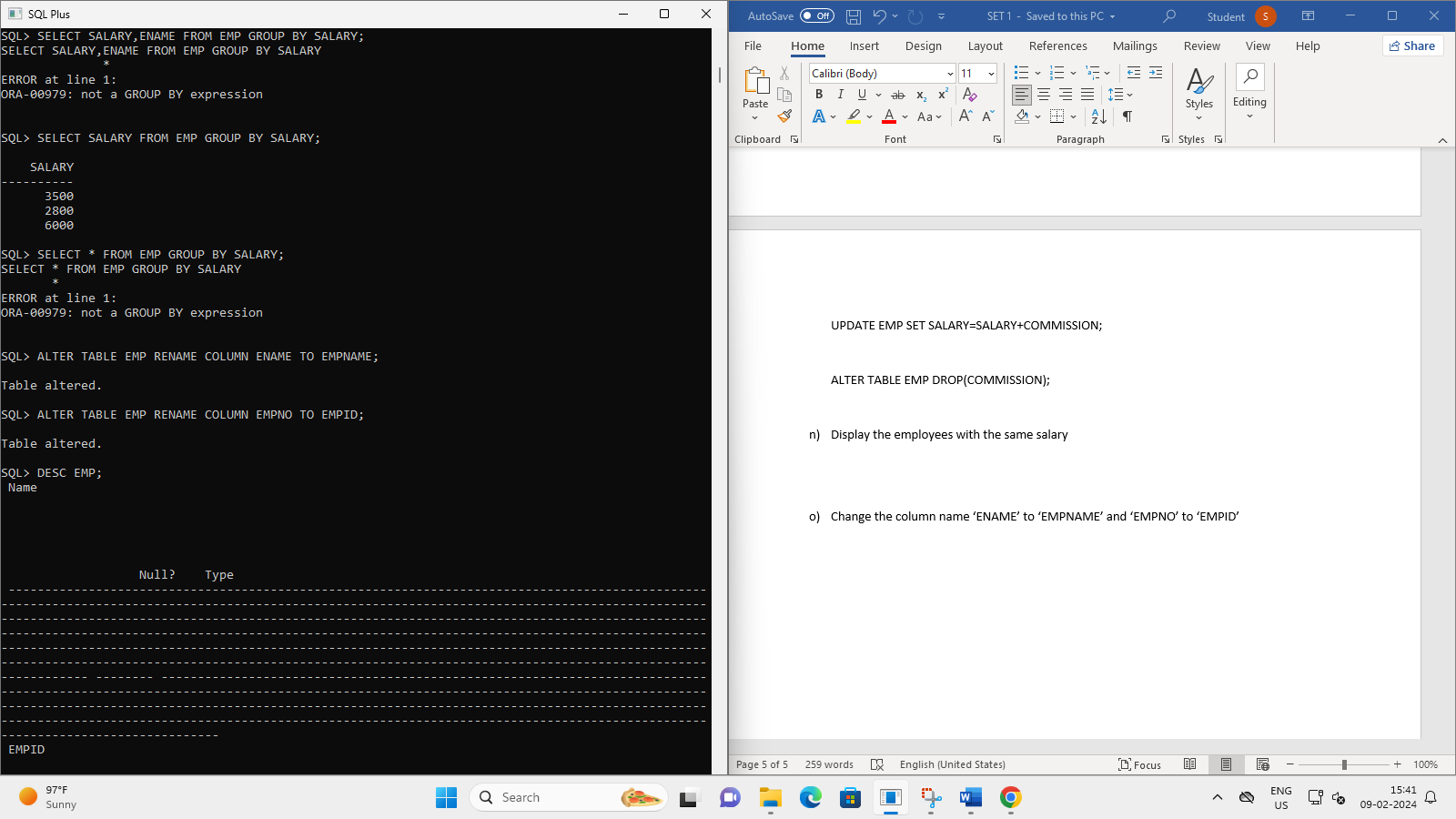
------------------------- ----------

SHAUN 6000

NEIL 6000



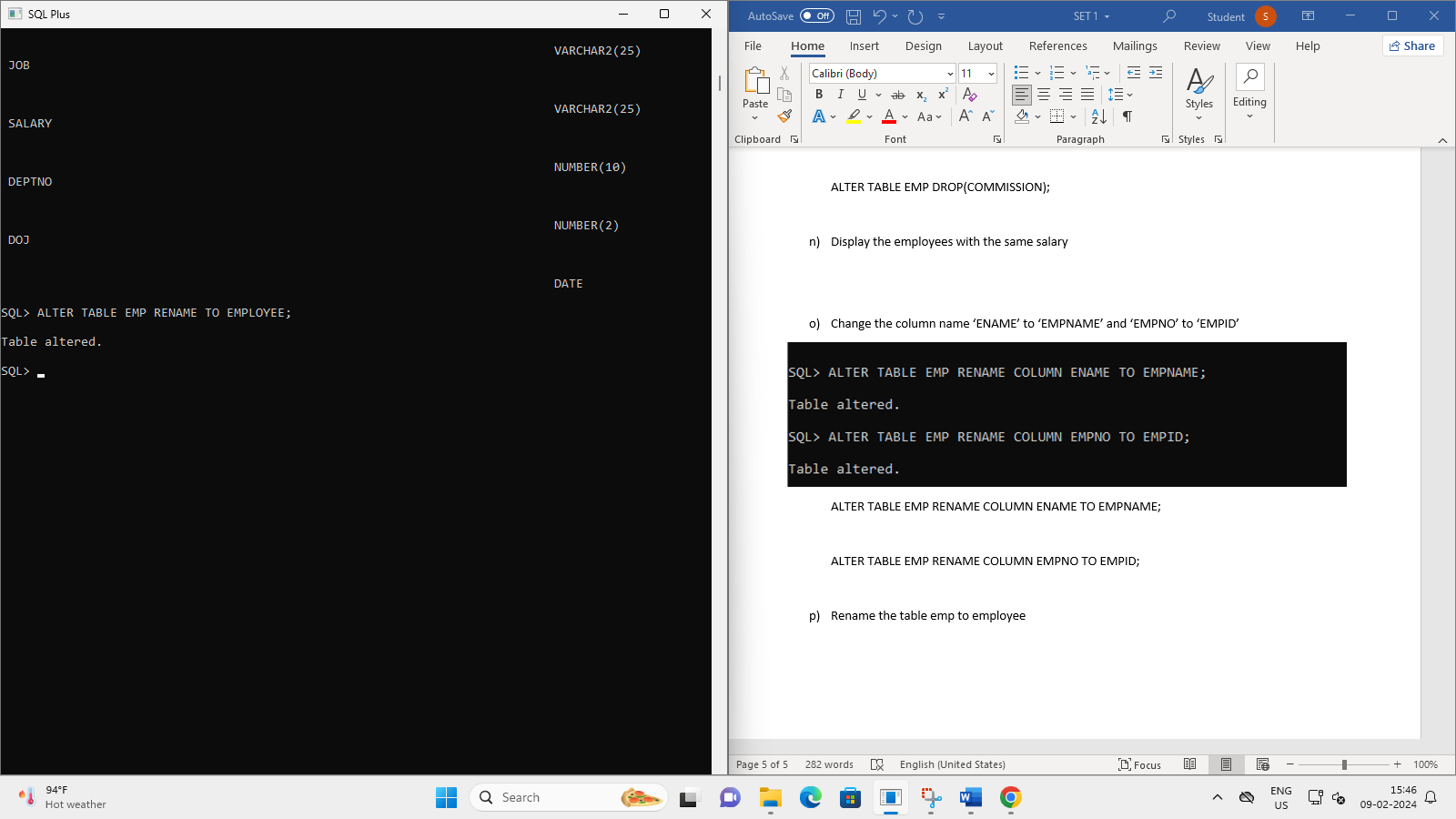
1. **Change the column name ‘ENAME’ to ‘EMPNAME’ and ‘EMPNO’ to ‘EMPID’**



ALTER TABLE EMP RENAME COLUMN ENAME TO EMPNAME;

ALTER TABLE EMP RENAME COLUMN EMPNO TO EMPID;

1. **Rename the table emp to employee**



ALTER TABLE EMP RENAME TO EMPLOYEE;

1. **Create the following tables and execute the queries given below**

**SAILORS**

|  |  |  |  |
| --- | --- | --- | --- |
| **sid** | **sname** | **rating** | **Age** |
| **22** | **Dustin** | **7** | **45** |
| **29** | **Brutas** | **1** | **33** |
| **31** | **Lubber** | **8** | **55** |
| **32** | **Andy** | **8** | **25** |
| **58** | **Rusty** | **10** | **35** |
| **64** | **Horatio** | **7** | **35** |
| **71** | **Zobra** | **10** | **16** |
| **74** | **Horatio** | **9** | **35** |
| **85** | **Art** | **3** | **26** |
| **95** | **Bob** | **3** | **64** |

**BOATS**

|  |  |  |
| --- | --- | --- |
| **bid** | **bname** | **color** |
| **101** | **Interlake** | **Blue** |
| **102** | **Interlake** | **Red** |
| **103** | **Clipper** | **Green** |
| **104** | **Marine** | **Red** |

**RESERVES**

|  |  |  |
| --- | --- | --- |
| **sid** | **bid** | **day** |
| **22** | **101** | **10/10/98** |
| **22** | **102** | **10/10/98** |
| **22** | **103** | **10/08/98** |
| **22** | **104** | **10/07/98** |
| **31** | **102** | **11/10/98** |
| **31** | **103** | **11/06/98** |
| **31** | **104** | **11/12/98** |
| **64** | **101** | **09/05/98** |
| **64** | **102** | **09/08/98** |
| **74** | **103** | **09/08/98** |

CREATE TABLE SAILORS(SID NUMBER(10) PRIMARY KEY,SNAME VARCHAR2(20), RATING NUMBER(10), AGE NUMBER(10));

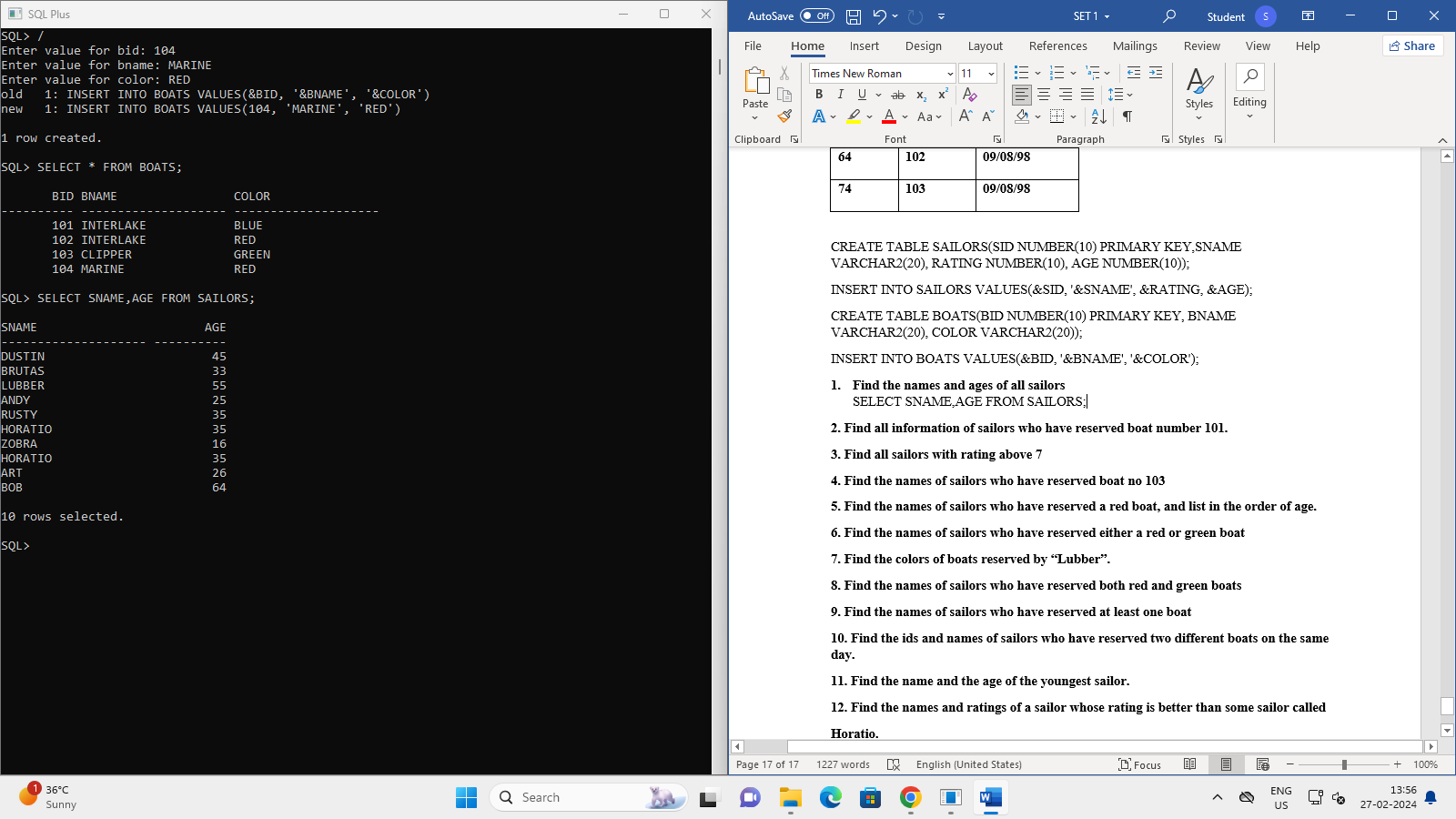
INSERT INTO SAILORS VALUES(&SID, '&SNAME', &RATING, &AGE);

CREATE TABLE BOATS(BID NUMBER(10) PRIMARY KEY, BNAME VARCHAR2(20), COLOR VARCHAR2(20));

INSERT INTO BOATS VALUES(&BID, '&BNAME', '&COLOR');

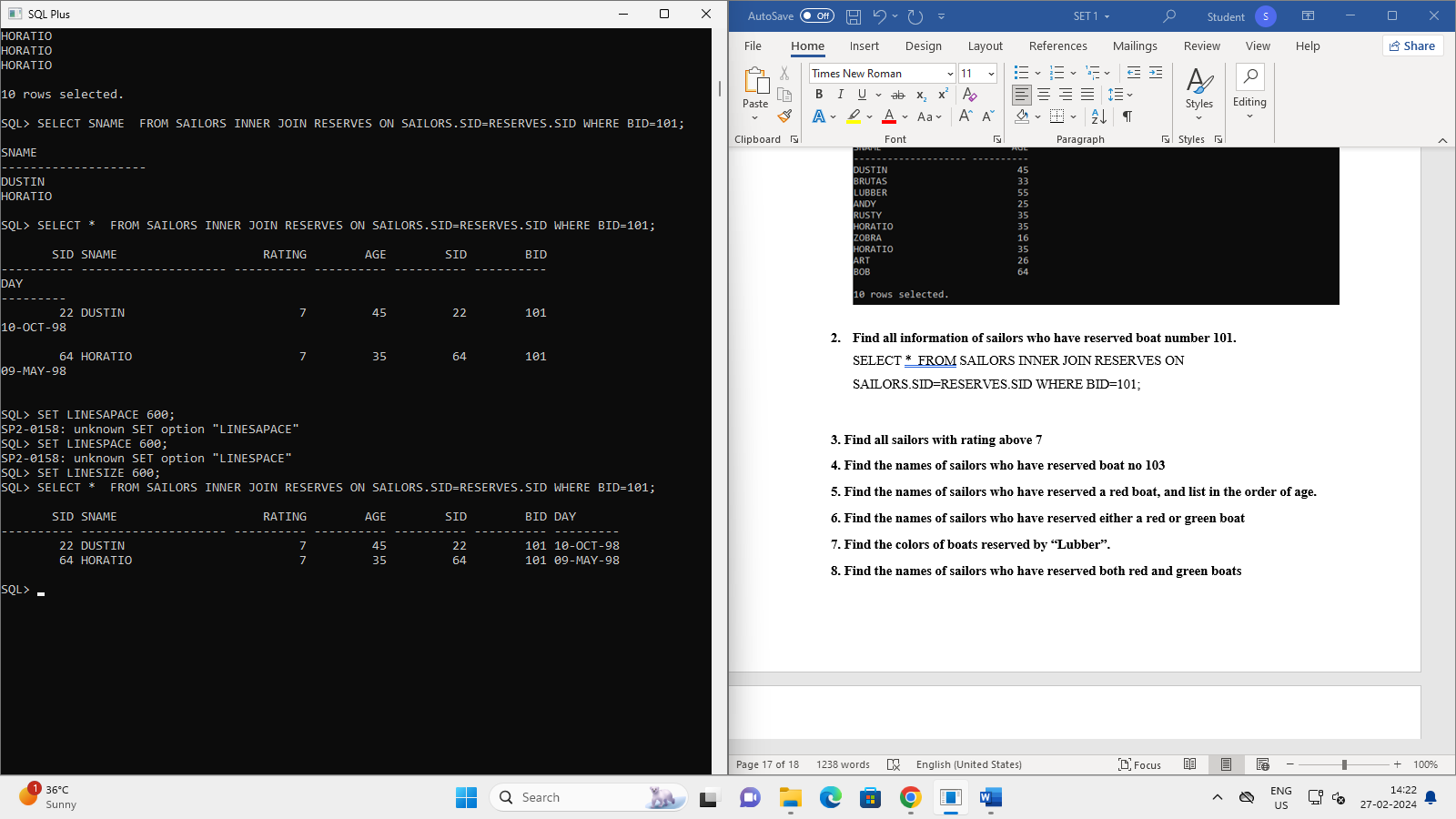
1. **Find the names and ages of all sailors**

SELECT SNAME,AGE FROM SAILORS;



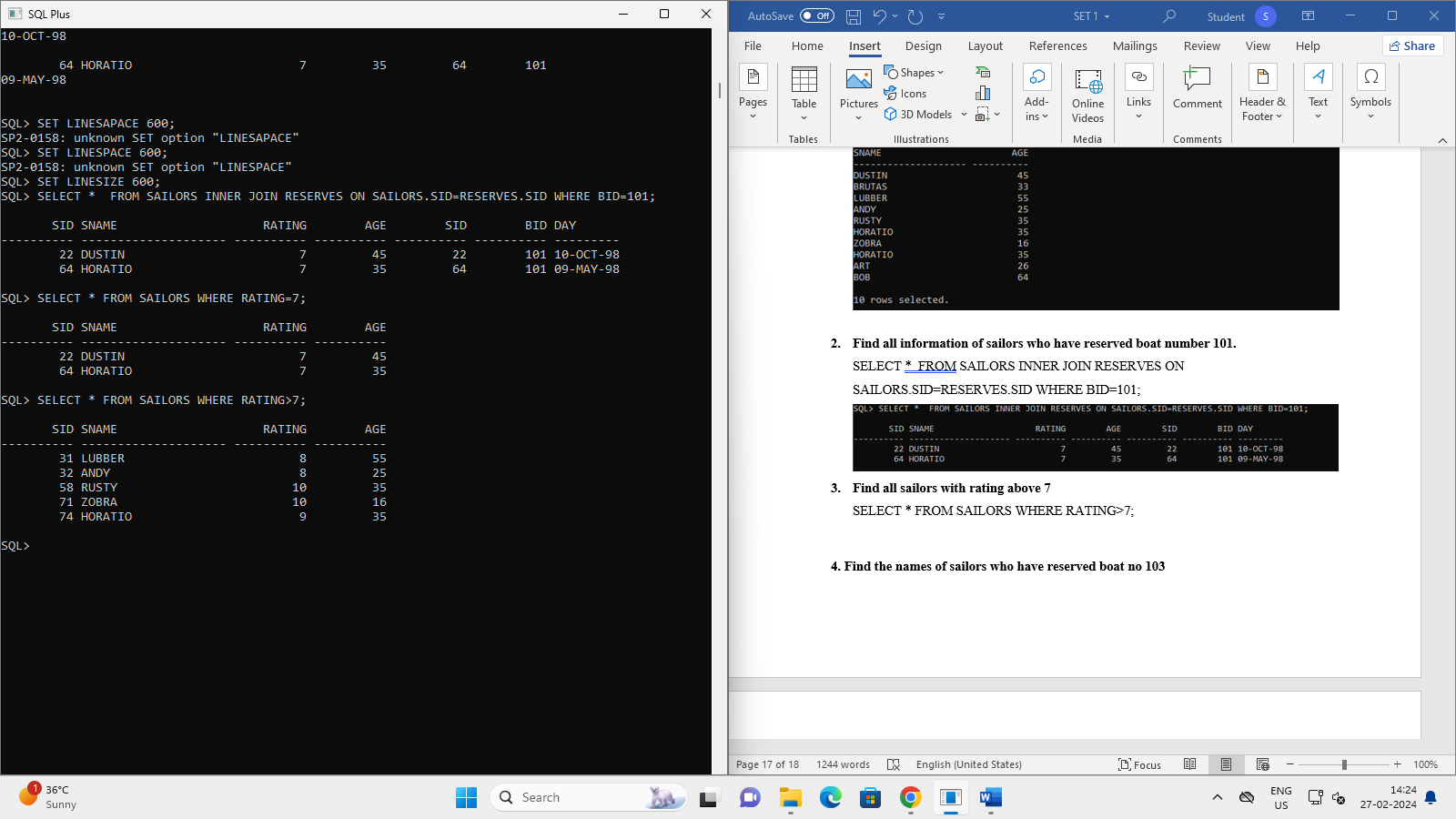
1. **Find all information of sailors who have reserved boat number 101.**

SELECT \* FROM SAILORS INNER JOIN RESERVES ON SAILORS.SID=RESERVES.SID WHERE BID=101;



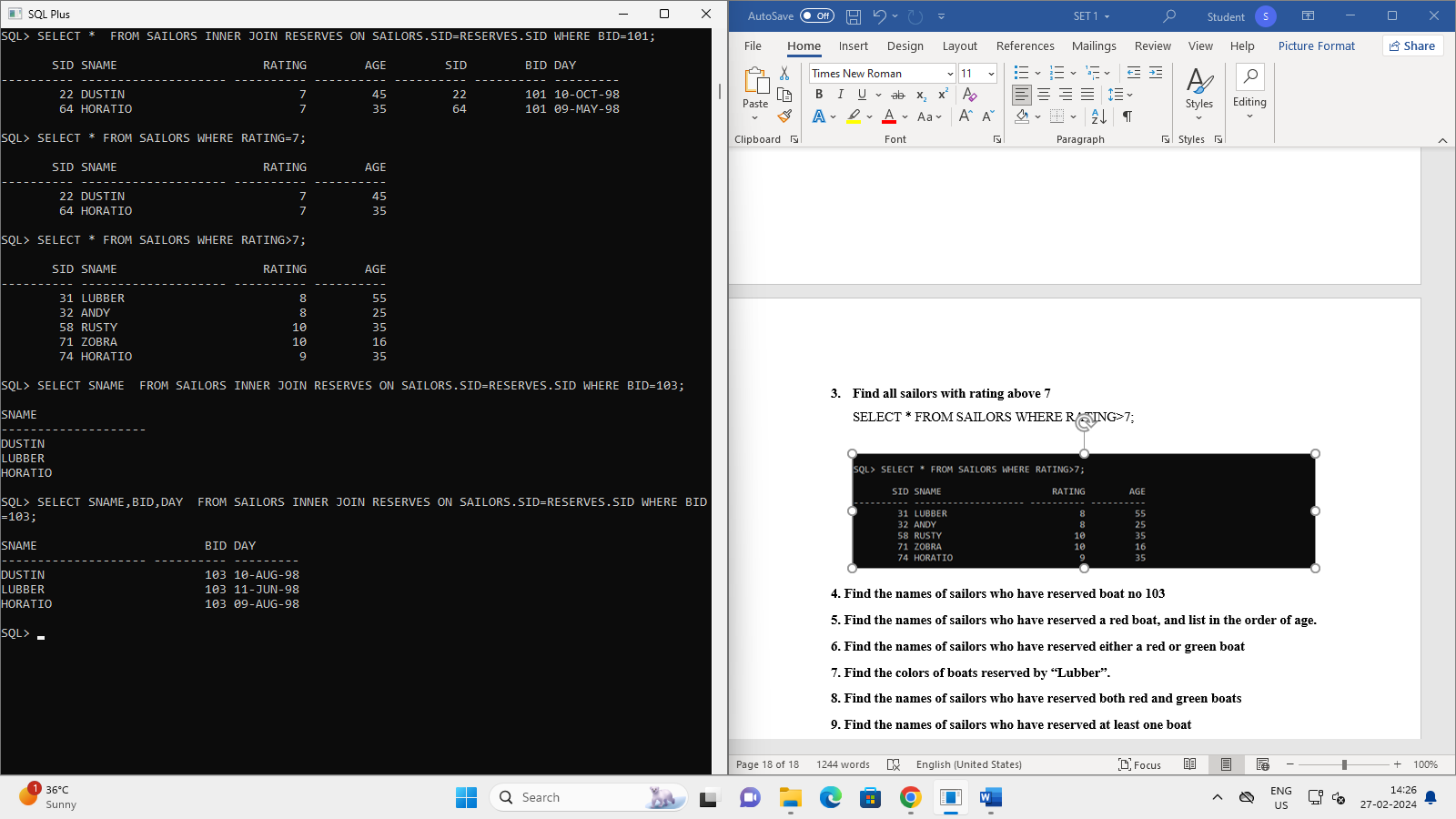
1. **Find all sailors with rating above 7**

SELECT \* FROM SAILORS WHERE RATING>7;



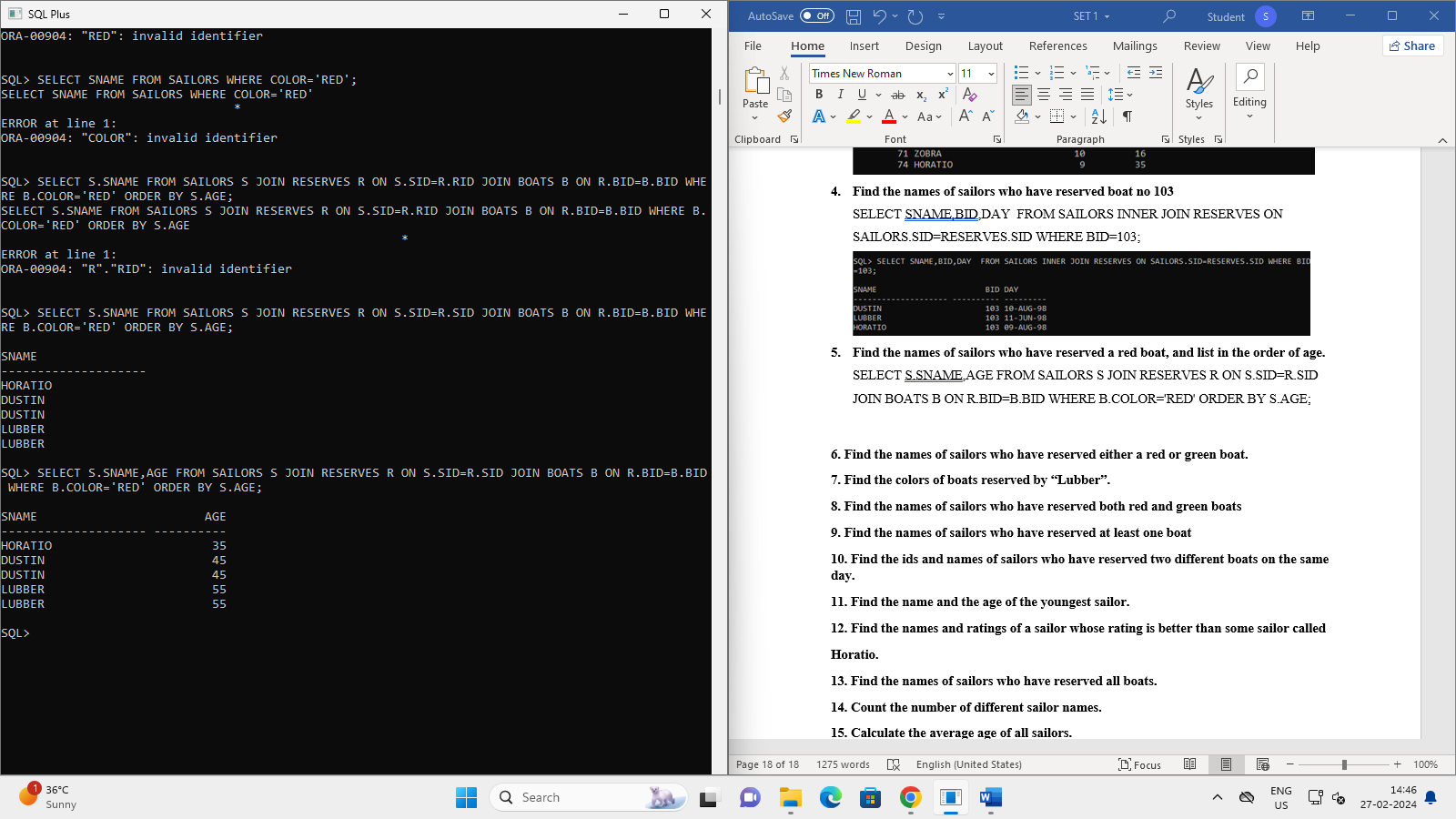
1. **Find the names of sailors who have reserved boat no 103**

SELECT SNAME,BID,DAY FROM SAILORS INNER JOIN RESERVES ON SAILORS.SID=RESERVES.SID WHERE BID=103;



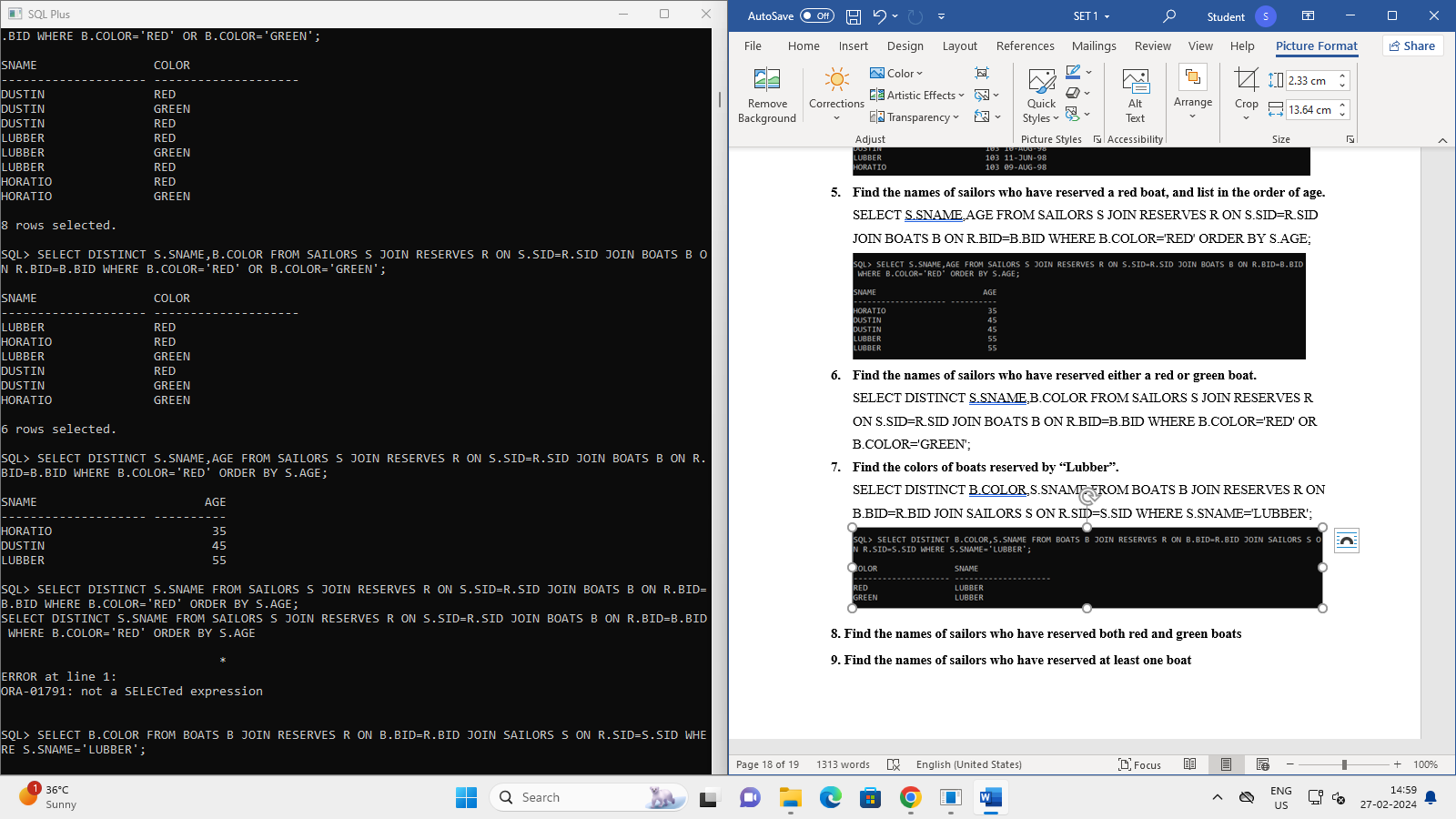
1. **Find the names of sailors who have reserved a red boat, and list in the order of age.**

SELECT S.SNAME,AGE FROM SAILORS S JOIN RESERVES R ON S.SID=R.SID JOIN BOATS B ON R.BID=B.BID WHERE B.COLOR='RED' ORDER BY S.AGE;



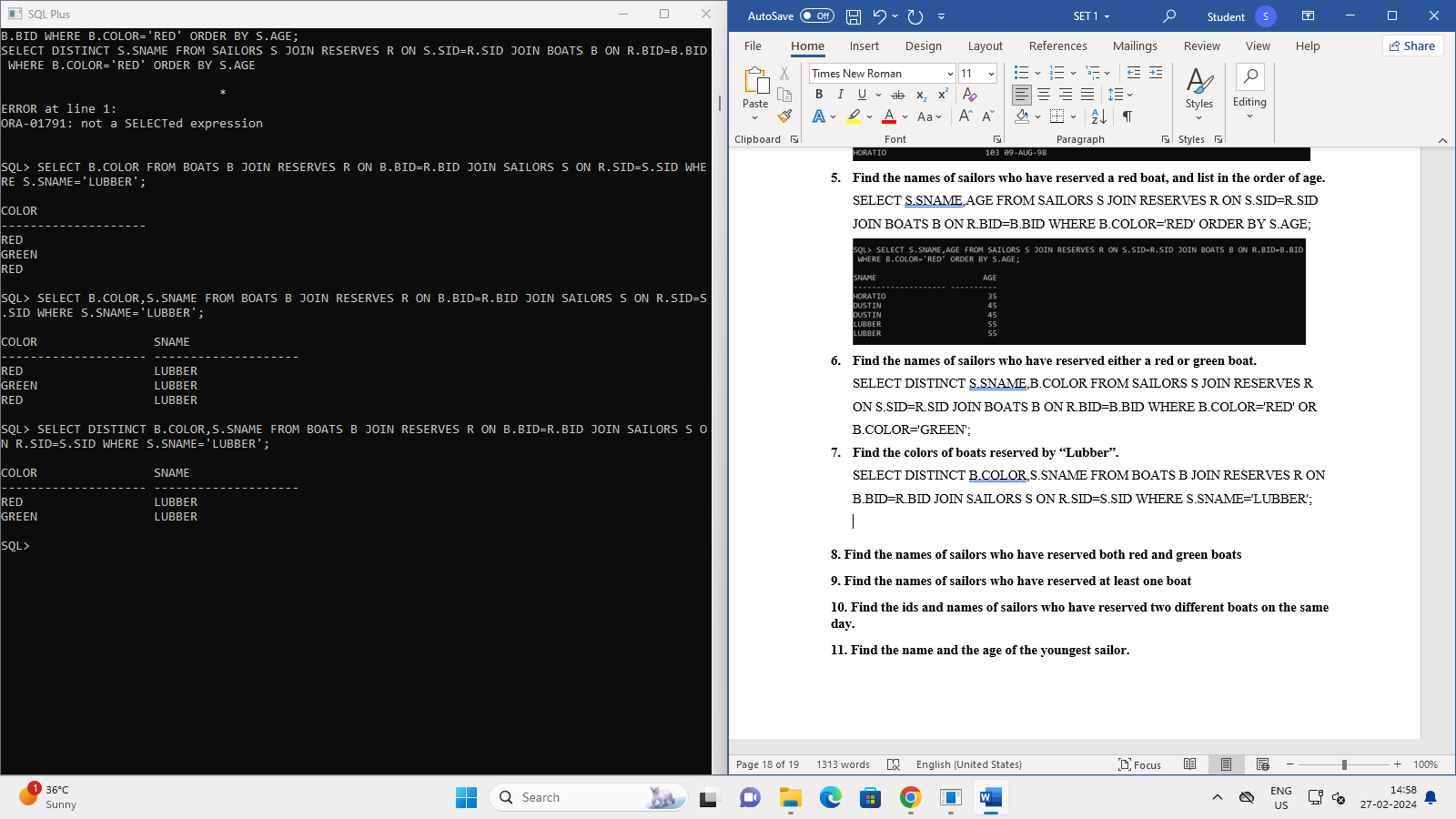
1. **Find the names of sailors who have reserved either a red or green boat.**

SELECT DISTINCT S.SNAME,B.COLOR FROM SAILORS S JOIN RESERVES R ON S.SID=R.SID JOIN BOATS B ON R.BID=B.BID WHERE B.COLOR='RED' OR B.COLOR='GREEN';



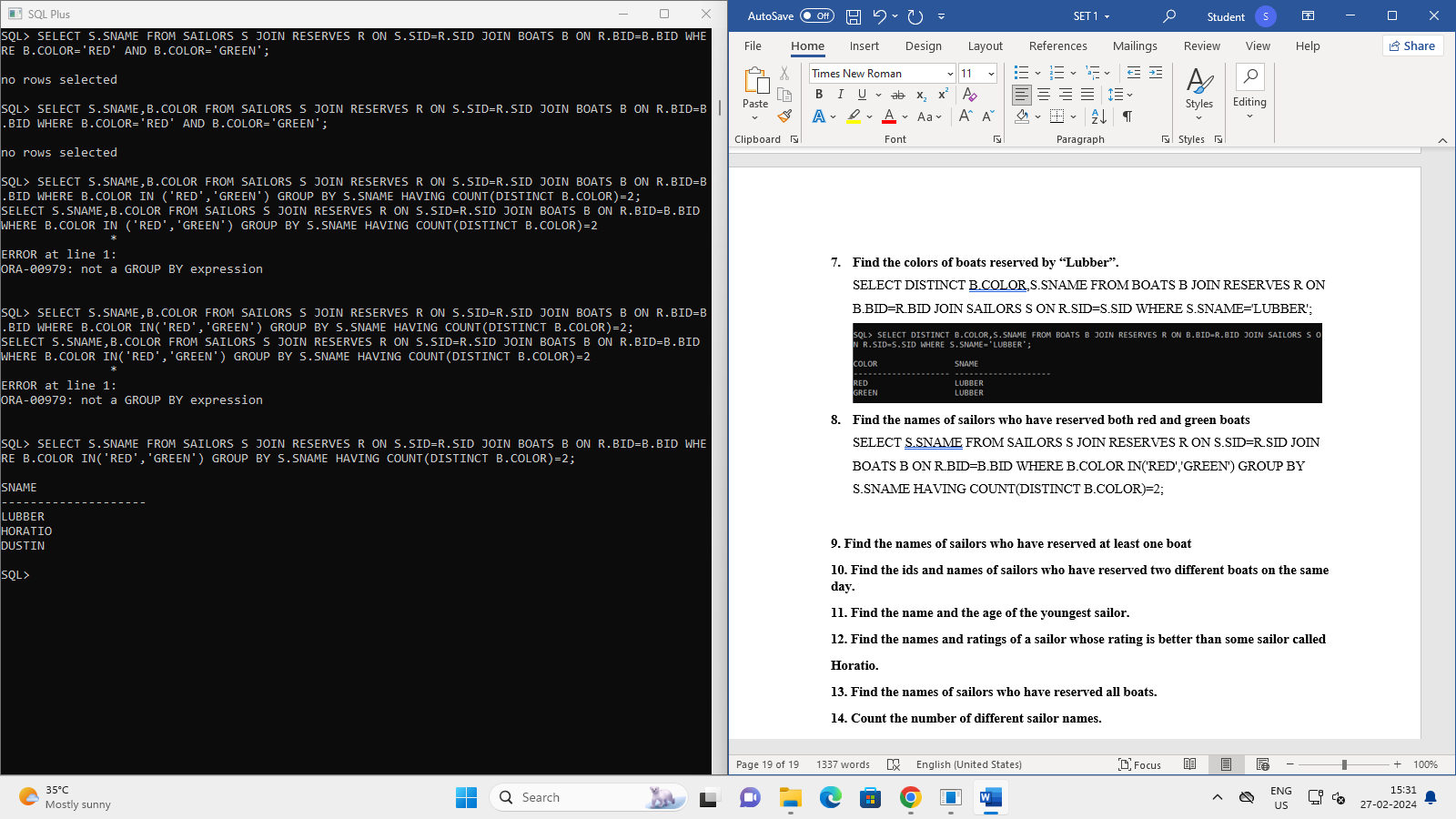
1. **Find the colors of boats reserved by “Lubber”.**

SELECT DISTINCT B.COLOR,S.SNAME FROM BOATS B JOIN RESERVES R ON B.BID=R.BID JOIN SAILORS S ON R.SID=S.SID WHERE S.SNAME='LUBBER';



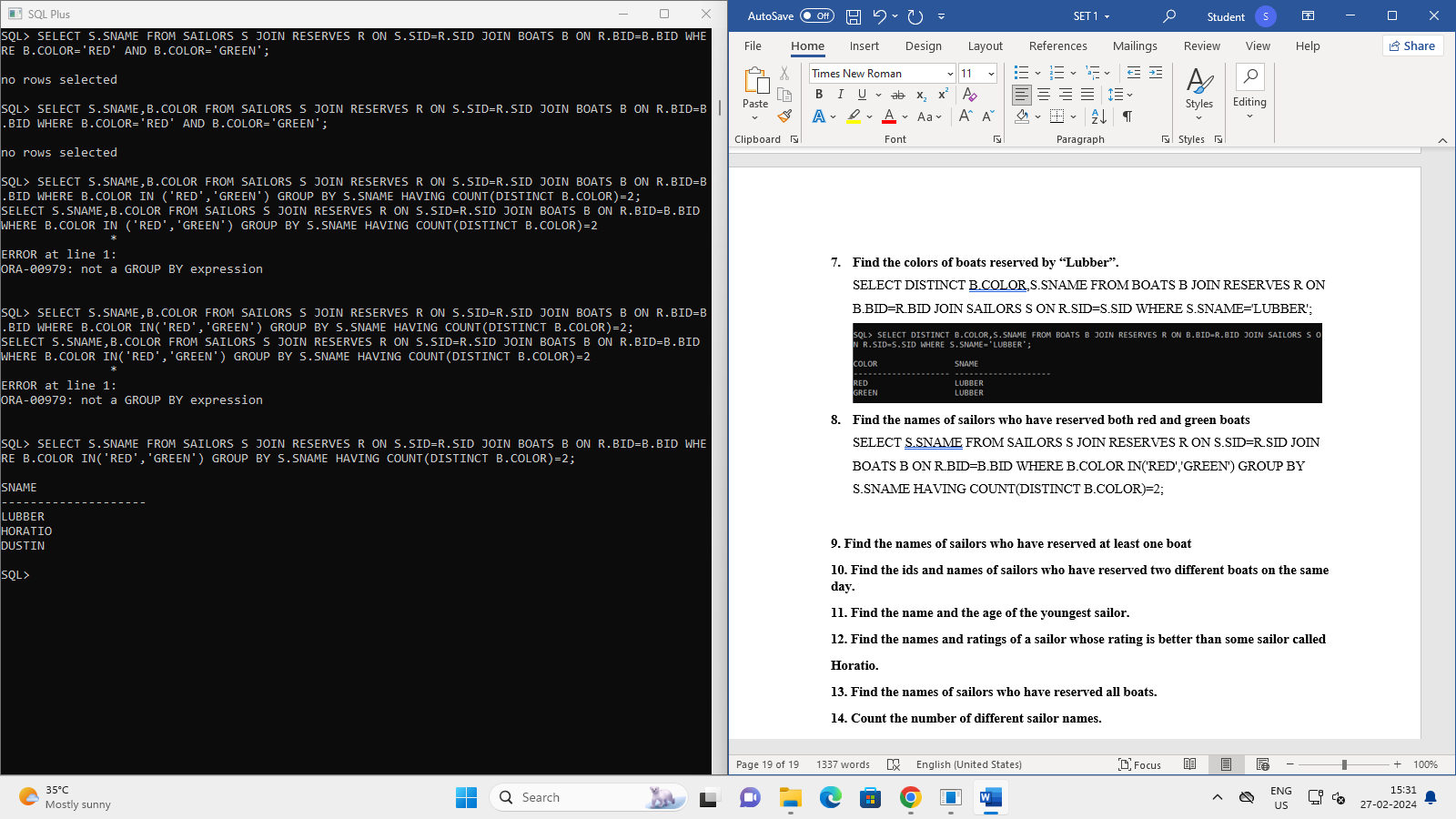
1. **Find the names of sailors who have reserved both red and green boats**

SELECT S.SNAME FROM SAILORS S JOIN RESERVES R ON S.SID=R.SID JOIN BOATS B ON R.BID=B.BID WHERE B.COLOR IN('RED','GREEN') GROUP BY S.SNAME HAVING COUNT(DISTINCT B.COLOR)=2;



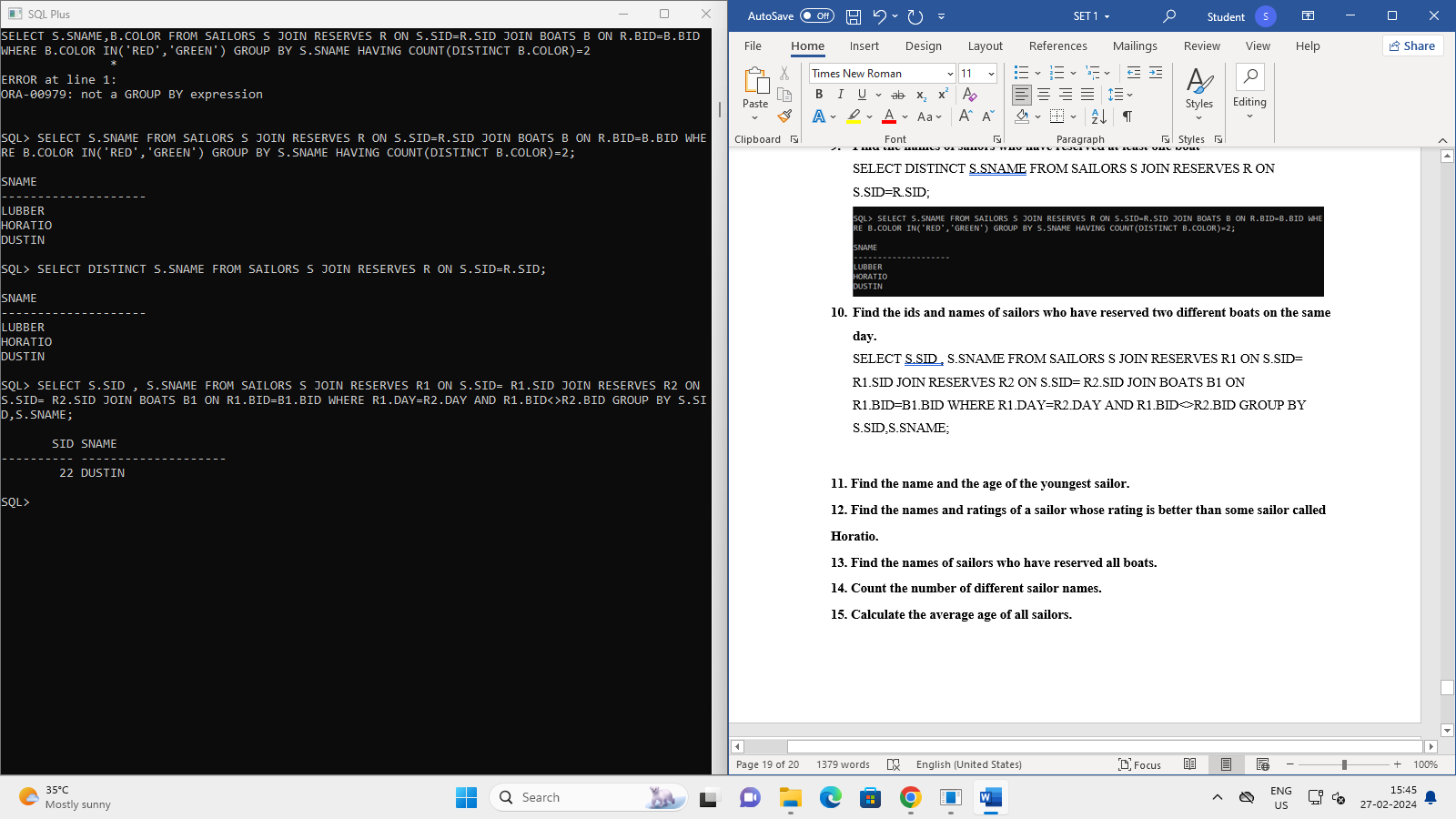
1. **Find the names of sailors who have reserved at least one boat**

SELECT DISTINCT S.SNAME FROM SAILORS S JOIN RESERVES R ON S.SID=R.SID;



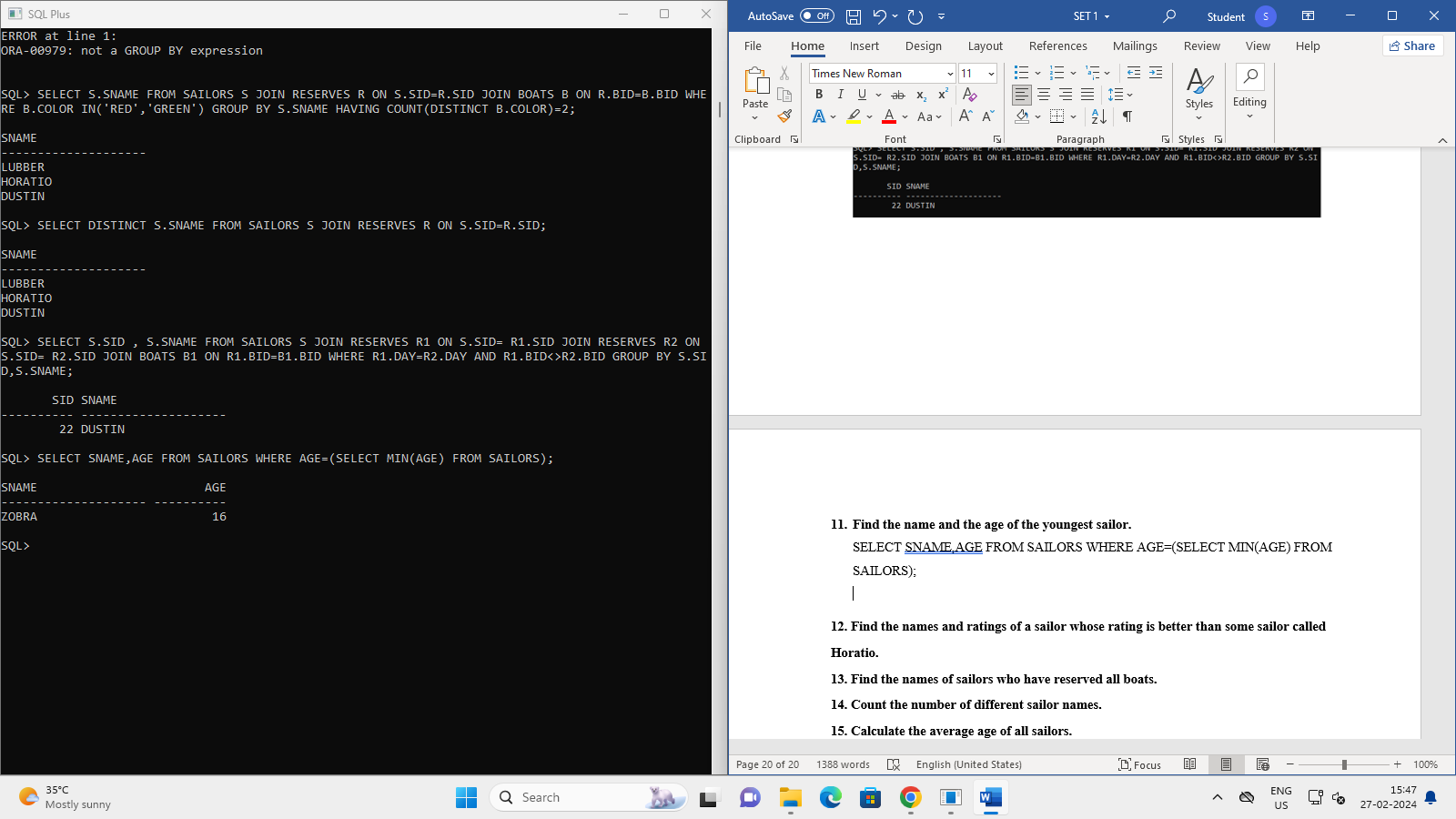
1. **Find the ids and names of sailors who have reserved two different boats on the same day.**

SELECT S.SID , S.SNAME FROM SAILORS S JOIN RESERVES R1 ON S.SID= R1.SID JOIN RESERVES R2 ON S.SID= R2.SID JOIN BOATS B1 ON R1.BID=B1.BID WHERE R1.DAY=R2.DAY AND R1.BID<>R2.BID GROUP BY S.SID,S.SNAME;



1. **Find the name and the age of the youngest sailor.**

SELECT SNAME,AGE FROM SAILORS WHERE AGE=(SELECT MIN(AGE) FROM SAILORS);

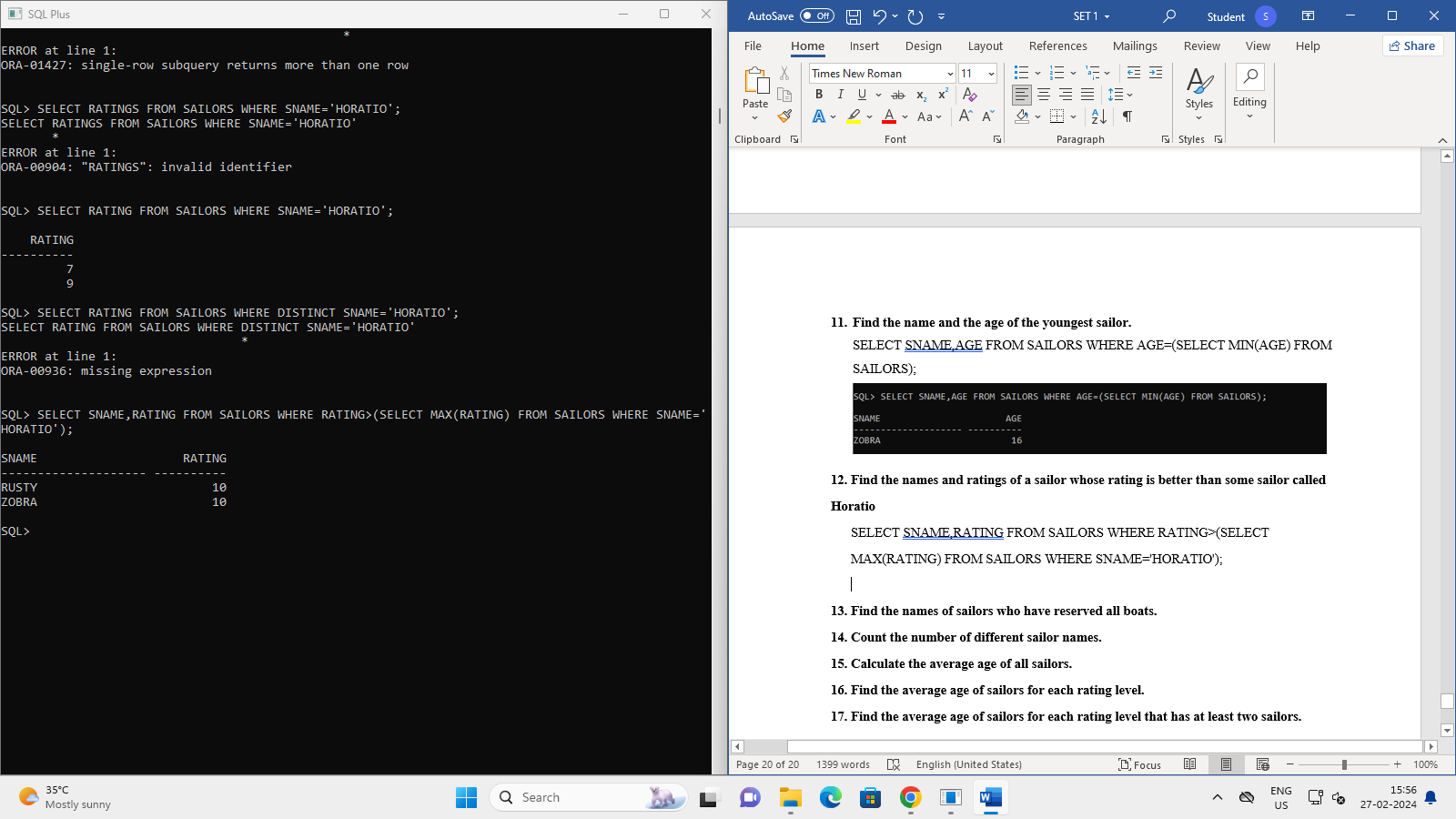


**12. Find the names and ratings of a sailor whose rating is better than some sailor called**

**Horatio**

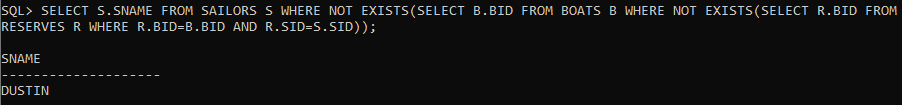
SELECT SNAME,RATING FROM SAILORS WHERE RATING>(SELECT

MAX(RATING) FROM SAILORS WHERE SNAME='HORATIO');



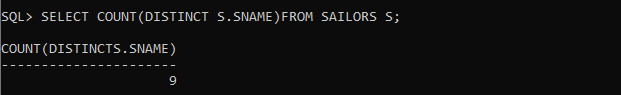
1. **Find the names of sailors who have reserved all boats.**

SELECT S.SNAME FROM SAILORS S WHERE NOT EXISTS(SELECT B.BID FROM BOATS B WHERE NOT EXISTS(SELECT R.BID FROM RESERVES R WHERE R.BID=B.BID AND R.SID=S.SID));



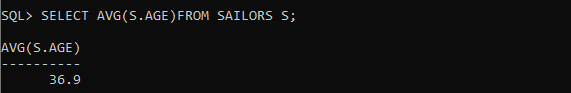
1. **Count the number of different sailor names.**

SELECT COUNT(DISTINCT S.SNAME)FROM SAILORS S;



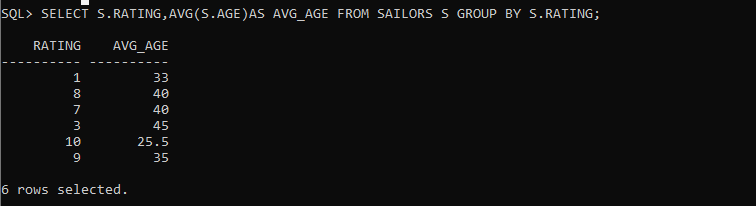
1. **Calculate the average age of all sailors.**

SELECT AVG(S.AGE)FROM SAILORS S;

****

1. **Find the average age of sailors for each rating level.**

SELECT S.RATING,AVG(S.AGE)AS AVG\_AGE FROM SAILORS S GROUP BY S.RATING;



1. **Find the average age of sailors for each rating level that has at least two sailors.**

SELECT S.RATING,AVG(S.AGE)AS AVG\_AGE FROM SAILORS S GROUP BY S.RATING HAVING COUNT(\*)>1;

