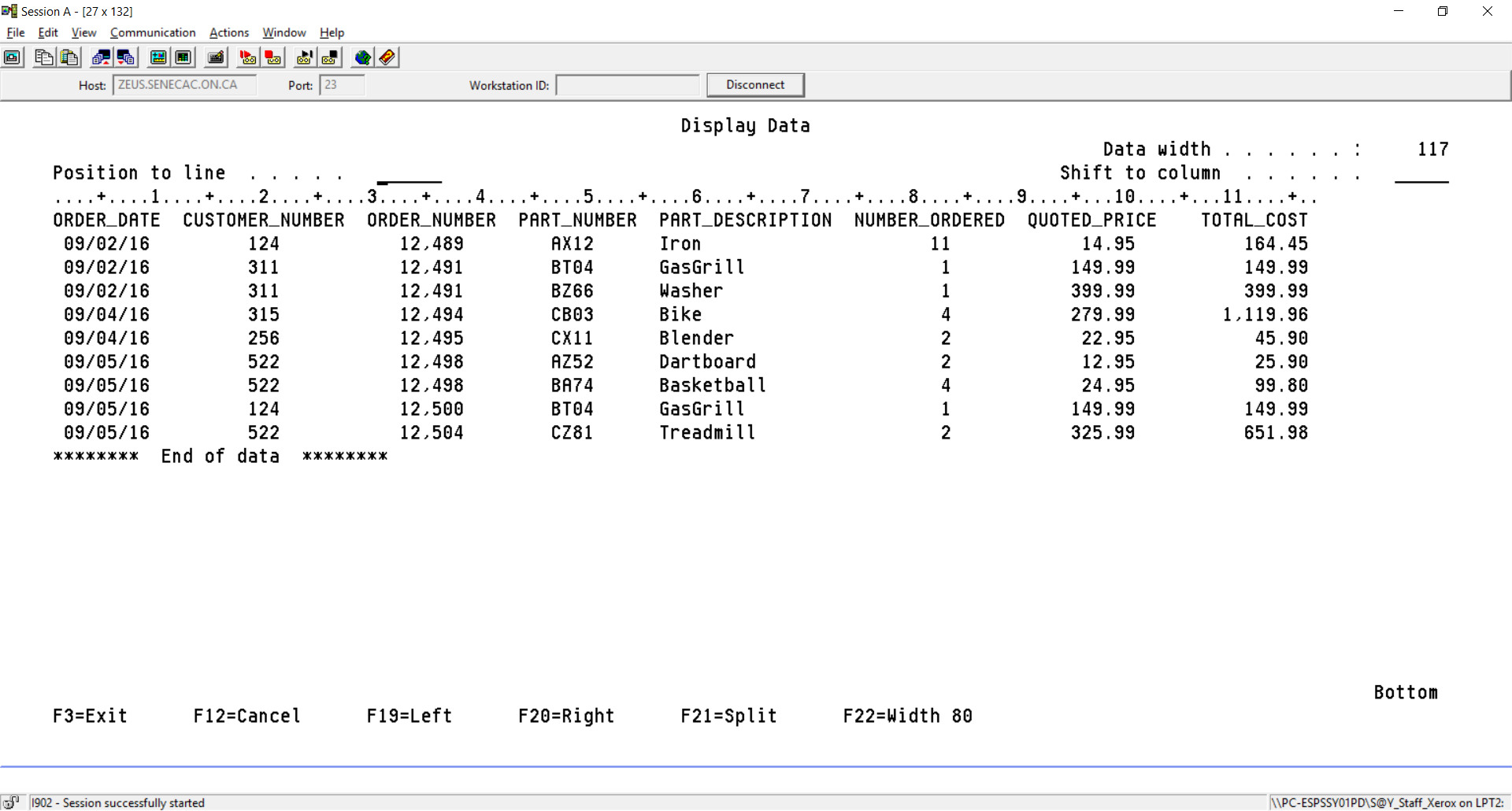
**Purpose – In this lab you will learn how to:**

* **ALTER a VIEW**
* **Distinguish and experience the options for a column. Not Null with Default, Not Null, or a different default than the natural one for the data type**
* **Use information from physical design documents to implement tables**
* **Grant permissions giving other users access to your database. This is often the responsibility of the Database Administrator (DBA) so consider yourself the DBA of your database!**

**Requirements to pass the lab:**

* **Hand in a system generated printout of the DSPFD command proving you have created the STUDENT table properly with all the constraints applied.**
* **Hand in a system generated printout of the DSPFD command proving you have created the PGM table properly with all the constraints applied.**
* **Provide screenshots indicating you have set permissions to your collection and the customer table correctly**
* **Answer the questions on the answer sheet**

1. In Lab 4 you created a VIEW called ORDER\_INFO. In that view you referred to a PART\_NUMBER and you did not know the actual part name. Delete your ORDER\_INFO view and recreate it so you can include the part description. (You may have created that view in your library (DB201???) instead of the collection (PREM…)



Unfortunately, a VIEW cannot be changed once it has been created. So, to add the part description in our view we go through the following steps:

1. Set the Schema to your collection name (PREMIER???)
2. DROP the View ===>DROP VIEW ???
3. Bring back the SQL command that created the VIEW
4. Make the changes to add the part description to the VIEW and
5. CREATE the VIEW again
6. On the answer sheet, provide the statement to create the revised Order\_Info view.

Use F9 a few times or use paging up and search for the statement used to create your view in the last lab. When you find the statement by paging up you need to place your cursor on the statement and press F9 to bring it down as the next statement to be executed. You now need to mention the PART table and the part description column.

Note that if your report is very long, then you have probably forgotten to use the WHERE clause to match the rows of the tables!! In that case you would have produced the dreaded CARTESIAN JOIN (every row of table 1 joined to every row of table 2).

**2. The three options for a column or field – NOT NULL WITH DEFAULT, NOT NULL, a different default than the natural one for a data type or nothing specified.**

You can bypass this section if you know the differences with the four possible choices above.

But ... you can also run it and prove it works as advertised.

If you were wondering about the options for a column, you could devise a little experiment that shows what happens with the four choices.

We have AONE, ATWO, ATHREE and AFOUR as column names for character fields. The main difference is choosing not null with default, not null, not adding a clause, or specifying a different default that the usual one for that column type after defining the type and size CHAR (5)

CREATE TABLE TESTING( AONE CHAR (5) NOT NULL WITH DEFAULT,

ATWO CHAR (5) NOT NULL,

ATHREE CHAR (5),

AFOUR CHAR(5) NOT NULL WITH DEFAULT 'XXXXX')

**Lets just put something in AONE and ignore ATWO, ATHREE and AFOUR**

INSERT INTO DB201B40/TESTING (AONE) VALUES('A')

Null values not allowed in column or variable ATWO.

**It did not work.**

**Lets just put something in ATHREE and ignore AONE ATWO and AFOUR**

INSERT INTO DB201B40/TESTING (ATHREE) VALUES('A')

Null values not allowed in column or variable ATWO

**It did not work.**

**Lets just put something in ATWO and ignore AONE, ATHREE and AFOUR**

INSERT INTO DB201B40/TESTING (ATWO) VALUES('A')

1 rows inserted in TESTING in DB201B40

**It worked. It is only necessary to supply a value for ATWO**.

**What was put in AONE, ATHREE and AFOUR?**

SELECT \* FROM TESTING

AONE ATWO ATHREE AFOUR

A - XXXXX

\*\*\*\*\*\*\*\* End of data \*\*\*\*\*\*\*\*

**AONE has spaces, ATWO has the letter ‘A’ ,ATHREE is showing '-' representing null and AFOUR has our specified default value.**

**Can you refer to null with the minus sign?**

DELETE FROM TESTING WHERE ATHREE = '-'

Row not found for DELETE

**No, you need to specify NULL.**

DELETE FROM TESTING WHERE ATHREE IS NULL

1 rows deleted from TESTING in DB201B40

You may want to drop the table and try three numeric fields and three date fields to see what the default values are.

When you say Not Null with default – what is the default for a date column when you do not specify a value when inserting a row?

Provide the answer for this on the answer sheet.

**3. Using information from physical design documents to implement tables**

Create the following two tables in your PREMIERxxx collection. (Don’t forget the constraints!)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Column | Data  Type | Length | PK | FK | NOT  NULL | Unique | Check | Default |
| Table: STUDENT |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
| Campus\_no | | Char | 1 | Y |  |  |  |  |  |
| Stud\_no | | Numeric | 3 | Y |  |  |  |  |  |
| Stud\_lname | | Char | 15 |  |  |  |  |  |  |
| Stud\_fname | | Char | 15 |  |  |  |  |  |  |
| Stud\_pgm | | Char | 3 |  | PGM(Pgm\_id) |  |  |  | ‘CPA’ |
| Locker\_no | | Numeric | 3 |  |  |  | Y |  |  |
|  | |  |  |  |  |  |  |  |  |
| Table: PGM | |  |  |  |  |  |  |  |  |
|  | |  |  |  |  |  |  |  |  |
| Pgm\_id | | Char | 3 | Y |  |  |  | ‘CPA’, ‘CPD’ |  |
| Pgm\_name | | Char | 40 |  |  | Y | Y |  |  |
| No\_semesters | | Numeric | 1 |  |  | Y |  | > = 0 | 6 |

Note: a 2-Part primary key constraint must be defined after the fields are defined, usually at the bottom of the CREATE TABLE command.

Insert the following data into your tables, be sure to enter the rows of the parent table before the rows of the child table:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Student | | | | | |
| 1 | 123 | Jane | Wong | CPD | 900 |
| 1 | 456 | David | Kasim | CPA | 905 |
|  | | | | | |
|  | | | | | |
| Pgm | | | | | |
| CPD | | Computer Programming | | | 4 |
| CPA | | Computer Programming and Analysis | | |  |
|  | | | | | |

If you had trouble inserting the last row, read on! When we want the system to insert a default value into a field We must use the INSERT command and specify exactly which fields we wish to supply values for. Enter the command as follows:  
  
INSERT INTO PREMIERxxx.PGM (Pgm\_id, Pgm\_name)   
 VALUES(‘CPA’, ‘Computer Programming and Analysis’)  
  
Now run a select command on the PGM file. What is listed as the No of Semesters for the CPA program?

What happens if you enter the following data into the Student table?

(1, 321, ‘Joe’ , ‘Chen’, ‘CPC’, 906)  
 (1, 654, ‘Cindy’, ‘Wong’, ‘CPA’, 905)

You should NOT be able to insert these two records! The Foreign Key should prevent the Joe Chen row from being added because the parent record is missing on the PGM table. The Cindy Wong row should be rejected because of a UNIQUE violation on Locker\_No. If the rows were not rejected then delete the two rows that you have just entered which should have been rejected using DELETE FROM STUDENT WHERE … on the green screen. You could also have deleted them using the Navigator. Correct your FOREIGN KEY and UNIQUE constraints and then try to enter the two records again.

**4. Granting Permissions giving other users access to your database and tables**

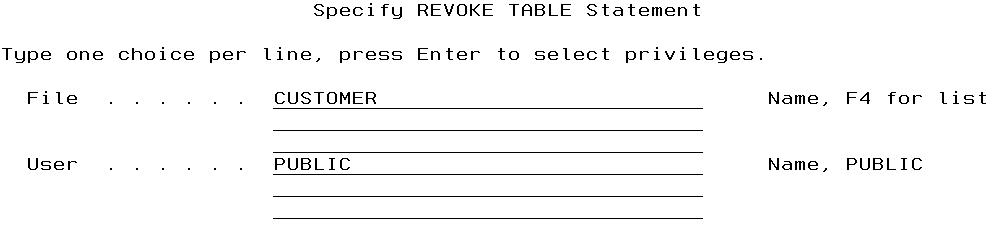
To give permission to use your schema objects to another user, you must give permission for both the collection AND the tables and/or VIEWs.

Try these security setting activities in interactive SQL:

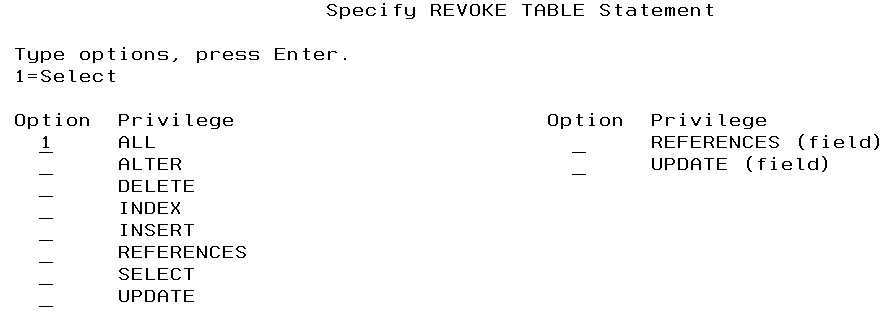
SET SCHEMA PREMIER???

The default security setting for the CUSTOMER table you created in your collection allows too much access. We will remove that access.

Type REVOKE, press F4 and select Table. Indicate the file name is CUSTOMER and you want to revoke PUBLIC privileges.



Press enter and on the next screen indicate you are revoking ALL



On the answer sheet, indicate what was the statement that was executed to do this?

We are going to allow various levels of access to your CUSTOMER and SALESREP tables. Here are some of the options available when allowing access.

**The following privileges can be used with the SQL GRANT statement for a table**:

ALL

Grants ALL table (file) privileges you currently have for all the tables or views (logical files) named in the ON clause. If ALL is not selected, you must use one or more of the keywords that follow.

Note: An error will result if the ALL privilege is selected with any of the other privileges.

ALTER

Grants the privilege to use the ALTER TABLE, COMMENT ON, and LABEL ON statements on the tables (files) or views (logical files) names in the ON clause.

DELETE

Grants the privilege to use the DELETE statement on the table (physical file) or view (logical file) named in the ON clause.

INDEX

Grants the privilege to use the CREATE INDEX statement on the table (physical file) named in the ON clause. Views may not be specified when granting INDEX privileges.

INSERT

Grants the privilege to use the INSERT statement on the table (physical file) or view (logical file) named in the ON clause. Shows up as ADD in iSeries Navigator and with the native DSPOBJAUT command.

REFERENCES

The user specified is granted REFERENCES privileges to the tables or views (logical files) specified in the ON clause.

REFERENCES (column)

The user specified is granted REFERENCES privileges to the columns selected from the tables specified in the ON clause.

SELECT

Grants the privilege to use the SELECT statement on the table (physical file) or view (logical file) named in the ON clause. Shows up as READ in iSeries Navigator and with the native DSPOBJAUT command.

UPDATE

Grants the privilege to use the UPDATE statement on the table (physical file) or view (logical file) named in the ON clause.

UPDATE (column)

Grants the privilege to use the UPDATE statement on columns selected from the table (physical file) named in the ON clause.

Apply the following permissions to your CUSTOMER and SALESREP tables. Where you see YOURINSTRUCTOR, put in your instructor’s user profile.

Instructor Id’s are as follows:

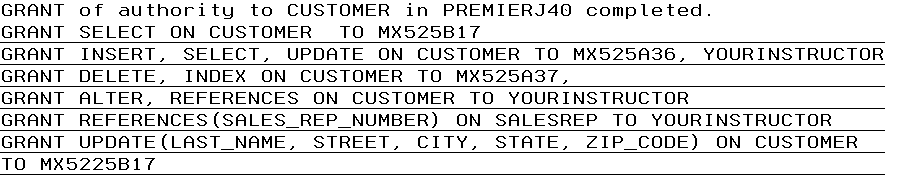
Lydia Li LYDIA\_LI

Parul Kantaria PKANTARIA  
Russell Pangborn PANGBORN  
Nazim Razavi NRAZAVI

Aliakbar Baharikhoob ALIAKBAR

Ayesha Manzer AMANZER

You will need to do them one at a time, and you should get a success message similar to the one showing at the top for each statement.



Note that user profiles get deleted from the system, so you instructor may give you a substitute profile for MX525A36, MX525A37 and MX525B17.

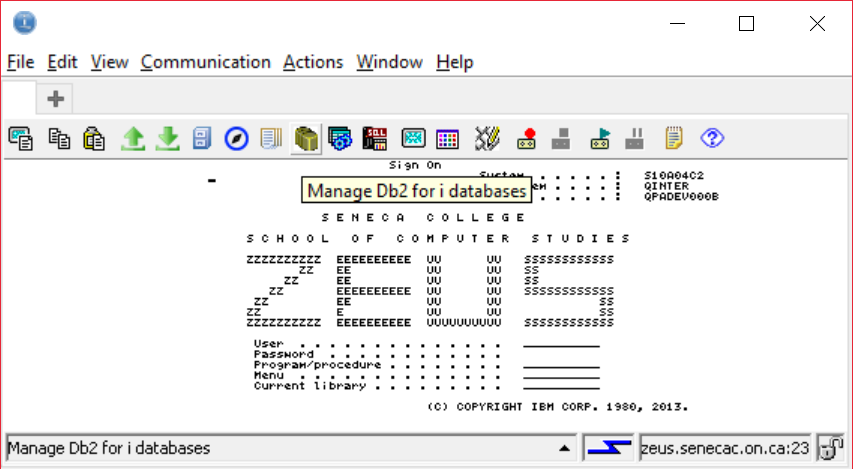
Remember, the public may have read, write, update and delete access to your SALESREP table by default when the table was created, so run the command that removes those privileges.

You will give all instructors access to your collection, but only allow your instructor to access certain tables in that collection.

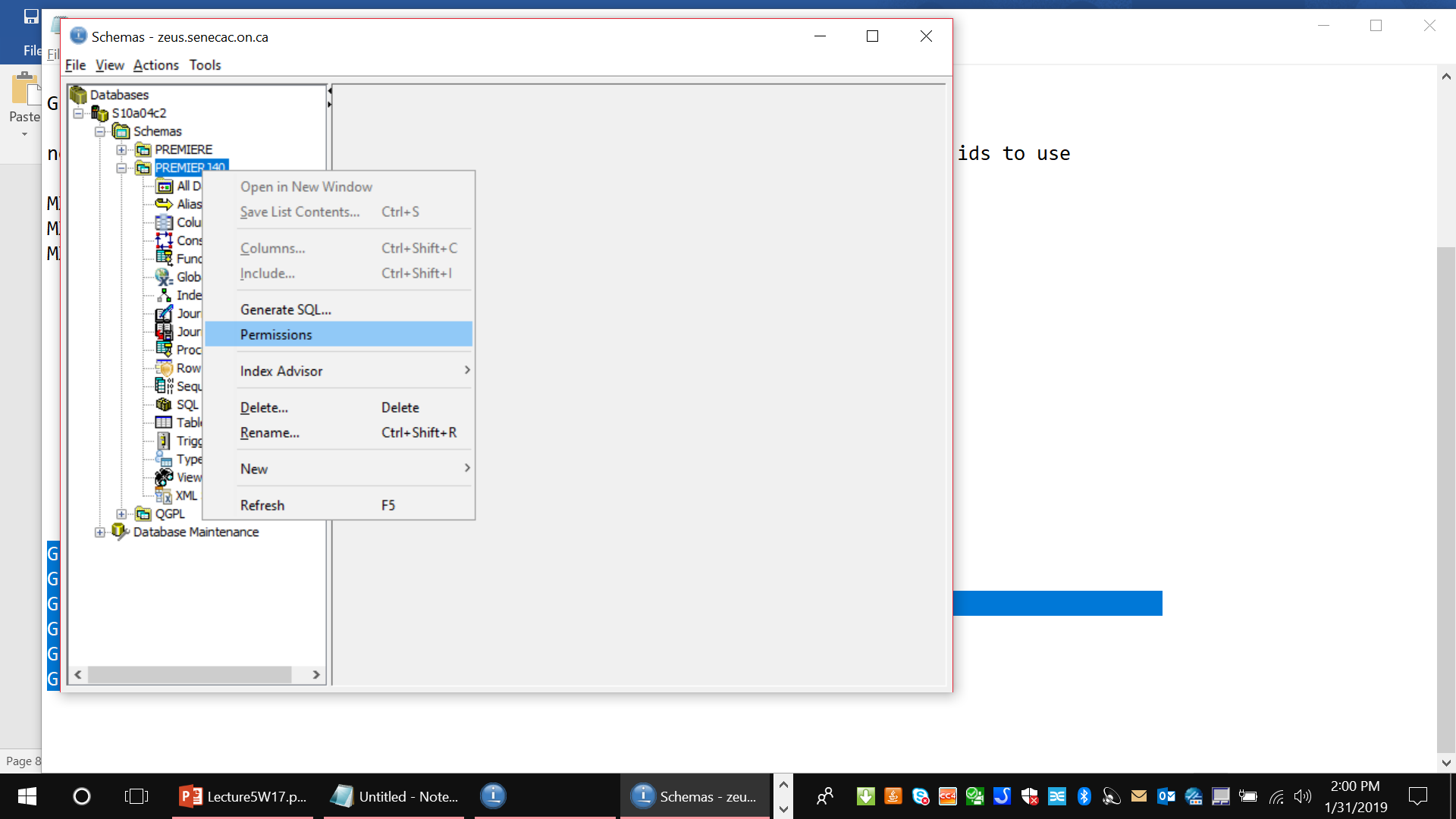
There is no SQL statement to allow access to your schema (collection, Library)

You can do this using a GUI in Access Client Solutions (ACS)

On the Access Client Solutions 5250 emulator screen click on the ninth icon at the top. It should say Manage DB2 for i Databases when you hover over it.



You have used this before. Make sure your collection is included in the Schema’s view. Then, right click on you collection name and select Permissions.



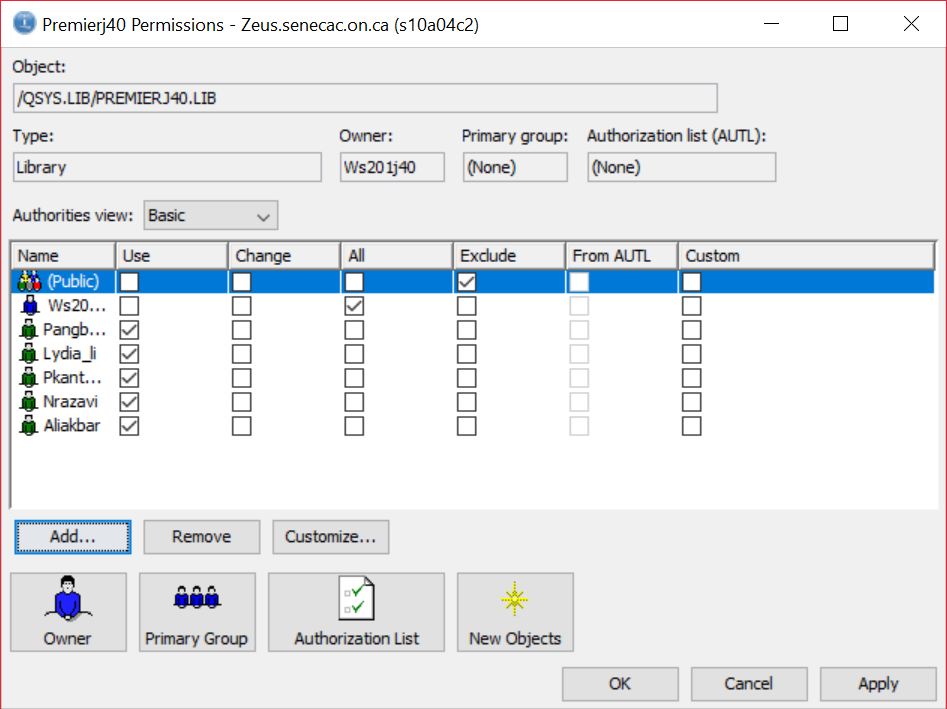
You need to make sure your collection name shows correctly under Object (when handing in a screenshot)

The Public will probably show as having Change authority. You will click on Exclude.

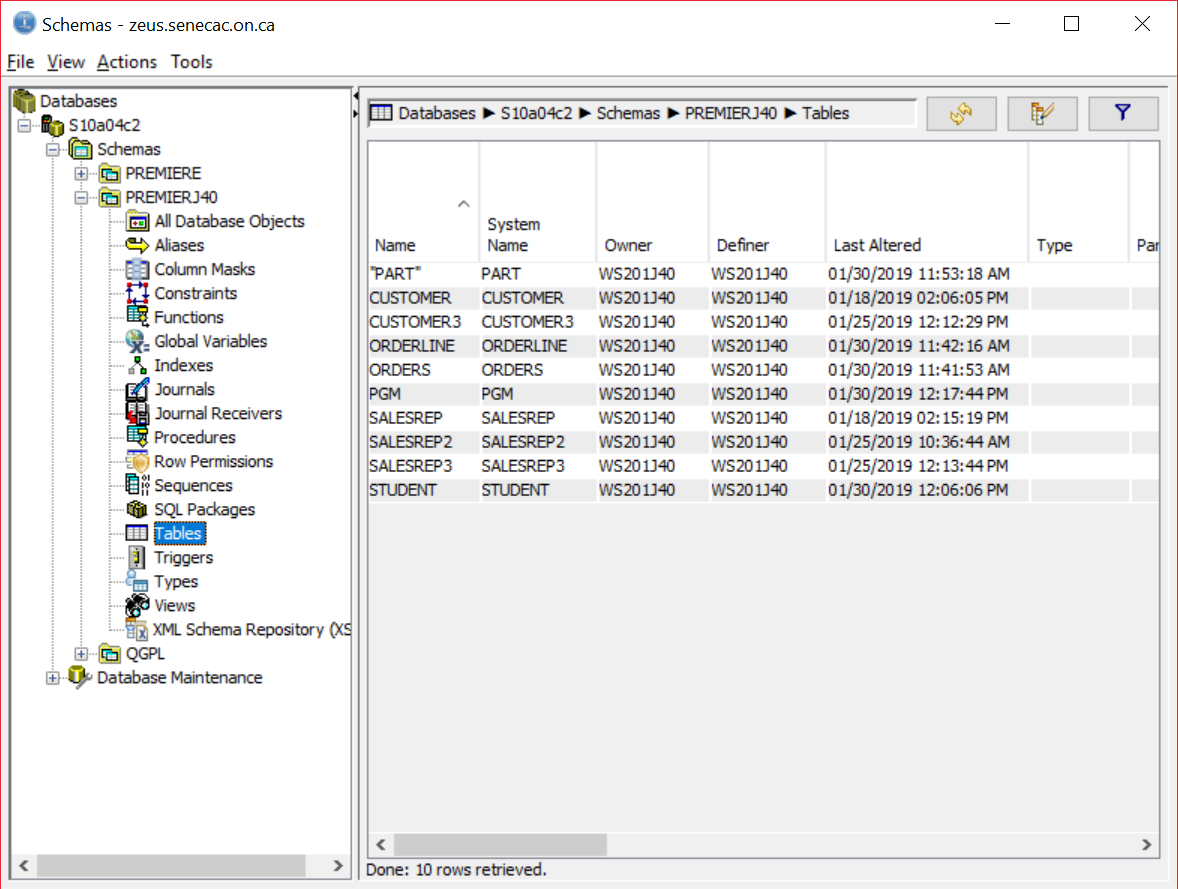
Be careful not to accidentally click on the line that indicates you have All access to this collection.

The ADD button can be used to add the instructors mentioned earlier in this lab. Since instructors change every semester, if your instructor is not on this list, add them here.

After you have added them, make sure you click on the Apply button on the lower right – or you may lose their added profiles to your collection.

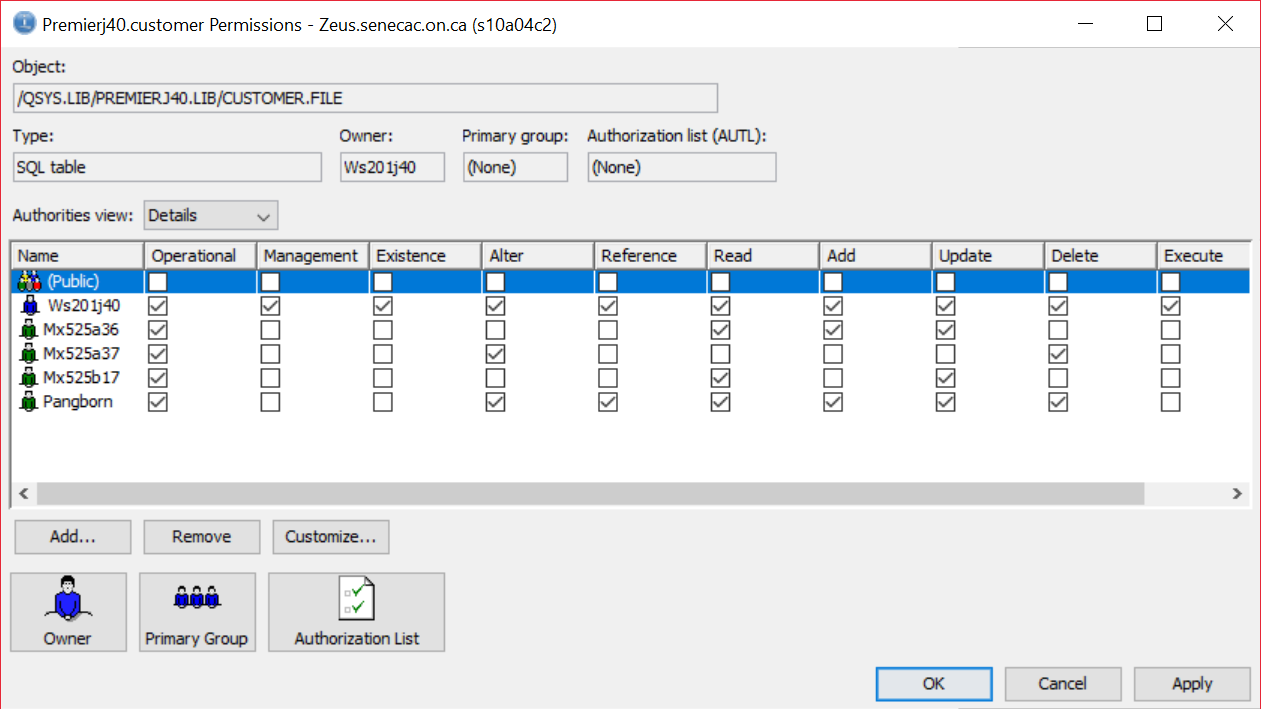


After you have set access to your collection in the GUI, take a look how the tables you set permissions for with SQL statements look.



My schema has a few versions of CUSTOMER and SALESREP - you will probably only have a single version.

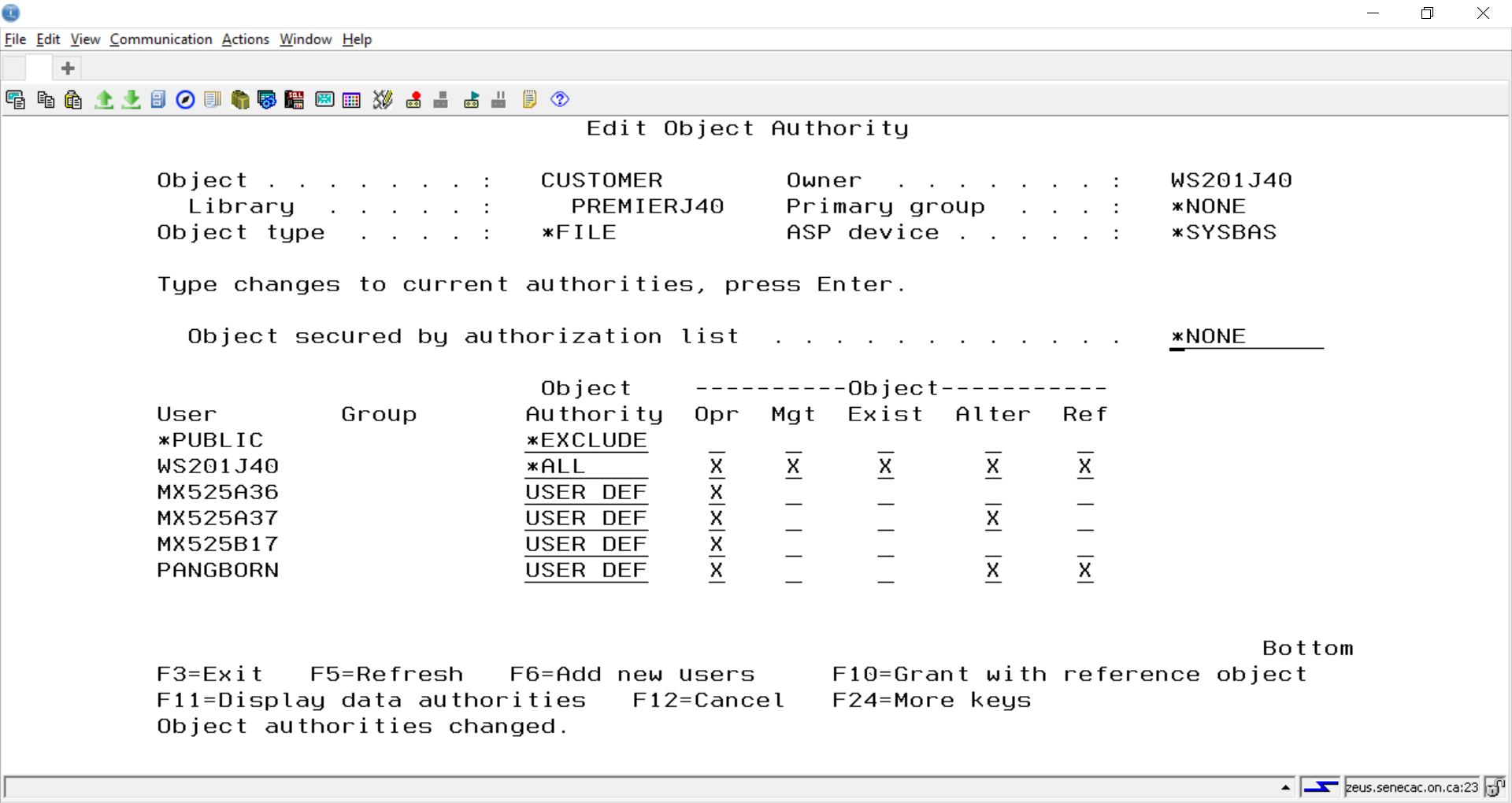
Notice the authorities view is set to Details. This is the screenshot you want to provide. It should refer to your object. PREMEIR???.Lib/CUSTOMER.FILE. On the answer sheet indicate one piece of information on how you set permissions to CUSTOMER that is not showing here.

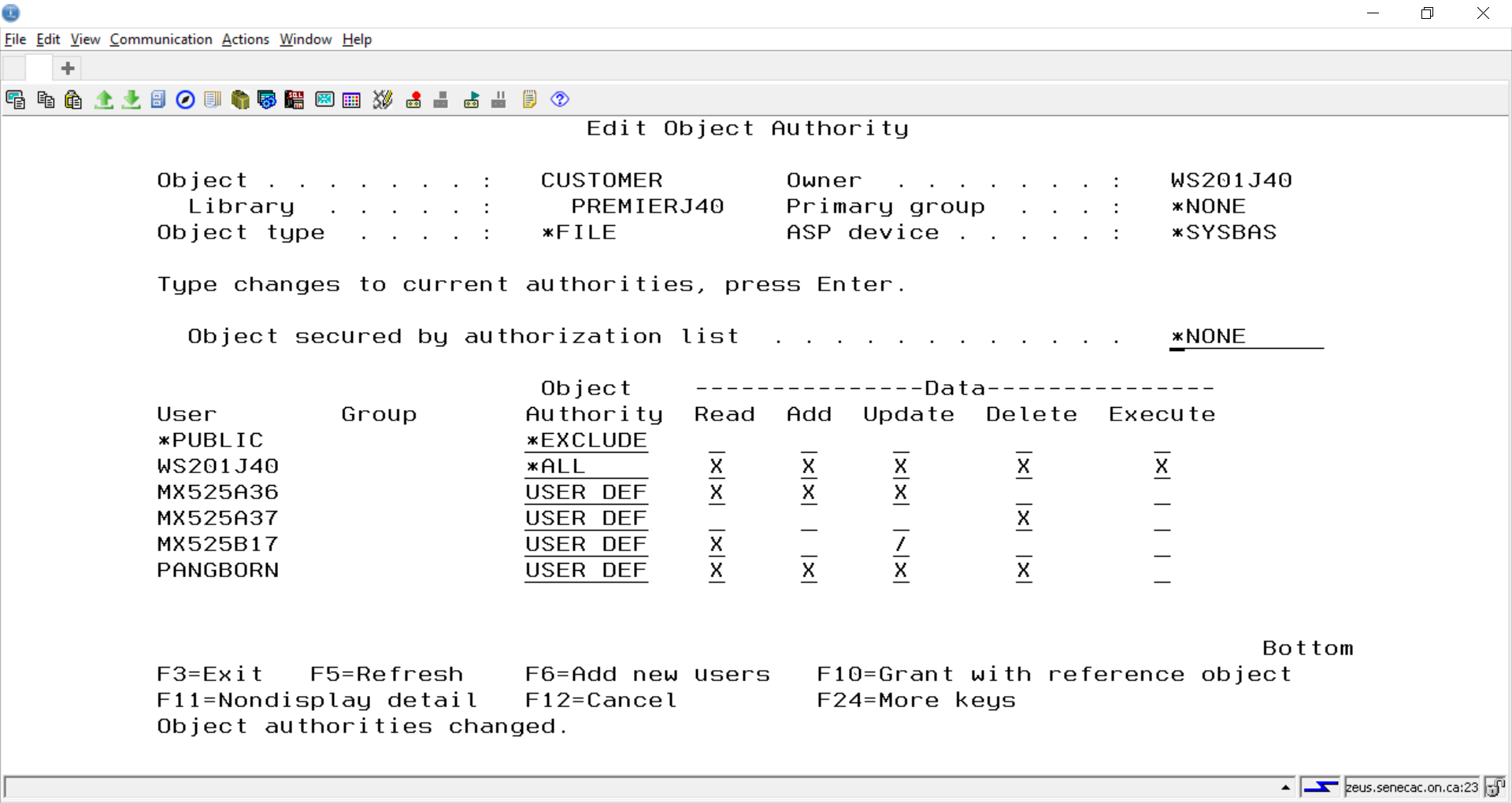


It is easier for some people to not run the GRANT or REVOKE SQL commands and click on the authorities you want to grant using the GUI. It is preferable to use the actual SQL statements when there are a lot of tables to create and have the permissions set in a script. You run the script once as needed and do not waste time clicking on individual settings.

At the command line (not in interactive SQL) you can use a native command to get to setting permissions.

==>EDTOBJAUT PREMIERJ40/CUSTOMER \*FILE You can cycle through several views with the F11 key and place an x or a blank to set permissions. Some settings may have been changed with these screenshots.





The native command to set access to your collection is:

==>EDTOBJAUT QSYS/PREMIERJ40 \*LIB

Security to your collection is important. Unfortunately, when you create a collection in SQL the default is the public has access to your schema. This is not acceptable for this subject. From this moment on, your PREMIER??? Collection and any other collection you create for a project should not be open to other students. We can run a program to determine if collections have the appropriate security setting. Each instructor may make a decision to penalize a student or a group for not protecting their work from being copied. Check with your instructor about any penalties for not protecting PREMIER??? at the completion of this lab.

Allowing access is a two stage process. First there is the collection access, second there is the table access.

The easiest way to limit access is to deny access to your collection. Then you do not have to worry about table access in the collection. If you do allow access to the collection but deny access to the table in the collection, then the user will not be able to access the table.

Let’s remove some authorities in interactive SQL

REVOKE SELECT ON CUSTOMER FROM MX525B17

REVOKE INSERT,SELECT, UPDATE ON CUSTOMER FROM MX525B36

REVOKE DELETE,INDEX ON CUSTOMER FROM MX525A37

When you are ready to take your screenshot of the permissions to CUSTOMER, you can change the setting back to match the screenshot by clicking in the appropriate boxes.

On a test or the exam, we will not be asking you to click on boxes. So be prepared to construct a GRANT or REVOKE SQL statement.