JAVA AWT BASED-Iris Based Age Classification system - SQL CONNECTIVITY USING JDBC

A
Report
Submitted in partial fulfilment of the
Requirements for the award of the Degree of

BACHELOR OF ENGINEERING

IN

INFORMATION TECHNOLOGY

By

Sheela Sai Rohith <1602-18-737-098>

Department of Information Technology

Vasavi College of Engineering (Autonomous)

Ibrahimbagh, Hyderabad-31

2020

BONAFIDE CERTIFICATE

This to Certify that the project report titled **"Online MOOC's year wise student database management system"** project work of **Mr.S.Sai Rohith** bearing Roll.no:**1602-18-737-098** who carried out this project under my supervision in the IV semester for the academic year 2019-2020.

Signature
Signature

external examiner ***internal examiner***

B.LEELAVATHY
Assistant Professor
Department of Information Technology

DBMS   ASSIGNMENT -2
Title: Iris based age classification system

IRIS BASED AGE CLASSIFICATION SYSTEM

Name : S.Sai Rohith   Roll Number : 1602-18-737-098

Abstract:

The average human eyeball is about one inch (25.4mm)in diameter, and weighs 0.25ounces (7.09gram). This is just slightly smaller than a regulation ping pong ball. The human iris ranges from 10.2mm to 13.0mm on average. by this diameter and range of vision we can find the age of the person if this values are collected from a healthy uninjured person.if we consider ratio of range of vision and retina diameter then it will decrease with the increase in the age."Age measurement is very difficult," says Dr Thomas Huang, the lead developer. "If you use the face to estimate age we can really get the apparent age, or how old a person looks."The researchers trained their computer algorithm using 1,600 different people with five pictures of each person, for a total of 8,000 images. The age of the people in the pictures ranged from one year to 93 years old.so by their research they have given ranges of retina_ratio and age of people with that ratio. with the help of that ratio this project is based on. with the help of that segeration we are going to estimate the age of people using this software.

INTRODUCTION

➢ REQUIREMENTS FOR IRIS BASED AGE CLASSIFICATION SYSTEM:

REQUIREMENT ANALYSIS

List of tables :

- COMPANY
- RETINA_SCANNER
- RETINA_SCANNER_PREPAREDBY
- SCAN
- PERSON
- REPORT

List of attributes with their domain types:

Company:

Company_id : number()

Name : varchar()

Address : varchar()

Rating : number()

Retina_scanner:

Scanner_id : number

Name : varchar()

Cost : number

Accuracy : number

Prepared_by:

Scanner_id : number

Company_id : number

Day : date

Person:

Person_id : number

Name : varchar()

Phone_no : number(10)

Address : varchar()

Scan:

Person_id : number

Scanner_id : number

Report_id : number

Report :

Report_id : number

Person_id : number

Age : number

Colour : varchar()

Retina_ratio : number

➢ SPECIFIC GOAL OF THE PROJECT:

The main goal to be achieved through this project was to provide a facility to classify people by their age without knowing their age from them. This is done with the help of their retina size. People who uses this application can konw their age with the help of their retina size.

The project also ensure that the details of the people are confidential and are stored in the database.

➢ Architecture and technology used:

SQL Plus is the most basic Oracle Database utility with a basic command-line interface, commonly used by users, administrators and programmers.

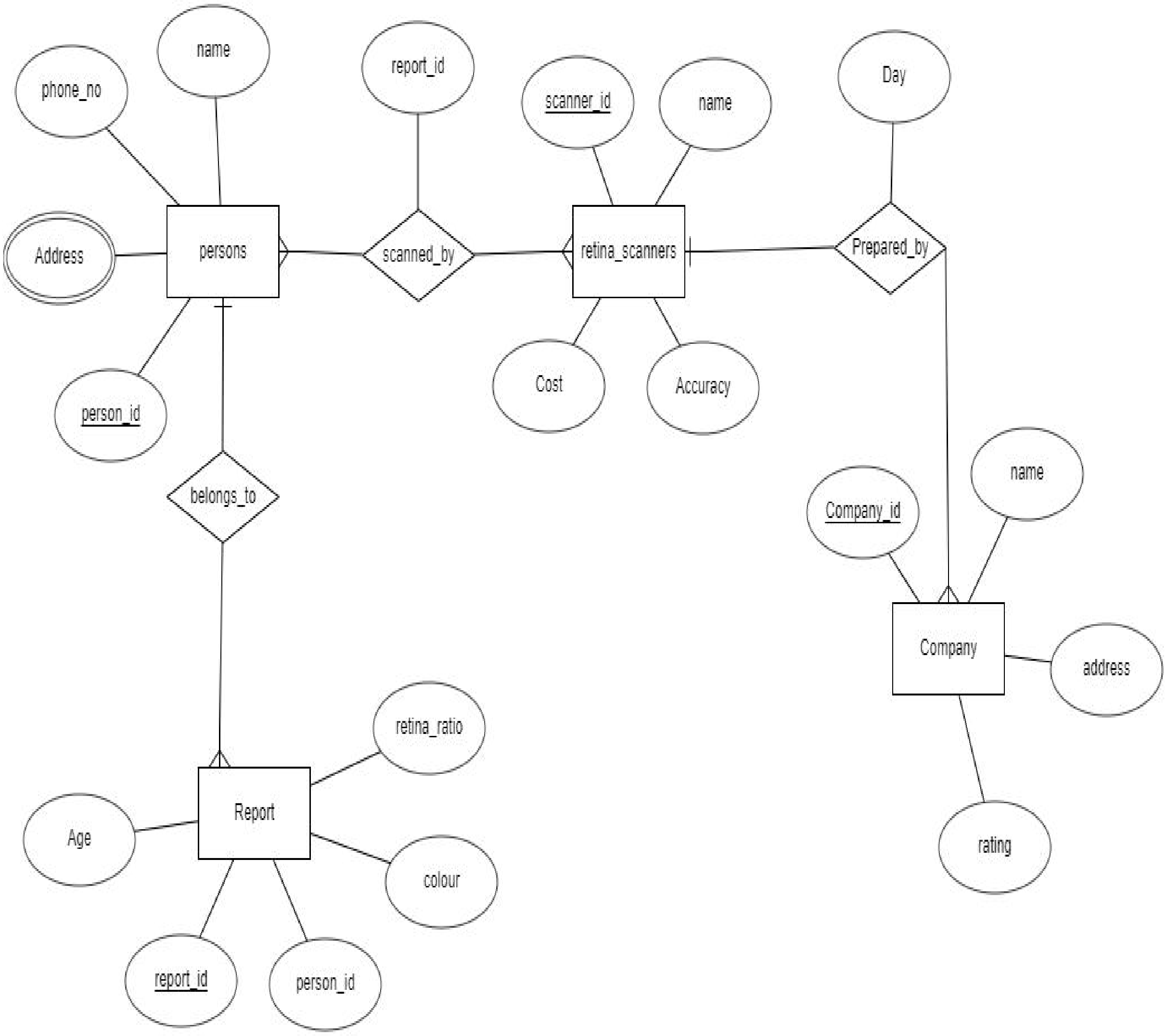
The interface of SQL Plus is used for creating the database. DDL and DML commands are implemented for operations being executed. The details of various Online MOOC's provider, courses, student, assignments, and results are stored in the form of tables in the database.

Eclipse is an integrated development environment(IDE) used in computer programming. It contains a base workspace and an extensible plug-in system for customizing the environment. Eclipse is written mostly in java and its primary use is for developing Java applications, but it may also be used to develop applications in other programming languages via plug-ins, including Erlang, JavaScripts etc.

The front end application code is written in "**Java**" using Eclipse. The portal for front end application is designed through Eclipse, runs and has the capacity to connect with the database which has data inserted using SQL.

➢ DESIGN:

i)<u>ER DIAGRAM</u>:



MAPPING CARDINALITIES AND PARTICIPATION CONSTRAINTS:

A company can prepare many retina scanners with different costs and different accuracy. But each retina scanner can be prepared by only one specific company only.

A person can be scan his eye any number of times with different scanners to know his age and generate many number of reports I.e., different report for each scan.

But a each report is belongs to only one person. That is person may repeat but report id should not repeat again.

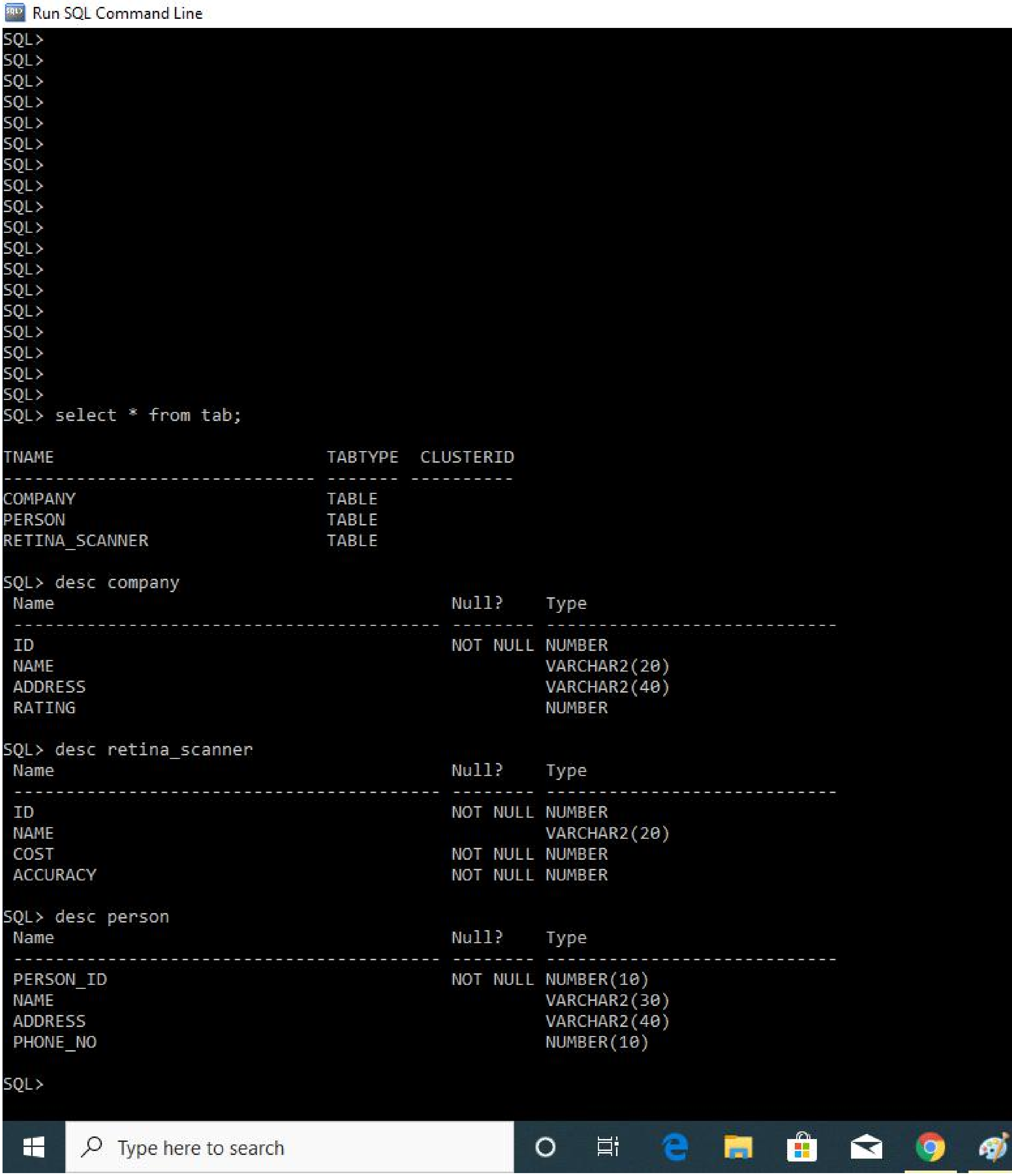
Address attribute in person table is multi variable attribute and all id's in each table are primary keys.

DDL Commands:

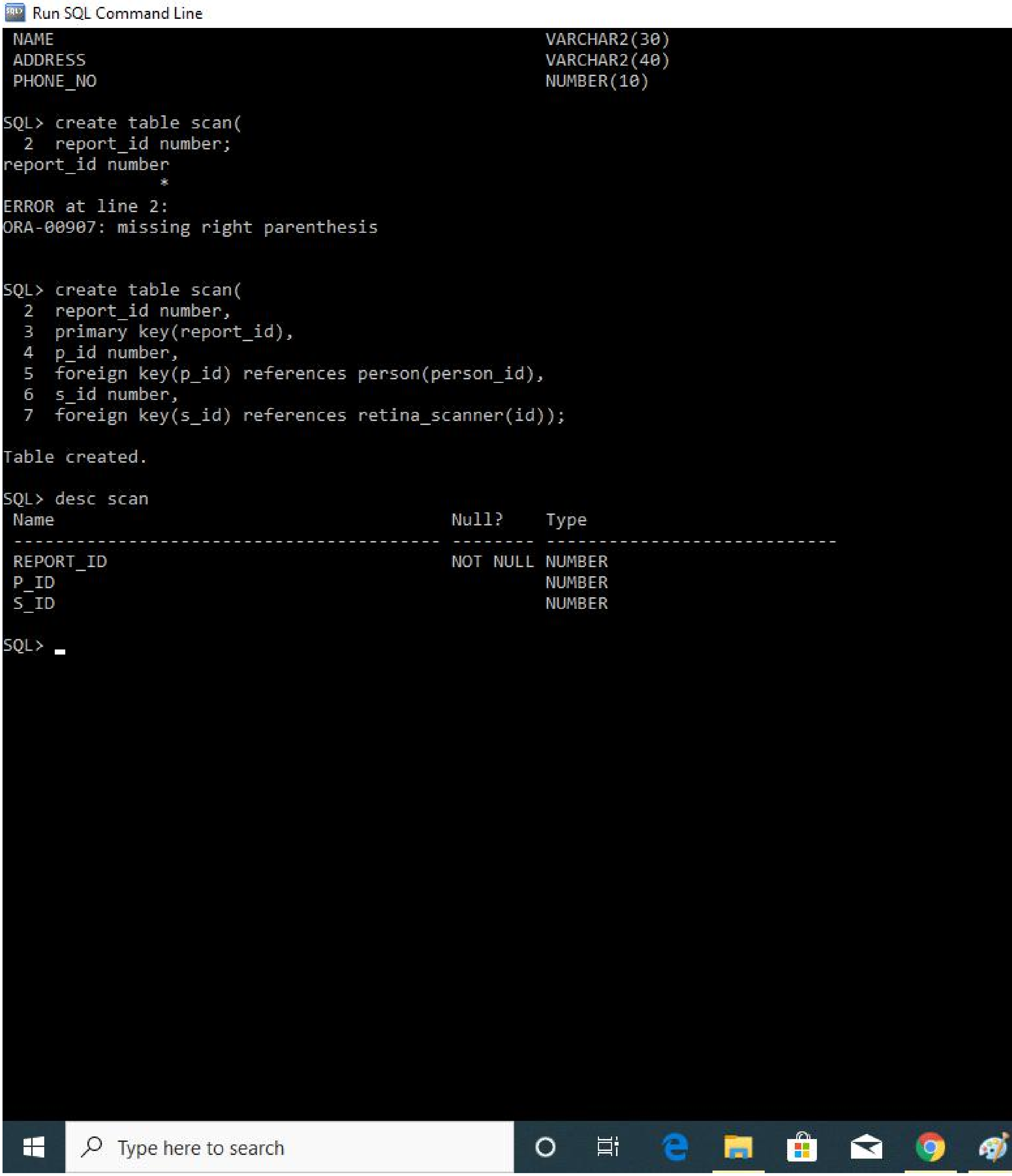
i) Creating all the required tables.

ii) Enforcing constraints to primary, forein key constraints.

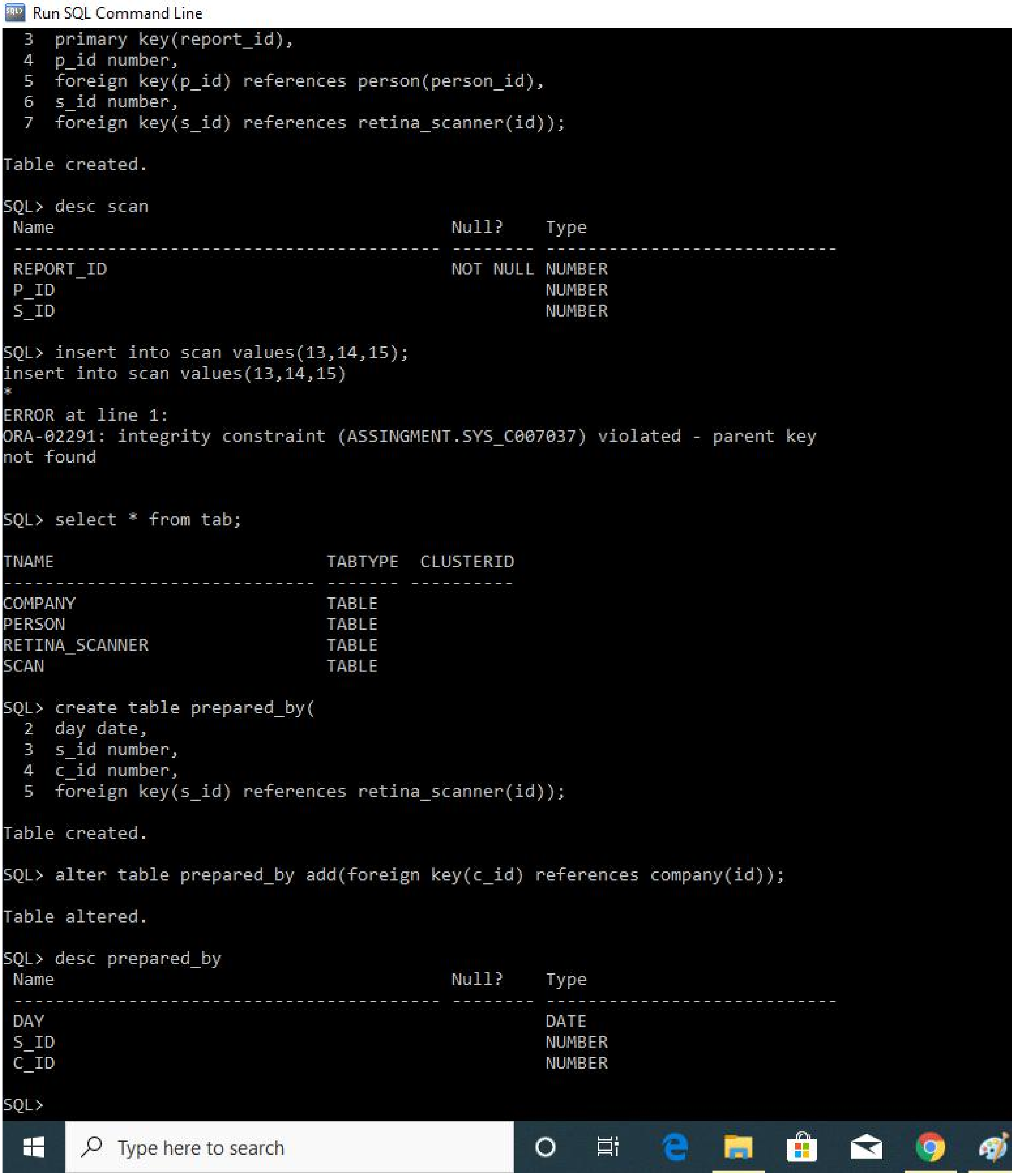
iii)



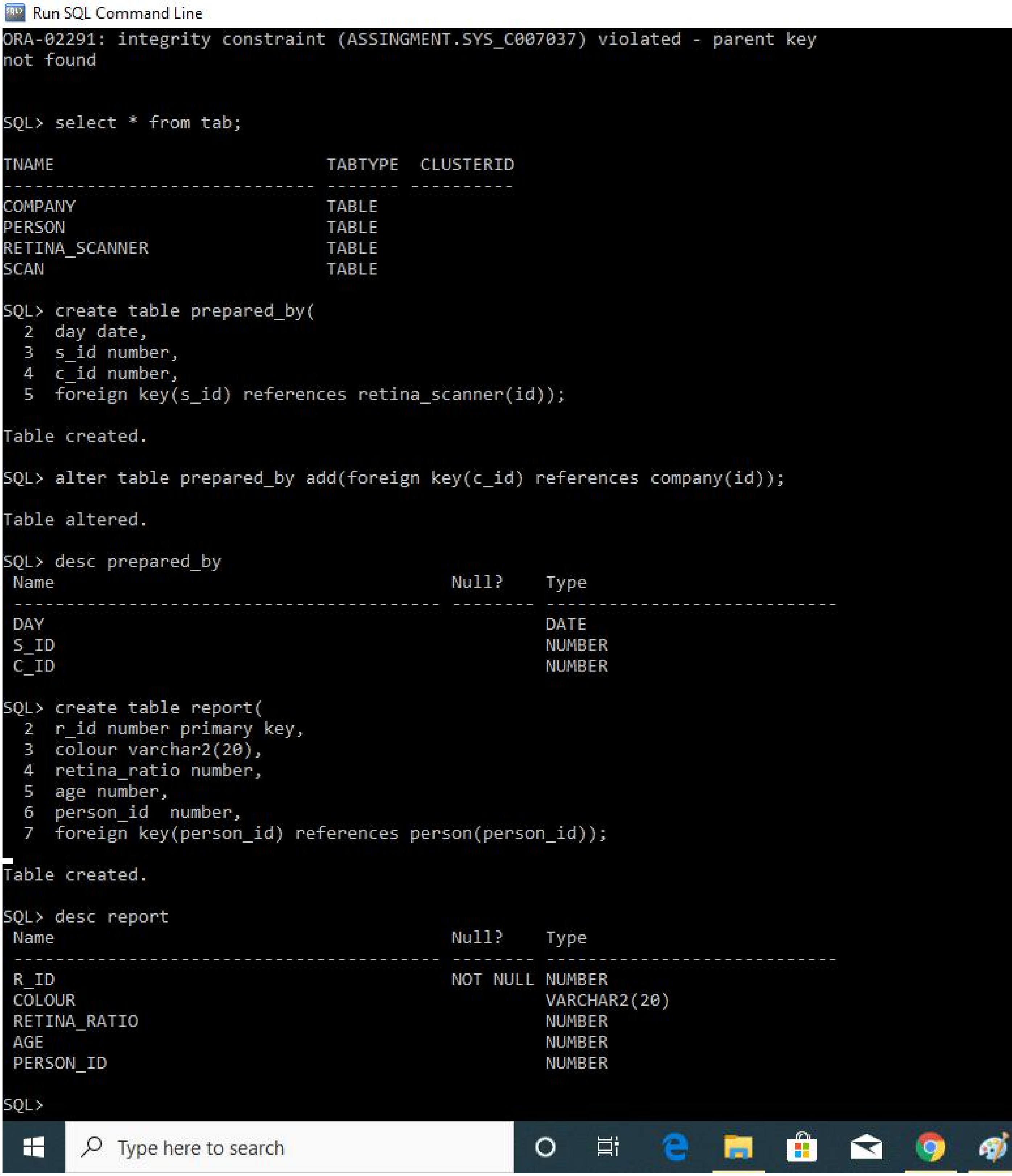
```
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL>
SQL> select * from tab;

TNAME                          TABTYPE  CLUSTERID
------------------------------ ------- ----------
COMPANY                        TABLE
PERSON                         TABLE
RETINA_SCANNER                 TABLE

SQL> desc company
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 ID                                        NOT NULL NUMBER
 NAME                                               VARCHAR2(20)
 ADDRESS                                            VARCHAR2(40)
 RATING                                             NUMBER

SQL> desc retina_scanner
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 ID                                        NOT NULL NUMBER
 NAME                                               VARCHAR2(20)
 COST                                      NOT NULL NUMBER
 ACCURACY                                  NOT NULL NUMBER

SQL> desc person
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 PERSON_ID                                 NOT NULL NUMBER(10)
 NAME                                               VARCHAR2(30)
 ADDRESS                                            VARCHAR2(40)
 PHONE_NO                                           NUMBER(10)

SQL>
```



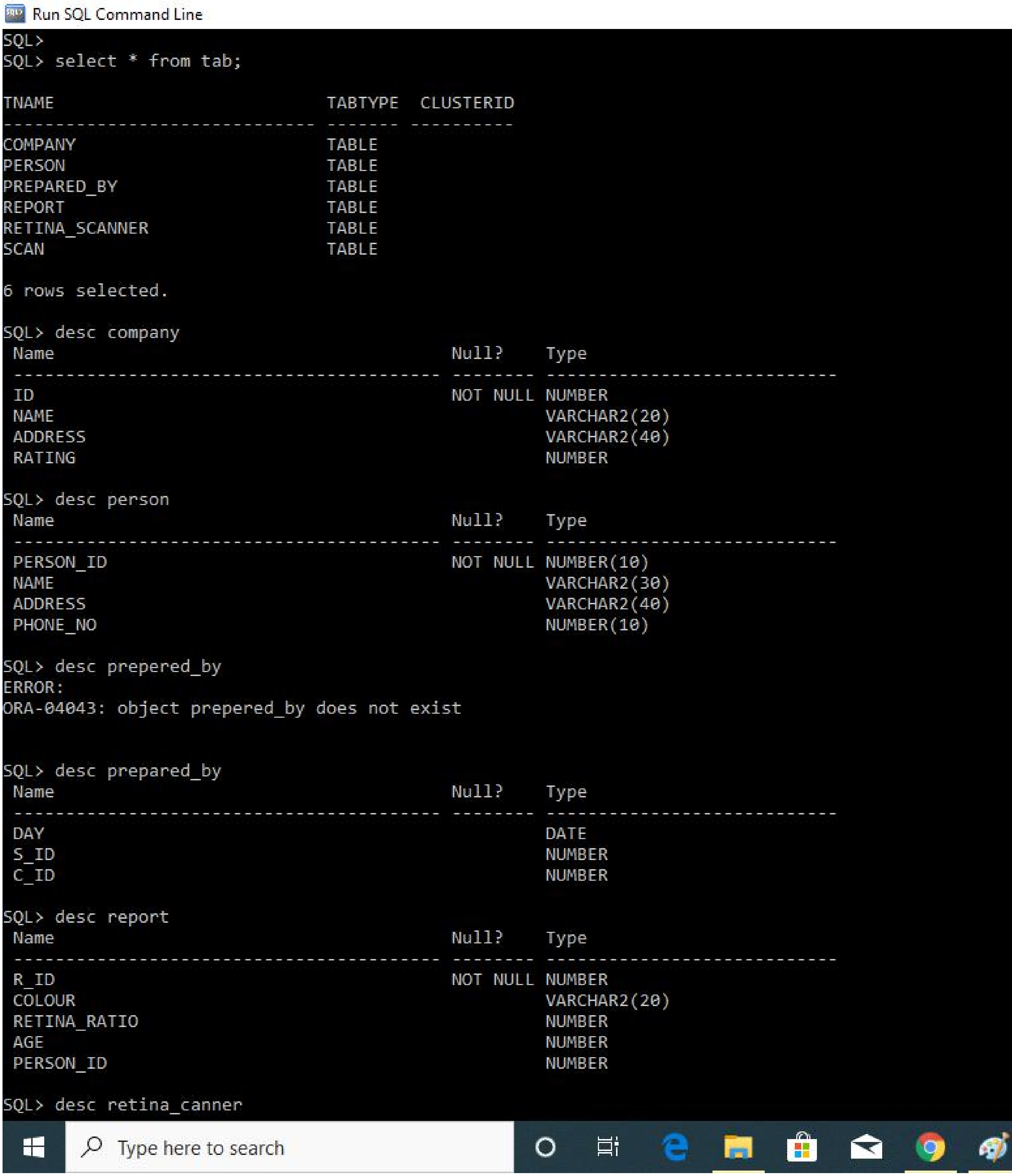
```
 NAME                                               VARCHAR2(30)
 ADDRESS                                            VARCHAR2(40)
 PHONE_NO                                           NUMBER(10)

SQL> create table scan(
  2  report_id number;
report_id number
               *
ERROR at line 2:
ORA-00907: missing right parenthesis


SQL> create table scan(
  2  report_id number,
  3  primary key(report_id),
  4  p_id number,
  5  foreign key(p_id) references person(person_id),
  6  s_id number,
  7  foreign key(s_id) references retina_scanner(id));

Table created.

SQL> desc scan
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 REPORT_ID                                 NOT NULL NUMBER
 P_ID                                               NUMBER
 S_ID                                               NUMBER

SQL> _
```



```
  3  primary key(report_id),
  4  p_id number,
  5  foreign key(p_id) references person(person_id),
  6  s_id number,
  7  foreign key(s_id) references retina_scanner(id));

Table created.

SQL> desc scan
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 REPORT_ID                                 NOT NULL NUMBER
 P_ID                                               NUMBER
 S_ID                                               NUMBER

SQL> insert into scan values(13,14,15);
insert into scan values(13,14,15)
*
ERROR at line 1:
ORA-02291: integrity constraint (ASSINGMENT.SYS_C007037) violated - parent key
not found


SQL> select * from tab;

TNAME                          TABTYPE  CLUSTERID
------------------------------ ------- ----------
COMPANY                        TABLE
PERSON                         TABLE
RETINA_SCANNER                 TABLE
SCAN                           TABLE

SQL> create table prepared_by(
  2  day date,
  3  s_id number,
  4  c_id number,
  5  foreign key(s_id) references retina_scanner(id));

Table created.

SQL> alter table prepared_by add(foreign key(c_id) references company(id));

Table altered.

SQL> desc prepared_by
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 DAY                                                DATE
 S_ID                                               NUMBER
 C_ID                                               NUMBER

SQL>
```

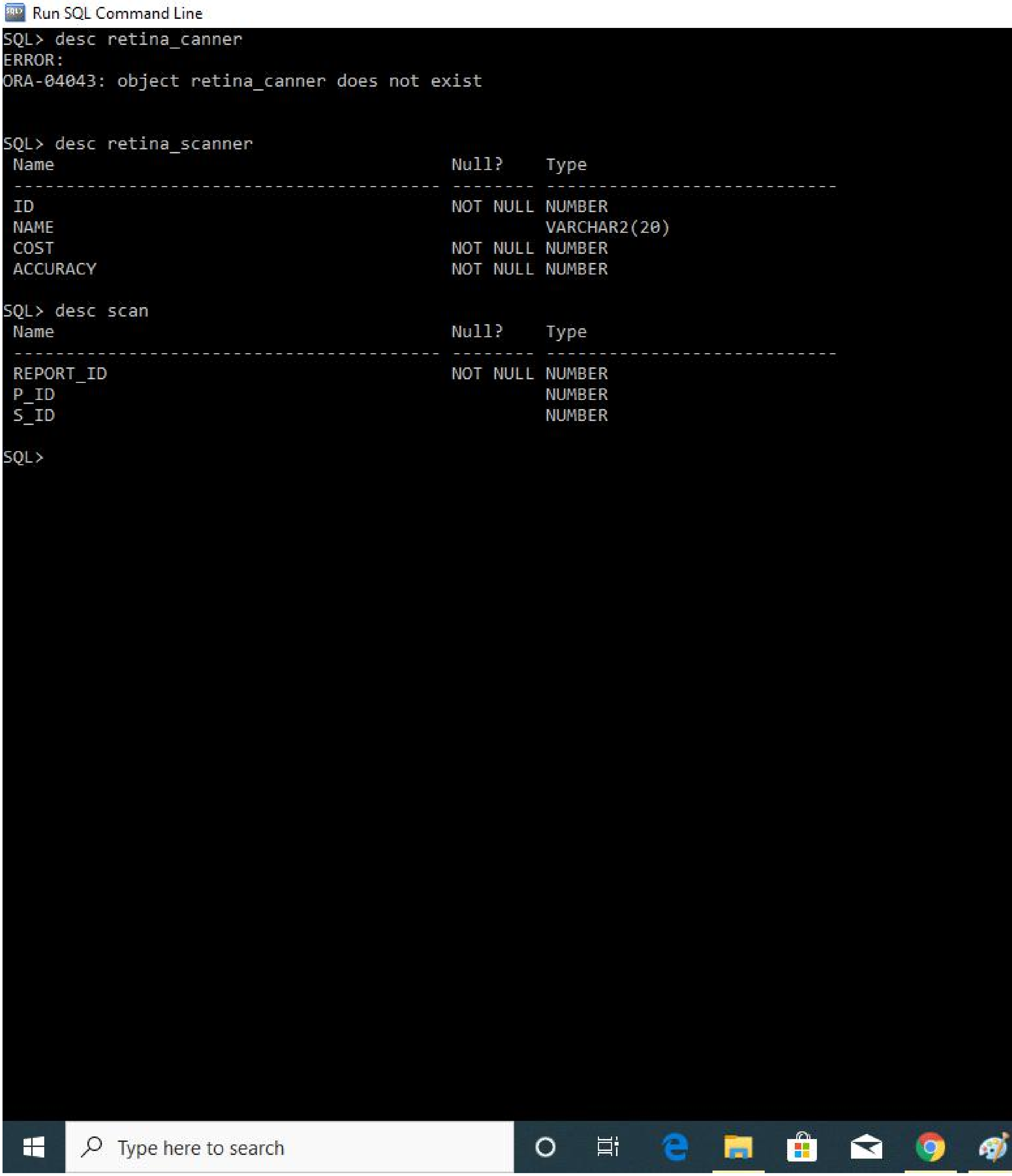


```
ORA-02291: integrity constraint (ASSINGMENT.SYS_C007037) violated - parent key
not found


SQL> select * from tab;

TNAME                          TABTYPE  CLUSTERID
------------------------------ ------- ----------
COMPANY                        TABLE
PERSON                         TABLE
RETINA_SCANNER                 TABLE
SCAN                           TABLE

SQL> create table prepared_by(
  2  day date,
  3  s_id number,
  4  c_id number,
  5  foreign key(s_id) references retina_scanner(id));

Table created.

SQL> alter table prepared_by add(foreign key(c_id) references company(id));

Table altered.

SQL> desc prepared_by
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 DAY                                                DATE
 S_ID                                               NUMBER
 C_ID                                               NUMBER

SQL> create table report(
  2  r_id number primary key,
  3  colour varchar2(20),
  4  retina_ratio number,
  5  age number,
  6  person_id  number,
  7  foreign key(person_id) references person(person_id));

Table created.

SQL> desc report
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 R_ID                                      NOT NULL NUMBER
 COLOUR                                             VARCHAR2(20)
 RETINA_RATIO                                       NUMBER
 AGE                                                NUMBER
 PERSON_ID                                          NUMBER

SQL>
```



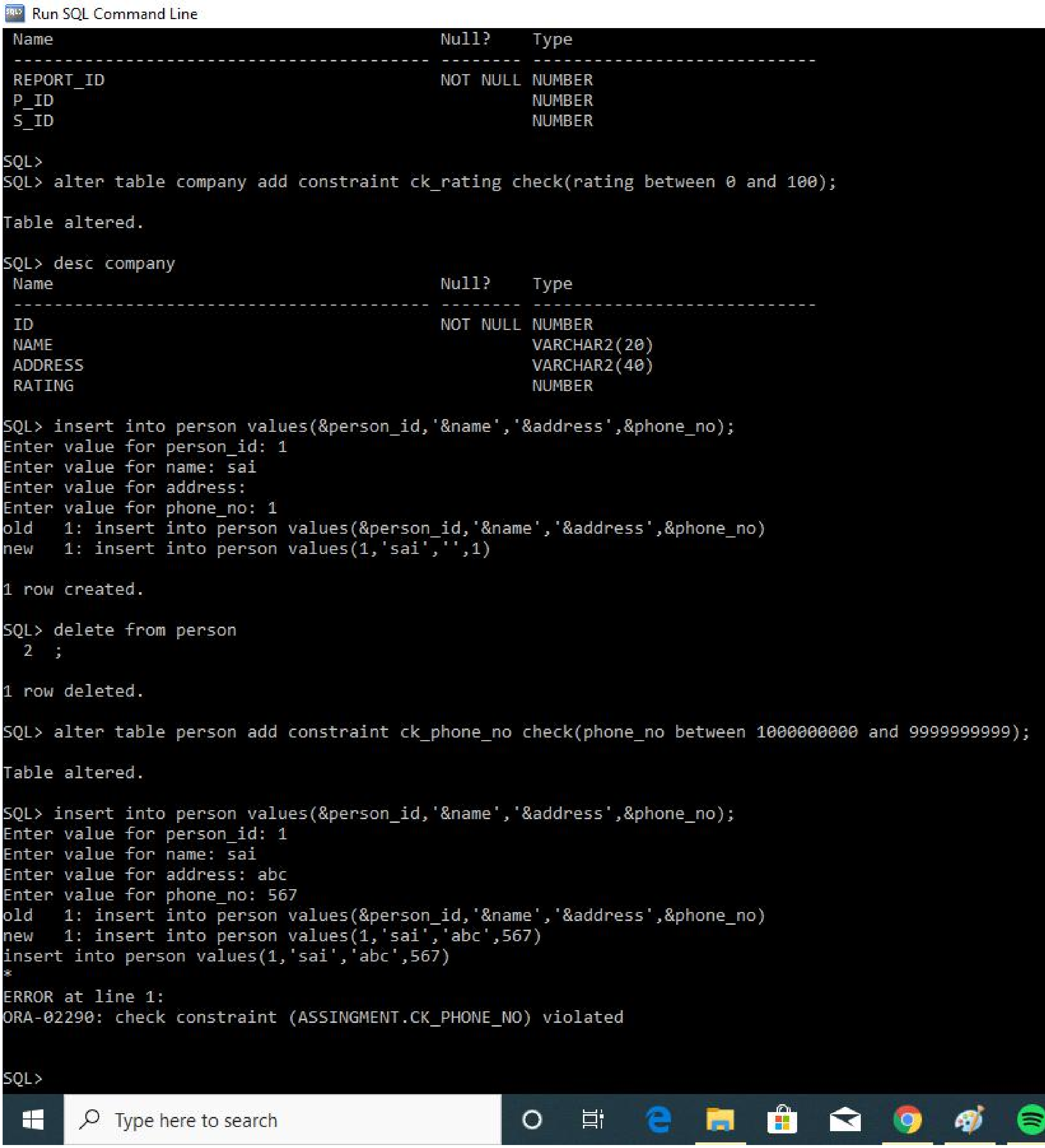
```
SQL>
SQL> select * from tab;

TNAME                          TABTYPE  CLUSTERID
------------------------------ ------- ----------
COMPANY                        TABLE
PERSON                         TABLE
PREPARED_BY                    TABLE
REPORT                         TABLE
RETINA_SCANNER                 TABLE
SCAN                           TABLE

6 rows selected.

SQL> desc company
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 ID                                        NOT NULL NUMBER
 NAME                                               VARCHAR2(20)
 ADDRESS                                            VARCHAR2(40)
 RATING                                             NUMBER

SQL> desc person
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 PERSON_ID                                 NOT NULL NUMBER(10)
 NAME                                               VARCHAR2(30)
 ADDRESS                                            VARCHAR2(40)
 PHONE_NO                                           NUMBER(10)

SQL> desc prepered_by
ERROR:
ORA-04043: object prepered_by does not exist


SQL> desc prepared_by
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 DAY                                                DATE
 S_ID                                               NUMBER
 C_ID                                               NUMBER

SQL> desc report
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 R_ID                                      NOT NULL NUMBER
 COLOUR                                             VARCHAR2(20)
 RETINA_RATIO                                       NUMBER
 AGE                                                NUMBER
 PERSON_ID                                          NUMBER

SQL> desc retina_canner
```

DBMS   ASSIGNMENT -2
Title: Iris based age classification system

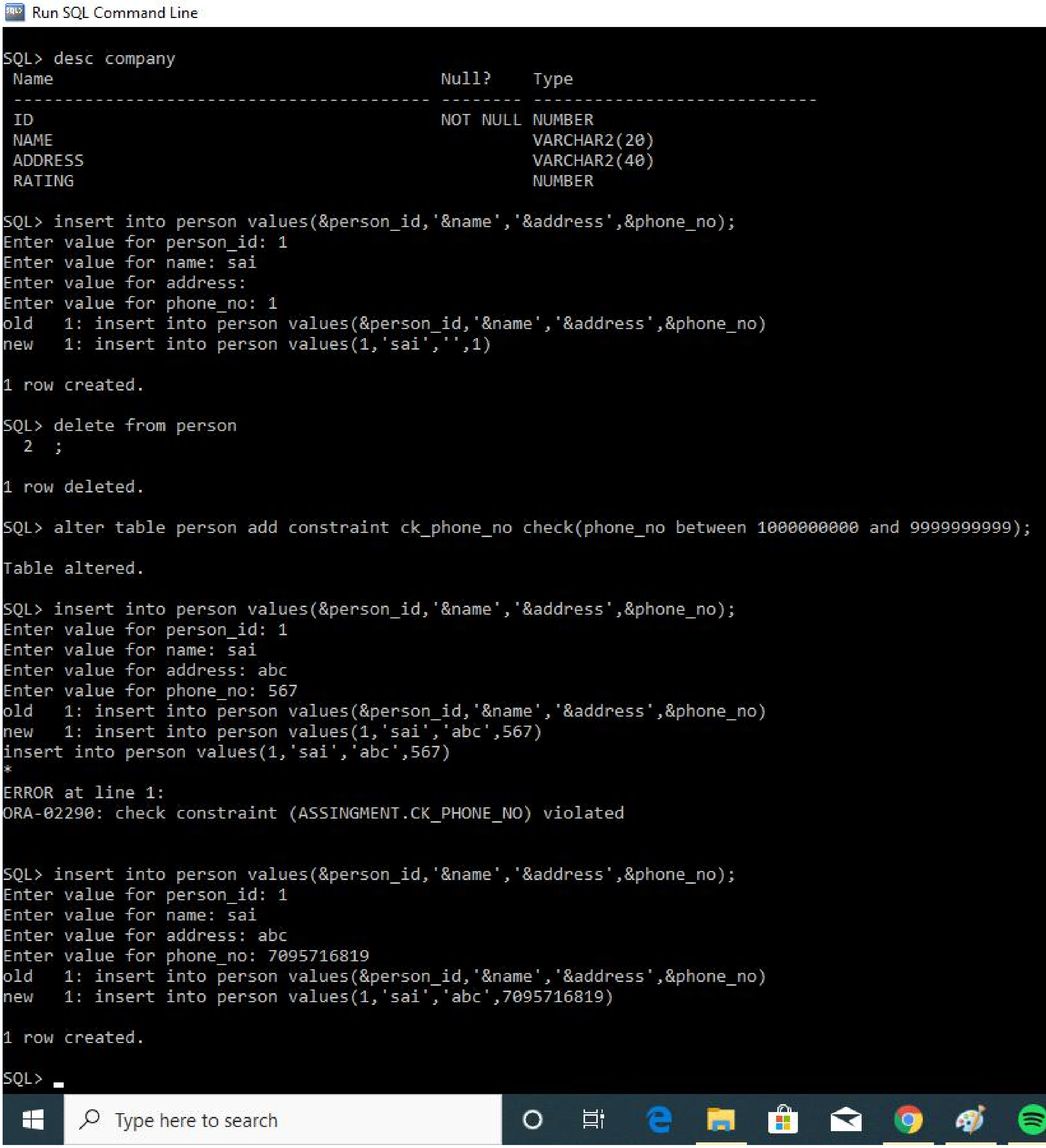


```
SQL> desc retina_canner
ERROR:
ORA-04043: object retina_canner does not exist


SQL> desc retina_scanner
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 ID                                        NOT NULL NUMBER
 NAME                                               VARCHAR2(20)
 COST                                      NOT NULL NUMBER
 ACCURACY                                  NOT NULL NUMBER

SQL> desc scan
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 REPORT_ID                                 NOT NULL NUMBER
 P_ID                                               NUMBER
 S_ID                                               NUMBER

SQL>
```

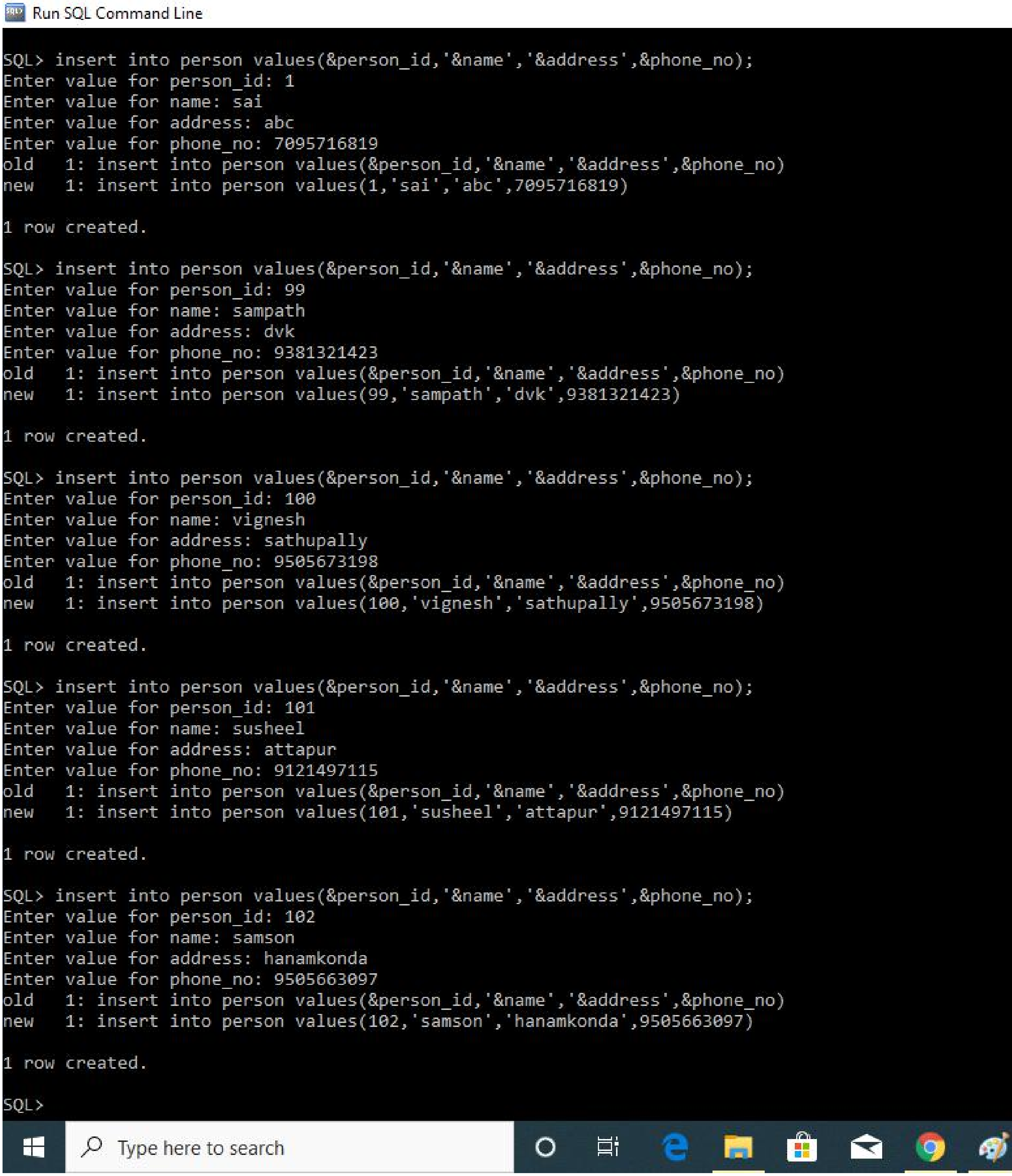
Name : S.Sai Rohith   Roll Number : 1602-18-737-098



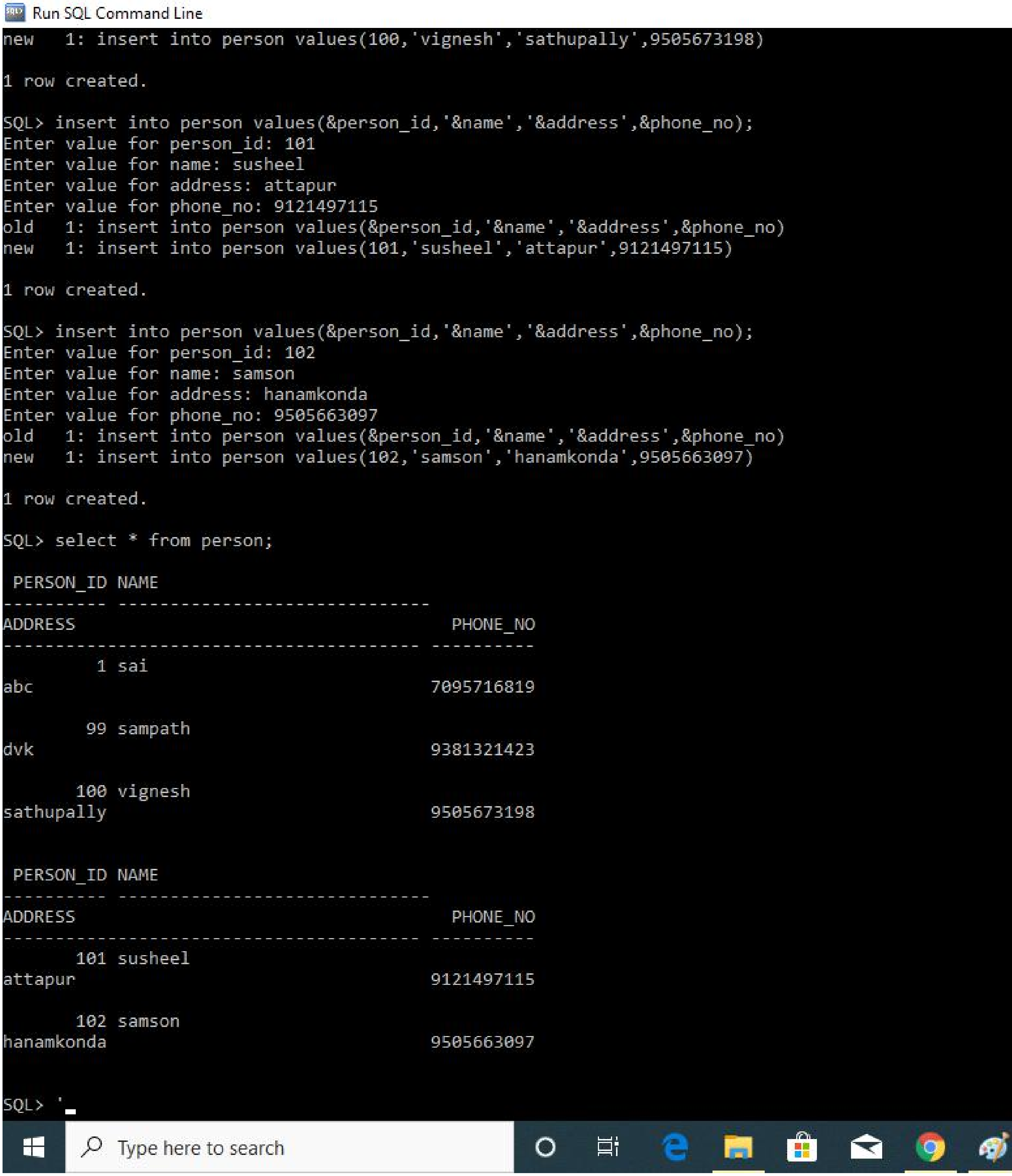
```
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 REPORT_ID                                 NOT NULL NUMBER
 P_ID                                               NUMBER
 S_ID                                               NUMBER

SQL>
SQL> alter table company add constraint ck_rating check(rating between 0 and 100);

Table altered.

SQL> desc company
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 ID                                        NOT NULL NUMBER
 NAME                                               VARCHAR2(20)
 ADDRESS                                            VARCHAR2(40)
 RATING                                             NUMBER

SQL> insert into person values(&person_id,'&name','&address',&phone_no);
Enter value for person_id: 1
Enter value for name: sai
Enter value for address:
Enter value for phone_no: 1
old   1: insert into person values(&person_id,'&name','&address',&phone_no)
new   1: insert into person values(1,'sai','',1)

1 row created.

SQL> delete from person
  2  ;

1 row deleted.

SQL> alter table person add constraint ck_phone_no check(phone_no between 1000000000 and 9999999999);

Table altered.

SQL> insert into person values(&person_id,'&name','&address',&phone_no);
Enter value for person_id: 1
Enter value for name: sai
Enter value for address: abc
Enter value for phone_no: 567
old   1: insert into person values(&person_id,'&name','&address',&phone_no)
new   1: insert into person values(1,'sai','abc',567)
insert into person values(1,'sai','abc',567)
*
ERROR at line 1:
ORA-02290: check constraint (ASSINGMENT.CK_PHONE_NO) violated


SQL>
```



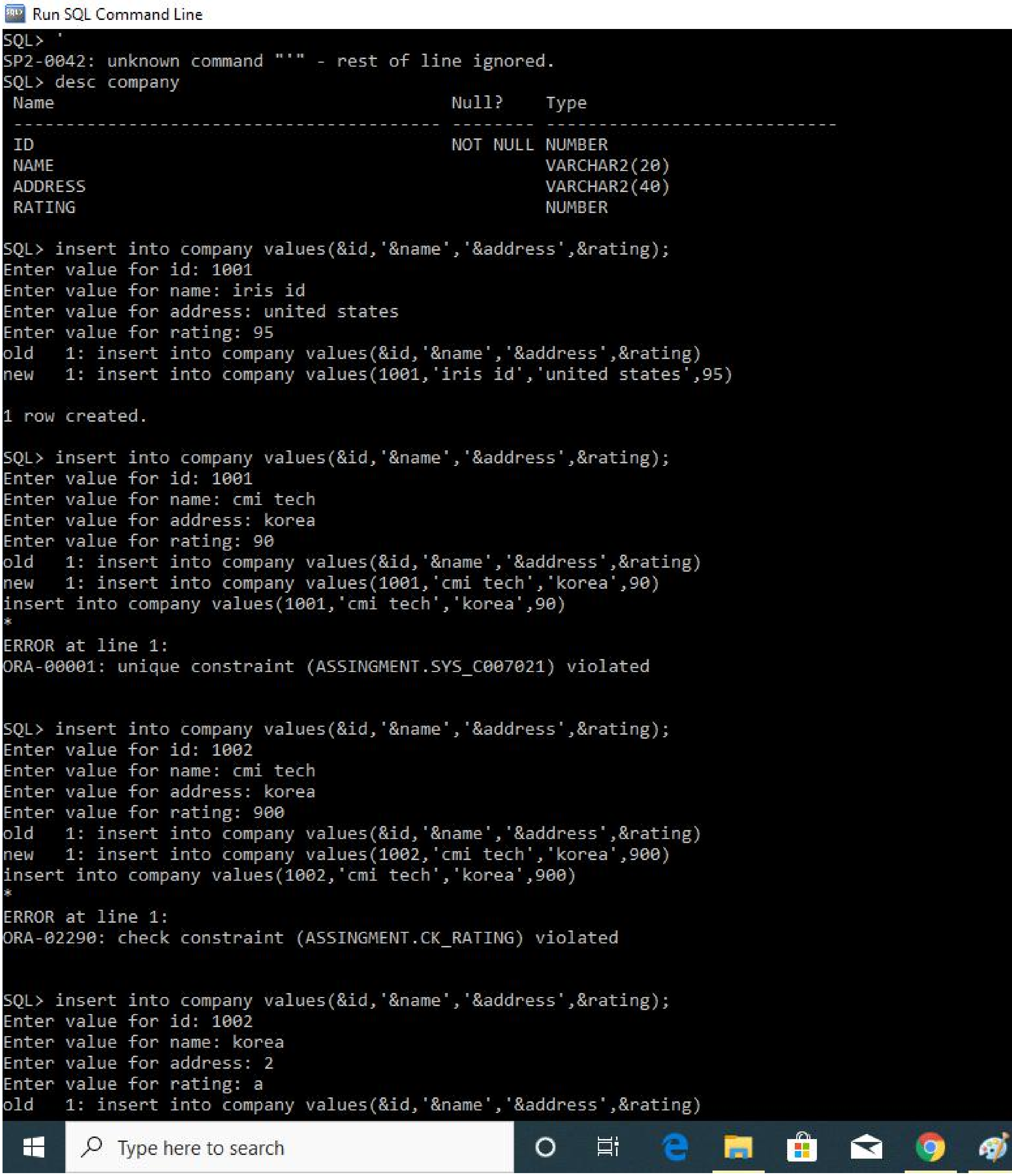
```
SQL> desc company
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 ID                                        NOT NULL NUMBER
 NAME                                               VARCHAR2(20)
 ADDRESS                                            VARCHAR2(40)
 RATING                                             NUMBER

SQL> insert into person values(&person_id,'&name','&address',&phone_no);
Enter value for person_id: 1
Enter value for name: sai
Enter value for address:
Enter value for phone_no: 1
old   1: insert into person values(&person_id,'&name','&address',&phone_no)
new   1: insert into person values(1,'sai','',1)

1 row created.

SQL> delete from person
  2  ;

1 row deleted.

SQL> alter table person add constraint ck_phone_no check(phone_no between 1000000000 and 9999999999);

Table altered.

SQL> insert into person values(&person_id,'&name','&address',&phone_no);
Enter value for person_id: 1
Enter value for name: sai
Enter value for address: abc
Enter value for phone_no: 567
old   1: insert into person values(&person_id,'&name','&address',&phone_no)
new   1: insert into person values(1,'sai','abc',567)
insert into person values(1,'sai','abc',567)
*
ERROR at line 1:
ORA-02290: check constraint (ASSINGMENT.CK_PHONE_NO) violated


SQL> insert into person values(&person_id,'&name','&address',&phone_no);
Enter value for person_id: 1
Enter value for name: sai
Enter value for address: abc
Enter value for phone_no: 7095716819
old   1: insert into person values(&person_id,'&name','&address',&phone_no)
new   1: insert into person values(1,'sai','abc',7095716819)

1 row created.

SQL> _
```



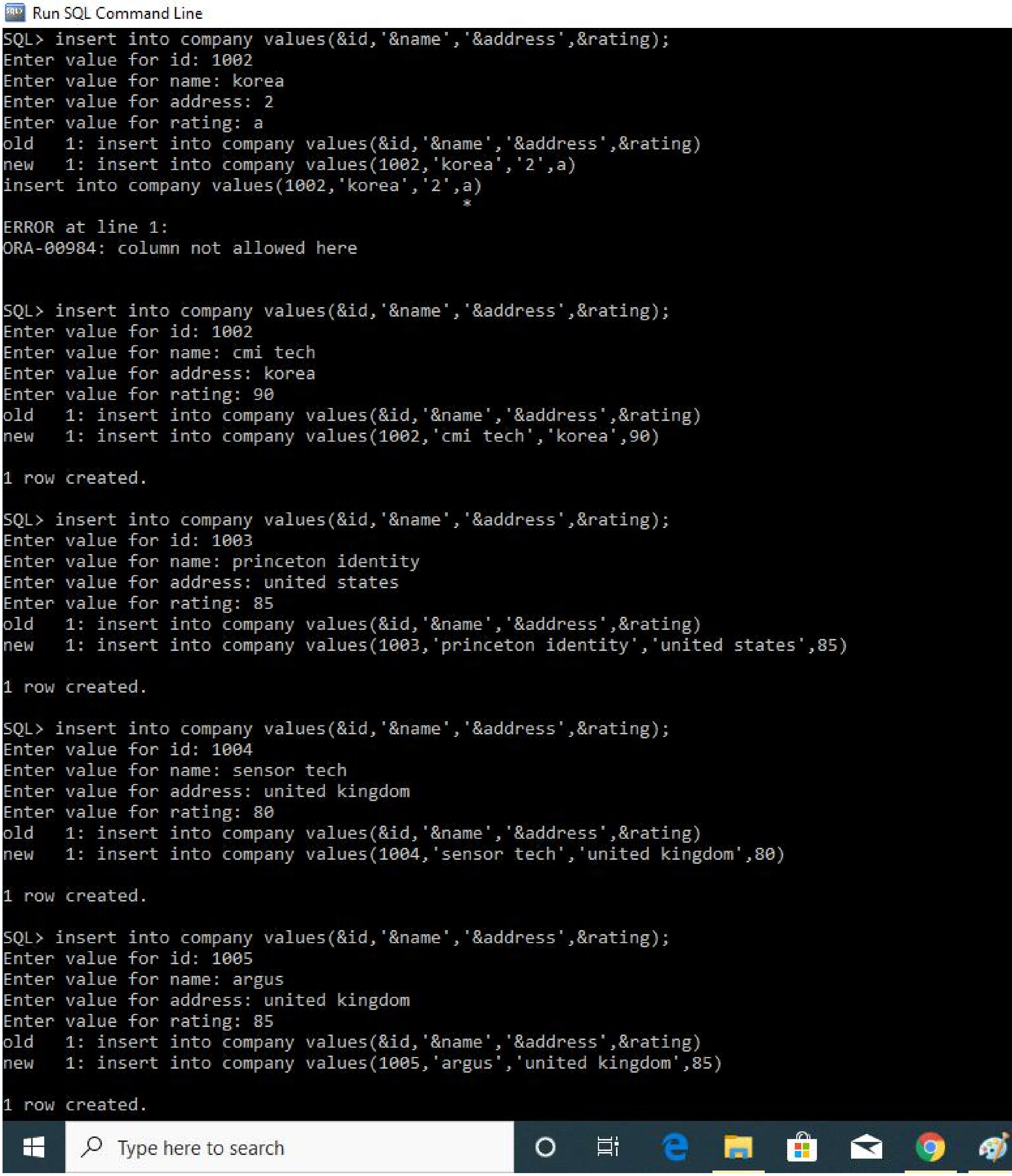
```
SQL> insert into person values(&person_id,'&name','&address',&phone_no);
Enter value for person_id: 1
Enter value for name: sai
Enter value for address: abc
Enter value for phone_no: 7095716819
old   1: insert into person values(&person_id,'&name','&address',&phone_no)
new   1: insert into person values(1,'sai','abc',7095716819)

1 row created.

SQL> insert into person values(&person_id,'&name','&address',&phone_no);
Enter value for person_id: 99
Enter value for name: sampath
Enter value for address: dvk
Enter value for phone_no: 9381321423
old   1: insert into person values(&person_id,'&name','&address',&phone_no)
new   1: insert into person values(99,'sampath','dvk',9381321423)

1 row created.

SQL> insert into person values(&person_id,'&name','&address',&phone_no);
Enter value for person_id: 100
Enter value for name: vignesh
Enter value for address: sathupally
Enter value for phone_no: 9505673198
old   1: insert into person values(&person_id,'&name','&address',&phone_no)
new   1: insert into person values(100,'vignesh','sathupally',9505673198)

1 row created.

SQL> insert into person values(&person_id,'&name','&address',&phone_no);
Enter value for person_id: 101
Enter value for name: susheel
Enter value for address: attapur
Enter value for phone_no: 9121497115
old   1: insert into person values(&person_id,'&name','&address',&phone_no)
new   1: insert into person values(101,'susheel','attapur',9121497115)

1 row created.

SQL> insert into person values(&person_id,'&name','&address',&phone_no);
Enter value for person_id: 102
Enter value for name: samson
Enter value for address: hanamkonda
Enter value for phone_no: 9505663097
old   1: insert into person values(&person_id,'&name','&address',&phone_no)
new   1: insert into person values(102,'samson','hanamkonda',9505663097)

1 row created.

SQL>
```



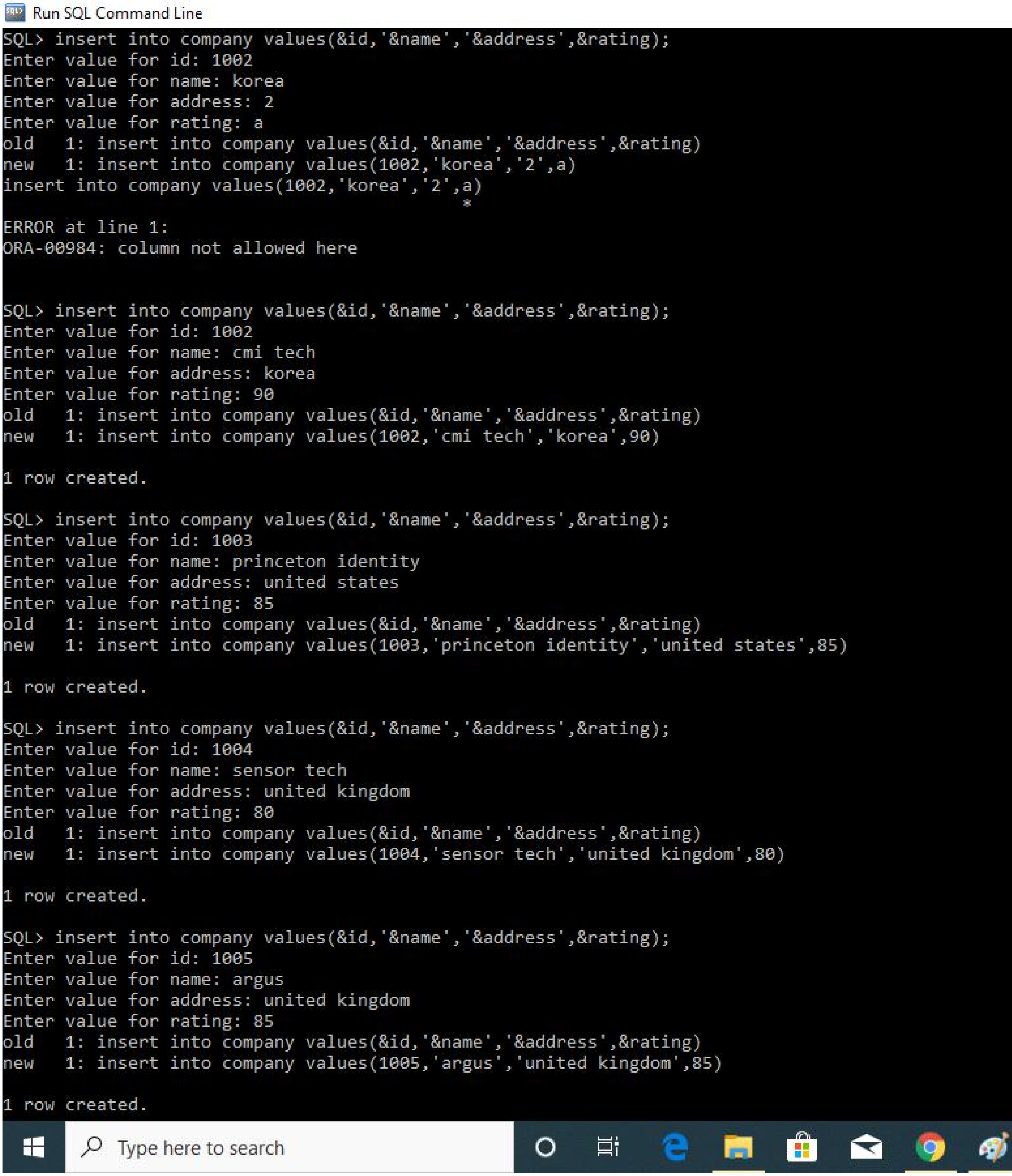
```
new   1: insert into person values(100,'vignesh','sathupally',9505673198)

1 row created.

SQL> insert into person values(&person_id,'&name','&address',&phone_no);
Enter value for person_id: 101
Enter value for name: susheel
Enter value for address: attapur
Enter value for phone_no: 9121497115
old   1: insert into person values(&person_id,'&name','&address',&phone_no)
new   1: insert into person values(101,'susheel','attapur',9121497115)

1 row created.

SQL> insert into person values(&person_id,'&name','&address',&phone_no);
Enter value for person_id: 102
Enter value for name: samson
Enter value for address: hanamkonda
Enter value for phone_no: 9505663097
old   1: insert into person values(&person_id,'&name','&address',&phone_no)
new   1: insert into person values(102,'samson','hanamkonda',9505663097)

1 row created.

SQL> select * from person;

 PERSON_ID NAME
---------- ------------------------------
ADDRESS                                      PHONE_NO
---------------------------------------- ----------
         1 sai
abc                                      7095716819

        99 sampath
dvk                                      9381321423

       100 vignesh
sathupally                               9505673198


 PERSON_ID NAME
---------- ------------------------------
ADDRESS                                      PHONE_NO
---------------------------------------- ----------
       101 susheel
attapur                                  9121497115

       102 samson
hanamkonda                               9505663097


SQL> '
```



```
SQL> '
SP2-0042: unknown command "'" - rest of line ignored.
SQL> desc company
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 ID                                        NOT NULL NUMBER
 NAME                                               VARCHAR2(20)
 ADDRESS                                            VARCHAR2(40)
 RATING                                             NUMBER

SQL> insert into company values(&id,'&name','&address',&rating);
Enter value for id: 1001
Enter value for name: iris id
Enter value for address: united states
Enter value for rating: 95
old   1: insert into company values(&id,'&name','&address',&rating)
new   1: insert into company values(1001,'iris id','united states',95)

1 row created.

SQL> insert into company values(&id,'&name','&address',&rating);
Enter value for id: 1001
Enter value for name: cmi tech
Enter value for address: korea
Enter value for rating: 90
old   1: insert into company values(&id,'&name','&address',&rating)
new   1: insert into company values(1001,'cmi tech','korea',90)
insert into company values(1001,'cmi tech','korea',90)
*
ERROR at line 1:
ORA-00001: unique constraint (ASSINGMENT.SYS_C007021) violated


SQL> insert into company values(&id,'&name','&address',&rating);
Enter value for id: 1002
Enter value for name: cmi tech
Enter value for address: korea
Enter value for rating: 900
old   1: insert into company values(&id,'&name','&address',&rating)
new   1: insert into company values(1002,'cmi tech','korea',900)
insert into company values(1002,'cmi tech','korea',900)
*
ERROR at line 1:
ORA-02290: check constraint (ASSINGMENT.CK_RATING) violated


SQL> insert into company values(&id,'&name','&address',&rating);
Enter value for id: 1002
Enter value for name: korea
Enter value for address: 2
Enter value for rating: a
old   1: insert into company values(&id,'&name','&address',&rating)
```

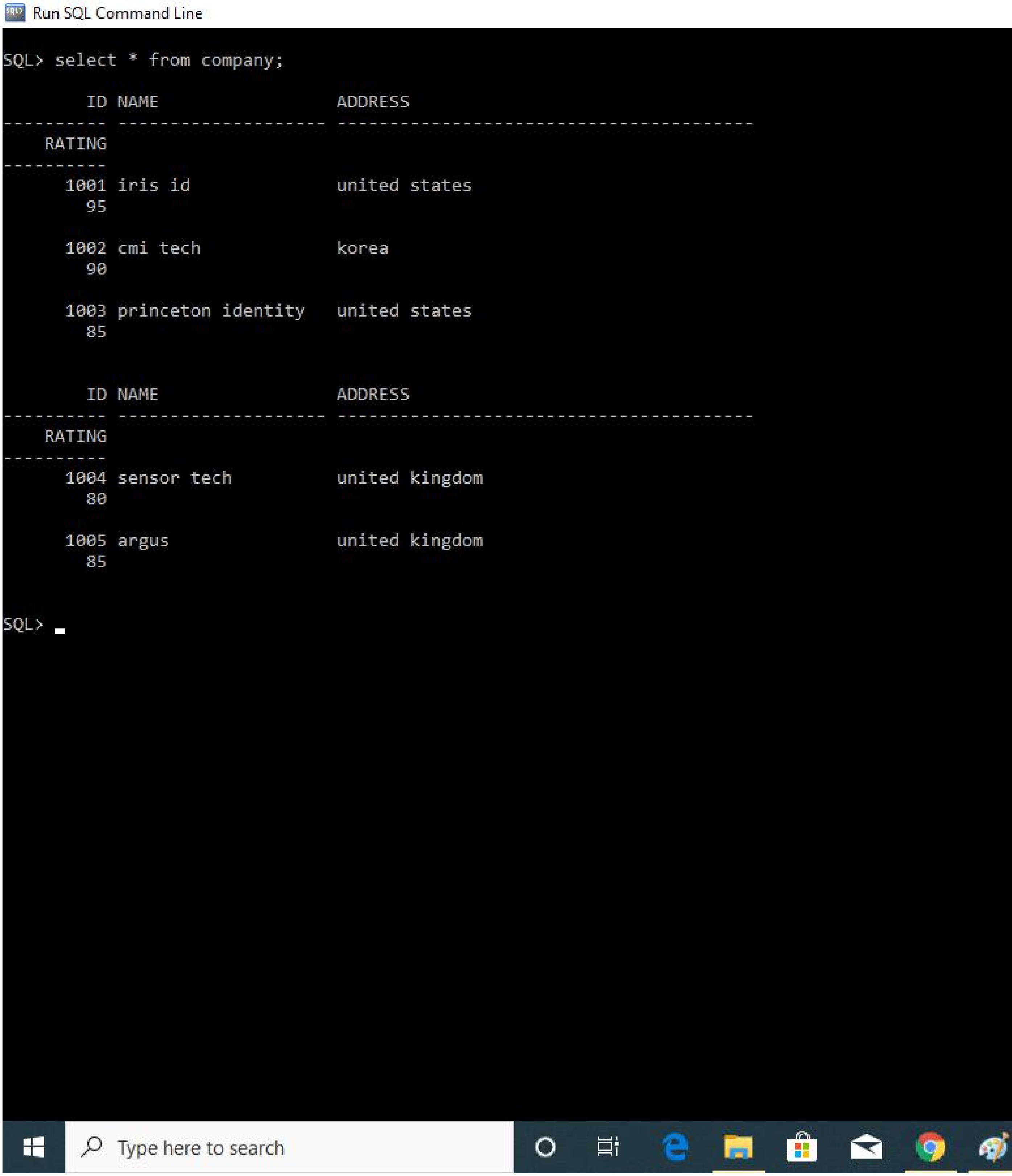


```
SQL> insert into company values(&id,'&name','&address',&rating);
Enter value for id: 1002
Enter value for name: korea
Enter value for address: 2
Enter value for rating: a
old   1: insert into company values(&id,'&name','&address',&rating)
new   1: insert into company values(1002,'korea','2',a)
insert into company values(1002,'korea','2',a)
                                            *

ERROR at line 1:
ORA-00984: column not allowed here


SQL> insert into company values(&id,'&name','&address',&rating);
Enter value for id: 1002
Enter value for name: cmi tech
Enter value for address: korea
Enter value for rating: 90
old   1: insert into company values(&id,'&name','&address',&rating)
new   1: insert into company values(1002,'cmi tech','korea',90)

1 row created.

SQL> insert into company values(&id,'&name','&address',&rating);
Enter value for id: 1003
Enter value for name: princeton identity
Enter value for address: united states
Enter value for rating: 85
old   1: insert into company values(&id,'&name','&address',&rating)
new   1: insert into company values(1003,'princeton identity','united states',85)

1 row created.

SQL> insert into company values(&id,'&name','&address',&rating);
Enter value for id: 1004
Enter value for name: sensor tech
Enter value for address: united kingdom
Enter value for rating: 80
old   1: insert into company values(&id,'&name','&address',&rating)
new   1: insert into company values(1004,'sensor tech','united kingdom',80)

1 row created.

SQL> insert into company values(&id,'&name','&address',&rating);
Enter value for id: 1005
Enter value for name: argus
Enter value for address: united kingdom
Enter value for rating: 85
old   1: insert into company values(&id,'&name','&address',&rating)
new   1: insert into company values(1005,'argus','united kingdom',85)

1 row created.
```



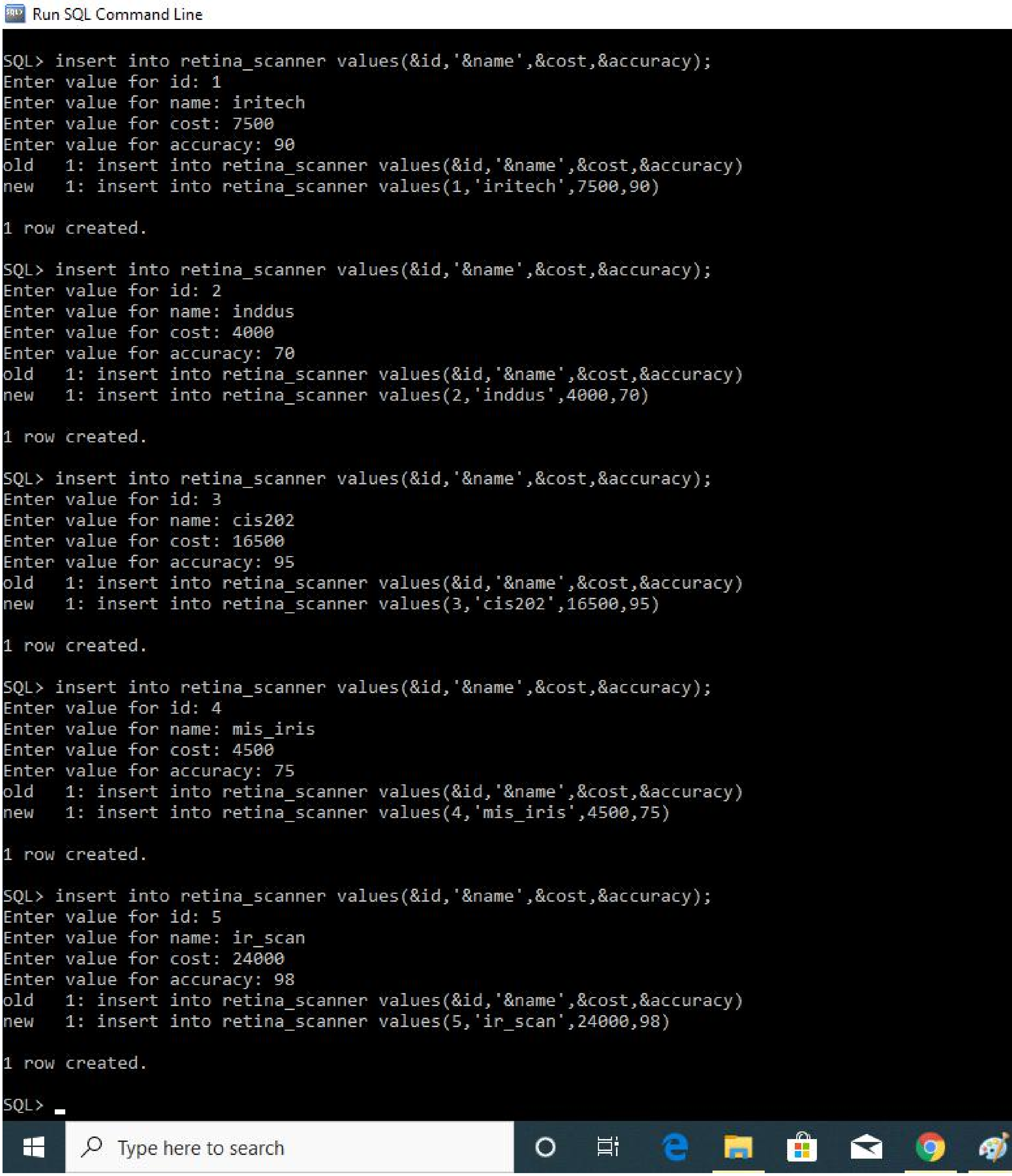
```
SQL> insert into company values(&id,'&name','&address',&rating);
Enter value for id: 1002
Enter value for name: korea
Enter value for address: 2
Enter value for rating: a
old   1: insert into company values(&id,'&name','&address',&rating)
new   1: insert into company values(1002,'korea','2',a)
insert into company values(1002,'korea','2',a)
                                            *

ERROR at line 1:
ORA-00984: column not allowed here


SQL> insert into company values(&id,'&name','&address',&rating);
Enter value for id: 1002
Enter value for name: cmi tech
Enter value for address: korea
Enter value for rating: 90
old   1: insert into company values(&id,'&name','&address',&rating)
new   1: insert into company values(1002,'cmi tech','korea',90)

1 row created.

SQL> insert into company values(&id,'&name','&address',&rating);
Enter value for id: 1003
Enter value for name: princeton identity
Enter value for address: united states
Enter value for rating: 85
old   1: insert into company values(&id,'&name','&address',&rating)
new   1: insert into company values(1003,'princeton identity','united states',85)

1 row created.

SQL> insert into company values(&id,'&name','&address',&rating);
Enter value for id: 1004
Enter value for name: sensor tech
Enter value for address: united kingdom
Enter value for rating: 80
old   1: insert into company values(&id,'&name','&address',&rating)
new   1: insert into company values(1004,'sensor tech','united kingdom',80)

1 row created.

SQL> insert into company values(&id,'&name','&address',&rating);
Enter value for id: 1005
Enter value for name: argus
Enter value for address: united kingdom
Enter value for rating: 85
old   1: insert into company values(&id,'&name','&address',&rating)
new   1: insert into company values(1005,'argus','united kingdom',85)

1 row created.
```

DBMS ASSIGNMENT -2
Title: Iris based age classification system



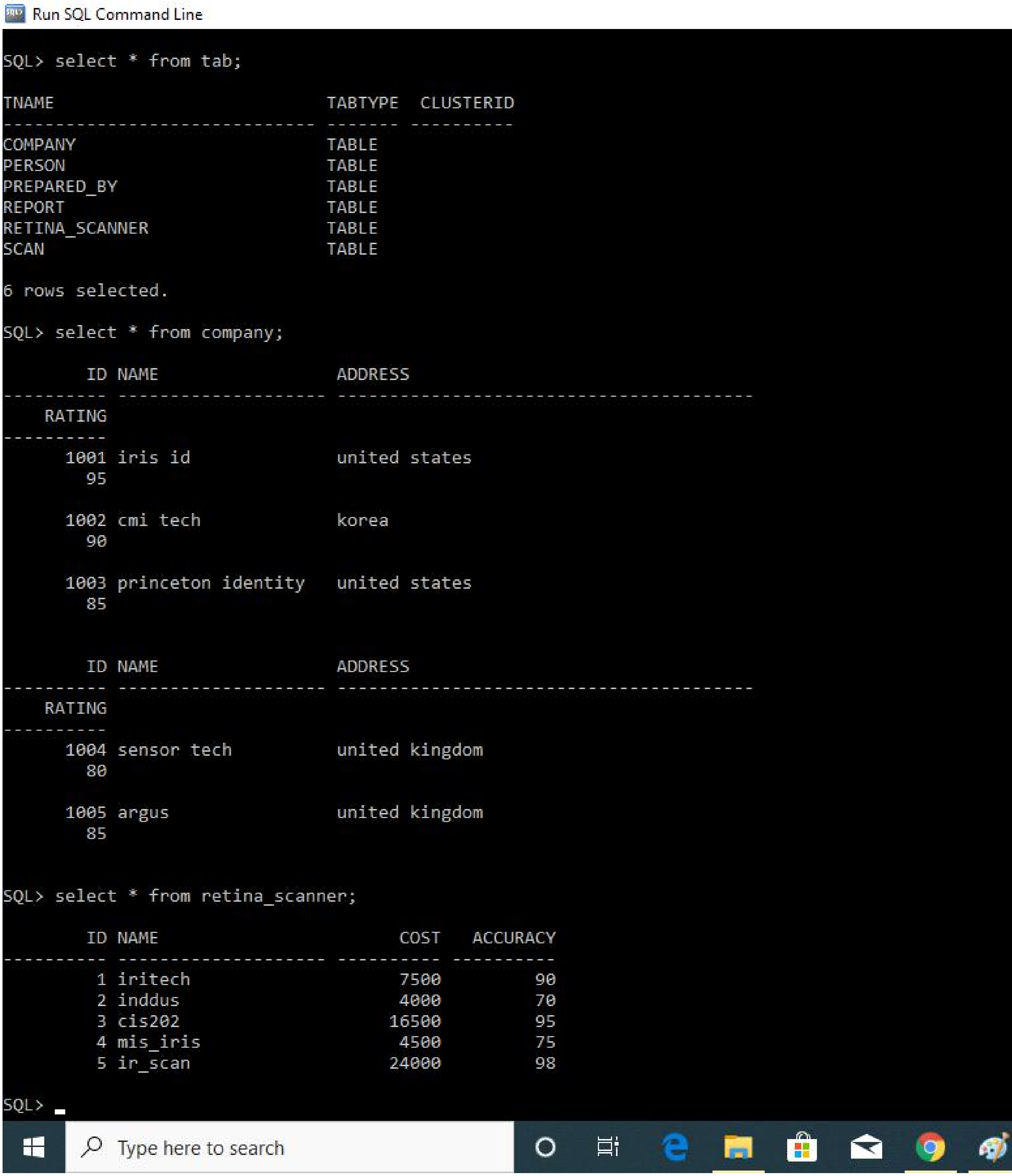
```
SQL> select * from company;

        ID NAME                 ADDRESS
---------- -------------------- ----------------------------------------
    RATING
----------
      1001 iris id              united states
        95

      1002 cmi tech             korea
        90

      1003 princeton identity   united states
        85


        ID NAME                 ADDRESS
---------- -------------------- ----------------------------------------
    RATING
----------
      1004 sensor tech          united kingdom
        80

      1005 argus                united kingdom
        85


SQL>
```

Name : S.Sai Rohith Roll Number : 1602-18-737-098



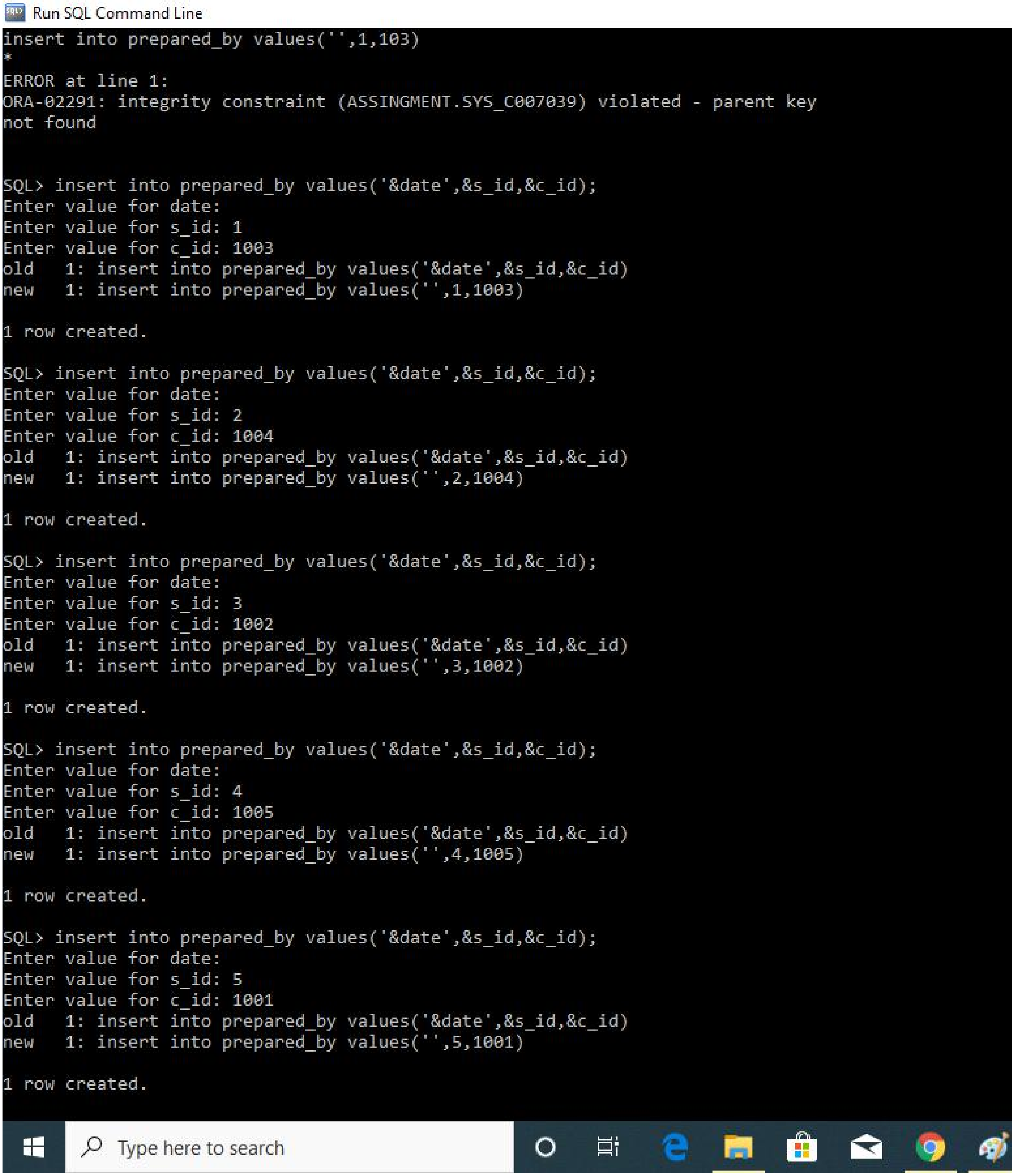
```
SQL> insert into retina_scanner values(&id,'&name',&cost,&accuracy);
Enter value for id: 1
Enter value for name: iritech
Enter value for cost: 7500
Enter value for accuracy: 90
old   1: insert into retina_scanner values(&id,'&name',&cost,&accuracy)
new   1: insert into retina_scanner values(1,'iritech',7500,90)

1 row created.

SQL> insert into retina_scanner values(&id,'&name',&cost,&accuracy);
Enter value for id: 2
Enter value for name: inddus
Enter value for cost: 4000
Enter value for accuracy: 70
old   1: insert into retina_scanner values(&id,'&name',&cost,&accuracy)
new   1: insert into retina_scanner values(2,'inddus',4000,70)

1 row created.

SQL> insert into retina_scanner values(&id,'&name',&cost,&accuracy);
Enter value for id: 3
Enter value for name: cis202
Enter value for cost: 16500
Enter value for accuracy: 95
old   1: insert into retina_scanner values(&id,'&name',&cost,&accuracy)
new   1: insert into retina_scanner values(3,'cis202',16500,95)

1 row created.

SQL> insert into retina_scanner values(&id,'&name',&cost,&accuracy);
Enter value for id: 4
Enter value for name: mis_iris
Enter value for cost: 4500
Enter value for accuracy: 75
old   1: insert into retina_scanner values(&id,'&name',&cost,&accuracy)
new   1: insert into retina_scanner values(4,'mis_iris',4500,75)

1 row created.

SQL> insert into retina_scanner values(&id,'&name',&cost,&accuracy);
Enter value for id: 5
Enter value for name: ir_scan
Enter value for cost: 24000
Enter value for accuracy: 98
old   1: insert into retina_scanner values(&id,'&name',&cost,&accuracy)
new   1: insert into retina_scanner values(5,'ir_scan',24000,98)

1 row created.

SQL> _
```

DBMS   ASSIGNMENT -2
Title: Iris based age classification system

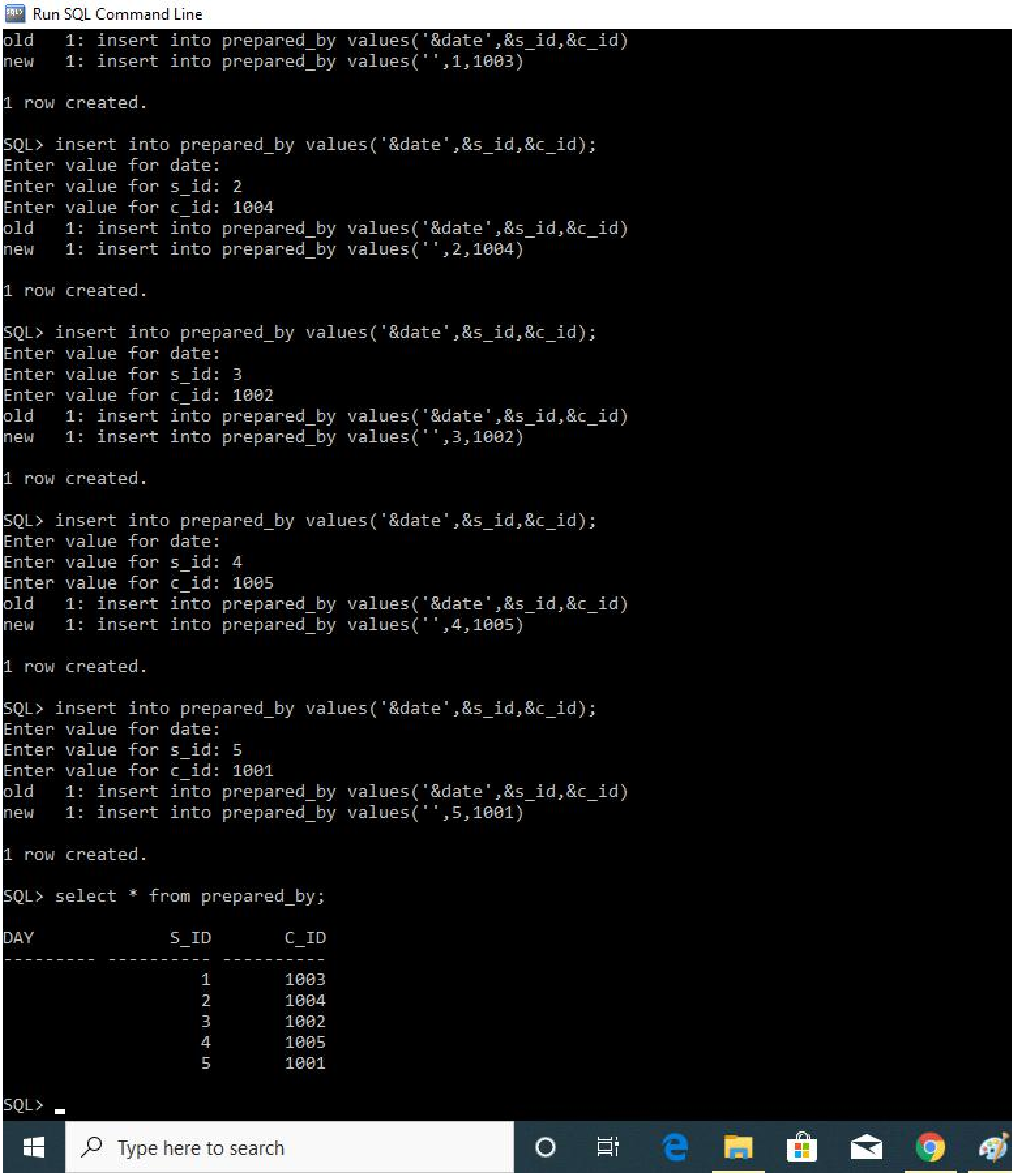


```
SQL> select * from tab;

TNAME                          TABTYPE  CLUSTERID
------------------------------ ------- ----------
COMPANY                        TABLE
PERSON                         TABLE
PREPARED_BY                    TABLE
REPORT                         TABLE
RETINA_SCANNER                 TABLE
SCAN                           TABLE

6 rows selected.

SQL> select * from company;

        ID NAME                 ADDRESS
---------- -------------------- ----------------------------------------
    RATING
----------
      1001 iris id              united states
        95

      1002 cmi tech             korea
        90

      1003 princeton identity   united states
        85


        ID NAME                 ADDRESS
---------- -------------------- ----------------------------------------
    RATING
----------
      1004 sensor tech          united kingdom
        80

      1005 argus                united kingdom
        85


SQL> select * from retina_scanner;

        ID NAME                       COST   ACCURACY
---------- -------------------- ---------- ----------
         1 iritech                    7500         90
         2 inddus                     4000         70
         3 cis202                    16500         95
         4 mis_iris                   4500         75
         5 ir_scan                   24000         98

SQL> _
```

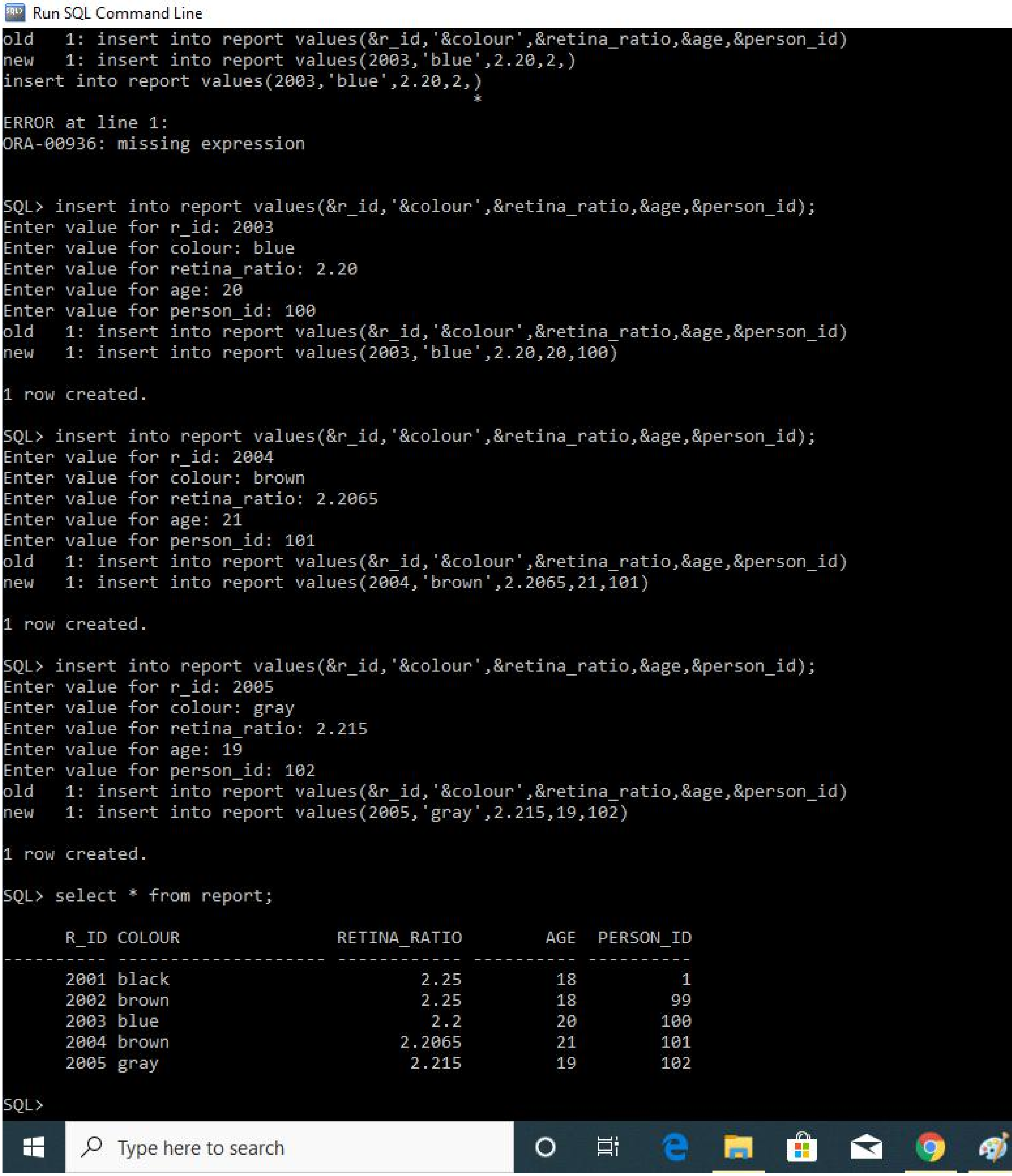
Name : S.Sai Rohith   Roll Number : 1602-18-737-098



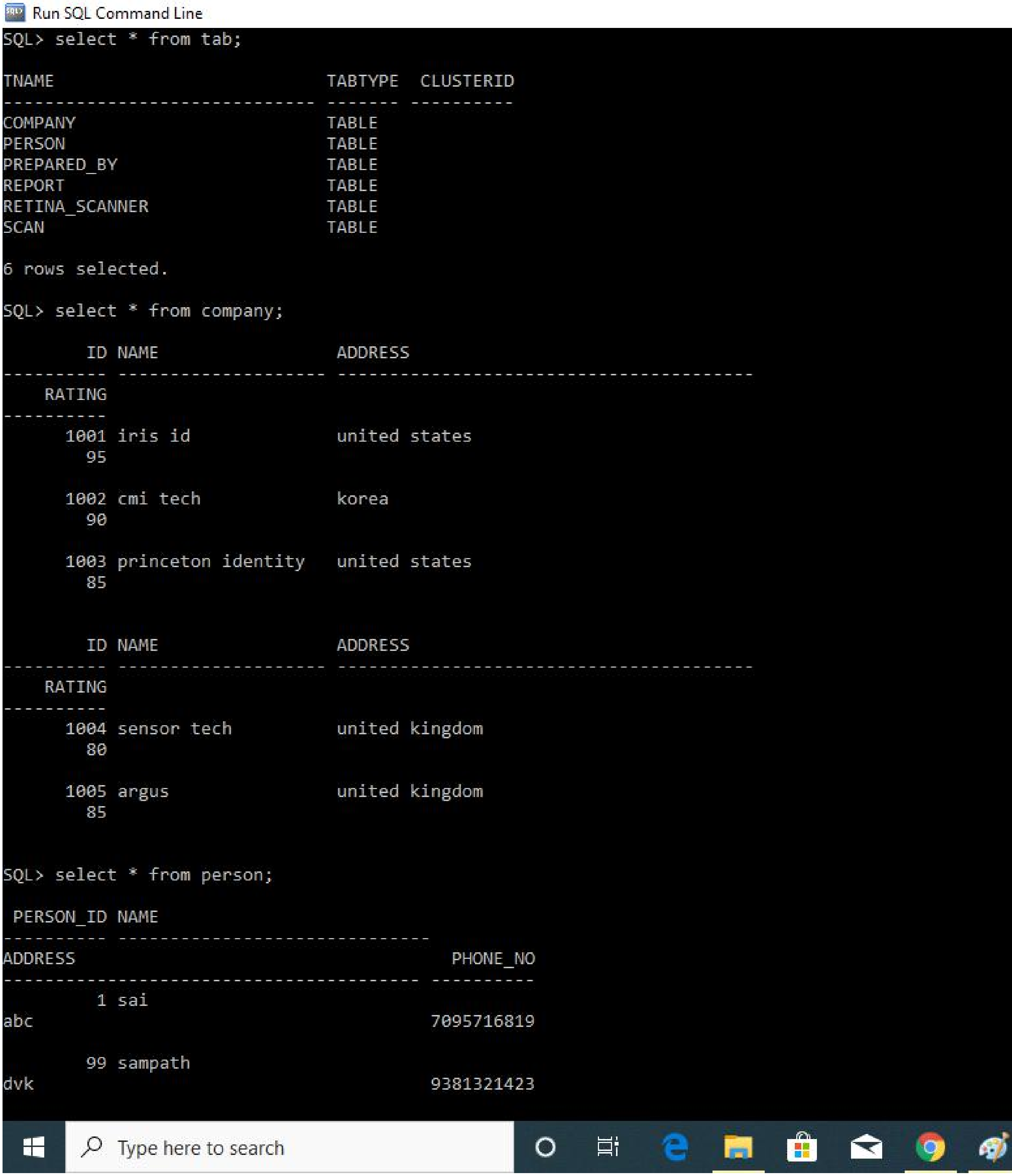
```
insert into prepared_by values('',1,103)
*
ERROR at line 1:
ORA-02291: integrity constraint (ASSINGMENT.SYS_C007039) violated - parent key
not found


SQL> insert into prepared_by values('&date',&s_id,&c_id);
Enter value for date:
Enter value for s_id: 1
Enter value for c_id: 1003
old   1: insert into prepared_by values('&date',&s_id,&c_id)
new   1: insert into prepared_by values('',1,1003)

1 row created.

SQL> insert into prepared_by values('&date',&s_id,&c_id);
Enter value for date:
Enter value for s_id: 2
Enter value for c_id: 1004
old   1: insert into prepared_by values('&date',&s_id,&c_id)
new   1: insert into prepared_by values('',2,1004)

1 row created.

SQL> insert into prepared_by values('&date',&s_id,&c_id);
Enter value for date:
Enter value for s_id: 3
Enter value for c_id: 1002
old   1: insert into prepared_by values('&date',&s_id,&c_id)
new   1: insert into prepared_by values('',3,1002)

1 row created.

SQL> insert into prepared_by values('&date',&s_id,&c_id);
Enter value for date:
Enter value for s_id: 4
Enter value for c_id: 1005
old   1: insert into prepared_by values('&date',&s_id,&c_id)
new   1: insert into prepared_by values('',4,1005)

1 row created.

SQL> insert into prepared_by values('&date',&s_id,&c_id);
Enter value for date:
Enter value for s_id: 5
Enter value for c_id: 1001
old   1: insert into prepared_by values('&date',&s_id,&c_id)
new   1: insert into prepared_by values('',5,1001)

1 row created.
```



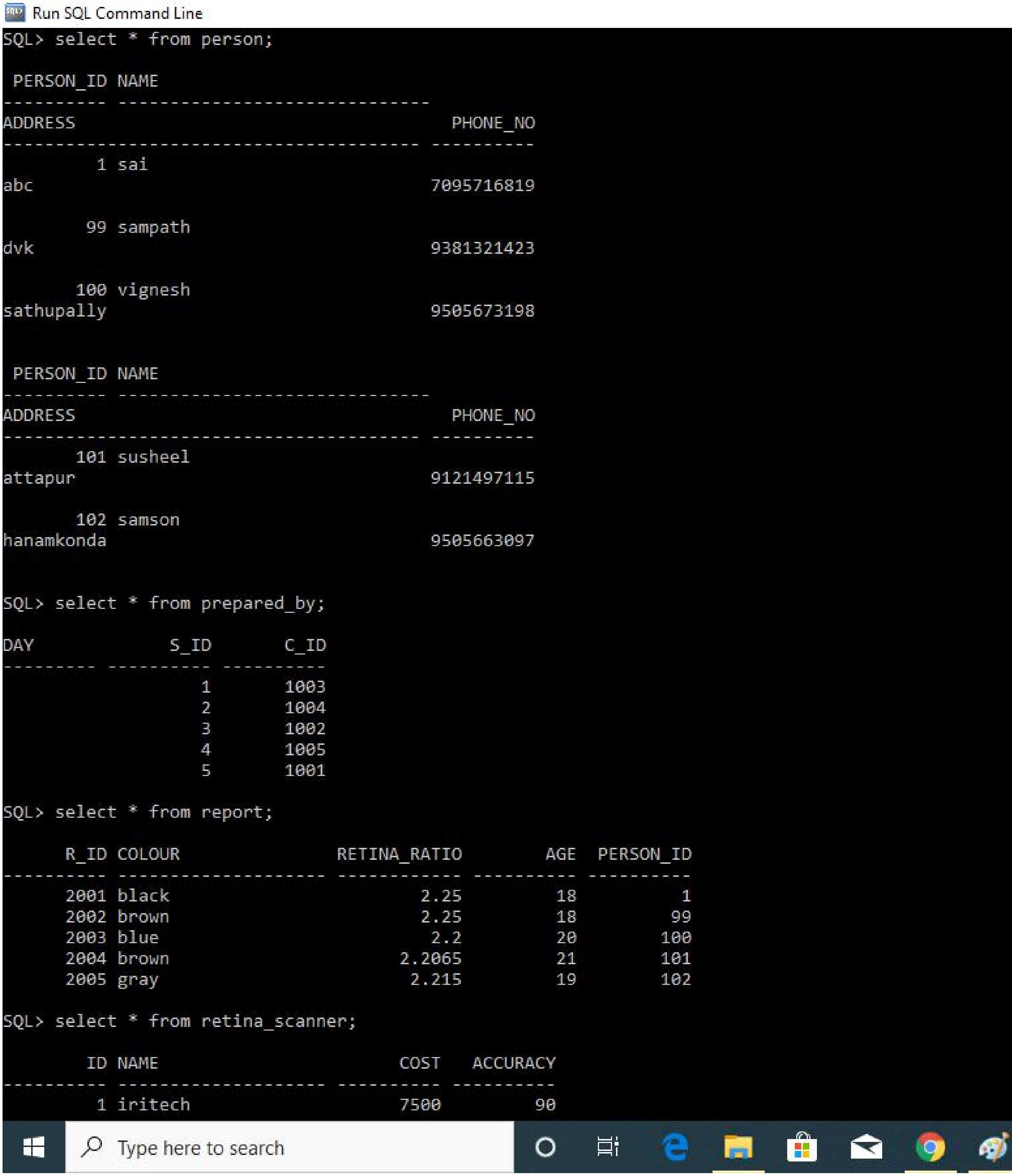
```
old   1: insert into prepared_by values('&date',&s_id,&c_id)
new   1: insert into prepared_by values('',1,1003)

1 row created.

SQL> insert into prepared_by values('&date',&s_id,&c_id);
Enter value for date:
Enter value for s_id: 2
Enter value for c_id: 1004
old   1: insert into prepared_by values('&date',&s_id,&c_id)
new   1: insert into prepared_by values('',2,1004)

1 row created.

SQL> insert into prepared_by values('&date',&s_id,&c_id);
Enter value for date:
Enter value for s_id: 3
Enter value for c_id: 1002
old   1: insert into prepared_by values('&date',&s_id,&c_id)
new   1: insert into prepared_by values('',3,1002)

1 row created.

SQL> insert into prepared_by values('&date',&s_id,&c_id);
Enter value for date:
Enter value for s_id: 4
Enter value for c_id: 1005
old   1: insert into prepared_by values('&date',&s_id,&c_id)
new   1: insert into prepared_by values('',4,1005)

1 row created.

SQL> insert into prepared_by values('&date',&s_id,&c_id);
Enter value for date:
Enter value for s_id: 5
Enter value for c_id: 1001
old   1: insert into prepared_by values('&date',&s_id,&c_id)
new   1: insert into prepared_by values('',5,1001)

1 row created.

SQL> select * from prepared_by;

DAY             S_ID       C_ID
--------- ---------- ----------
                   1       1003
                   2       1004
                   3       1002
                   4       1005
                   5       1001

SQL> _
```



```
old   1: insert into report values(&r_id,'&colour',&retina_ratio,&age,&person_id)
new   1: insert into report values(2003,'blue',2.20,2,)
insert into report values(2003,'blue',2.20,2,)
                                             *
ERROR at line 1:
ORA-00936: missing expression


SQL> insert into report values(&r_id,'&colour',&retina_ratio,&age,&person_id);
Enter value for r_id: 2003
Enter value for colour: blue
Enter value for retina_ratio: 2.20
Enter value for age: 20
Enter value for person_id: 100
old   1: insert into report values(&r_id,'&colour',&retina_ratio,&age,&person_id)
new   1: insert into report values(2003,'blue',2.20,20,100)

1 row created.

SQL> insert into report values(&r_id,'&colour',&retina_ratio,&age,&person_id);
Enter value for r_id: 2004
Enter value for colour: brown
Enter value for retina_ratio: 2.2065
Enter value for age: 21
Enter value for person_id: 101
old   1: insert into report values(&r_id,'&colour',&retina_ratio,&age,&person_id)
new   1: insert into report values(2004,'brown',2.2065,21,101)

1 row created.

SQL> insert into report values(&r_id,'&colour',&retina_ratio,&age,&person_id);
Enter value for r_id: 2005
Enter value for colour: gray
Enter value for retina_ratio: 2.215
Enter value for age: 19
Enter value for person_id: 102
old   1: insert into report values(&r_id,'&colour',&retina_ratio,&age,&person_id)
new   1: insert into report values(2005,'gray',2.215,19,102)

1 row created.

SQL> select * from report;

      R_ID COLOUR               RETINA_RATIO        AGE  PERSON_ID
---------- -------------------- ------------ ---------- ----------
      2001 black                        2.25         18          1
      2002 brown                        2.25         18         99
      2003 blue                          2.2         20        100
      2004 brown                      2.2065         21        101
      2005 gray                        2.215         19        102

SQL>
```

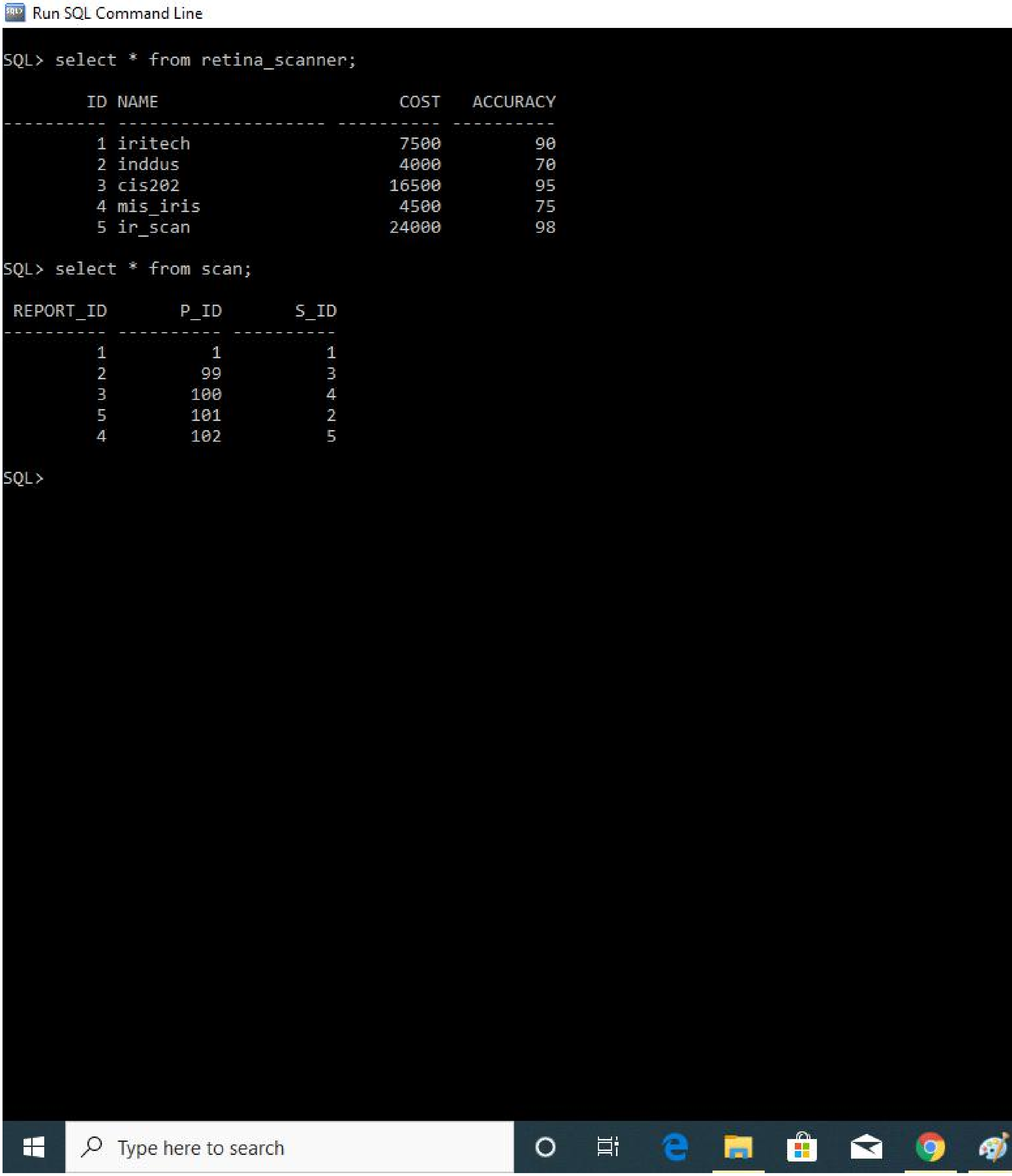


```
SQL> select * from tab;

TNAME                          TABTYPE  CLUSTERID
------------------------------ ------- ----------
COMPANY                        TABLE
PERSON                         TABLE
PREPARED_BY                    TABLE
REPORT                         TABLE
RETINA_SCANNER                 TABLE
SCAN                           TABLE

6 rows selected.

SQL> select * from company;

        ID NAME                 ADDRESS
---------- -------------------- ----------------------------------------
    RATING
----------
      1001 iris id              united states
        95

      1002 cmi tech             korea
        90

      1003 princeton identity   united states
        85


        ID NAME                 ADDRESS
---------- -------------------- ----------------------------------------
    RATING
----------
      1004 sensor tech          united kingdom
        80

      1005 argus                united kingdom
        85


SQL> select * from person;

 PERSON_ID NAME
---------- ------------------------------
ADDRESS                                    PHONE_NO
---------------------------------------- ----------
         1 sai
abc                                      7095716819

        99 sampath
dvk                                      9381321423
```



```
SQL> select * from person;

 PERSON_ID NAME
---------- ------------------------------
ADDRESS                                    PHONE_NO
---------------------------------------- ----------
         1 sai
abc                                      7095716819

        99 sampath
dvk                                      9381321423

       100 vignesh
sathupally                               9505673198


 PERSON_ID NAME
---------- ------------------------------
ADDRESS                                    PHONE_NO
---------------------------------------- ----------
       101 susheel
attapur                                  9121497115

       102 samson
hanamkonda                               9505663097


SQL> select * from prepared_by;

DAY             S_ID       C_ID
--------- ---------- ----------
                   1       1003
                   2       1004
                   3       1002
                   4       1005
                   5       1001

SQL> select * from report;

      R_ID COLOUR               RETINA_RATIO        AGE  PERSON_ID
---------- -------------------- ------------ ---------- ----------
      2001 black                        2.25         18          1
      2002 brown                        2.25         18         99
      2003 blue                          2.2         20        100
      2004 brown                      2.2065         21        101
      2005 gray                        2.215         19        102

SQL> select * from retina_scanner;

        ID NAME                       COST   ACCURACY
---------- -------------------- ---------- ----------
         1 iritech                    7500         90
```

DBMS   ASSIGNMENT -2
Title: Iris based age classification system



```
SQL> select * from retina_scanner;

        ID NAME                           COST   ACCURACY
---------- -------------------- ---------- ----------
         1 iritech                        7500         90
         2 inddus                         4000         70
         3 cis202                        16500         95
         4 mis_iris                       4500         75
         5 ir_scan                       24000         98

SQL> select * from scan;

 REPORT_ID       P_ID       S_ID
---------- ---------- ----------
         1          1          1
         2         99          3
         3        100          4
         5        101          2
         4        102          5

SQL>
```

Name : S.Sai Rohith   Roll Number : 1602-18-737-098

Implementation

- Front end programs:

1)Insert a Company:

```
package company;

import java.awt.*;
import java.awt.event.*;
import java.sql.*;

import javax.swing.JOptionPane;
public class AddCompany extends Panel
{
        Button AddCompanyButton;
        TextField sidText, snameText, ratingText, addressText;
        TextArea errorText;
        Connection connection;
        Statement statement;
        public AddCompany()
        {
                try
                {
                        Class.forName("oracle.jdbc.driver.OracleDriver");
                }
                catch (Exception e)
                {
                        System.err.println("Unable to find and load driver");
                        System.exit(1);
                }
                connectToDB();

        }

        public void connectToDB()
    {
                try
                {
                  connection =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","assingment",
"vasavi");
                  statement = connection.createStatement();
                  statement.executeUpdate("commit");

                }
                catch (SQLException connectException)
                {
                  System.out.println(connectException.getMessage());
                  System.out.println(connectException.getSQLState());
                  System.out.println(connectException.getErrorCode());
```

```
               System.exit(1);
             }
    }
       public void buildGUI()
       {
             //Handle Insert Account Button
             AddCompanyButton = new Button("Add Company");
             AddCompanyButton.addActionListener(new ActionListener()
             {
                    public void actionPerformed(ActionEvent e)
                    {
                          try
                          {

       if(Integer.getInteger(sidText.getText())==null)
                                       throw new NumberFormatException();
                              //Double.getDouble(ratingText.getText());
                            //String query = "INSERT INTO company
(ID,NAME,Address,RATING) VALUES (2,'sai rohith','abc colony',20)";

                            String query= "INSERT INTO company VALUES(" +
sidText.getText() + ", " + "'" + snameText.getText() + "','" + addressText.getText()
+ "'," + ratingText.getText() + ")";
                            int i = statement.executeUpdate(query);
                            statement.executeUpdate("commit");
                            errorText.append("\nInserted " + i + " rows
successfully");
                          }
                          catch (SQLException insertException)
                          {
                            displaySQLErrors(insertException);
                          }
                          catch(Exception ex)
                          {
                                 JOptionPane.showMessageDialog(null, "sid
should be only a number");
                          }
                    }
             });
             sidText=new TextField(15);
             snameText = new TextField(15);
             ratingText = new TextField(15);
             addressText = new TextField(15);


             errorText = new TextArea(10, 40);
             errorText.setEditable(false);

             Panel first = new Panel();
             first.setLayout(new GridLayout(4, 2));
             first.add(new Label("Company ID:"));
             first.add(sidText);
             first.add(new Label("Name:"));
             first.add(snameText);
             first.add(new Label("Rating:"));
             first.add(ratingText);
             first.add(new Label("Address:"));
             first.add(addressText);
             first.setBounds(125,90,200,100);

             Panel second = new Panel(new GridLayout(4, 1));
             second.add(AddCompanyButton);
        second.setBounds(125,220,150,100);

             Panel third = new Panel();
             third.add(errorText);
             third.setBounds(125,320,300,200);

             setLayout(null);

             add(first);
             add(second);
             add(third);
             setSize(500, 600);
             setVisible(true);
             System.out.println("hello");
       }


       private void displaySQLErrors(SQLException e)
       {
             errorText.append("\nSQLException: " + e.getMessage() + "\n");
             errorText.append("SQLState:     " + e.getSQLState() + "\n");
             errorText.append("VendorError:  " + e.getErrorCode() + "\n");
       }
}
```

2)Update a Company:

```
package company;



import java.awt.*;

import java.awt.event.*;

import java.sql.*;
```

```
public class UpdateCompany extends Panel
{
    Button updateCompanyButton;
    List companyIDList;
    TextField sidText, snameText, ratingText, addressText;
    TextArea errorText;
    Connection connection;
    Statement statement;
    ResultSet rs;

    public UpdateCompany()
    {
        try
        {
            Class.forName("oracle.jdbc.driver.OracleDriver");
        }
        catch (Exception e)
        {
            System.err.println("Unable to find and load driver");
            System.exit(1);
        }
        connectToDB();
    }

    public void connectToDB()
{
        try
        {
          connection =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","assingment","vasavi");
          statement = connection.createStatement();


        }
        catch (SQLException connectException)
        {
          System.out.println(connectException.getMessage());
          System.out.println(connectException.getSQLState());
          System.out.println(connectException.getErrorCode());
          System.exit(1);
        }
}


    public void loadCompanies()
    {
        try
        {
            companyIDList.removeAll();
          rs = statement.executeQuery("SELECT ID FROM company");
          while (rs.next())
          {
            companyIDList.add(rs.getString("ID"));
```

```
                    }

                }

                catch (SQLException e)

                {

                    displaySQLErrors(e);

                }

        }


        public void buildGUI()

        {

                companyIDList = new List(10);

                    loadCompanies();

                    add(companyIDList);


                    companyIDList.addItemListener(new ItemListener()

                    {

                            public void itemStateChanged(ItemEvent e)

                            {

                                    try

                                    {

                                            rs = statement.executeQuery("SELECT * FROM company
where ID ="+companyIDList.getSelectedItem());

                                            rs.next();

                                            sidText.setText(rs.getString("ID"));

                                            snameText.setText(rs.getString("NAME"));

                                            ratingText.setText(rs.getString("RATING"));
                    addressText.setText(rs.getString("address"));
                }
                catch (SQLException selectException)
                {
                    displaySQLErrors(selectException);
                }
            }
        });



        updateCompanyButton = new Button("Update Company");
        updateCompanyButton.addActionListener(new ActionListener()
        {
            public void actionPerformed(ActionEvent e)
            {
                try
                {
                    Statement statement = connection.createStatement();
                    int i = statement.executeUpdate("UPDATE company "
                    + "SET name='" + snameText.getText() + "', "
                    + "rating=" + ratingText.getText() + ", "
                    + "address ='"+ addressText.getText() + "' WHERE id = "
                    + companyIDList.getSelectedItem());
                    errorText.append("\nUpdated " + i + " rows successfully");
                    i = statement.executeUpdate("commit");
                    loadCompanies();
                }
                catch (SQLException insertException)
                {
                    displaySQLErrors(insertException);
                }
            }
        });

        sidText = new TextField(15);
        sidText.setEditable(false);
        snameText = new TextField(15);
        ratingText = new TextField(15);
        addressText = new TextField(15);

        errorText = new TextArea(10, 40);
        errorText.setEditable(false);

        Panel first = new Panel();
        first.setLayout(new GridLayout(4, 2));
        first.add(new Label("Company ID:"));
        first.add(sidText);
        first.add(new Label("Name:"));
        first.add(snameText);
```

```
                first.add(new Label("Rating:"));

                first.add(ratingText);

                first.add(new Label("Address:"));

                first.add(addressText);



                Panel second = new Panel(new GridLayout(4, 1));

                second.add(updateCompanyButton);



                Panel third = new Panel();

                third.add(errorText);



                add(first);

                add(second);

                add(third);



                setSize(500, 600);

                setLayout(new FlowLayout());

                setVisible(true);



        }



        private void displaySQLErrors(SQLException e)

        {

                errorText.append("\nSQLException: " + e.getMessage() + "\n");

                errorText.append("SQLState:          " + e.getSQLState() + "\n");
```

```
			errorText.append("VendorError:    " + e.getErrorCode() + "\n");

		}




}
```

3)Delete a Company

```
package company;

import java.awt.*;

import java.awt.event.*;

import java.sql.*;

public class DeleteCompany extends Panel

{

	Button deleteCompanyButton;

	List companiesIDList;

	TextField sidText, snameText, ratingText, addressText;

	TextArea errorText;

	Connection connection;

	Statement statement;

	ResultSet rs;


	public DeleteCompany()

	{
```

```
                try

                {

                        Class.forName("oracle.jdbc.driver.OracleDriver");

                }

                catch (Exception e)

                {

                        System.err.println("Unable to find and load driver");

                        System.exit(1);

                }

                connectToDB();

        }


        public void connectToDB()

     {

                try

                {

                  connection =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","assingment","vasavi");

                  statement = connection.createStatement();


                }

                catch (SQLException connectException)

                {

                  System.out.println(connectException.getMessage());

                  System.out.println(connectException.getSQLState());

                  System.out.println(connectException.getErrorCode());
                System.exit(1);
            }
    }

        public void loadCompanies()
        {
            try
            {

                    companiesIDList.removeAll();
                rs = statement.executeQuery("SELECT * FROM company");
                while (rs.next())
                {
                    companiesIDList.add(rs.getString("ID"));
                }
            }
            catch (SQLException e)
            {
                displaySQLErrors(e);
            }
        }

        public void buildGUI()
        {
            companiesIDList = new List(10);
```

```
                    loadCompanies();

                    add(companiesIDList);



                    //When a list item is selected populate the text fields

                    companiesIDList.addItemListener(new ItemListener()

                    {

                              public void itemStateChanged(ItemEvent e)

                              {

                                        try

                                        {

                                                  rs = statement.executeQuery("SELECT * FROM
company");

                                                  while (rs.next())

                                                  {

                                                            if
(rs.getString("ID").equals(companiesIDList.getSelectedItem()))

                                                            break;

                                                  }

                                                  if (!rs.isAfterLast())

                                                  {

                                                            sidText.setText(rs.getString("ID"));

                                                            snameText.setText(rs.getString("NAME"));

                                                            ratingText.setText(rs.getString("RATING"));

                                                            addressText.setText(rs.getString("Address"));

                                                  }

                                        }
```

```
                catch (SQLException selectException)

                {

                    displaySQLErrors(selectException);

                }

            }

        });




        deleteCompanyButton = new Button("Delete Company");

        deleteCompanyButton.addActionListener(new ActionListener()

        {

            public void actionPerformed(ActionEvent e)

            {

                try

                {

                    Statement statement = connection.createStatement();

                    int i = statement.executeUpdate("DELETE FROM
company WHERE ID = "

                            + companiesIDList.getSelectedItem());

                    errorText.append("\nDeleted " + i + " rows
successfully");

                    sidText.setText(null);

                    snameText.setText(null);

                    ratingText.setText(null);

                    addressText.setText(null);

                    statement.executeUpdate("commit");
                    loadCompanies();
                }
                catch (SQLException insertException)
                {
                    displaySQLErrors(insertException);
                }
            }
        });

        sidText = new TextField(15);
        snameText = new TextField(15);
        ratingText = new TextField(15);
        addressText = new TextField(15);

        errorText = new TextArea(10, 40);
        errorText.setEditable(false);

        Panel first = new Panel();
        first.setLayout(new GridLayout(4, 2));
        first.add(new Label("Company ID:"));
        first.add(sidText);
        first.add(new Label("Name:"));
        first.add(snameText);
        first.add(new Label("Rating:"));
        first.add(ratingText);
        first.add(new Label("Address:"));
        first.add(addressText);

        Panel second = new Panel(new GridLayout(4, 1));
        second.add(deleteCompanyButton);

        Panel third = new Panel();
        third.add(errorText);

        add(first);
        add(second);
        add(third);

        setSize(450, 600);
        setLayout(new FlowLayout());
        setVisible(true);

    }



    private void displaySQLErrors(SQLException e)
    {
        errorText.append("\nSQLException: " + e.getMessage() + "\n");
        errorText.append("SQLState:          " + e.getSQLState() + "\n");
```

```
                    errorText.append("VendorError:    " + e.getErrorCode() + "\n");

          }




}
```

4) Main FrontEndApplication

```
import java.awt.*;

import java.awt.event.*;


import javax.swing.*;

import javax.swing.JOptionPane;

import company.*;

import person.AddPerson;

import person.DeletePerson;

import person.UpdatePerson;

import report.AddReport;

import report.DeleteReport;

import report.UpdateReport;

import scanner.AddScanner;
import scanner.DeleteScanner;
import scanner.UpdateScanner;
;class frontPage extends JFrame implements ActionListener
{
        String msg = "";
        Label ll;
        CardLayout cardLO;


        AddCompany add;
        UpdateCompany ups;
        DeleteCompany dels;
        AddPerson addP;
        UpdatePerson upP;
        DeletePerson delP;
        AddScanner addS;
        UpdateScanner upS;
        DeleteScanner delS;
        AddReport addR;
        DeleteReport delR;
        UpdateReport upR;


        Panel home,welcome;


        public frontPage()
        {
                cardLO = new CardLayout();
```

```
                        home = new Panel();

                        home.setLayout(cardLO);




                        ll = new Label();

                        ll.setAlignment(Label.CENTER);

                        ll.setText("Welcome to Retina based age classigication System 
database");


                        welcome = new Panel();

                        welcome.add(ll);


                        add=new AddCompany();add.buildGUI();

                        ups = new UpdateCompany();    ups.buildGUI();

                        dels = new DeleteCompany();   dels.buildGUI();

                        addP = new AddPerson();addP.buildGUI();

                        upP = new UpdatePerson();upP.buildGUI();

                        delP=new DeletePerson();delP.buildGUI();

                        addS=new AddScanner();addS.buildGUI();

                        upS=new UpdateScanner();upS.buildGUI();

                        delS=new DeleteScanner();delS.buildGUI();

                        addR=new AddReport();addR.buildGUI();

                        delR=new DeleteReport();delR.buildGUI();

                        upR=new UpdateReport();upR.buildGUI();
```

```
home.add(welcome, "Welcome");
home.add(add, "Add Company");
home.add(ups, "Update Company");
home.add(dels, "Delete Company");
home.add(addP, "Add Person");
home.add(upP, "Update Person");
home.add(delP,"Delete Person");
home.add(addS,"Add Scanner");
home.add(upS,"Update Scanner");
home.add(delS,"Delete Scanner");
home.add(addR,"Add Report");
home.add(delR,"Delete Report");
home.add(upR,"Update Report");


add(home);


MenuBar mbar = new MenuBar();
setMenuBar(mbar);


Menu Compnay = new Menu("Company ");
MenuItem item1, item2, item3;
Compnay.add(item1 = new MenuItem("Add Company"));
Compnay.add(item2 = new MenuItem("View Company"));
Compnay.add(item3 = new MenuItem("Delete Company"));
mbar.add(Compnay);
```

```
Menu Scanner = new Menu("Scanner");

MenuItem item4, item5, item6;

Scanner.add(item4 = new MenuItem("Add Scanner"));

Scanner.add(item5 = new MenuItem("View Scanners"));

Scanner.add(item6 = new MenuItem("Delete Scanner"));

mbar.add(Scanner);


Menu Person = new Menu("Person");

MenuItem item7, item8, item9;

Person.add(item7 = new MenuItem("Add Person"));

Person.add(item8 = new MenuItem("View Persons"));

Person.add(item9 = new MenuItem("Delete Person"));

mbar.add(Person);


Menu Report = new Menu("Report");

MenuItem item10, item11, item12;

Report.add(item10 = new MenuItem("Add Report"));

Report.add(item11 = new MenuItem("View Report"));

Report.add(item12 = new MenuItem("Delete Report"));

mbar.add(Report);


item1.addActionListener(this);

item2.addActionListener(this);

item3.addActionListener(this);

item4.addActionListener(this);

item5.addActionListener(this);
```

```
			item6.addActionListener(this);

			item7.addActionListener(this);

			item8.addActionListener(this);

			item9.addActionListener(this);

			item10.addActionListener(this);

			item11.addActionListener(this);

			item12.addActionListener(this);



			addWindowListener(new WindowAdapter(){

				public void windowClosing(WindowEvent we)

				{

					quitApp();

				}

			});


			setTitle("Retina Based Age Classification System");

			setSize(500, 600);

			setVisible(true);


	}


	public void actionPerformed(ActionEvent ae)

	{

		String arg = ae.getActionCommand();

		if(arg.equals("Add Company"))
```

```
                 {

                        cardLO.show(home, "Add Company");

            }


                 else if(arg.equals("View Company"))

                 {

                        cardLO.show(home, "Update Company");

                        ups.loadCompanies();

                 }


                 else if(arg.equals("Delete Company"))

                 {

                        cardLO.show(home, "Delete Company");

                        dels.loadCompanies();

                 }


                 else if(arg.equals("Add Person"))

                 {

                        cardLO.show(home, "Add Person");

                 }

                 else if(arg.equals("View Persons"))

                 {

                        cardLO.show(home, "Update Person");

                        upP.loadPersons();

                 }

                 else if(arg.equals("Delete Person"))
```

```
{

            cardLO.show(home, "Delete Person");

            delP.loadPersons();

}

else if(arg.equals("Add Scanner"))

{

            cardLO.show(home, "Add Scanner");

}

else if(arg.equals("View Scanners"))

{

            cardLO.show(home, "Update Scanner");

            upS.loadScanners();

}

else if(arg.equals("Delete Scanner"))

{

            cardLO.show(home, "Delete Scanner");

            delS.loadScanners();

}

else if(arg.equals("Add Report"))

{

            cardLO.show(home, "Add Report");

}

else if(arg.equals("Delete Report"))

{

            cardLO.show(home, "Delete Report");

            delR.loadReports();
```

```
                    }
                    else if(arg.equals("View Report"))
                    {
                            cardLO.show(home, "Update Report");
                            upR.loadReports();
                    }



            }
            private void quitApp () {


                            try {
                                            int reply = JOptionPane.showConfirmDialog (this,
                                                    "Are you really want to exit\nFrom ReTina
Scanner System?",
                                                    "RetinaSystem - Exit",
JOptionPane.YES_NO_OPTION, JOptionPane.PLAIN_MESSAGE);


                                            if (reply == JOptionPane.YES_OPTION) {
                                            setVisible (false);
                                            dispose();
                                            System.out.println ("Thanks for Using Retina based
age classification System\nAuthor - sai rohith sheela");
                                            System.exit (0);
                                            }
                                    else if (reply == JOptionPane.NO_OPTION) {

        setDefaultCloseOperation(JFrame.DO_NOTHING_ON_CLOSE);
```

```
                        }

                }

                catch (Exception e) {}

        }

        public static void main(String ... args)

        {

                new frontPage();

        }

}
```

Connectivity with the Database:

Java Database Connectivity (JDBC) is an application programming interface (API) for the programming language Java, which defines how a client may access a database. It is a Java-based data access technology used for Java database connectivity. It is part of the Java Standard Edition platform, from Oracle Corporation. It provides methods to query and update data in a database and is oriented towards relational databases.

Block of code for JAVA- SQL connectivity with JDBC:

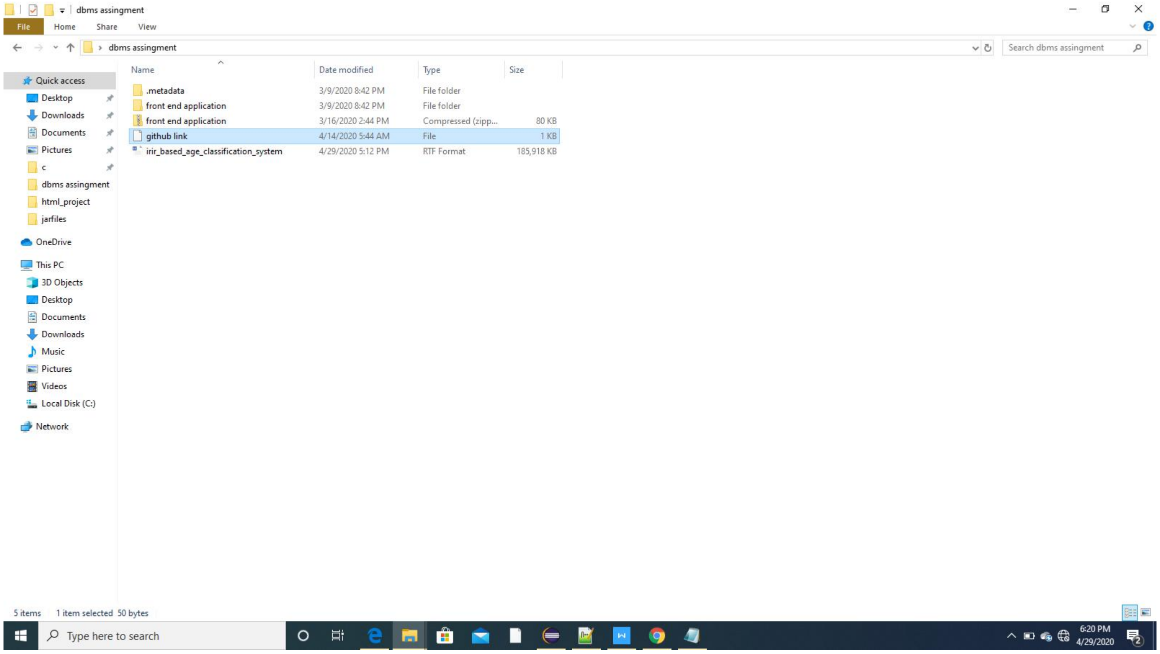
```
public void connectToDB()
    {
             try
             {
               connection =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","assingment",
"vasavi");
               statement = connection.createStatement();

             }
             catch (SQLException connectException)
             {
               System.out.println(connectException.getMessage());
               System.out.println(connectException.getSQLState());
               System.out.println(connectException.getErrorCode());
               System.exit(1);
             }

    }
```

GITHUB LINK:

https://github.com/sheelasairohith/dbma_assingment

Folder Structure:

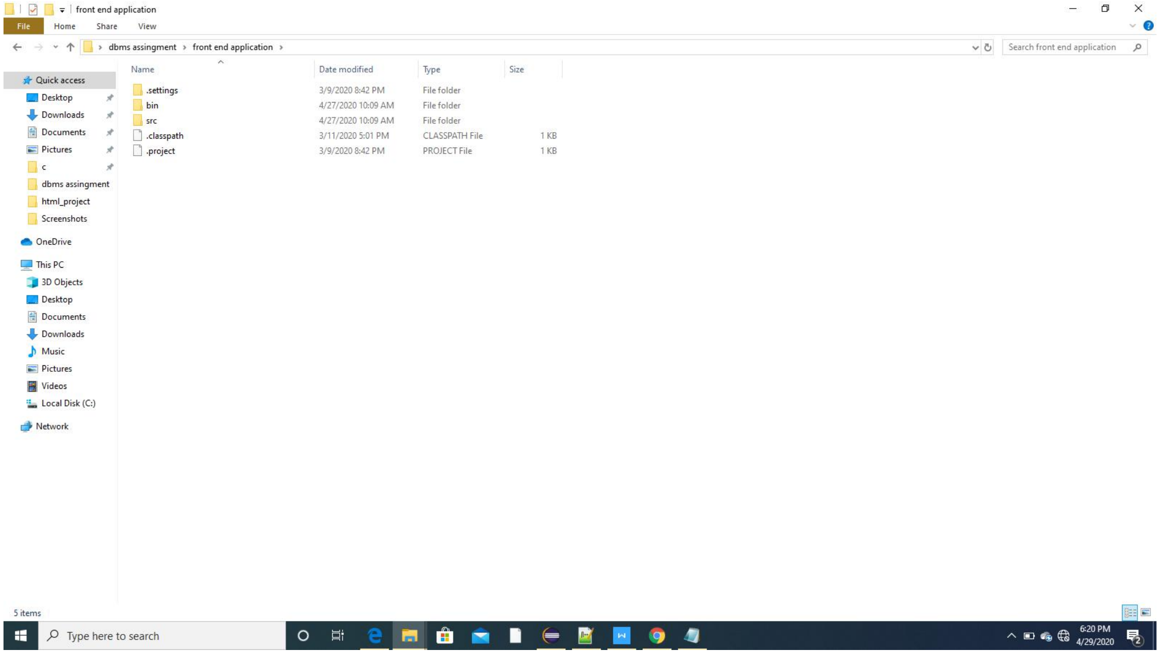
This project contains a folder named src in which it has 1 .java files which belongs to main front end application and this src folder contains 4 additional folders where each folder contains code to insert, delete, update for erquried tables present in the backend. By which we can navigate easily to reach the java code and we can make changes easily.



Name	Date modified	Type	Size
.metadata	3/9/2020 8:42 PM	File folder	
front end application	3/9/2020 8:42 PM	File folder	
front end application	3/16/2020 2:44 PM	Compressed (zipp...	80 KB
github link	4/14/2020 5:44 AM	File	1 KB
irir_based_age_classification_system	4/29/2020 5:12 PM	RTF Format	185,918 KB

5 items   1 item selected 50 bytes

NOTE: in this select front end application folder to see next page.

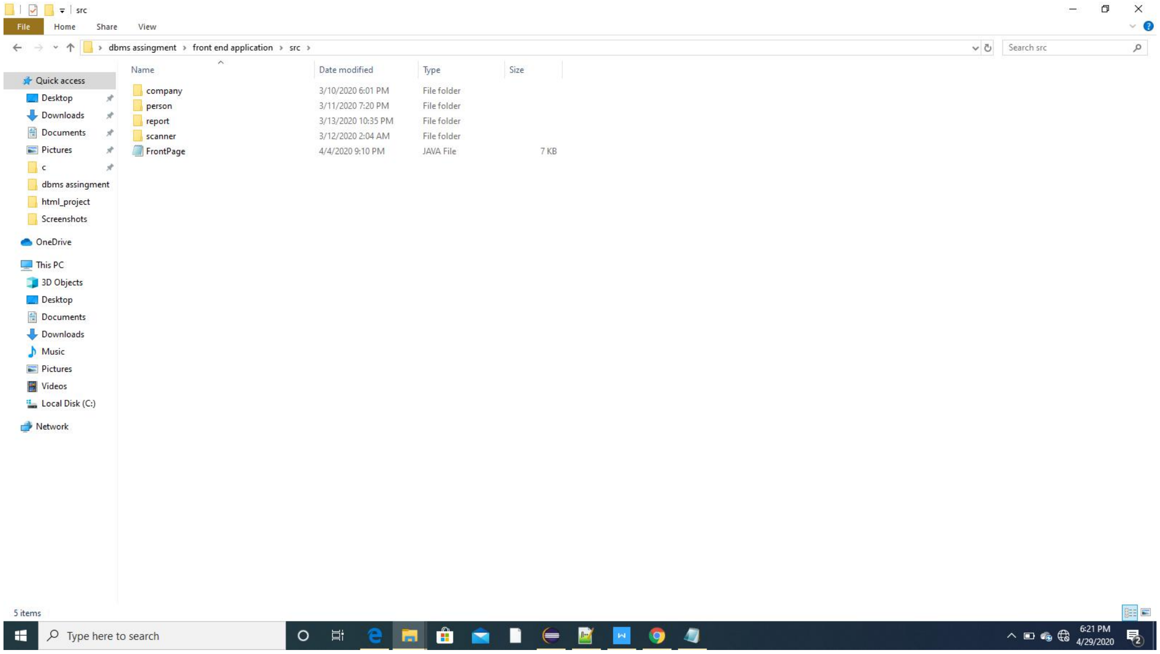


Name	Date modified	Type	Size
.settings	3/9/2020 8:42 PM	File folder	
bin	4/27/2020 10:09 AM	File folder	
src	4/27/2020 10:09 AM	File folder	
.classpath	3/11/2020 5:01 PM	CLASSPATH File	1 KB
.project	3/9/2020 8:42 PM	PROJECT File	1 KB

5 items

NOTE: now select src to view below folders.

DBMS ASSIGNMENT -2
Title: Iris based age classification system

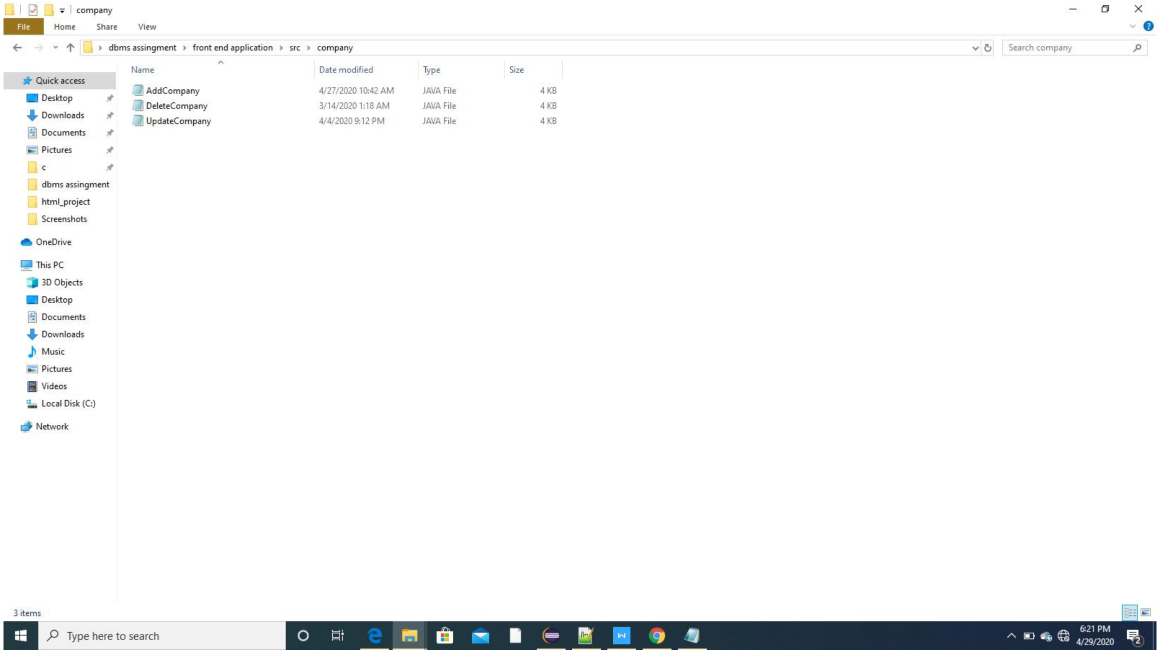


› dbms assingment › front end application › src ›

Name	Date modified	Type	Size
company	3/10/2020 6:01 PM	File folder	
person	3/11/2020 7:20 PM	File folder	
report	3/13/2020 10:35 PM	File folder	
scanner	3/12/2020 2:04 AM	File folder	
FrontPage	4/4/2020 9:10 PM	JAVA File	7 KB

5 items

NOTE: in this each folder contains 3 .java files for example if you select folder called company then it will be like..



› dbms assingment › front end application › src › company

Name	Date modified	Type	Size
AddCompany	4/27/2020 10:42 AM	JAVA File	4 KB
DeleteCompany	3/14/2020 1:18 AM	JAVA File	4 KB
UpdateCompany	4/4/2020 9:12 PM	JAVA File	4 KB

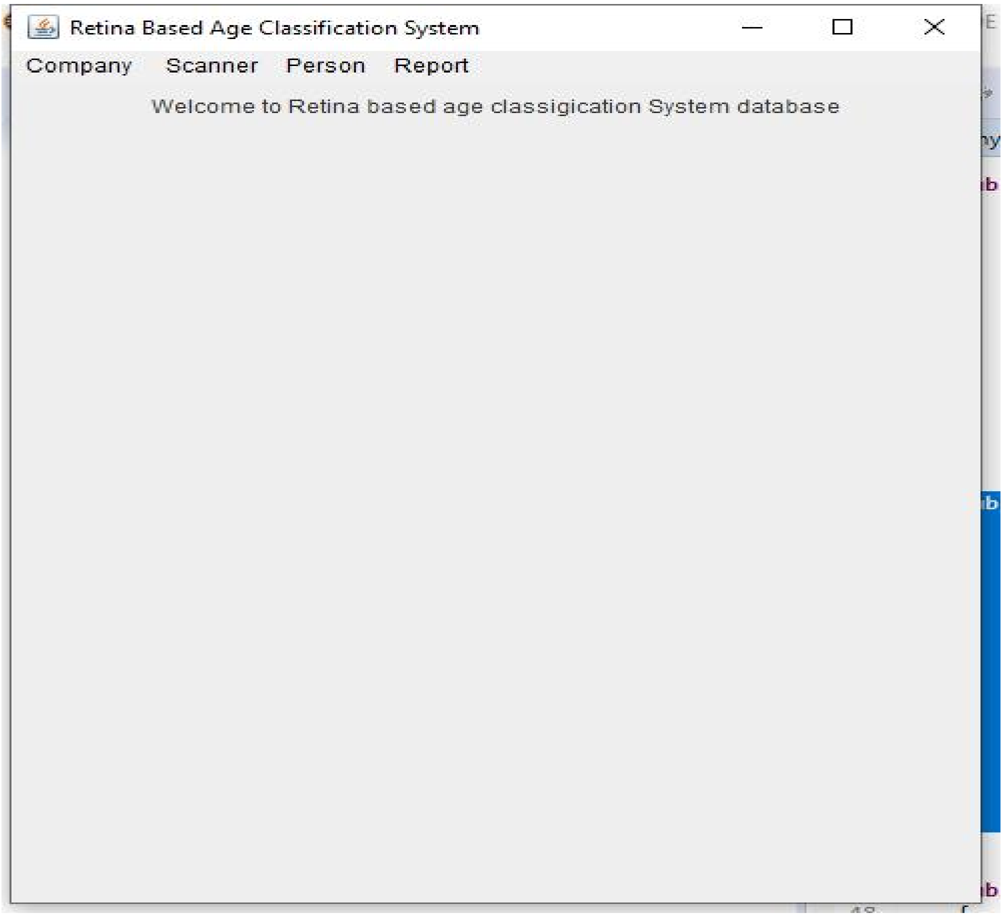
3 items

Name : S.Sai Rohith Roll Number : 1602-18-737-098

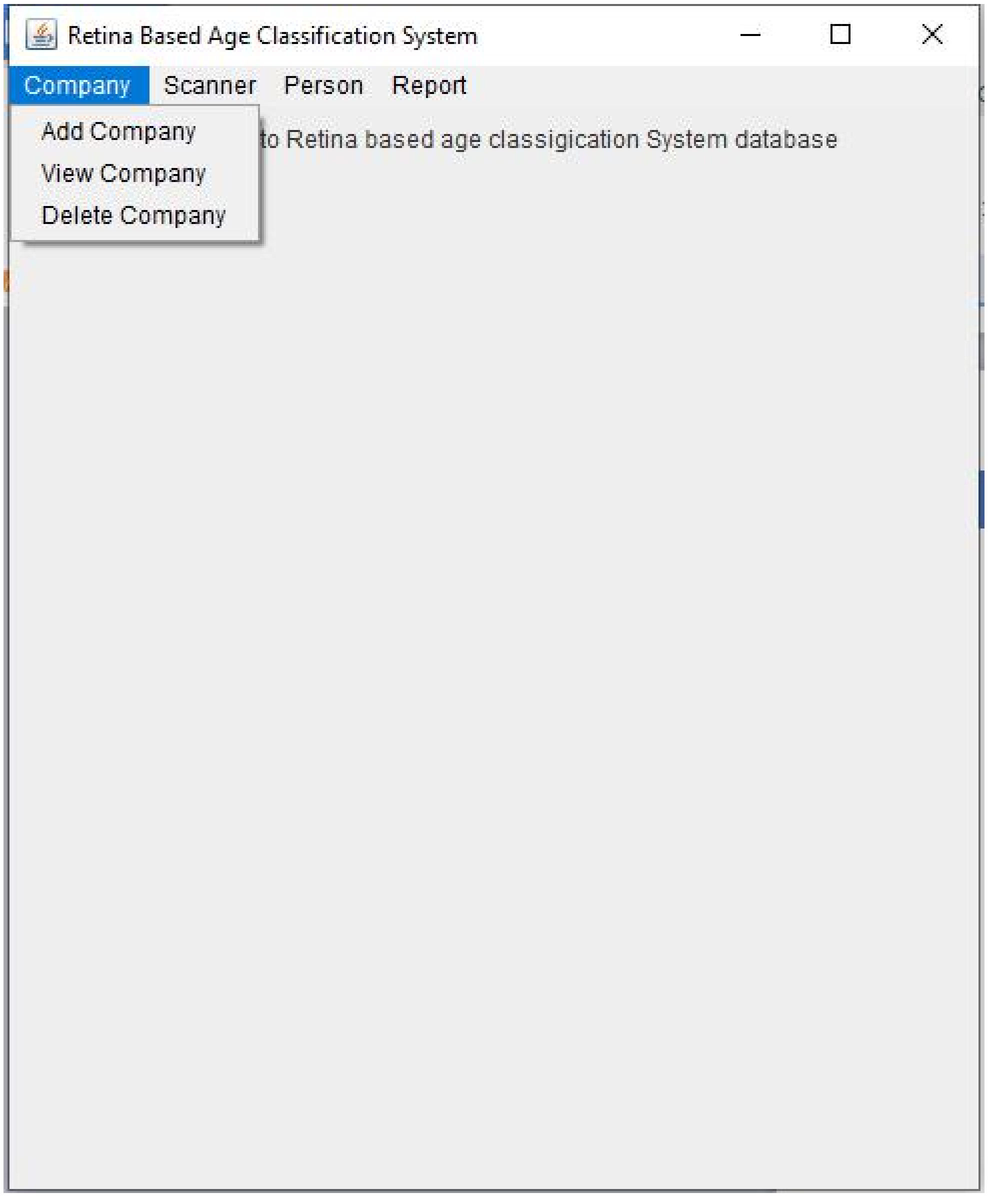
TESTING

The program runs for execution of three basic operations of insertion, update and delete on 5 different table. Along with this, it also has a output column which gives the information about how many rows have been edited. Errors, syntactical or exceptional will be shown if occurred.

HOME PAGE:

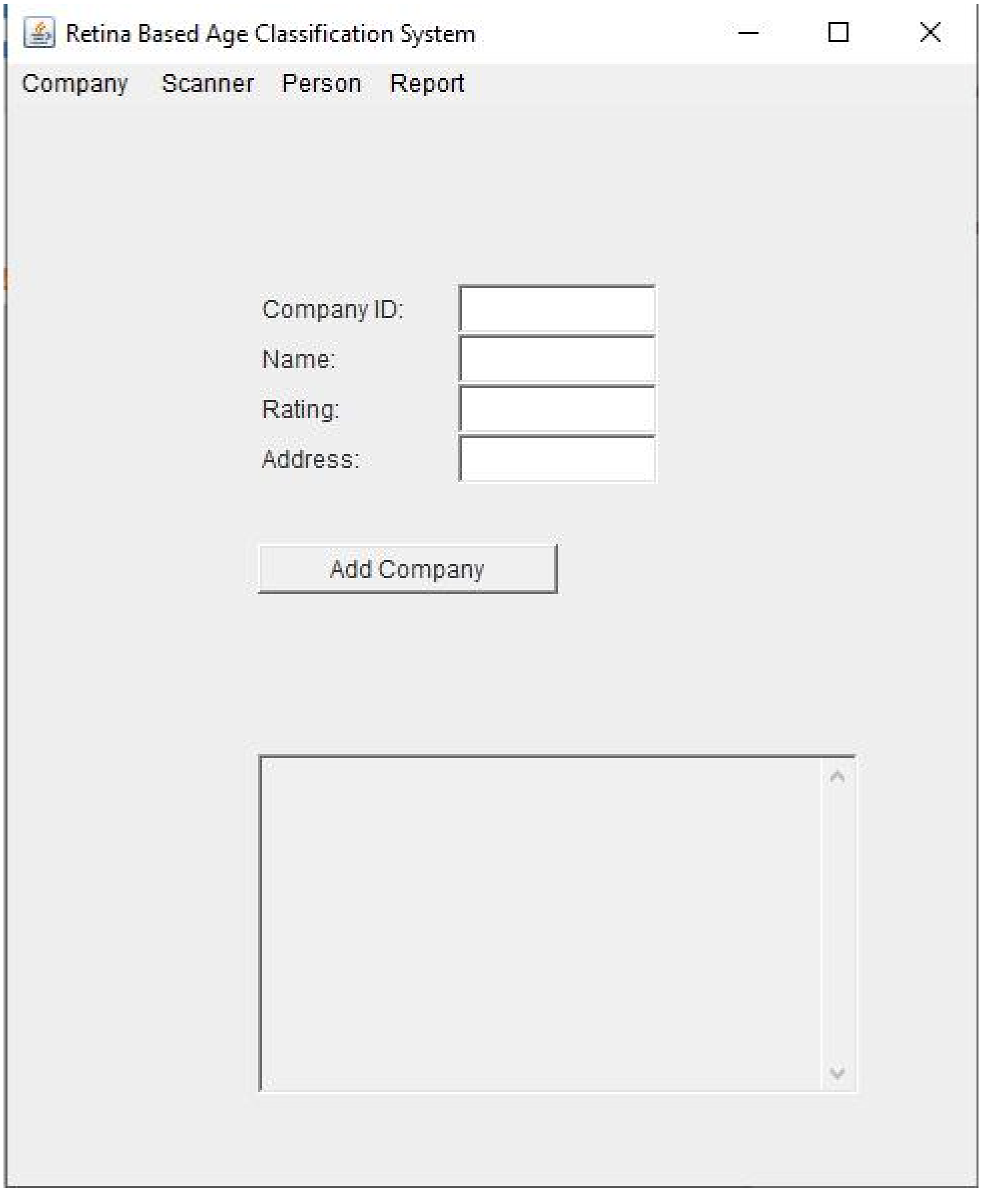


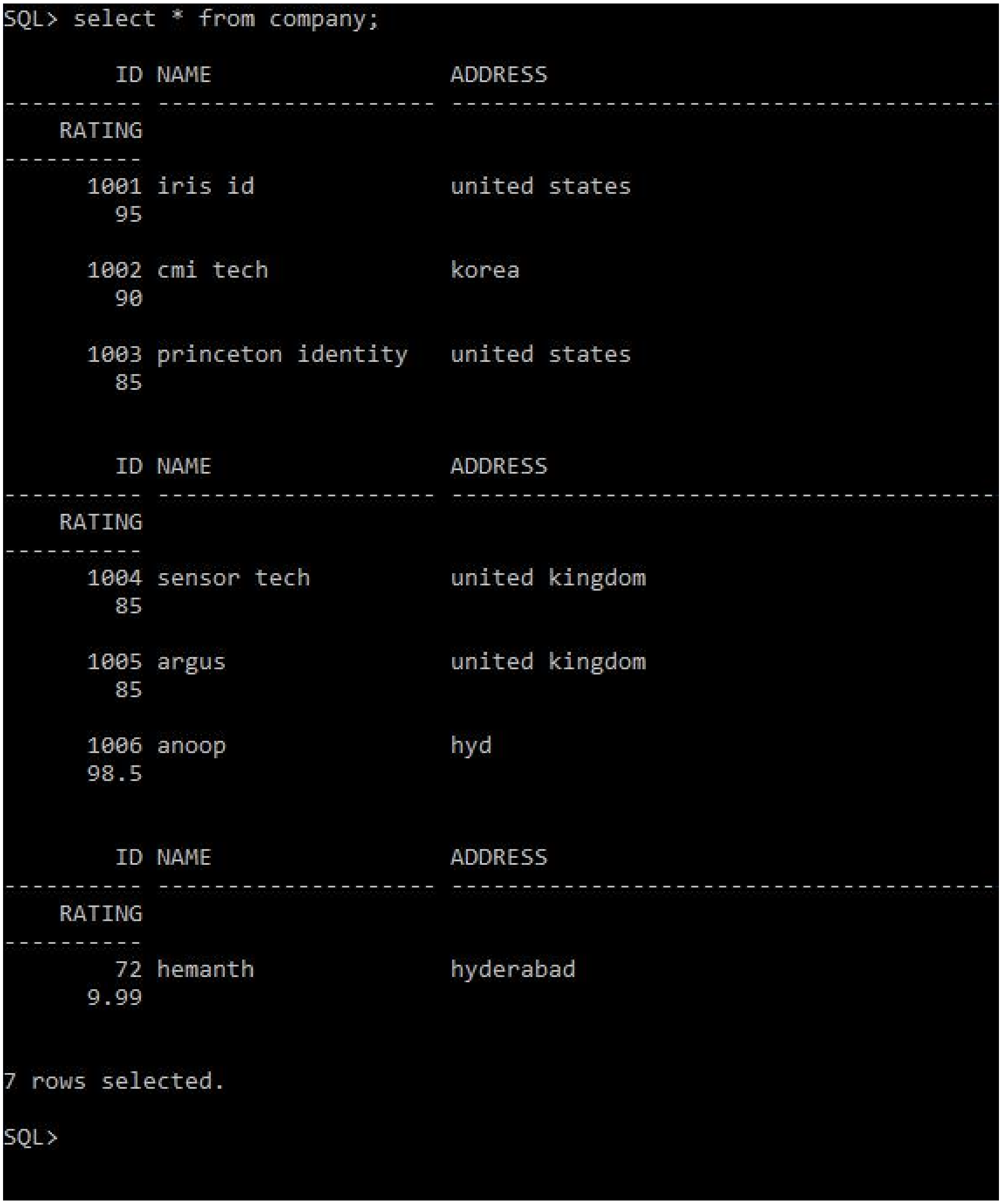
Welcome to Retina based age classigication System database



to Retina based age classigication System database

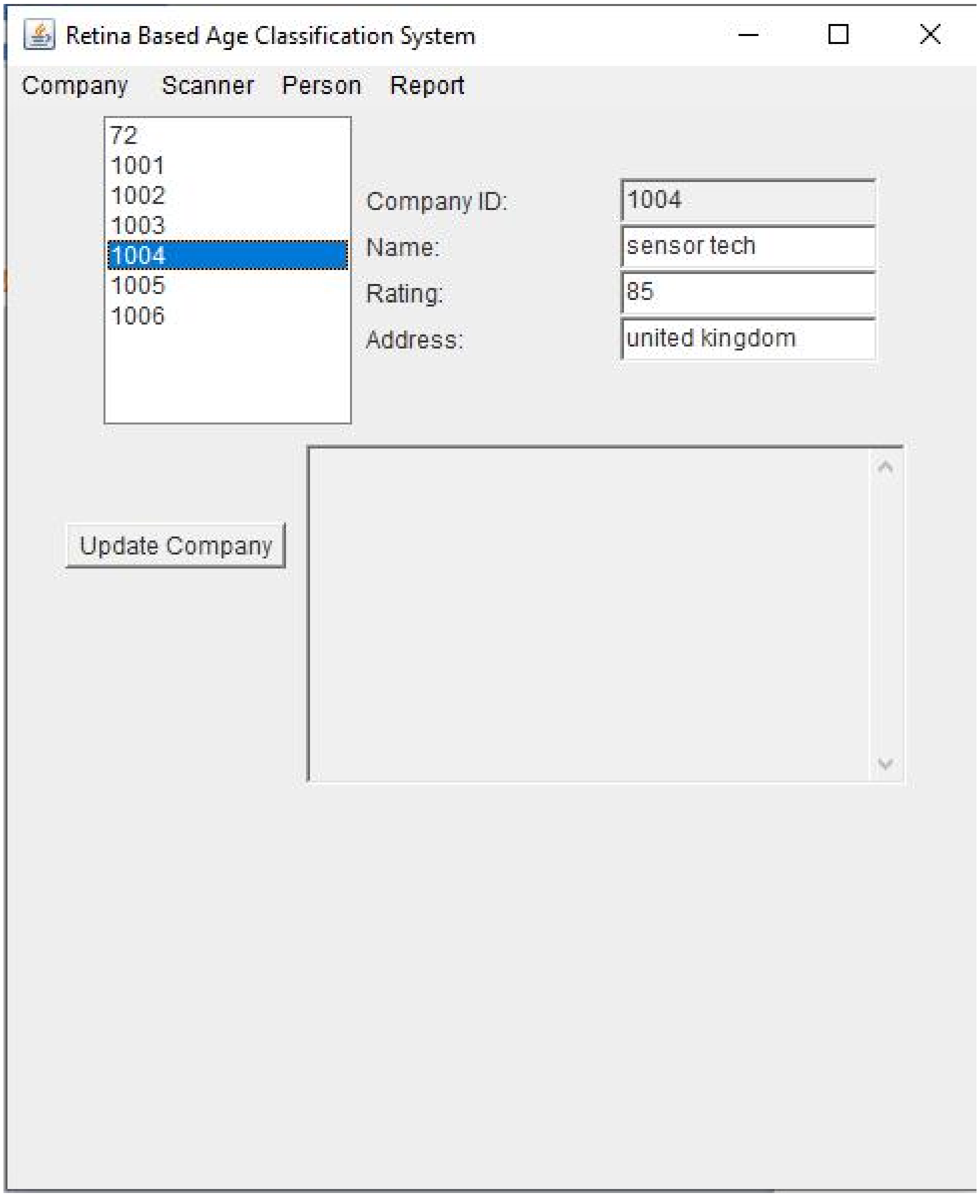
INSERT Company:


Retina Based Age Classification System
Company
Scanner
Person
Report
Company ID:
Name:
Rating:
Address:
Add Company

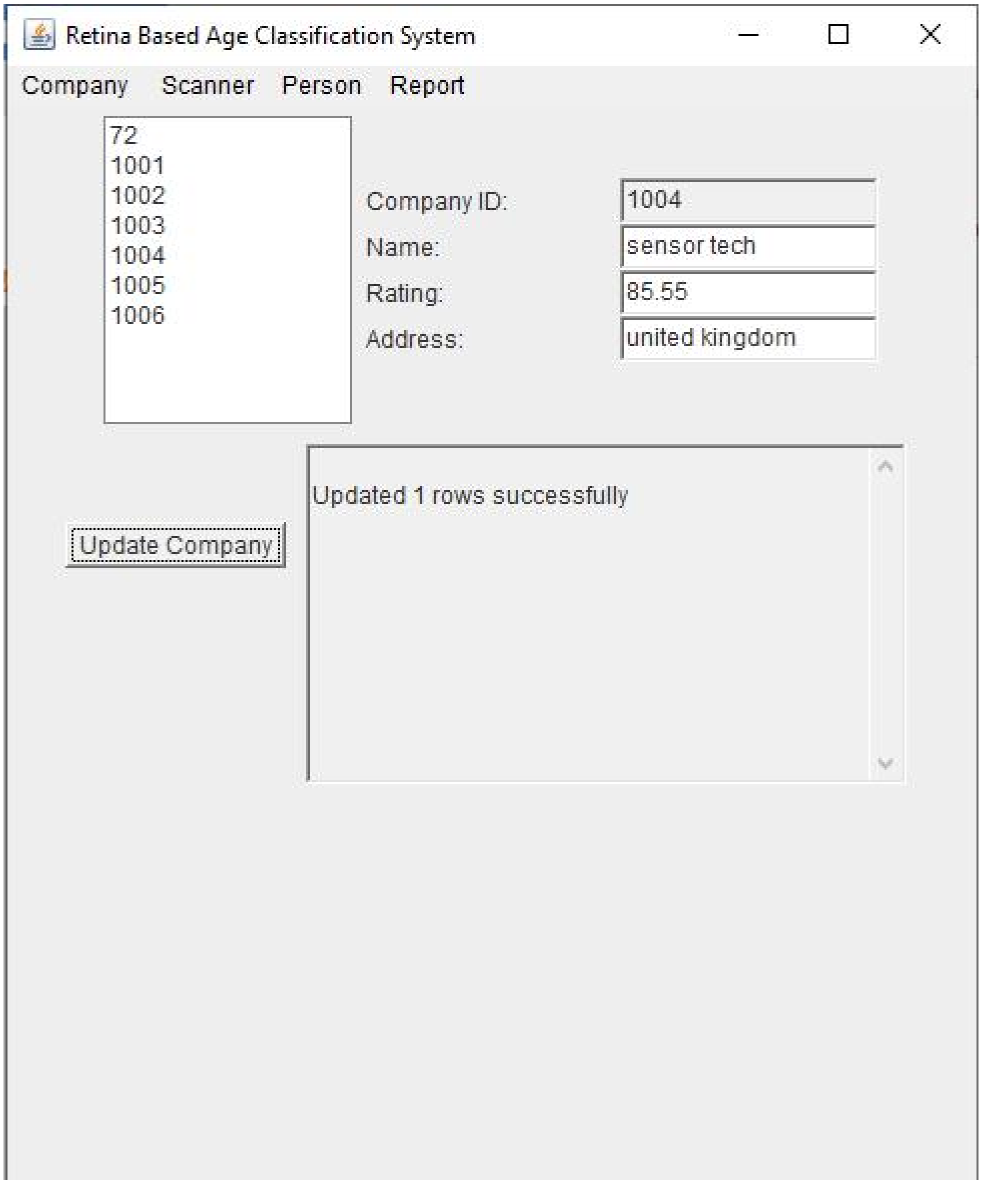


```
SQL> select * from company;

        ID NAME                 ADDRESS
---------- -------------------- ----------------------------------------
    RATING
----------
      1001 iris id              united states
        95

      1002 cmi tech             korea
        90

      1003 princeton identity   united states
        85


        ID NAME                 ADDRESS
---------- -------------------- ----------------------------------------
    RATING
----------
      1004 sensor tech          united kingdom
        85

      1005 argus                united kingdom
        85

      1006 anoop                hyd
      98.5


        ID NAME                 ADDRESS
---------- -------------------- ----------------------------------------
    RATING
----------
        72 hemanth              hyderabad
      9.99


7 rows selected.

SQL>
```

UPDATE STUDENT:



Company ID: 1004
Name: sensor tech
Rating: 85
Address: united kingdom



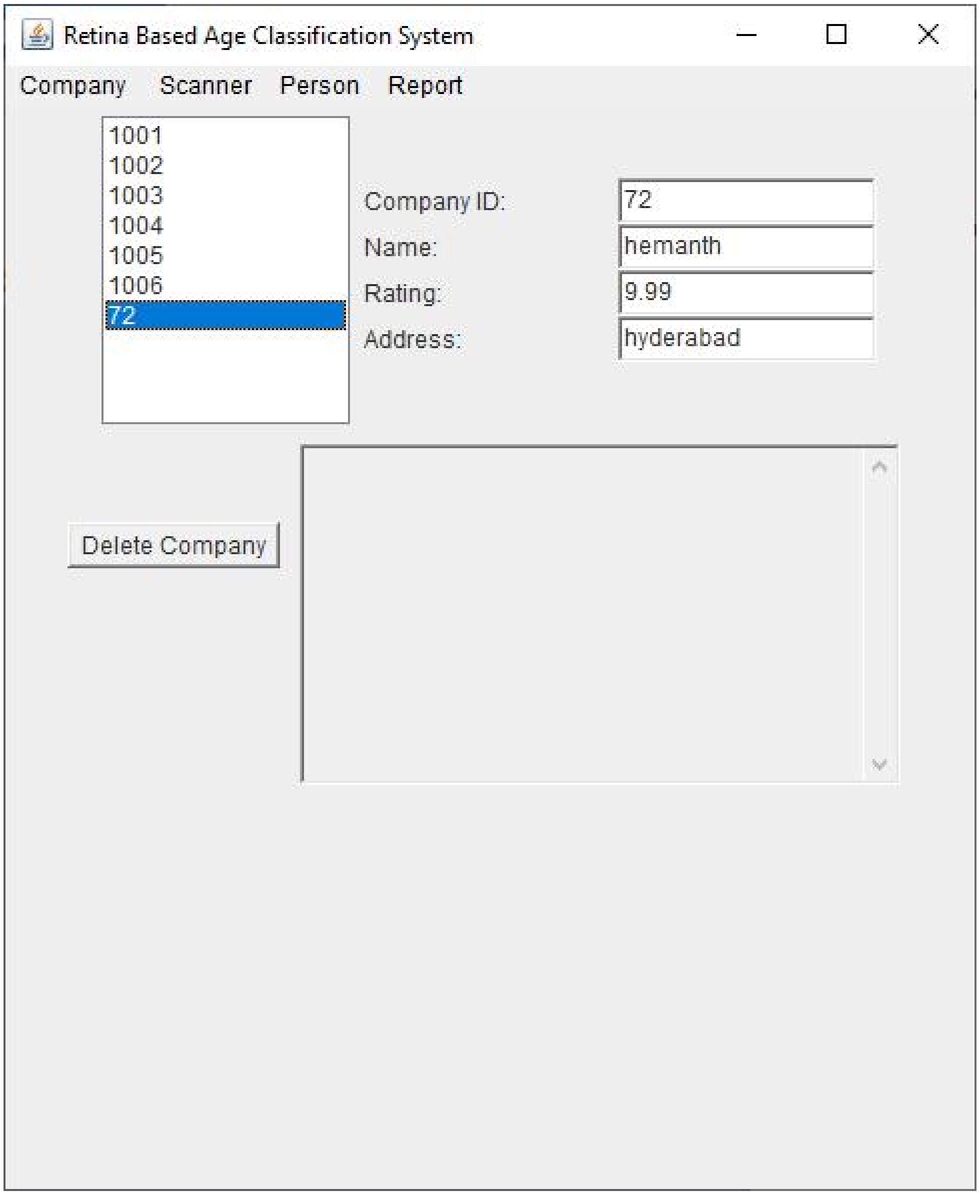
72
1001
1002
1003
1004
1005
1006

Company ID: 1004
Name: sensor tech
Rating: 85.55
Address: united kingdom

```
Updated 1 rows successfully
```

DELETE STUDENT:

Before selecting delete company button:



1001
1002
1003
1004
1005
1006
72

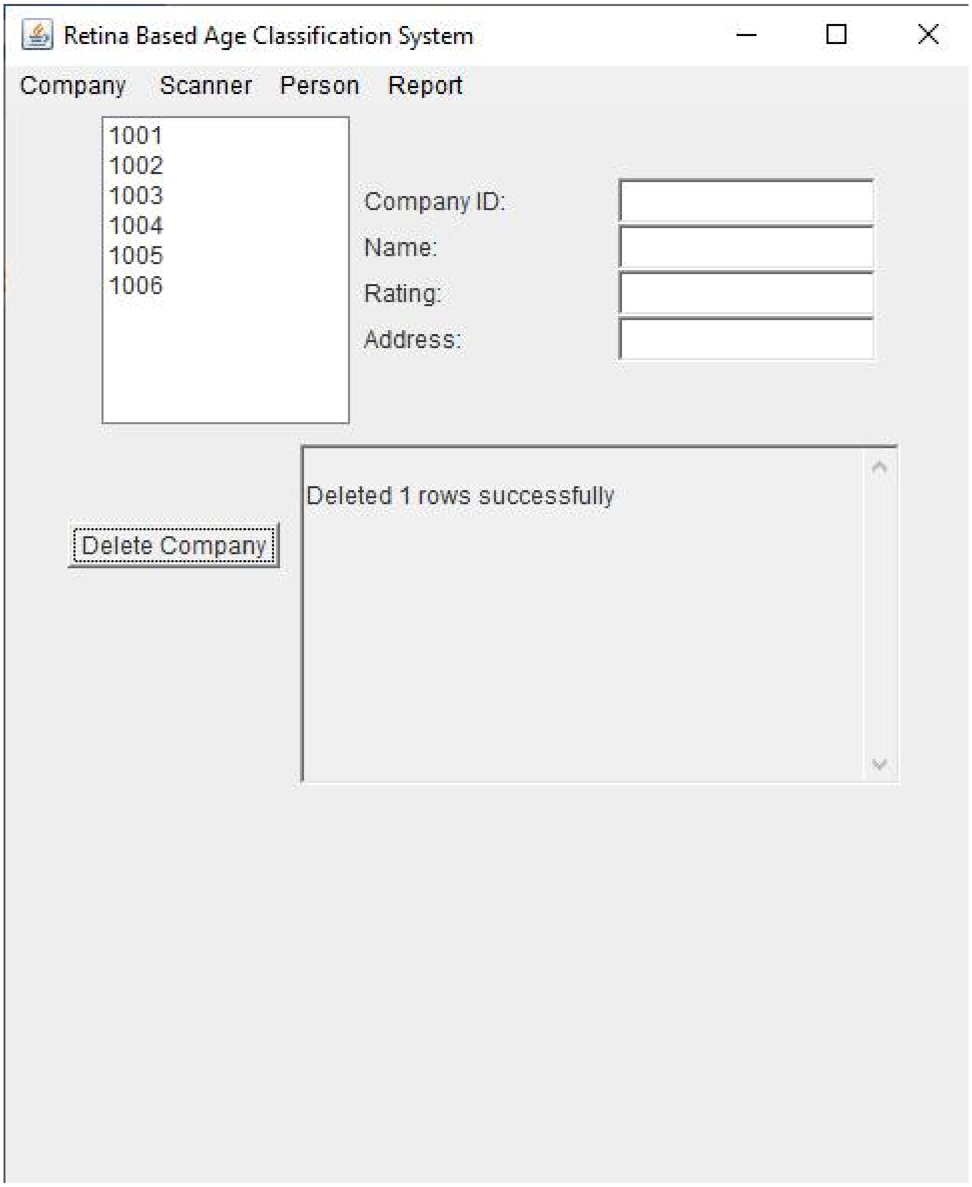
Company ID: 72

Name: hemanth

Rating: 9.99

Address: hyderabad

After selecting delete company button:



1001
1002
1003
1004
1005
1006

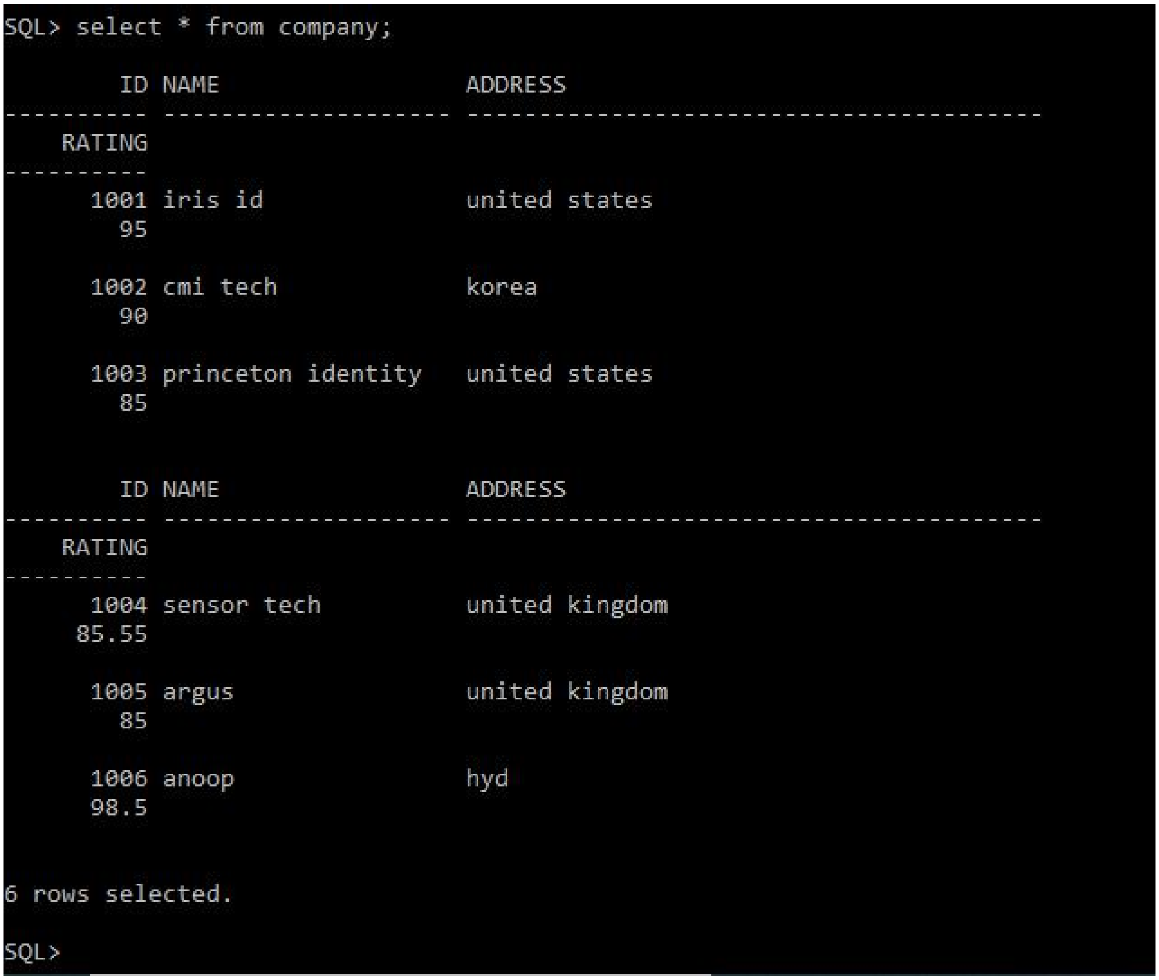
Company ID:

Name:

Rating:

Address:

```
Deleted 1 rows successfully
```

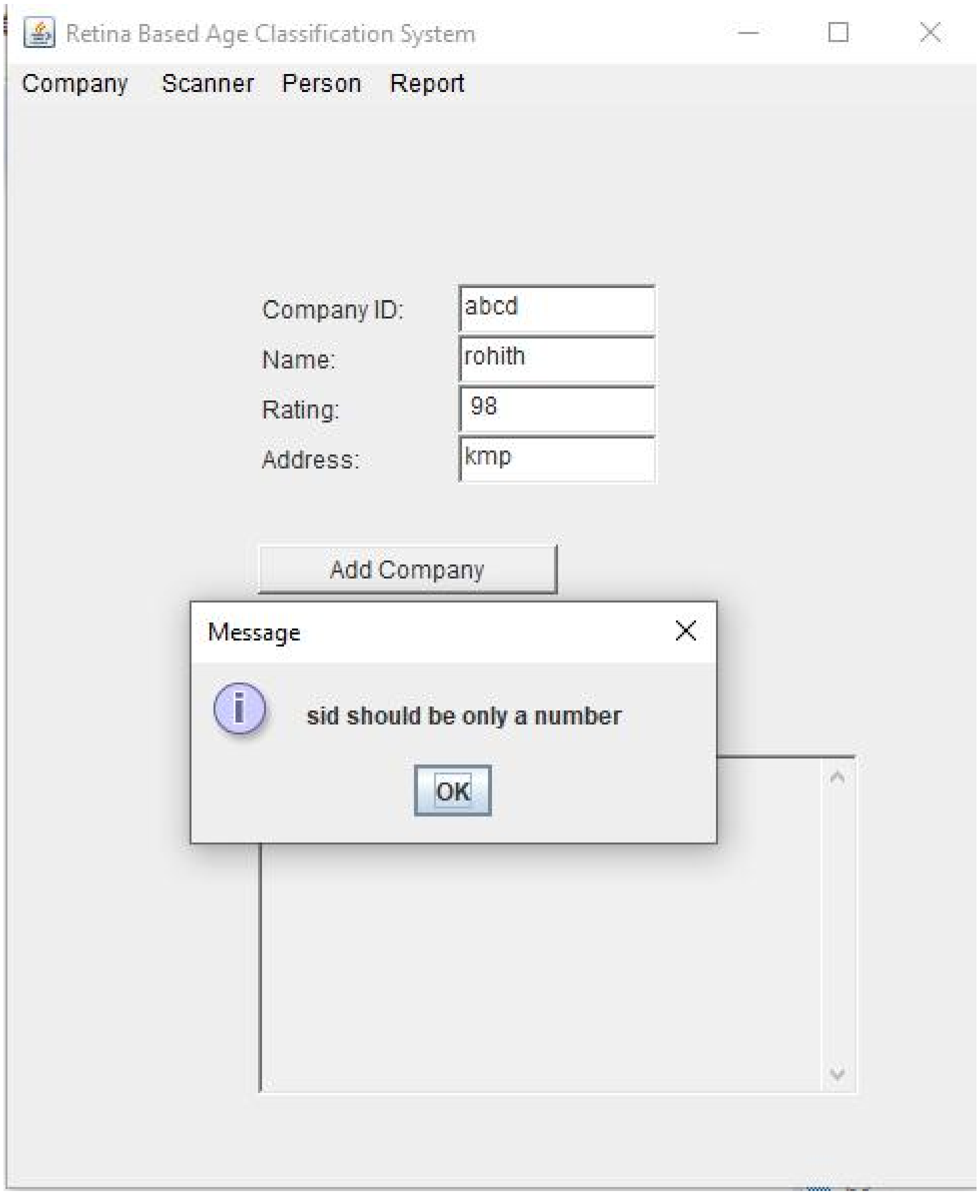


```
SQL> select * from company;

        ID NAME                 ADDRESS
---------- -------------------- ----------------------------------------
    RATING
----------
      1001 iris id              united states
        95

      1002 cmi tech             korea
        90

      1003 princeton identity   united states
        85


        ID NAME                 ADDRESS
---------- -------------------- ----------------------------------------
    RATING
----------
      1004 sensor tech          united kingdom
     85.55

      1005 argus                united kingdom
        85

      1006 anoop                hyd
      98.5


6 rows selected.

SQL>
```

Extra features present in the application:

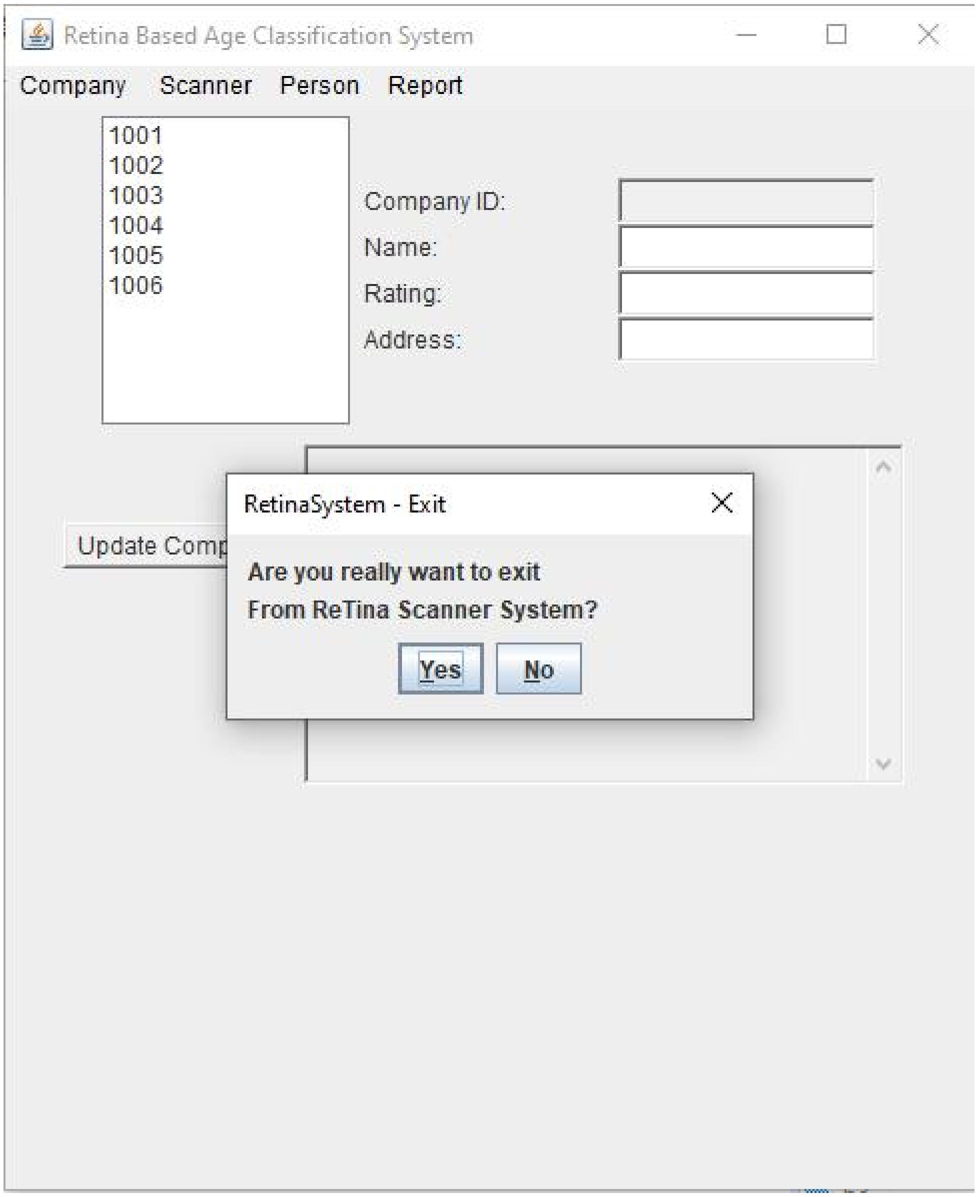
1) Many times unknowingly we may give mismatching data types in the text fields like id’s, etc. So in this application we will show warning to the users when they give wrong data with the help of jOptionPane present in the swing package.

2)Many times unknowingly we may click exit button and then we realize we have clicked exit and we need to start the work from first. To avoid this problem when a user click exit opion we will show a option like do you really want to exit if he click yes then only we will close the application.



Company ID: abcd
Name: rohith
Rating: 98
Address: kmp

sid should be only a number



Are you really want to exit
From ReTina Scanner System?

RESULTS

The DML commands, Insert, update and delete for one of the tables in given below:

For company table: (in java, as per the application)

Insert:

```
"INSERT INTO company VALUES(" + sidText.getText() + ", " + "'" +
snameText.getText() + "','" + addressText.getText() + "'," + ratingText.getText()
+ ")";
```

Update:

```
"UPDATE company "+ "SET name='" + snameText.getText() + "', "+
"rating=" + ratingText.getText() + ", "+ "address ='"+ addressText.getText() + "'
WHERE id = "+ companyIDList.getSelectedItem()
```

Delete:

```
"DELETE FROM company WHERE ID = "+ companiesIDList.getSelectedItem()
```

REFERENCES

1. http://sociallearningcommunity.com/10-of-the-best-mooc-providers/
2. https://en.wikipedia.org/wiki/List_of_MOOC_providers
3. https://github.com/sheelasairohith/dbma assingment